

# Planning and Control of Uncertain Cooperative Mobile Manipulator-Endowed Systems under Temporal-Logic Tasks

Christos K. Verginis



**Planning and Control of Uncertain Cooperative  
Mobile Manipulator-Endowed Systems under  
Temporal-Logic Tasks**

CHRISTOS K. VERGINIS

PhD Thesis  
Stockholm, Sweden, 2020

KTH Royal Institute of Technology  
School of Electrical Engineering  
and Computer Science  
Division of Decision and Control Systems  
SE-100 44 Stockholm  
Sweden

TRITA-EECS-AVL-2020:20  
ISBN: 978-91-7873-503-7

Akademisk avhandling som med tillstånd av Kungliga Tekniska högskolan  
framlägges till offentlig granskning för avläggande av May 2020 i .

© Christos K. Verginis, May 2020

Tryck: Universitetsservice US AB

## Abstract

Control and planning of multi-agent systems is an active and increasingly studied topic of research, with many practical applications such as rescue missions, security, surveillance, and transportation. This thesis addresses the planning and control of multi-agent systems under temporal logic tasks. The considered systems concern complex, robotic, manipulator-endowed systems, which can coordinate in order to execute complicated tasks, including object manipulation/transportation. Motivated by real life scenarios, we take into account high-order dynamics subject to model uncertainties and unknown disturbances. Our approach is based on the integration of tools from the areas of multi-agent systems, intelligent control theory, cooperative object manipulation, discrete abstraction design of multi-agent-object systems, and formal verification.

The first part of the thesis is devoted to the design of continuous control protocols for the cooperative object manipulation/transportation by multiple robotic agents, and the relation of rigid cooperative manipulation schemes to multi-agent formation. We propose first a variety of centralized and decentralized control algorithms that do not employ force/torque information at the contact points and take into account both cases of rigid and rolling grasping points, dynamic uncertainties in the object's and agents' model, and potential constraint satisfaction, such as obstacle avoidance and input saturation. Next, we tackle the problem of robust formation control for a class of multi-agent systems and we analyze the relation between formation rigidity theory and rigid cooperative manipulation.

In the second part of the thesis, we develop control schemes for the continuous coordination of multi-agent complex systems with uncertain dynamics. We first study the motion planning problem and propose novel adaptive control schemes for the collision-free navigation of single- and multi-agent spherical systems in obstacle-cluttered environments. Next, we focus on the leader-follower coordination problem of spherical multi-agent systems. More specifically, we design a robust adaptive decentralized control scheme for the successful navigation of a designated leader to a predefined point, while guaranteeing collision avoidance and connectivity maintenance properties. Finally, we design a closed-form robust barrier function-based control protocol for the collision avoidance of multiple 3D ellipsoidal agents.

The third part of the thesis is focused on the planning and control of multi-agent and multi-agent-object systems subject to complex tasks expressed as temporal logic formulas. We tackle first the case of local independent tasks for multi-agent systems, and by using previous results on multi-agent constrained navigation, we design a discrete abstraction of the agents' motion in the workspace and synthesize decentralized control policies that satisfy the agents' specifications. Next, in addition to the robotic agents,

we take into account complex tasks to be satisfied by unactuated objects. We design a discrete abstraction that simulates the behavior of the agents and the objects in the workspace and we synthesize controllers for the agents that take into account both theirs and the objects' specifications.

The fourth and final part of the thesis focuses on several extension schemes for single-agent setups. Firstly, we consider the problem of single-agent motion planning under timed temporal tasks in an obstacle-cluttered environment. Using previous results on collision-free timed navigation, we develop a novel control policy that guarantees satisfaction of the agent's timed tasks as well as asymptotic optimality with respect to energy resources. Secondly, we tackle the motion planning problem for high-dimensional complex systems with uncertain dynamics in obstacle-cluttered environments. We integrate intelligent control techniques with sampling-based motion planning algorithms to guarantee the safe navigation of the system to a predefined goal, while compensating for the model inaccuracies. Finally, we develop a novel control protocol that achieves asymptotic reference tracking for an unknown control affine system, while respecting at the same time funnel constraints.

## Sammanfattning

Reglering och planering av multiagent-system är ett aktivt och växande forskningsfält med en rad praktiska tillämpningar såsom räddningsuppdrag, övervakning, säkerhet och transport. Denna avhandling adresserar planering och reglering av multiagent-system med temporallogiska uppgifter. De berörda systemen är komplexa, robotiska, manipulatorbaserade system, vilka kan samarbeta för att utföra komplicerade uppgifter, bland annat manipulation och transport av objekt. Motiverade av verkliga scenarier tar vi hänsyn till högnivå dynamik med osäkerheter och okända störningar. Vårt angreppssätt baseras på integration av redskap från följande områden: multiagent-system, intelligent regler teknik, samarbete objekt-manipulation, diskret abstrakt design av multi-agent-objekt system och formell verifiering.

Avhandlingens första del tillägnas design av kontinuerliga protokoll för samarbete manipulering och transporter objekt utförd av flera robotagenter, och relationen av rigida samarbetskravande manipulationsuppgifter. Vi föreslår först några centralisera och decentralisera regleralgoritmer som saknar information om kraft och moment i kontaktpunkterna, men som tar hänsyn till både fasta och rullande greppunkter, dynamiska osäkerheter i objektets och agentens modell, samt möjlighet till att uppfylla villkor såsom att undvika hinder och mätning av insignaler. Som ett nästa steg behandlar vi reglering för robust formering för en klass av multi-agent system och vi analyserar relationen mellan teori för formationsrigiditet och rigid samarbetskravande manipulation.

I avhandlingens andra del utvecklar vi regleralgoritmer för kontinuerlig koordinering av komplexa multi-agent system med osäker dynamik. Vi betraktar först rörelseplanering och föreslår nya adaptiva regleralgoritmer för kollisionsfri navigering av enkel- och sfäriska- multiagent-system i hinderfyllda miljöer. Vi fokuserar sedan på ledar-följare koordinering av sfäriska multiagent-system. Mer specifikt så designar vi robust adaptiv decentralisera reglering för framgångsrik navigation av en utnämnd ledare till en förutbestämt punkt, samtidigt som vi garanterar att kollisioner kan undvikas och att sammankoppling upprätthålls. Slutligen designar vi ett robust reglerprotokoll på återkopplad form baserat på barriärfunktion för kollisionsundvikande av multipla ellipsoidformade 3D-agenter.

Avhandlingens tredje del fokuserar på planering och reglering av multiagent och multiagent-objekt system med komplexa uppgifter uttryckta med formler på temporallogisk form. Vi behandlar först fallet med lokala oberoende uppgifter för multiagent-system, och genom att använda tidigare resultat från begränsad navigering av multiagent-system designar vi en diskret abstraktion av agentens rörelse i arbetsytan och syntetisera decentralisera reglerpolicyer som uppfyller agentens specifikationer. Förutom

robotagenterna tar vi sedan även hänsyn till komplexa uppgifter som utförs av opåverkade objekt. Vi designar en diskret abstraktion som simulerar agenternas beteenden och objekten i arbetsytan och vi syntetiseras regulatorer som tar hänsyn både till agenternas och objektens specifikationer.

Den fjärde och sista delen av avhandlingen fokuserar på flera utvidgningar för singelagentfallet. Först betraktar vi rörelseplanering för singelagenter under temporaala uppgifter i en hinderfyllt miljö. Genom att använda tidigare resultat från kollisionsfri tidsbegränsad navigering utvecklar vi en ny reglerpolicy som garanterat uppfyller agentens tidsbegränsningar och är asymptotiskt optimal med avseende på energikällor. Sedan angriper vi rörelseplaneringproblemet för mångdimensionella komplexa system med osäker dynamik i hinderfyllda miljöer. Vi integrerar intelligenta tekniker för reglering med samplingsbaserade rörelseplaneringsalgoritmer för att garantera säker navigering av systemet till ett förutbestämt mål, samtidigt som vi kompenseras för modellfel. Slutligen utvecklar vi nya regleringsprotokoll som uppnår asymptotisk referensföljning för ett okänt affint system, samtidigt som trattformade begränsningar uppfylls.

# Acknowledgements

First and foremost, I would like to express my gratitude to my supervisor Prof. Dimos Dimarogonas for his valuable support, guidance and encouragement; 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 thank my co-supervisor Prof. Danica Kragic for her great insight, knowledge and assistance, and acknowledge the Knut och Alice Wallenberg Foundation, which funded partially my work.

Special thanks go to my advisor during my stay in Houston, Prof. Lydia Kavraki. She has been very kind and supportive, always providing significant research insights and motivation. The experience I gained is invaluable and I consider it a privilege to have worked with her group.

I thank all my colleagues (current and former) at the Decision and Control Systems for creating a nice atmosphere, and for all the exciting, boring, difficult, happy, and sad moments we shared. Special thanks go to my colleagues Alex, Lars, and Wences for our joint works and their continuous assistance and support, as well as to Pedro R., Pedro P., Pian, Peter, Sebastian, Antonio, Andrea, Xiao, Wei, Fei, Maria, Luis, Pierre-Jean, Leonardo, Dionysis, Christoforos, Shahab, Dimitris, Souleimane for our exciting group meetings and to Sofie for her MITL implementations. I would especially like to thank Dimitris Boskos for discussing with me all my strange research and notation questions and teaching me how to be technically rigorous. I am also grateful to Jana for conveying her excitement for temporal logics. Big thanks go to Ziwei, Matteo, Yu, Imran, Cristina, Nicola, Francesca and Akash for our great collaboration during their master theses and for being patient with me. I would also like to thank all the professors and the administrative staff of the department. Special thanks to Silvia, Felicia, Anneli, Christer for their valuable help, and to Emmy for answering my vast amount of questions and making hence this thesis possible.

Thank you David, for organizing the after work activities, and Mladen, Jezdimir, Rui, Rijad, Ines, and all the other kitchen regulars for our endless lunch discussions. Joana, thank you for pushing us to learn Swedish, our little

group was successful while it lasted. I am grateful for Robert, Matias, Emma, and Elis for their positive spirit and for being great companions in our travels abroad. I am glad to have met the Italian attitude of Antonio, Riccardo, Valerio, Marco, and Demia. Thank you Peter for showing me stereograms and how to eat chestnut puree. Thank you Vahan for the khinkali. I would also like to thank my great roommates, Pedro Roque, Goncalo, Manne, Rong, and of course, Pedro Pereira, whose need for technical rigorousness motivated me and contributed to my compulsion. Thank you Dionisi, for always passing by to discuss random things, we have missed you. Thanks also to the SML people. Thank you Aldo for your positive attitude and inexplicably contagious laughter, Robin and Kuba, for refusing to give up on the robotic arms and my controllers, and Pedro, for upgrading the SML and pushing me (not very successfully though) to experiment with real robots.

I would further like to thank my Co4Robots partners and friends. Thank you Dimitri, Wei, Christos, and Pouria for your help in the project organization and implementation. Thank you Pedro, Sergio, Michali, Kosta Alevizo, Kosta Roditaki, Alexandre, Philipp, Alessandro, George, Claudio, and Meng for all our experiences during this project. I further thank the rest of the CSL people, Prof. Kyriakopoulos, Costas Vrohidis, and Panagiotis Vlantis for our collaboration during the project as well as my master thesis, and of course Babis, for exposing me to intelligent control systems and teaching me Lyapunov tricks. Babi, I wouldn't be here without you.

I would like to thank my friends in Stockholm. Spyropoule and Charisi, thank you for all the weekend lunches, fikas, and drinks. Alex, thank you for being there for me when I started, for being a good friend, and for always having the answer in all my bureaucratic and application questions. Pedro, thank you for being a great colleague, roommate, and friend. Thank you for co-piloting Co4robots and for the great time in all our trips abroad. Thank you for helping me with all my hardware and software questions, and in general, for putting up with me. Thank you for trying to put a band together and motivating me to play music again. Special thanks go to Wences and Oliv for all our shared moments and to Lars for co-organizing buffet Fridays and our fruitful discussions. I would also like to thank Dimitris, Vagelis, Psilos, Christoforos and Danilo for all the basketball moments we had so far and their incompetence in guarding me (please don't kick me out of the team). I hope Francisco is proud of us.

I would further like to thank my friends in Greece and the rest of the world. Thank you Pano, Ilia, Tonia, Thanasi, and Faidona for everything we have been through together all these years. Thank you for your support and for being great friends.

To my family, mother and brother, thank you from the bottom of my heart, for shaping my personality and always believing in me. You made me who I am and are always there for me. I am truly proud of you.

Finally, I would like to express my utmost gratitude to my life partner, Stella, for constantly inspiring and motivating me all these years. And putting up with me. Stellou, this thesis could not have been written without your limitless moral support and unconditional love and it is yours as much as it is mine.

*Christos Verginis*  
Stockholm, Sweden  
April 2020.



*To my father*



# Contents

<b>Abstract</b>	<b>3</b>
<b>Sammanfattning</b>	<b>5</b>
<b>Acknowledgements</b>	<b>7</b>
<b>List of Abbreviations</b>	<b>17</b>
<b>List of Symbols</b>	<b>19</b>
<b>List of Figures</b>	<b>21</b>
<b>1 Introduction</b>	<b>29</b>
1.1 Thesis Outline and Contributions . . . . .	32
<b>2 Cooperative Object Manipulation</b>	<b>39</b>
2.1 Introduction . . . . .	40
2.2 Rigid Contacts . . . . .	44
2.3 Rolling Contacts . . . . .	90
2.4 Conclusion . . . . .	114
<b>3 Formation Control and Rigid Cooperative Manipulation</b>	<b>117</b>
3.1 Introduction . . . . .	118
3.2 Formation Control in $\mathbb{SE}(3)$ . . . . .	120
3.3 Cooperative Manipulation via Internal Force Regulation: A Rigidity Theory Approach . . . . .	141
3.4 Conclusion . . . . .	173
<b>4 Continuous Coordination of Multi-Agent Systems</b>	<b>175</b>
4.1 Introduction . . . . .	176
4.2 Adaptive Robot Navigation with Collision Avoidance Subject to 2nd-order Uncertain Dynamics . . . . .	180
4.3 Adaptive Leader-Follower Coordination with Transient Constraints . . . . .	209

4.4	Closed-Form Collision Avoidance of Ellipsoidal Multi-Agent Systems . . . . .	221
4.5	Conclusion . . . . .	231
<b>5</b>	<b>Abstractions of Multi-Agent and Multi-Agent-Object Systems</b> . . . . .	<b>235</b>
5.1	Introduction . . . . .	235
5.2	Decentralized Motion Planning with Collision Avoidance for a Team of UAVs under High Level Goals . . . . .	238
5.3	Robust Decentralized Abstractions for Multiple Mobile Manipulators . . . . .	248
5.4	Timed Abstractions for Distributed Cooperative Manipulation . . . . .	251
5.5	Planning and Control for Multi-Robot-Object Systems under Temporal Logic Formulas . . . . .	272
5.6	Conclusion . . . . .	293
<b>6</b>	<b>Single-Agent Extensions</b> . . . . .	<b>295</b>
6.1	Introduction . . . . .	295
6.2	Reconfigurable Motion Planning and Control in Obstacle Cluttered Environments under Timed Temporal Tasks . . . . .	299
6.3	Sampling-based Motion Planning for Uncertain High-dimensional Systems via Adaptive Control . . . . .	307
6.4	Asymptotic Tracking of Nonsmooth Feedback Stabilizable Unknown Systems with Prescribed Transient Response . . . . .	322
6.5	Conclusion . . . . .	333
<b>7</b>	<b>Summary and Future Research Directions</b> . . . . .	<b>337</b>
7.1	Summary . . . . .	337
7.2	Future Research Directions . . . . .	338
<b>A</b>	<b>Dynamical Systems</b> . . . . .	<b>341</b>
A.1	Lipschitz Continuous Systems . . . . .	341
A.2	Systems with Discontinuous Right-Hand-Side . . . . .	344
A.3	Reduction Principle . . . . .	346
<b>B</b>	<b>Funnel Control</b> . . . . .	<b>349</b>
<b>C</b>	<b>Navigation Functions</b> . . . . .	<b>353</b>
C.1	Multirobot Navigation Functions (MRNFs) . . . . .	353
C.2	Decentralized Navigation Functions (DNFs) . . . . .	354
<b>D</b>	<b>Nonlinear Model Predictive Control</b> . . . . .	<b>357</b>
<b>E</b>	<b>Graph Theory and Rigid Frameworks</b> . . . . .	<b>361</b>

<b>E.1</b>	<b>Graph Theory . . . . .</b>	<b>361</b>
<b>E.2</b>	<b>Rigidity Theory . . . . .</b>	<b>362</b>
<b>F</b>	<b>Temporal Logics-based Task Specification . . . . .</b>	<b>365</b>
<b>F.1</b>	<b>Task Specification in LTL . . . . .</b>	<b>365</b>
<b>F.2</b>	<b>Task Specification in MITL . . . . .</b>	<b>366</b>
<b>G</b>	<b>Useful Properties . . . . .</b>	<b>369</b>



# List of Abbreviations

DNF	Decentralized Navigation Function
DoF	Degree of Freedom
FHOCP	Finite Horizon Optimal Control Problem
KRNF	Koditschek-Rimon navigation function
LTL	Linear Temporal Logic
MILP	Mixed-Integer Linear Programming
MITL	Metric Interval Temporal Logic
MTL	Metric Temporal Logic
MPC	Model Predictive Control
MRNF	Multirobot Navigation Function
NMPC	Nonlinear Model Predictive Control
OCP	Optimal Control Problem
ODE	Ordinary Differential Equation
PPC	Prescribed Performance Control
RoI	Regions of Interest
RPF	Relation Proximity Function
RVF	Relation Verification Function
STL	Signal Temporal Logic
TBA	Timed Büchi Automata
TS	Transition System
TWTL	Time Window Temporal Logic
UAV	Unmanned Aerial Vehicle
WTS	Weighted Transition System



# List of Symbols

$\mathbb{N}$	Set of natural numbers
$\mathbb{Q}$	Set of rational numbers
$\mathbb{R}$	Set of real numbers
$\mathbb{R}_{\geq 0}$	Set of non-negative real numbers
$\mathbb{R}_{> 0}$	Set of positive real numbers
$\mathbb{S}^{n-1}$	Unit sphere in $\mathbb{R}^n$
$\text{SO}(3)$	Special orthogonal group in 3 dimensions
$\text{SE}(3)$	Special Euclidean group in 3 dimensions
$a \times b$	Cross-product between two vectors $a, b \in \mathbb{R}^3$
$S(x) \in \mathbb{R}^{3 \times 3}$	Skew-symmetric matrix of vector $x \in \mathbb{R}^3$ satisfying $S(a)b = a \times b$ , for any vectors $a, b \in \mathbb{R}^3$
$\lambda_{\min}(A) \in \mathbb{R}$	Minimum eigenvalue of a matrix $A \in \mathbb{R}^{n \times n}$
$\lambda_{\max}(A) \in \mathbb{R}$	Maximum eigenvalue of a matrix $A \in \mathbb{R}^{n \times n}$
$\sigma_{\min}(A) \in \mathbb{R}$	Minimum singular value of a matrix $A \in \mathbb{R}^{n \times n}$
$\partial \mathcal{A}$	Boundary of a set $\mathcal{A}$
$\text{Int}(\mathcal{A})$	Interior of a set $\mathcal{A} \subset \mathbb{R}^n$
$\bar{\mathcal{A}}$	Closure of a set $\mathcal{A} \subset \mathbb{R}^n$
$x \succeq y$	Element-wise inequality for vectors $x, y \in \mathbb{R}^n$
$(\alpha_1, \dots, \alpha_n)^\omega$	Infinite sequence created by repeating $\alpha_1, \dots, \alpha_n$
$\mathcal{B}(p, y) \subset \mathbb{R}^n$	An open ball with center $p \in \mathbb{R}^n$ and radius $y \in \mathbb{R}_{> 0}$
$A \oplus B$	Kronecker sum of the matrices $A \in \mathbb{R}^{n \times n}, B \in \mathbb{R}^{m \times m}$
$A \otimes B$	Kronecker product of the matrices $A \in \mathbb{R}^{m \times n} B \in \mathbb{R}^{p \times q}$
$a \cdot b$	Quaternion product of the quaternions $a, b \in \mathbb{S}^3$
$a^+$	Quaternion conjugate of $a \in \mathbb{S}^3$
$\text{tr}(A)$	Trace of a matrix $A \in \mathbb{R}^{n \times n}$
$\det(A)$	Determinant of a matrix $A \in \mathbb{R}^{n \times n}$
$\ A\ _F$	Frobenius norm of matrix $A \in \mathbb{R}^{n \times n}$
$\text{span}(A)$	Span of matrix $A \in \mathbb{R}^{n \times n}$
$\text{rank}(A)$	Rank of matrix $A \in \mathbb{R}^{n \times n}$
$\text{null}(A)$	Nullspace of matrix $A \in \mathbb{R}^{n \times n}$
$\text{adj}(A)$	Adjugate of matrix $A \in \mathbb{R}^{n \times n}$

$A^\dagger$	Moore-Penrose pseudo-inverse of matrix $A \in \mathbb{R}^{n \times n}$
$\dim(\mathbb{A})$	Dimension of vector space $\mathbb{A}$
$\nabla_x f(), \nabla_x^2 f()$	Gradient and Hessian, respectively, of function $f() \in \mathbb{R}$ with respect to $x \in \mathbb{R}^n$
$\ x\ _1 :=  \sum_i^n x_i $	$\ell_1$ norm of vector $x = [x_1, \dots, x_n]^\top \in \mathbb{R}^n$
$\ x\  := \sqrt{x^\top x}$	Euclidean norm of vector $x = [x_1, \dots, x_n]^\top \in \mathbb{R}^n$
$\ A\  := \sqrt{\lambda_{\max}(A^\top A)}$	Induced norm of matrix $A \in \mathbb{R}^{n \times n}$
$\mathbf{1}_n \in \mathbb{R}^n$	The column vector with all entries 1 (subscript often omitted)
$I_n \in \mathbb{R}^{n \times n}$	The unit matrix of dimension $n$
$0_{m \times n} \in \mathbb{R}^{m \times n}$	The $m \times n$ matrix with all entries zeros (subscript often omitted)
$\text{diag}(a_1, \dots, a_n)$	Diagonal (block-diagonal) matrix with scalars (matrices) $a_i$ in the main diagonal (block diagonal)
$\mathbf{e}_h \in \mathbb{R}^3$	Vector of one in the $h \in \{1, 2, 3\}$ element and zeros everywhere else
$\text{sgn} : \mathbb{R} \rightarrow \{-1, 0, 1\}$	The sign function defined by $\text{sgn}(x) = 1$ , if $x > 0$ , $\text{sgn}(x) = -1$ , if $x < 0$ , and $\text{sgn}(x) = 0$ , if $x = 0$
$\text{sgn} : \mathbb{R}^n \rightarrow \{-1, 0, 1\}^n$	$\text{sgn}(x) = [\text{sgn}(x_1), \dots, \text{sgn}(x_n)]^\top$ for $x = [x_1, \dots, x_n]^\top \in \mathbb{R}^n$
$\text{SGN} : \mathbb{R} \rightarrow [-1, 1]$	The set-valued sign function defined by $\text{SGN}(x) = 1$ , if $x > 0$ , $\text{SGN}(x) = -1$ , if $x < 0$ , and $\text{SGN}(x) \in [0, 1]$ , if $x = 0$
$\text{SGN} : \mathbb{R}^n \rightarrow [-1, 1]^n$	$\text{SGN}(x) = [\text{SGN}(x_1), \dots, \text{SGN}(x_n)]^\top$ for $x = [x_1, \dots, x_n]^\top \in \mathbb{R}^n$
$\text{sat} : \mathbb{R} \rightarrow [-1, 1]$	The saturation function $\text{sat}(x) = x$ if $ x  \leq 1$ and $\text{sat}(x) = \frac{x}{ x }$ if $ x  > 1$

# List of Figures

1.1 A humanoid robot moving to an environment consisting of 6 rooms and 3 corridor regions. In room $R_6$ there exists a ball that the robot can grab.	30
2.1 Two robotic agents rigidly grasping an object.	44
2.2 Simulation results for the control scheme of Section 2.2.3: (a): The position errors $e_p(t)$ ; (b): The quaternion errors $e_\varphi(t)$ , $\ e_\varepsilon(t)\ $ ; (c) The velocity errors $e_{v_f}(t)$ , $\forall t \in [0, 40]$ . A zoomed version of the steady-state response has been included in all plots.	57
2.3 The adaptation error norms $\ e_{\vartheta_i}(t)\ $ , $i \in \mathcal{N}$ , $\ e_{\vartheta_O}(t)\ $ (a), $\ e_{d_i}(t)\ $ , $i \in \mathcal{N}$ , $\ e_{d_O}(t)\ $ (b), of the control scheme of Section 2.2.3 $\forall t \in [0, 40]$ .	58
2.4 The agents' joint torques $\tau_i(t)$ , $i \in \mathcal{N}$ , (in (a)-(d), respectively) of the control scheme of Section 2.2.3 $\forall t \in [0, 40]$ , and the motor saturation (with black), which has not been plotted in (a), (b), (d) for better visualization.	59
2.5 Experimental results for the control scheme of Section 2.2.3: (a): The position errors $e_p(t)$ ; (b): The quaternion errors $e_\varphi(t)$ , $e_\varepsilon(t)$ ; (c) The velocity errors $e_{v_f}(t)$ , $\forall t \in [0, 70]$ .	60
2.6 The norms of the adaptation signals $e_{\vartheta_i}(t)$ , $\forall i \in \{1, 2\}$ (left) and $e_{\vartheta_O}(t)$ , (right) $\forall t \in [0, 70]$ of the experiment of the controller in Section 2.2.3.	60
2.7 The agents' joint torques of the experiment of the controller in Section 2.2.3, for $t \in [0, 70]$ , with their respective limits (with black).	61

2.8 Simulation results for the controller of Section 2.2.4 with (in blue) and without (in green) taking into account input constraints; Top: The position errors $e_{s_x}(t)$ , $e_{s_y}(t)$ , $e_{s_z}(t)$ (with blue and green, respectively) along with the respective performance functions (with red); Bottom: The orientation errors $e_{s_\phi}(t)$ , $e_{s_\theta}(t)$ , $e_{s_\psi}(t)$ (with blue and green, respectively) along with the respective performance functions (with red), $\forall t \in [0, 40]$ . Zoomed versions of the transient and steady-state response have been included for all plots. . . . .	70
2.9 The velocity errors $e_v(t)$ along with the respective performance functions (with red) for the controller of Section 2.2.4, $\forall t \in [0, 40]$ . . . . .	70
2.10 The agents' joint torques $\tau_i(t)$ , $i \in \mathcal{N}$ , (in (a)-(d), respectively) of the control scheme of Section 2.2.4 $\forall t \in [0, 40]$ by taking into account input constraints. . . . .	72
2.11 The agents' joint torques $\tau_i(t)$ , $i \in \mathcal{N}$ , (in (a)-(d), respectively) of the control scheme of Section 2.2.4 $\forall t \in [0, 40]$ without taking into account input constraints, $\forall t \in [0, 0.001]$ . . . . .	73
2.12 Experimental results for the controller of Section 2.2.4; Top: the pose errors $e_{s_x}(t)$ , $e_{s_z}(t)$ , $e_{s_\theta}(t)$ (with blue) along with the respective performance functions (with red); Bottom: The velocity errors $e_{v_x}(t)$ , $e_{v_z}(t)$ , $e_{v_\theta}(t)$ (with blue) along with the respective performance functions (with red), $\forall t \in [0, 70]$ . . . . .	74
2.13 The agents' joint torques of the experiment of the controller in Section 2.2.4, $\forall t \in [0, 70]$ , with their respective limits (with black). . . . .	74
2.14 The errors of the object for $t \in [0, 80]$ seconds. . . . .	80
2.15 The errors of robotic agents for $t \in [0, 80]$ seconds. . . . .	81
2.16 The control inputs of the actuators of the robotic agents $u_i(t)$ , $\forall t \in [0, 80]$ seconds. . . . .	81
2.17 The error states of agent 1. . . . .	89
2.18 The error states of agent 2. . . . .	89
2.19 The error states of agent 3. . . . .	90
2.20 The control inputs of agent 1 with $-10 \leq u_{1,j}(t) \leq 10$ , $\forall j \in \{1, \dots, 4\}$ . . . . .	90
2.21 The control inputs of agent 2 with $-10 \leq u_{2,j}(t) \leq 10$ , $\forall j \in \{1, \dots, 4\}$ . . . . .	91
2.22 The control inputs of agent 3 with $-10 \leq u_{3,j}(t) \leq 10$ , $\forall j \in \{1, \dots, 4\}$ . . . . .	91
2.23 A robotic agent in contact with a rigid object via a rolling contact. . . . .	92
2.24 Initial configuration of the system that consists of three mobile manipulators and a rigid object. . . . .	104
2.25 The evolution of the position error, $e_{p_o}(t)$ , $\forall t \in [0, 50]$ . . . . .	105
2.26 The evolution of $e_\varphi(t)$ , $e_\epsilon(t)$ , $\forall t \in [0, 50]$ . . . . .	105

2.27	The evolution of $\hat{\vartheta}_{\mathcal{R}_o}(t)$ , $\hat{\vartheta}_{\mathcal{R}}(t)$ , $\forall t \in [0, 50]$ .	105
2.28	The required friction to prevent slip for the three agents. The black dashed line represents $\mu_f = 0.9$ .	106
2.29	The resulting inputs $\tau(t)$ of the agents, $\forall t \in [0, 50]$ .	106
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 $\varsigma_i > \varsigma_j$ are their sensing ranges.	122
3.2	The distance error signal of the edge (1, 2).	141
3.3	The distance error signal of the edge (2, 3).	141
3.4	The distance error signal of the edge (2, 4).	142
3.5	The orientation error signal of the edge (1, 2).	142
3.6	The orientation error signal of the edge (2, 3).	143
3.7	The orientation error signal of the edge (2, 4).	143
3.8	The distance between the agents.	144
3.9	The control input signals of agent 1.	144
3.10	The control input signals of agent 2.	145
3.11	The control input signals of agent 3.	145
3.12	The control input signals of agent 4.	146
3.13	Illustration of bearing rigidity. The networks in (a) are not bearing rigid because the same inter-neighbor bearings may lead to different geometric patterns of the networks, for example, a square on the left and a rectangle on the right. The networks in (b) are bearing rigid because the same inter-neighbor bearings imply the same geometric pattern though the networks may differ in terms of translation and scale.	147
3.14	Two agents rigidly grasping an object in a 1D scenario.	157
3.15	Four UR5 robotic arms rigidly grasping an object. The red counterpart represents a desired object pose at $t = 0$ .	170
3.16	The error metrics $e_p(t)$ , $e_o(t)$ , $e_v(t)$ , respectively, top to bottom, for the two choices $G_1^*$ and $G_2^*$ and $t \in [0, 15]$ seconds.	171
3.17	The resulting control inputs $\tau_i(t)$ for $G_1^*$ (left) and $G_2^*$ (right), $\forall i \in \{1, \dots, 4\}$ and $t \in [0, 15]$ seconds.	171
3.18	The norm of the internal forces $\ h_{\text{int}}(t)\ $ (as computed via (3.51)) for the two cases of $G^*$ and $t \in [0, 15]$ seconds.	172
3.19	The norms of the resulting control inputs, $\ \tau_i(t)\ $ for $G_1^*$ (with blue) and $G_2^*$ (with red), $\forall i \in \{1, \dots, 4\}$ , and $t \in [0, 15]$ seconds.	172
3.20	The norm of the internal force error $\ e_{\text{int}}(t)\ $ , when using $G_1^*$ and for $t \in [0, 15]$ seconds.	172

4.1 A 2D example of the workspace $\mathcal{W}$ with 50 obstacles $\mathcal{O}_j$ , $j \in \{1, \dots, 50\}$ . The blue asterisks indicate potential initial configurations of the robot and the obstacles have been enlarged with the robot radius $r$ . The red asterisk indicates a potential goal robot position.	182
4.2 A workspace with two star-shaped obstacles. The blue asterisk indicates the center of the robot and the obstacles have been enlarged with the robot radius $r$ through the Minkowski sum. The red asterisk indicates a potential goal robot position.	192
4.3 The resulting trajectories $x(t)$ , $t \in [0, 100]$ seconds, from the initial points $-(5, 5)$ , $(-6, 4.5)$ , and $(3.5, -7)$ to the destination $(5, 5)$ . Left: without any disturbances. Right: with bounded disturbance $d(x, v, t)$ .	205
4.4 Left: The resulting input signals $u(t) = (u_x(t), u_y(t))$ , $t \in [0, 100]$ seconds, for the 2D trajectories of Fig. 4.3. Right: The resulting adaptation signals $\hat{\alpha}(t)$ , $\hat{m}(t)$ , $t \in [0, 100]$ seconds, for the 2D trajectories of Fig. 4.3. The extra subscript $d$ corresponds to the model where a bounded disturbance vector $d(x, v, t)$ was included.	206
4.5 The resulting trajectories $x(t)$ , $t \in [0, 100]$ seconds, from the initial points $-(5, 5)$ , $(-6, 4.5)$ , and $(3.5, -7)$ to the destination $(5, 5)$ .	207
4.6 Left: The resulting input signals $u(t) = (u_x(t), u_y(t), u_z(t))$ , $t \in [0, 100]$ seconds, for the 3D trajectories of Fig. 4.5. Right: The resulting adaptation signals $\hat{\alpha}(t)$ , $\hat{m}(t)$ , $t \in [0, 100]$ seconds, for the 3D trajectories of Fig. 4.5.	207
4.7 Left: The resulting trajectory $x(t)$ , $t \in [0, 500]$ seconds, from the initial points $-(5, 5)$ to the destination $(3, 4)$ , in the 2D star world workspace. Right: The respective trajectory in the transformed sphere world.	208
4.8 The input and adaptations signals $u(t)$ , $\hat{\alpha}(t)$ , $\hat{m}(t)$ , for the 2D (a) and 3D (b) star world workspaces, for $[0, 500]$ and $[0, 200]$ seconds, respectively.	209
4.9 Left: The resulting trajectory $x(t)$ , $t \in [0, 200]$ seconds, from the initial points $-(4, 4, 2)$ to the destination $(1, 2, 2)$ , in the 3D star world workspace. Right: The respective trajectory in the transformed sphere world.	209
4.10 The initial configurations of the multi-agent scenario. The obstacles are depicted as filled red disks whereas the agents as circles. The destinations are shown with asterisk.	210
4.11 The resulting signals $\ x_i(t) - x_{d,i}\ $ , $\forall i \in \mathcal{N}$ , shown to converge to zero for the multi-agent scenario.	210
4.12 The resulting trajectories of the agents $x_i(t)$ in the 2D workspace, $\forall i \in \mathcal{N}, t \in [0, 870]$ seconds, for the multi-agent scenario.	211

4.13 The signal $\beta_{\min}(t)$ , which stays strictly positive, for all $t \in [0, 870]$ , implying that inter-agent collisions and agent-obstacle collisions are avoided.	211
4.14 The initial positions of the 6 UAVs, along with the desired leader goals $x_{d,k}$ , $k \in \{1, \dots, 4\}$ , and the edge set $\mathcal{E}_0$ .	220
4.15 (a): The leader signal $\ s_e(t)\  + \ e_{v_1}(t)\ $ , which converges to zero for every navigation objective; (b) the product $\prod_{k \in \bar{\mathcal{K}}} \frac{1}{\beta_{\epsilon,k}(\nu_k(t))} \prod_{l \in \mathcal{K}_0} \frac{1}{\beta_{n,l}(\nu_l(t))}$ , which remains bounded, proving thus the collision and connectivity properties (the zero values stem from the computer's lower numerical limits); (c) the adaptation signals $\prod_{i \in \{1, \dots, 6\}} \ \hat{\theta}_i(t)\ $ , $\prod_{i \in \{1, \dots, 6\}} \hat{d}_{b_i}(t)$ , $\prod_{i \in \{1, \dots, 6\}} \hat{a}_i(t)$ , which remain bounded, $\forall t \in [0, 277]$ s.	220
4.16 The motion of the multi-agent system as the leader navigates to $x_{d,1}$ (a), ..., $x_{d,4}$ (d). The connectivity of $\mathcal{E}_0$ is also pictured via straight lines.	222
4.17 The resulting control inputs $u_i(t)$ , $i \in \{1, \dots, 6\}$ , $t \in [0, 277]$ s.	223
4.18 Ellipsoid approximation of (a) the rigid links of a robotic manipulator, (b) a mobile robot (top and front view).	223
4.19 The evolution of agent trajectories $\forall t \in [0, 20]$ sec.	232
4.20 Top: The evolution of the minimum of the functions $\min_{k \in \bar{\mathcal{K}}} \{\beta_k(t)\}$ . Bottom: The evolution of the signals $\gamma_i(t)$ and $v_i(t)$ , $\forall i \in \mathcal{N}$ , $\forall t \in [0, 20]$ sec.	233
4.21 The control inputs of the agents $u_i(t)$ , $\forall t \in [0, 20]$ sec, $i \in \mathcal{N}$ .	234
 5.1 Bounding sphere of an aerial vehicle.	238
5.2 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.	243
5.3 The resulting 3-dimensional control signals of the 3 agents for the simulation studies. Top: agent 1, middle: agent 2, bottom: agent 3.	244
5.4 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 $\pi_1, \dots, \pi_4$ .	244
5.5 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.	246

5.6 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): $\pi_2 \rightarrow_1 \pi_4, \pi_4 \rightarrow_2 \pi_3$ , (b), (e): $\pi_4 \rightarrow_1 \pi_3, \pi_3 \rightarrow_2 \pi_2$ , (c), (f): $\pi_3 \rightarrow_1 \pi_2, \pi_2 \rightarrow_2 \pi_4$ .	247
5.7 The resulting 2-dimensional control signals of the 2 agents for the first experimental scenario. Top: agent 1, bottom: agent 2.	247
5.8 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 $\pi_1, \dots, \pi_3$ .	249
5.9 The resulting 2-dimensional control signals of the 2 agents for the second experimental scenario. Top: agent 1, bottom: agent 2.	249
5.10 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): $\pi_1 \rightarrow_1 \pi_2, \pi_2 \rightarrow_2 \pi_1$ , (b), (e): $\pi_2 \rightarrow_1 \pi_3, \pi_1 \rightarrow_2 \pi_2$ , (c), (f): $\pi_3 \rightarrow_1 \pi_2, \pi_2 \rightarrow_2 \pi_1$ .	250
5.11 An agent that consists of $\ell_i = 3$ rigid links.	250
5.12 (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 $\pi_1$ to $\pi_2$ , agent 2 from $\pi_2$ to $\pi_1$ , and agent 3 from $\pi_1$ to $\pi_3$ . (c): The second transition of the agents in the workspace. Agent 1 transits from $\pi_2$ to $\pi_1$ , agent 2 from $\pi_1$ to $\pi_2$ , and agent 3 from $\pi_3$ to $\pi_2$ .	259
5.13 The obstacle functions $\beta_i, i \in \{1, 2, 3\}$ , which remain strictly positive.	260
5.14 The resulting control inputs $\tau_i, \forall i \in \{1, 2, 3\}$ for the two transitions.	261
5.15 The parameter deviations $\tilde{\alpha}_i, \forall i \in \{1, 2, 3\}$ , which are shown to be bounded.	261
5.16 An example of the system shown in Fig. 2.1 in the configuration that produces $\hat{L}$ .	263
5.17 The workspace partition according to the bounding box of the coupled system.	263
5.18 Top view of a transition between two adjacent regions $\pi_j$ and $\pi_{j'}$ . Since $p_o \in \mathcal{B}(p_{j,j'}(t), l_0)$ , we conclude that $\mathcal{S}_q \subset \mathcal{B}(p_o, \hat{L}) \subset \mathcal{B}(p_{j,j'}(t), l_0 + \hat{L}) \subset \pi_j \cup \pi_{j'}$ .	265
5.19 The aerial robots employed in the simulation rigidly grasping an object.	270

5.20 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.	270
5.21 (a): The overall desired object trajectory (with red), the actual object trajectory (with black), the domain specified by $\mathcal{B}(p_{r_j, r_{j'}}(t), l_0)$ , $\forall j \in \{1, \dots, 10\}$ (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.	273
5.22 The pose errors $e_s(t)$ (with blue) along with the performance functions $\rho_s(t)$ (with red).	274
5.23 The velocity errors $e_v(t)$ (with blue) along with the performance functions $\rho_v(t)$ (with red).	275
5.24 The resulting control inputs $\tau_i = [f_{B_i}^\top, \mu_{B_i}^\top, \tau_{\alpha_{i,1}}, \tau_{\alpha_{i,2}}]$ for $i = 1$ and $i = 2$ ; $f_{B_i}, \mu_{B_i}, \tau_{\alpha_i}$ are the quadrotor base forces and torques and the manipulator torque commands, respectively.	276
5.25 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.	289
5.26 The transition $\pi_{s,1} \rightarrow_s \pi_{s,2}$ (a), that corresponds to the navigation of the agents $\pi_1 \rightarrow_1 \pi_2, \pi_3 \rightarrow_2 \pi_1, \pi_4 \rightarrow_3 \pi_1$ .	292
5.27 The transition $\pi_{s,3} \rightarrow_s \pi_{s,4}$ (b), that corresponds to the transportation $\pi_1 \xrightarrow[T]{\{2,3\}} \pi_3$ .	293
 6.1 Workspace overview. The red discs correspond to the three regions of interest. The plotted paths are the resulting trajectories from the first out of the five executions of the suffix.	308
6.2 (top) The transition times calculated before each transition; filled marks correspond to actually used transition times. (bottom) The resulting timed run of the transition system.	309
6.3 A UR5 robotic arm in an obstacle-cluttered environment with 4 targets.	319
6.4 Box plots showing the execution time (top) and the nodes (bottom) created of the three algorithms (in logarithmic scale) for the four paths (organized in two groups of two (left and right)); ‘+’ indicate the outliers.	320
6.5 Box plots showing the execution time (a) and number of nodes (b) created for the kinodynamic RRT in logarithmic scale (for the first two joints and the path $(0, 0) \rightarrow (-\frac{\pi}{18}, \frac{\pi}{4})$ ).	321

6.6 The error values $e_{r_j}(t) = e_{r_j}(t) - q_{d,r_j}^i(t)$ for the adaptive controller (a) and the PID one (b). . . . .	321
6.7 The evolution of the errors $e_p(t)$ (top), $e_v(t)$ (bottom), depicted with blue, along with the performance functions $\rho_p(t)$ , $\rho_v(t)$ , depicted with red, $\forall t \in [0, 60]$ sec. . . . .	333
6.8 The evolution of the transformed errors $\varepsilon_p(t)$ , $\varepsilon_v(t)$ , $\forall t \in [0, 60]$ sec. . . . .	334
6.9 The evolution of the control inputs $u(t) = [u_1(t), u_2(t)]^\top$ , $\forall t \in [0, 60]$ sec. . . . .	334
6.10 The evolution of the adaptation signal $\hat{d}(t)$ , $\forall t \in [0, 60]$ sec. . . . .	335
B.1 Illustration of funnel control, where the error $e(t)$ is confined in the prescribed funnel defined by the functions $\rho_L(t)$ , $\rho_U(t)$ . . . . .	351

# Chapter 1

## Introduction

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* [? ? ? ? ? ], in which all the agents are required to converge to a specific point, and *formation* [? ? ], 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* [? ? ? ] and/or *collision avoidance* [? ].

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.

Another topic that has troubled researchers the last decades is the control



**Figure 1.1:** A humanoid robot moving to an environment consisting of 6 rooms and 3 corridor regions. In room  $R_6$  there exists a ball that the robot can grab.

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 state? 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.

The aforementioned problems become even more challenging when one takes into account system uncertainty. The dynamic model of real robotic systems cannot be accurately known by the user/designer, since it includes terms that might not be easy to identify, e.g., dynamic parameters (mass, inertia), friction, and other external disturbances. This becomes more apparent as the complexity of the considered systems increases (consider, e.g., a mobile robot vs a 6-DoF robotic manipulator). These uncertainties are expected to affect the performance of the system, and since they cannot be accurately canceled by the control design, the latter must render the closed-loop system robust to them [? ].

Motivated by the above discussion, this thesis aims at solving the problem of decentralized motion- and task-planning of uncertain 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 robust 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 first the problem of cooperative object manipulation. We design control protocols for the centralized and decentralized cooperative manipulation of an object grasped by multiple robotic agents by means of rigid as well as rolling contacts, possibly subject to model uncertainties. Moreover, we study the relation of rigid cooperative manipulation with rigid formation control, and design a novel control algorithm for the latter problem. Secondly, again in view of transition design for multi-agent systems, we develop numerous control protocols for the coordination of multi-agent systems, including multi-agent navigation and leader-follower coordination, subject to collision and connectivity constraints as well as model uncertainties. The third part draws from the previous ones to design well defined discrete abstractions for multi-agent and multi-agent-objects systems. In that way, we allow the expression of complex desired tasks as temporal logic specifications, for which we provide

controller synthesis. Finally, we study some problems for single-agent setups, including timed temporal specifications in an obstacle-cluttered environment, integration of intelligent control protocols with sampling-based motion planning algorithms for complex uncertain systems, as well as asymptotic stability properties with funnel constraint satisfaction.

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 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.

## 1.1 Thesis Outline and Contributions

In this Section, we provide the outline of the thesis and indicate the contributions of each chapter. The thesis is divided into *four main parts*, the first three of which aiming to solve the aforementioned problems, and the final one discussing single-agent extensions.

- The *first part* consists of Chapters 2 and 3. In this part, we first tackle the problem of cooperative manipulation of an object grasped by several robotic agents. We propose a variety of control algorithms, combining centralized and decentralized setups, rigid and rolling contacts, adaptive and Model Predictive control techniques, as well as incorporation of collision avoidance techniques with workspace obstacles. Next, motivated by its application to cooperative manipulation, we propose a novel control algorithm for the formation stabilization of a multi-agent team. Moreover, we explicitly study the relation of rigid formation control with robotic cooperative manipulation schemes through rigid contacts.
- The *second part* consists of Chapter 4. In this part, we develop continuous control algorithms for multi-agent coordination under model uncertainties. More specifically, we propose first a novel adaptive control protocol for the single- and multi-agent collision-free navigation in an obstacle-cluttered workspace subject to uncertain dynamics and spherical shapes. Next, we tackle the leader-follower coordination problem. We develop an adaptive control algorithm for the leader navigation to a predefined goal while guaranteeing inter-agent collision avoidance and connectivity maintenance. Finally, motivated by real robotic structures, we present a closed form control protocol that achieves collision avoidance among *ellipsoidal* agents, while compensating at the same time for the uncertain dynamics.

- The *third part* consists of Chapter 5. In this chapter, we design well-defined abstractions for multi-agent and multi-agent-object systems in discretized workspaces. This allows us to define complex tasks as temporal logic formulas and employ formal verification methodologies to synthesize control protocols. The discretized abstractions include both fully partitioned workspaces as well as discretization based on predefined regions of the workspace. We use control methodologies from the previous chapters as well as newly designed ones.
- The *fourth and final part*, consisting of Chapter 6, considers some challenging extensions for single-agent systems. Firstly, we tackle the problem of the optimal motion planning under *timed* temporal logic specifications in an obstacle-cluttered environment. We use previous results on collision-free *timed* navigation and we develop a novel reconfigurable framework that guarantees asymptotically optimal behavior. Secondly, we integrate adaptive control techniques with sampling-based algorithms for the motion planning problem of complex high-dimensional systems. We propose a two-layer approach that compensates for the system uncertainties and guarantees the collision-free navigation to the goal via a geometric path in an extended free space. Finally, we develop a novel control algorithm that guarantees asymptotic stability of a general class of uncertain systems subject to funnel constraints.

## Chapter 2

This chapter addresses the problem of cooperative manipulation of a single object by multiple robotic agents. We present first four control algorithm for the case of *rigid* contact grasps. The first two are decentralized, adaptive closed-form techniques that aim to guarantee trajectory tracking by the object's center of mass while compensating for model uncertainties and external disturbances, and imposing predefined performance on the closed-loop system. Next, we design a centralized and a decentralized control protocol using the Nonlinear Model Predictive Control methodology, which guarantee object transportation to a desired pose, while complying with other constraints, such as obstacle avoidance and input saturation. Finally, we consider the case of rolling contacts. We design a centralized control protocol that guarantees object trajectory tracking, robust to model uncertainties and center of mass location. Moreover, we propose a novel algorithm for contact maintenance of the agents with the object. By employing event-triggered agent communication, we extend the latter scheme to a decentralized version. It is noteworthy that none of the aforementioned approaches relies on

force/torque sensor information. The covered material is based on the following contributions [? ? ? ? ? ?]:

- C. K. Verginis, M. Mastellaro and D. V. Dimarogonas, “Robust quaternion-based cooperative manipulation without force/torque information”, IFAC-PapersOnLine, 50(1), pp. 1754-1759, Toulouse, France, 2017.
- 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, vol. 28, no. 3, pp. 713-729, 2020.
- A. Nikou, C. K. Verginis and D. V. Dimarogonas, “A nonlinear model predictive control scheme for cooperative manipulation with singularity and collision avoidance”, IEEE Mediterranean Conference on Control and Automation (MED), pp. 707-712, Valletta, Malta, 2017.
- C. K. Verginis, A. Nikou and D. V. Dimarogonas, “Communication-based decentralized cooperative object transportation using nonlinear model predictive control”, IEEE European Control Conference (ECC), pp. 733-738, Limassol, Cyprus, 2018.
- C. K. Verginis, W. S. Cortez and D. V. Dimarogonas, “Adaptive Cooperative Manipulation with Rolling Contacts”, to appear in the American Control Conference (ACC), Denver, Colorado, USA, 2020.
- C. K. Verginis, W. S. Cortez and D. V. Dimarogonas, “Decentralized adaptive Cooperative Manipulation with Rolling Contacts”, under preparation.

## Chapter 3

This chapter presents first a novel control protocol for the formation control of tree graphs in  $\mathbb{SE}(3)$ . The control laws are decentralized 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 and certain predefined functions characterize the transient and steady-state performance of the closed loop system. Next, we study the relation between rigid cooperative manipulation and rigid formations. By doing so, we provide novel conditions for the internal force-free cooperative manipulation based on the rigidity matrix of the underlying multi-agent system. The covered material is based on the following contributions [? ? ? ]:

- C. K. Verginis, A. Nikou and D. V. Dimarogonas, “Robust formation control in  $\mathbb{SE}(3)$  for tree-graph structures with prescribed transient and steady state performance”, *Automatica* 103 (2019): 538-548.
- C. K. Verginis and D. V. Dimarogonas: “Energy-Optimal Cooperative Manipulation via Provable Internal-Force Regulation”, to appear in the IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 2020.
- C. K. Verginis, D. Zelazo, and D. V. Dimarogonas: “Cooperative Manipulation via Internal Force Regulation: A Rigidity Theory Perspective”, Under Review. Arxiv Link: <https://arxiv.org/pdf/1911.01297.pdf>

## Chapter 4

This chapter tackles the problem of multi-agent coordination in the following ways. Firstly, we consider the problem of single- and multi-agent navigation in an obstacle-cluttered spherical environment under 2nd-order uncertain dynamics. We propose an adaptive control scheme that guarantees the single-agent collision-free navigation to the goal from almost all initial conditions while compensating for the uncertain dynamics, which is then extended it to a decentralized priority-based multi-agent case. Secondly, we consider the leader-following coordination problem in the following sense. A leader agent aims at navigating to a pre-specified pose, while the entire team has to avoid collision with each other, as well as maintain connectivity. We develop a decentralized adaptive control protocol, compensating again for dynamic uncertainties, to guarantee accomplishment of the aforementioned specifications. The algorithms above consider spherical agents, which might be a conservative over-simplification when it comes to real robots. Therefore, we finally develop an adaptive control methodology that guarantees collision avoidance among *ellipsoidal* agents. We propose a novel closed-form function that encodes collisions among 3D ellipsoids and combine it with an adaptive control law that compensates for the model uncertainties. The covered material is based on the following contributions [? ? ? ]:

- C. K. Verginis and D. V. Dimarogonas: “Adaptive Robot Navigation with Collision Avoidance Subject to 2nd-order Uncertain Dynamics”, Under Review.
- C. K. Verginis and D. V. Dimarogonas, “Adaptive Leader-Follower Coordination of Lagrangian Multi-AgentSystems under Transient Constraints”, IEEE Conference on Decision and Control (CDC), pp. 3833-3838, Nice, France, 2019.

- C. K. Verginis and D. V. Dimarogonas, “Closed-Form Barrier Functions for Multi-Agent Ellipsoidal Systems with Uncertain Lagrangian Dynamics”, IEEE Control System Letters, pp. 727-732, 2019.

## Chapter 5

This chapter addresses the motion planning problem for multi-agent and multi-agent-object systems under high level complex tasks expressed as temporal logic formulas. We first focus on local temporal logic specifications for each agent individually. We use previous results to derive well-defined discrete abstractions based on pre-defined regions of interest in the workspace, possibly by accounting for collision and connectivity constraints. We use then standard formal verification techniques to derive paths that satisfy the independent tasks. Next, apart from the agents, we consider that unactuated objects have to satisfy certain temporal logic tasks. The robotic agents are now responsible for satisfying the objects’ tasks, except for their own. We use again previous results to derive discrete abstractions of the coupled system’s motion, based on both regions of interest as well as a complete workspace partition. We then apply the same formal verification-based strategy to obtain discrete paths that satisfy the agents’ and the object’s goals. These results are based on [? ? ? ? ? ? ?]:

- C. K. Verginis, Z. Xu and D. V. Dimarogonas, “Decentralized motion planning with collision avoidance for a team of UAVs under high level goals”, IEEE International Conference on Robotics and Automation (ICRA), pp. 781-787, Singapore, 2017.
- C. K. Verginis and D. V. Dimarogonas, “Robust decentralized abstractions for multiple mobile manipulators”, IEEE Conference on Decision and Control (CDC), pp. 2222-2227, Melbourne, Australia, 2017.
- C. K. Verginis and D. V. Dimarogonas, “Distributed Cooperative Manipulation under Timed Temporal Specifications”, American Control Conference (ACC), pp. 1358-1363, Seattle, USA, 2017.
- C. K. Verginis and D. V. Dimarogonas, “Timed abstractions for distributed cooperative manipulation”, Autonomous Robots, 42, no. 4 (2018): 781-799.
- C. K. Verginis and D. V. Dimarogonas, “Multi-agent motion planning and object transportation under high level goals”, IFAC-PapersOnLine, 50(1), pp. 15816-15821, Toulouse, France, 2017.
- C. K. Verginis, and D. V. Dimarogonas, “Motion and cooperative transportation planning for multi-agent systems under temporal logic

formulas”, BOSCH AI Conference, 2018.  
Arxiv Link: <https://arxiv.org/pdf/1803.01579.pdf>.

## Chapter 6

This chapter addresses some challenging extensions for single-agent setups. Firstly, we consider the problem of motion planning under timed temporal tasks for a mobile robot in an obstacle-cluttered environment. We use previous results in collision-free timed navigation and develop a novel timed automata-based reconfiguration algorithm that achieves the satisfaction of the task in an asymptotically energy-optimal way. Secondly, we consider the motion planning problem for complex high-dimensional systems (e.g., robotic manipulators) with uncertain dynamics in obstacle-cluttered environments. We integrate in an innovative way sampling-based motion planning algorithms and adaptive control to provide a two layer framework that guarantees the safe navigation of the robot to its goal, while compensating for its uncertain dynamics. Finally, we consider the tracking problem for a class of uncertain nonlinear systems under funnel constraints. We develop a novel adaptive control protocol that achieves *asymptotic* tracking while complying to the funnel specifications and without using any model information. These results are based on [? ? ? ?]:

- C. K. Verginis, K. Vrohidis, C. P. Bechlioulis, K. J. Kyriakopoulos, and D. V. Dimarogonas, “Reconfigurable Motion Planning and Control in Obstacle Cluttered Environments under Timed Temporal Tasks”, IEEE International Conference on Robotics and Automation (ICRA), pp. 951-957, Montreal, Canada, 2019.
- C. K. Verginis, D. V. Dimarogonas, and L. E. Kavraki, “Sampling-based Motion Planning for Uncertain High-dimensional Systems via Adaptive Control”, submitted to the Workshop on the Algorithmic Foundations of Robotics (WAFR), Oulu, Finland, 2020.
- C. K. Verginis and D. V. Dimarogonas, “Asymptotic Stability of Uncertain Lagrangian Systems with Prescribed Transient Response”, IEEE Conference on Decision and Control, pp. 7037-7042, Nice, France, 2019.
- C. K. Verginis and D. V. Dimarogonas, “Asymptotic Tracking of Second-order Nonsmooth Feedback Stabilizable Unknown Systems with Prescribed Transient Response”, under Review.

Finally, in Chapter 7 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 [? ? ? ? ? ? ?]:

- A. Nikou, C. K. Verginis and D. V. Dimarogonas, “Robust distance-based formation control of multiple rigid bodies with orientation alignment”, IFAC-PapersOnLine, 50(1), pp. 15458-15463, Toulouse, France, 2017.
- 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”, IEEE International Conference on Decision and Control (CDC), pp. 411-416, 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), pp. 2997-3002, Melbourne, Australia, 2017.
- A. Nikou, C. K. Verginis, S. Heshmati-alamdar 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), pp. 990-995, Melbourne, Australia, 2017.
- J. Wei, C. K. Verginis, J. Wu, D. V. Dimarogonas, H. Sandberg, and K. H. Johansson, “Asymptotic and Finite-Time Almost Global Attitude Tracking: Representations Free Approach”, European Control Conference (ECC), pp. 3126-3131, Limassol, Cyprus, 2018.
- T. Pan, C. K. Verginis, A. M. Wells, D. V. Dimarogonas, and L. E. Kavraki: “Augmenting Control Policies with Motion Planning for Robust and Safe Multi-robot Navigation”, submitted to the IEEE International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 2020.
- N. Lissandrini, C. K. Verginis, P. Roque, A. Cenedese, and D. V. Dimarogonas: “Decentralized Nonlinear MPC for Robust Cooperative Manipulation by Heterogeneous Aerial-Ground Robots”, submitted to the IEEE International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 2020.

## Chapter 2

# Cooperative Object Manipulation

As mentioned in the previous chapter, cooperative manipulation of objects by autonomous robotic agents is of paramount importance in creating discrete representations of multi-object-robot systems as well as autonomizing item transportation tasks.

This chapter addresses the problem of cooperative manipulation of a single object by multiple robotic agents. We consider first the case where the agents grasp an object by means of *rigid contacts*, and we present four novel control methodologies for the trajectory tracking by the object's center of mass, without the need for force/torque feedback at the grasping points. 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. Load sharing coefficients are also introduced to account for potential differences in the agents' power capabilities. Thirdly, we turn to optimization techniques and use Nonlinear Model Predictive Control (NMPC) to guarantee convergence of the object's center of mass to a *fixed* pose, both in a centralized and a communication-based decentralized framework. These approaches also guarantee collision avoidance properties among the robotic agents and potential workspace obstacles as well as avoidance of kinematic/representation singularities.

Secondly, we consider the problem of object manipulation by means of *rolling contacts*. We present a centralized control algorithm that achieves trajectory tracking by the object as well as a decentralized extension using event-triggered communication, still without using force/torque feedback. Both schemes employ adaptive control ideas to compensate for potential uncertainties in the agents' and the object's dynamic parameters and do not use information regarding the object's center of mass, since the tracking concerns an observable point on the object. Contact slip avoidance is also

guaranteed by novel optimization algorithms. Simulation and experimental results support the theoretical findings.

## 2.1 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 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, [? ? ? ? ? ? ? ? ? ? ]). Impedance and hybrid force/position control is the most common methodology used in the related literature [? ], 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., [? ]), which can render the use of force/torque sensors unnecessary. Force/Torque sensor-free methodologies can be found in [? ? ? ], which have inspired the dynamic modeling in this work. Moreover, [? ] uses an external force estimator, without employing force sensors, [? ] presents a force sensor-free control protocol with gain tuning, and [? ] considers the object regulation problem without force/torque feedback. Finally, force/torque sensor-free methodologies are developed in [? ], where the robot dynamics are not taken into account, and in [? ], 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 repre-

sentation 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 [? ? ?] as well as in [?], 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 [?] with a gain tuning scheme, in [?], where the object regulation problem is considered, and in [?], [?]. An estimation of parameters is included in [? ?], whereas [?] and [?] employ fuzzy mechanisms to compensate for model uncertainties. In [? ?] the authors develop a task-oriented adaptive control protocol using observers. Kinematic uncertainties and joint limits are handled in [?], [?], and [?], respectively.

An internal force and load distribution analysis is performed in [?]; [?] employs a leader-follower scheme, and [?] develops a decentralized force consensus algorithm. Furthermore, [?] introduces hybrid modeling of cooperative manipulation schemes and [?] includes intermittent contact; [?] proposes a kinematic-based multi-robot manipulation scheme, and [? ?] address the problem from a formation-control point of view. In [?] a navigation function-based approach is used, and object manipulation by aerial robots is considered in [? ? ?].

Another interesting direction regarding cooperative manipulation is the *safe* transportation of an object in an obstacle-cluttered environment. In standard manipulation tasks, 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 [?] 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.

Other important properties that concern robotic manipulators are the 1) input saturation constraints, naturally characterizing real actuators, and 2) 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 agents cannot move, must be always avoided, especially when dealing with task-space control in the end-effector [? ]. As already mentioned before, *representation* singularities can also occur in the mapping from coordinate rates to angular velocities of a rigid body. Typical control schemes cannot guarantee satisfaction of a task while provably avoiding input saturations or singularities.

The aforementioned properties can be considered as an instance of constrained-based control, which 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) [? ], where a constrained optimization problem is solved for a finite horizon in the future, providing a prediction of the state evolution. 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. [? ? ? ? ? ? ? ? ? ? ]), which we use in this chapter for the problem of the constraint cooperative object manipulation.

All the aforementioned approaches rely on the assumption that each robotic agent is *rigidly* attached to the object, allowing it to apply any force/torque at the contact point. This rigidity assumption is highly restrictive as it only applies to objects on which a rigid grasp can be formed, excluding, e.g., objects with smooth surfaces or large boxes/spheres (e.g., packages), which cannot be rigidly grasped by a simple gripper. Non-rigid/rolling contacts, on the other hand, increase the number of objects that can be grasped, increase the workspace of the system, and allow for modular manipulation scenarios in which robots can be swapped in/out to adjust the grasp online. Note that, by employing rolling contacts, the cooperative manipulation problem becomes similar to robotic grasping [? ] albeit with moving “fingers.”

Rolling contacts complicate the problem as each contact may only apply a force that respects friction cone constraints to prevent slip, instead of an arbitrary wrench associated with rigid contacts [? ]. Early robotic grasping approaches required exact knowledge of the agent’s dynamics [? ?

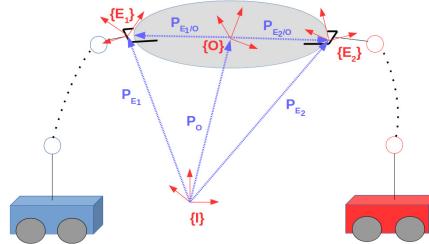
]. Other recent techniques are robust to model uncertainties, but neglect rolling effects or dynamics [? ? ? ], while other more sensor-deprived approaches assume the object is weightless [? ? ]. The approach from [? ] assumes a priori bounded states, which does not apply to mobile manipulators that can be potentially considered. Adaptive control schemes that have also been developed require force and contact location sensing, and assume boundedness of the uncertain parameter estimates [? ? ], or are limited to set-point (constant reference) manipulation [? ].

Furthermore, for collaborative manipulation using rolling contacts, it is critical to ensure the object does not slip. This is neglected by most of the aforementioned approaches, which assume either rigid grasps or simply no slip without guarantees. Methods of ensuring slip prevention are developed typically by solving an optimization problem online [? ? ? ]. However, [? ? ] neglect the dynamics of the system, which may perturb the system and cause slip. The approach in [? ] uses a conservative bound on the dynamics, which overcompensates the amount of force required to hold the object. Finally, most related works consider accurate knowledge of the object center of mass, which can be difficult to obtain in practice, especially in cases of complicated object shapes.

The contribution of this chapter consists of the following: Firstly, we introduce two novel close-form 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. In particular, we develop first 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 that avoids undesired representation singularities. Then, we propose a decentralized model-free control scheme that guarantees *predefined* transient and steady-state performance for the object's center of mass.

Secondly, we use NMPC to 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.

Thirdly, we propose an adaptive control protocol for the trajectory tracking by an observable point on an object that is manipulated by  $N$  robotic agents in terms of *rolling contacts*, also without using force/torque measurements at the grasping points. We develop a centralized as well as a decentralized event-triggered communication-based control scheme. Both schemes include the adaptive and quaternion modeling attributes of the rigid grasp schemes, and are robust to uncertainties of the object's center of mass pose, since the tracking concerns an observable a priori selected point



**Figure 2.1:** Two robotic agents rigidly grasping an object.

on the object. Novel algorithms that guarantee contact slip avoidance are also developed. We provide detailed stability analyses for all the proposed schemes, whose validity is verified by using simulation and experimental results.

## 2.2 Rigid Contacts

Consider  $N$  fully actuated robotic agents rigidly grasping an object (see Fig. 2.1). 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.

### 2.2.1 System Model

We derive in this section the model of the system object-robots.

We denote by  $q_i, \dot{q}_i \in \mathbb{R}^{n_i}$ , with  $n_i \in \mathbb{N}, \forall i \in \mathcal{N} := \{1, \dots, N\}$ , the generalized joint-space variables and their time derivatives of agent  $i$ , with  $q_i := [q_{i1}, \dots, q_{in_i}]$ . 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 orientation of the  $i$ th end-effector, denoted by  $p_{E_i}$  and  $\eta_{E_i}$ , respectively, can be derived by the forward kinematics and are smooth functions of  $q_i$ , i.e.  $p_{E_i} := p_{E_i}(q_i) : \mathbb{R}^{n_i} \rightarrow \mathbb{R}^3$ ,  $\eta_{E_i} := \eta_{E_i}(q_i) : \mathbb{R}^{n_i} \rightarrow \mathbb{T}$ , where  $\mathbb{T}$  is an appropriate orientation space. The differential equation describing the dynamics of each agent is [?]:

$$B_i \ddot{q}_i + C_{q_i} \dot{q}_i + g_{q_i} + d_{q_i} = \tau_i - J_i^\top h_i, \quad (2.1)$$

where  $B_i := B_i(q_i) : \mathbb{R}^{n_i} \rightarrow \mathbb{R}^{n_i \times n_i}$  is the positive definite inertia matrix,  $C_{q_i} := C_{q_i}(q_i, \dot{q}_i) : \mathbb{R}^{2n_i} \rightarrow \mathbb{R}^{n_i \times n_i}$  is the Coriolis matrix,  $g_{q_i} := g_{q_i}(q_i) : \mathbb{R}^{n_i} \rightarrow \mathbb{R}^{n_i}$  is the joint-space gravity term,  $d_{q_i} := d_{q_i}(q_i, \dot{q}_i, t) : \mathbb{R}^{2n_i} \times \mathbb{R}_{\geq 0} \rightarrow$

$\mathbb{R}^{n_i}$  is a bounded vector representing unmodeled friction, uncertainties and external disturbances,  $h_i \in \mathbb{R}^6$  is the vector of generalized forces that agent  $i$  exerts on the grasping point with the object and  $\tau_i = [\tau_{i,1}, \dots, \tau_{i,n_i}]^\top \in \mathbb{R}^{n_i}$  is the vector of joint torques, acting as control inputs,  $\forall i \in \mathcal{N}$ .

The generalized velocity of each agent's end-effector  $v_i := [\dot{p}_{E_i}^\top, \omega_{E_i}^\top]^\top \in \mathbb{R}^6$ , where  $\omega_{E_i} \in \mathbb{R}^3$  is the respective angular velocity, can be considered as a transformed state through the differential kinematics  $v_i = J_i \dot{q}_i$  [?], where  $J_i := J_i(q_i) : \mathbb{R}^{n_i} \rightarrow \mathbb{R}^{6 \times n_i}$  is a smooth function representing the geometric Jacobian matrix,  $\forall i \in \mathcal{N}$  [?]. The latter leads also to

$$\dot{v}_i = J_i \ddot{q}_i + \dot{J}_i \dot{q}_i. \quad (2.2)$$

We define also the sets  $S_i := \{q_i \in \mathbb{R}^{n_i} : \det(J_i(q_i)J_i(q_i)^\top) > 0\}$ , which contains all the singularity-free configurations. By employing the differential kinematics as well as (2.2), we obtain from (2.1) the transformed task space dynamics [?]:

$$M_i \ddot{v}_i + C_i v_i + g_i + d_i = u_i - h_i, \quad (2.3)$$

with the corresponding task space terms  $M_i := M_i(q_i) : S_i \rightarrow \mathbb{R}^{6 \times 6}$ ,  $C_i := C_i(q_i, \dot{q}_i) : S_i \times \mathbb{R}^{n_i} \rightarrow \mathbb{R}^{6 \times 6}$ ,  $g_i := g_i(q_i) : S_i \rightarrow \mathbb{R}^6$ ,  $d_i := d_i(q_i, \dot{q}_i, t) : S_i \times \mathbb{R}^{n_i} \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^6$  and  $u_i = [u_{i,1}, \dots, u_{i,6}]^\top \in \mathbb{R}^6$  being the task space wrench, related to  $\tau_i$  via  $\tau_i = J_i^\top u_i + (I_{n_i} - J_i^\top \tilde{J}_i^\top) \tau_{i0}$ , where  $\tilde{J}_i$  is a generalized inverse of  $J_i$  [?];  $\tau_{i0}$  concerns redundant agents ( $n_i > 6$ ) and does not contribute to end-effector forces.

The agent task-space dynamics (2.3) can be written in vector form as:

$$M \ddot{v} + C v + g + d = u - h, \quad (2.4)$$

where  $v := [v_1^\top, \dots, v_N^\top]^\top \in \mathbb{R}^{6N}$ ,  $M := M(q) := \text{diag}\{[M_i]_{i \in \mathcal{N}}\} \in \mathbb{R}^{6N \times 6N}$ ,  $C := C(q, \dot{q}) := \text{diag}\{[C_i]_{i \in \mathcal{N}}\} \in \mathbb{R}^{6N \times 6N}$ ,  $h := [h_1^\top, \dots, h_N^\top]^\top$ ,  $u := [u_1^\top, \dots, u_N^\top]^\top$ ,  $g := g(q) := [g_1^\top, \dots, g_N^\top]^\top$ ,  $d := d(q, \dot{q}, t) := [d_1^\top, \dots, d_N^\top]^\top \in \mathbb{R}^{6N}$ .

Regarding the object, we denote by  $x_o := [p_o^\top, \eta_o^\top]^\top \in \mathbb{M} := \mathbb{R}^3 \times \mathbb{T}$ ,  $v_o := [\dot{p}_o^\top, \omega_o^\top]^\top \in \mathbb{R}^6$  the pose and generalized velocity of its center of mass;  $\eta_o$  here denotes explicitly Euler angles  $\eta_o := [\phi_o, \theta_o, \psi_o]^\top \in \mathbb{T} = \mathbb{R}^3$ . We consider the following second order dynamics, which can be derived based on the Newton-Euler formulation:

$$\dot{x}_o = J_o v_o, \quad (2.5a)$$

$$M_o \ddot{v}_o + C_o v_o + g_o + d_o = h_o, \quad (2.5b)$$

where  $M_o := M_o(\eta_o) : \mathbb{T} \rightarrow \mathbb{R}^{6 \times 6}$  is the positive definite inertia matrix,  $C_o := C_o(\eta_o, \omega_o) : \mathbb{T} \times \mathbb{R}^6 \rightarrow \mathbb{R}^{6 \times 6}$  is the Coriolis matrix,  $g_o \in \mathbb{R}^6$  is the gravity vector,  $d_o := d_o(x_o, \dot{x}_o, t) : \mathbb{M} \times \mathbb{R}^6 \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^6$  a bounded vector representing modeling uncertainties and external disturbances, and  $h_o \in \mathbb{R}^6$

is the vector of generalized forces acting on the object's center of mass. Moreover,  $J_O := J_O(\eta_O) : \mathbb{T} \rightarrow \mathbb{R}^{6 \times 6}$  is the object representation Jacobian  $J_O(\eta_O) := \text{diag}\{I_3, J_{O_\eta}\}$ , where  $J_{O_\eta} := J_{O_\eta}(\eta_O) : \mathbb{T} \rightarrow \mathbb{R}^{3 \times 3}$ :

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

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

$$\|J_O(\eta_O)\| = \sqrt{\frac{|\sin(\theta_O)|+1}{1-\sin^2(\theta_O)}}, \quad (2.6a)$$

$$\|J_O(\eta_O)^{-1}\| = \sqrt{1 + \sin(\theta_O)} \leq \sqrt{2}, \quad (2.6b)$$

$\forall \eta_O \in \mathbb{T}$ . We also denote by  $R_O := R_O(\eta_O) : \mathbb{T} \rightarrow \mathbb{SO}(3)$  the object's rotation matrix.

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  $\eta_O$  can be transformed to the unit quaternion  $\zeta_O = [\varphi_O, \epsilon_O^\top]^\top \in \mathbb{S}^3$ , where  $\varphi_O \in [-1, 1]$  and  $\epsilon_O \in \mathbb{R}^3$  are the scalar and vector parts, respectively [? ]. The dynamics of  $\zeta_O$  can be proven to satisfy [? ]:

$$\dot{\zeta}_O = \frac{1}{2}E(\zeta_O)\omega_O \quad (2.7a)$$

$$\omega_O = 2E(\zeta_O)^\top \dot{\zeta}_O, \quad (2.7b)$$

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

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

and hence it holds that  $E(\zeta)^\top E(\zeta) = I_3, \forall \zeta \in \mathbb{S}^3$ . It can be also shown that

$$\dot{\omega}_O = 2E(\zeta_O)^\top \ddot{\zeta}_O.$$

In view of Fig. 2.1 one concludes that the pose of the agents and the object's center of mass are related as

$$p_{E_i}(q_i) = p_O + R_i(q_i)p_{E_i/O}^{E_i}, \quad (2.8a)$$

$$\eta_{E_i}(q_i) = \eta_O + \eta_{E_i/O}, \quad (2.8b)$$

$\forall i \in \mathcal{N}$ , where  $R_i := R_i(q_i) : \mathbb{R}^{n_i} \rightarrow \mathbb{SO}(3)$  is the  $i$ 's end-effector rotation matrix, and  $p_{E_i/O}^{E_i}, \eta_{E_i/O} \in \mathbb{R}^3$  are the *constant* distance and orientation offset vectors between  $\{O\}$  and  $\{E_i\}$ . Following (2.8), 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

$$v_i = J_{O_i} v_O, \quad (2.9)$$

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

$$J_{O_i}(x) := \begin{bmatrix} I_3 & -S(R_i(x)p_{E_i/O}^{E_i}) \\ 0 & I_3 \end{bmatrix}, \forall x \in \mathbb{R}^{n_i}, \quad (2.10)$$

which is always full-rank. Moreover, from (2.9), one obtains

$$\dot{v}_i = J_{O_i} \dot{v}_O + \dot{J}_{O_i} v_O. \quad (2.11)$$

In addition, it can be proved for  $J_{O_i}$  that

$$\|J_{O_i}(q_i)\| \leq \|p_{O/E_i}^{E_i}\| + 1, \forall q_i \in \mathbb{R}^{n_i}, i \in \mathcal{N}, \quad (2.12)$$

which will be used in the subsequent analysis.

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

$$h_O = Gh, \quad (2.13)$$

where  $G := G(q) : \mathbb{R}^n \rightarrow \mathbb{R}^{6 \times 6N}$ , with  $G(q) := [J_{O_1}^\top, \dots, J_{O_N}^\top]$ , is the full row-rank grasp matrix. By substituting (2.4) into (2.13), we obtain:

$$h_O = G(u - M\dot{v} - Cv - g - d),$$

which, after substituting (2.9), (2.11), (2.5), and rearranging terms, yields the overall system coupled dynamics:

$$\widetilde{M}\dot{v}_O + \widetilde{C}v_O + \widetilde{g} + \widetilde{d} = Gu, \quad (2.14)$$

where

$$\widetilde{M} := \widetilde{M}(x) := M_O + GMG^\top \quad (2.15a)$$

$$\widetilde{C} := \widetilde{C}(x) := C_O + GCG^\top + GM\dot{G}^\top \quad (2.15b)$$

$$\widetilde{g} := \widetilde{g}(x) := g_O + Gg. \quad (2.15c)$$

$$\widetilde{d} := \widetilde{d}(x, t) := d_O + Gd \quad (2.15d)$$

and  $x$  is the overall state  $x := [q^\top, \dot{q}^\top, \eta_O^\top, \omega_O^\top]^\top \in \mathbb{X} := \mathbb{S} \times \mathbb{R}^{n+3} \times \mathbb{T}$ ,  $\mathbb{S} := \mathbb{S}_1 \times \dots \times \mathbb{S}_N$ . Moreover, the following Lemma is necessary for the following analysis.

**Lemma 2.1.** *The matrix  $\dot{\widetilde{M}}(x)$  is symmetric and positive definite and the matrix  $\dot{\widetilde{M}}(x) - 2\widetilde{C}(x)$  is skew symmetric, i.e.,*

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

*Proof.* The matrices  $M_O$  and  $M_i$  are symmetric and positive definite,  $\forall i \in \mathcal{N}$  and the matrices  $\dot{M}_i - 2C_i$ ,  $M_O - 2C_O$  are skew-symmetric,  $\forall i \in \mathcal{N}$  [? ], which leads to the skew-symmetry of  $\dot{M} - 2C$ . Therefore, since  $G$  is full row-rank, we can conclude the symmetry and positive definiteness of  $\dot{\widetilde{M}}$ . Regarding the skew symmetry of  $\dot{\widetilde{M}} - 2\widetilde{C}$ , we define first  $A := A(x) := \dot{G}MG^\top$ , and we have from (2.15b):

$$\dot{\widetilde{M}} - 2\widetilde{C} = \dot{M}_O - 2C_O + G(\dot{M} - 2C)G^\top + A - A^\top,$$

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

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

$$\underline{m}I_6 \leq \dot{\widetilde{M}}(x) \leq \bar{m}I_6, \quad (2.16)$$

$\forall x \in \mathbb{X}$ , where  $\underline{m}$  and  $\bar{m}$  are positive unknown constants.

## 2.2.2 Problem Statement - Uncertain Model

The general problem treated in this chapter is the tracking of a pose/trajectory by the object. We first assume that the object and robot models (2.5), (2.3) are uncertain, i.e., they are not fully available for feedback in the control design. Officially, the problem we are aiming to solve for the rigid contact case is the following:

**Problem 2.1.** Given a desired bounded object smooth pose trajectory specified by  $x_d := x_d(t) := [(p_d)^\top, (\eta_d)^\top]^\top := [(p_d(t))^\top, (\eta_d(t))^\top]^\top : \mathbb{R}_{\geq 0} \rightarrow \mathbb{M}$ ,  $\eta_d := [\varphi_d, \theta_d, \psi_d] := [\varphi_d(t), \theta_d(t), \psi_d(t)] : \mathbb{R}_{\geq 0} \rightarrow \mathbb{T}$ , with bounded first and second derivatives, determine a continuous time-varying control law  $u$  in (2.14) such that

$$\lim_{t \rightarrow \infty} \begin{bmatrix} p_O(t) - p_d(t) \\ \eta_O(t) - \eta_d(t) \end{bmatrix} = 0$$

To solve the aforementioned problem, we need the following assumptions regarding the agent feedback and the kinematic singularities.

**Assumption 2.1.** (Feedback) Each agent  $i \in \mathcal{N}$  has continuous feedback of its own state  $q_i, \dot{q}_i$ .

**Assumption 2.2.** (Object geometry) Each agent  $i \in \mathcal{N}$  knows the constant offsets  $p_{E_i/O}^{E_i}$  and  $\eta_{E_i/O}, \forall i \in \mathcal{N}$ .

**Assumption 2.3.** (Kinematic singularities) The robotic agents operate away from kinematic singularities, i.e.,  $q_i(t)$  evolves in a closed subset of  $S_i, \forall i \in \mathcal{N}$ .

Assumption 2.1 is realistic for real manipulation systems, since on-board sensors can provide accurately the measurements  $q_i, \dot{q}_i$ . The object geometrical characteristics in Assumption 2.2 can be obtained by on-board sensors, whose inaccuracies are not modeled here and constitute part of future work. Finally, Assumption 2.3 states that the  $q_i$  that achieve  $x_O(t) = \bar{x}_d(t), \forall t \in \mathbb{R}_{\geq 0}$  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  $p_{E_i/O}^{E_i}$  and  $\eta_{E_i/O}$ , it can compute  $J_{O_i}(q_i)$  from (2.10), and  $x_O, v_O$  by inverting (2.8) and (2.9), respectively. Consequently, each agent can then compute the object unit quaternion  $\zeta_O$  as well as  $\zeta_O$ .

Note that, due to Assumption 2.2 and the grasp rigidity, the object-agents configuration is similar to a single closed-chain robot. The considered multi-agent setup, however, renders the problem more challenging, since the agents must calculate their own control signal in a decentralized manner, without communicating with each other. Moreover, each agent needs to compensate its own part of the (possibly uncertain/unknown) dynamics of the coupled dynamic equation (2.14), while respecting the rigidity kinematic constraints.

We present next two control schemes for the solution of Problem 2.1. 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  $\tilde{d}(x, t)$  in (2.14). The first control scheme is presented in Section 2.2.3 and is based on quaternion feedback and adaptation laws, while the second control scheme is given in Section 2.2.4 and is inspired by the Prescribed Performance Control (PPC) methodology introduced in [?].

### 2.2.3 Adaptive Control with Quaternion Feedback

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

**Assumption 2.4.** (Uncertainties/Disturbance parameterization) There exist constant *unknown* vectors  $\bar{d}_o \in \mathbb{R}^{\mu_o}, \bar{d}_i \in \mathbb{R}^\mu$  and known functions  $\delta_o := \delta_o(x_o, \dot{x}_o, t) : \mathbb{M} \times \mathbb{R}^6 \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^{6 \times \mu_o}, \delta_i := \delta_i(q_i, \dot{q}_i, t) : \mathbb{R}^{2n_i} \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^{6 \times \mu}$ , such that  $d_o(x_o, \dot{x}_o, t) = \delta_o(x_o, \dot{x}_o, t)\bar{d}_o, d_i(q_i, \dot{q}_i, t) = \delta_i(q_i, \dot{q}_i, t)\bar{d}_i$ ,  $\forall q_i, \dot{q}_i \in \mathbb{R}^{n_i}, x_o \in \mathbb{M}, \dot{x}_o \in \mathbb{R}^6, t \in \mathbb{R}_{\geq 0}, i \in \mathcal{N}$ , where  $\delta_o(x_o, \dot{x}_o, t)$  and  $\delta_i(q_i, \dot{q}_i, t)$  are continuous in  $(x_o, \dot{x}_o)$  and  $(q_i, \dot{q}_i)$ , respectively, and uniformly bounded in  $t$ .

The aforementioned assumption is motivated by the use of Neural Networks for approximating unknown functions in compact sets [? ]. More specifically, any continuous function  $f(x) : \mathbb{R}^n \rightarrow \mathbb{R}^m$  can be approximated on a known compact set  $X \subset \mathbb{R}^n$  by a Neural Network equipped with  $N$  Radial Basis Functions (RBFs)  $\Phi(x)$  and using unknown ideal constant connection weights that are stored in a matrix  $\Theta \in \mathbb{R}^{N \times m}$  as  $f(x) = \Theta^\top \Phi(x) + \varepsilon(x); \Theta^\top \Phi(x)$  represents the parametric uncertainty and  $\varepsilon(x)$  represents the unknown nonparametric uncertainty, which is bounded as  $\|\varepsilon(x)\| \leq \bar{\varepsilon}$  in  $X$ . In our case, the functions  $\delta_o, \delta_i$  play the role of the known function  $\Phi(x)$  and  $\bar{d}_o, \bar{d}_i$  and  $\mu, \mu_o$  represent the unknown constants  $\Theta$  and the number of layers of the Neural Network, respectively. Nevertheless, in view of Neural Network approximation, Assumption 4 implies that the nonparametric uncertainty is zero and that  $d_o$  and  $d_i$  are *known* functions of *time*. These properties can be relaxed with non-zero bounded nonparametric uncertainties and *unknown* but bounded time-dependent disturbances, i.e.  $d_i(q_i, \dot{q}_i, t) = \delta_{i,q}(q_i, \dot{q}_i)\bar{d}_i + d_{i,t}(t) + \varepsilon_{i,q}(q_i, \dot{q}_i)$  and  $d_o(x_o, \dot{x}_o, t) = \delta_{o,x}(x_o, \dot{x}_o)\bar{d}_o + d_{o,t}(t) + \varepsilon_{o,x}(x_o, \dot{x}_o)$ , where  $d_{i,t}, d_{o,t}, \varepsilon_{i,q}, \varepsilon_{o,x}$  are bounded. In that case, instead of asymptotic convergence of the pose to the desired one, we can show convergence of the respective errors to a compact set around the origin. For more details on Neural Network approximation and adaptive control with illustrative examples, we refer the reader to [? , Ch. 12].

The desired Euler angle orientation vector  $\eta_d : \mathbb{R}_{\geq 0} \rightarrow \mathbb{T}$  is transformed first to the unit quaternion  $\zeta_d := \zeta_d(t) : \mathbb{R}_{\geq 0} \rightarrow \mathbb{S}^3$  [? ]. 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:

$$e_p := p_o - p_d.$$

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  $e_\zeta = [e_\varphi, e_\epsilon^\top]^\top \in \mathbb{S}^3$  be the unit quaternion describing the orientation error. Then, it holds that [?],

$$e_\zeta := \zeta_d \cdot \zeta_o^+ = \begin{bmatrix} \varphi_d \\ \epsilon_d \end{bmatrix} \cdot \begin{bmatrix} \varphi_o \\ -\epsilon_o \end{bmatrix},$$

and yields

$$e_\zeta = \begin{bmatrix} e_\varphi \\ e_\epsilon \end{bmatrix} := \begin{bmatrix} \varphi_o \varphi_d + \epsilon_o^\top \epsilon_d \\ \varphi_o \epsilon_d - \varphi_d \epsilon_o + S(\epsilon_o) \epsilon_d \end{bmatrix}.$$

By employing the quaternion dynamics (see (2.7a)) and certain properties of skew-symmetric matrices [?], it can be shown that the error dynamics of  $e_p, e_\varphi$  are:

$$\dot{e}_p = \dot{p}_o - \dot{p}_d \quad (2.17a)$$

$$\dot{e}_\varphi = \frac{1}{2} e_\epsilon^\top e_\omega \quad (2.17b)$$

$$\dot{e}_\epsilon = -\frac{1}{2} [e_\varphi I_3 + S(e_\epsilon)] e_\omega - S(e_\epsilon) \omega_d, \quad (2.17c)$$

where  $e_\omega := \omega_o - \omega_d$  is the angular velocity error, with  $\omega_d = 2E(\zeta_d)^\top \dot{\zeta}_d$ , as indicated by (2.7b).

Due to the ambiguity of unit quaternions, when  $\zeta_o = \zeta_d$ , then  $e_\zeta = [1, 0_3^\top]^\top \in \mathbb{S}^3$ . If  $\zeta_o = -\zeta_d$ , then  $e_\zeta = [-1, 0_3^\top]^\top \in \mathbb{S}^3$ , which, however, represents the same orientation. Therefore, the control objective established in Problem 2.1 is equivalent to

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

The left hand side of (2.3), after employing (2.9) and (2.11), becomes

$$M_i \ddot{v}_i + C_i \dot{v}_i + g_i + d_i = M_i \left( J_{O_i} \ddot{v}_o + \dot{J}_{O_i} v_o \right) + C_i J_{O_i} v_o + g_i + d_i.$$

which, according to Assumption 2.4 and the fact that the manipulator dynamics can be linearly parameterized with respect to dynamic parameters [?], becomes

$$M_i J_{O_i} \ddot{v}_o + \left( M_i \dot{J}_{O_i} + C_i J_{O_i} \right) v_o + g_i + d_i = Y_i \vartheta_i + \delta_i \bar{d}_i,$$

$\forall i \in \mathcal{N}$ , 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  $Y_i := Y_i(q_i, \dot{q}_i, v_o, \dot{v}_o) : \mathcal{S} \times \mathbb{R}^{n_i+12} \rightarrow \mathbb{R}^{6 \times \ell}$  are known regressor matrices, independent of  $\vartheta_i, i \in \mathcal{N}$ . Without loss of generality, we assume here that

the dimension of  $\vartheta_i$  is the same,  $\ell$  for all the agents. Similarly, the dynamical terms of the left hand side of (2.5b) can be written as

$$M_O \dot{v}_O + C_O v_O + g_O + d_O = Y_O \vartheta_O + \delta_O \bar{d}_O,$$

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  $Y_O := Y_O(\eta_O, \omega_O, v_O, \dot{v}_O) : \mathbb{T} \times \mathbb{R}^{15} \rightarrow \mathbb{R}^{6 \times \ell_O}$  is a known regressor matrix, independent of  $\vartheta_O$ . It is worth noting that the choice for  $\ell$  and  $\ell_O$  is not unique. In view of the aforementioned expressions, the left-hand side of (2.14) can be written as:

$$\widetilde{M} \dot{v}_O + \widetilde{C} v_O + \widetilde{g} + \widetilde{d} = Y_O \vartheta_O + \delta_O \bar{d}_O + G \left( \widetilde{Y} \vartheta + \widetilde{\delta} \bar{d} \right) \quad (2.18)$$

where  $\widetilde{Y} := \widetilde{Y}(q, \dot{q}, v_O, \dot{v}_O) := \text{diag}\{[Y_i]_{i \in \mathcal{N}}\} \in \mathbb{R}^{6N \times N\ell}$ ,  $\vartheta := [\vartheta_1^\top, \dots, \vartheta_N^\top]^\top \in \mathbb{R}^{N\ell}$ ,  $\widetilde{\delta} := \widetilde{\delta}(q, \dot{q}, t) := \text{diag}\{[\delta_i]_{i \in \mathcal{N}}\} \in \mathbb{R}^{6N \times N\mu}$ , and  $\bar{d} := [\bar{d}_1^\top, \dots, \bar{d}_N^\top]^\top \in \mathbb{R}^{N\mu}$ .

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

$$e_{\vartheta_O} := \vartheta_O - \hat{\vartheta}_O \quad (2.19a)$$

$$e_\vartheta := \begin{bmatrix} e_{\vartheta_1} \\ \vdots \\ e_{\vartheta_N} \end{bmatrix} := \begin{bmatrix} \vartheta_1 - \hat{\vartheta}_1 \\ \vdots \\ \vartheta_N - \hat{\vartheta}_N \end{bmatrix} = \vartheta - \hat{\vartheta}. \quad (2.19b)$$

In the same vein, we introduce the states  $\hat{d}_O \in \mathbb{R}^{\mu_O}$  and  $\hat{d}_i \in \mathbb{R}^\mu$  that correspond to the estimates of  $\bar{d}_O$  and  $\bar{d}_i$ , respectively, by agent  $i \in \mathcal{N}$ , and the corresponding stack vector  $\hat{d} := [\hat{d}_1^\top, \dots, \hat{d}_N^\top]^\top \in \mathbb{R}^{N\mu}$ , for which we also formulate the associated errors as

$$e_{d_O} := \bar{d}_O - \hat{d}_O \in \mathbb{R}^{\mu_O} \quad (2.20a)$$

$$e_d := \begin{bmatrix} e_{d_1} \\ \vdots \\ e_{d_N} \end{bmatrix} := \begin{bmatrix} \bar{d}_1 - \hat{d}_1 \\ \vdots \\ \bar{d}_N - \hat{d}_N \end{bmatrix} = \bar{d} - \hat{d} \in \mathbb{R}^{N\mu}. \quad (2.20b)$$

Next, we design the reference velocity

$$v_f := v_d - K_f e = \begin{bmatrix} \dot{p}_d - k_p e_p \\ \omega_d + k_\zeta e_\epsilon \end{bmatrix} \quad (2.21)$$

where  $v_d := [p_d^\top, \omega_d^\top]^\top$ ,  $e := [e_p^\top, -e_\epsilon^\top]^\top \in \mathbb{R}^6$ , and  $K_f := \text{diag}\{k_p, k_\zeta\}$ , with  $k_p, k_\zeta$  positive control gains. We also introduce the respective velocity error  $e_{v_f}$  as

$$e_{v_f} := v_o - v_f, \quad (2.22)$$

and design the adaptive control law  $u_i$  in (2.14), for each agent  $i \in \mathcal{N}$ , as  $u_i : \mathbb{U}_f \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^6$  with

$$u_i := u_i(\chi_f, t) := Y_{f_i} \hat{\vartheta}_i + \delta_i \hat{d}_i + J_{M_i} \left[ Y_{f_O} \hat{\vartheta}_O - e - K_v e_{v_f} + \delta_O \hat{d}_O \right], \quad (2.23)$$

where  $\mathbb{U}_f := \mathbb{S}_i \times \mathbb{T} \times \mathbb{R}^{12+\ell+\ell_O+\mu+\mu_O}$ ,  $\chi_f := [q_i^\top, \eta_O^\top, e^\top, e_{v_f}^\top, \hat{\vartheta}_i^\top, \hat{\vartheta}_O^\top, \hat{d}_i^\top, \hat{d}_O^\top]^\top$ ,  $Y_{f_i} := Y_i(q_i, \dot{q}_i, v_f, \dot{v}_f)$ ,  $Y_{f_O} := Y_O(\eta_O, \omega_O, v_f, \dot{v}_f)$ ,  $K_v$  is a positive definite gain matrix and  $J_{M_i} := J_{M_i}(q_i) : \mathbb{R}^{n_i} \rightarrow \mathbb{R}^{6 \times 6}$  are the matrices [?]

$$J_{M_i}(q_i) = \begin{bmatrix} m_i^*(m_O^*)^{-1} I_3 & m_i^*(J_O^*)^{-1} S(R_O(\eta_{E_i}(q_i) - \eta_{E_i/O}) p_{O/E_i}^O) \\ 0 & J_i^*(J_O^*)^{-1} \end{bmatrix}, \quad (2.24)$$

for some positive coefficients  $m_i^* \in \mathbb{R}_{>0}$  and positive definite matrices  $J_i^* \in \mathbb{R}^{3 \times 3}$ ,  $\forall i \in \mathcal{N}$ , satisfying

$$\begin{aligned} m_O^* &= \sum_{i \in \mathcal{N}} m_i^*, \quad \sum_{i \in \mathcal{N}} R_i(q_i) p_{E_i/O}^{E_i} m_i^* = 0 \\ J_O^* &= \sum_{i \in \mathcal{N}} J_i^* - \sum_{i \in \mathcal{N}} m_i^* S(R_i(q_i) p_{E_i/O}^{E_i})^2. \end{aligned}$$

In addition, we design the following adaptation laws:

$$\dot{\hat{\vartheta}}_i = -\gamma_i Y_{f_i}^\top J_{O_i} e_{v_f}, \quad (2.25a)$$

$$\dot{\hat{d}}_i = -\beta_i \delta_i^\top J_{O_i} e_{v_f} \quad (2.25b)$$

$$\dot{\hat{\vartheta}}_O = -\gamma_O Y_{f_O}^\top e_{v_f} \quad (2.25c)$$

$$\dot{\hat{d}}_O = -\beta_O \delta_O^\top e_{v_f}, \quad (2.25d)$$

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

The control and adaptation laws can be written in vector form

$$u = \tilde{Y}_f \hat{\vartheta} + \tilde{\delta} \hat{d} + G_M^+ \left[ Y_{f_O} \hat{\vartheta}_O - e + \delta_O \hat{d}_O - K_v e_{v_f} \right] \quad (2.26a)$$

$$\dot{\hat{\vartheta}} = -\Gamma \tilde{Y}_f^\top G^\top e_{v_f} \quad (2.26b)$$

$$\dot{\hat{d}} = -B_g \tilde{\delta}^\top G^\top e_{v_f} \quad (2.26c)$$

$$\dot{\hat{\vartheta}}_O = -\gamma_O Y_{f_O}^\top e_{v_f} \quad (2.26d)$$

$$\dot{\hat{d}}_O = -\beta_O \delta_O^\top e_{v_f}, \quad (2.26e)$$

where  $\tilde{Y}_f := \tilde{Y}(q, \dot{q}, v_f, \dot{v}_f)$ ,  $G_M^+ := G_M^+(q) := [J_{M_1}^\top, \dots, J_{M_N}^\top]^\top \in \mathbb{R}^{6N \times 6}$ ,  $B_g := \text{diag}\{[\beta_i I_\mu]_{i \in \mathcal{N}}\}$ , and  $\Gamma := \text{diag}\{[\gamma_i I_\ell]_{i \in \mathcal{N}}\}$ . The matrix  $G_M^+(q)$  was introduced in [?], where it was proved that it yields a load distribution that is free of internal forces. The parameters  $m_O^*, m_i^*$  are used to distribute the object's needed effort (the term that right multiplies  $G_M^+(q)$  in (2.26a)) to the agents.

**Remark 2.1 (Decentralized manner (adaptive controller)).** Notice from (2.23) and (2.25) that the overall control protocol is decentralized in the sense that the agents calculate their own control signals without communicating with each other. In particular, the control gains and the desired trajectory can be transmitted off-line to the agents, which can compute the object's pose and velocity, and hence the signals  $e, v_f, e_{v_f}$  from the inverse kinematics. For the computation of  $J_{M_i}$ , each agent needs knowledge of the offsets  $p_{E_i/O}^{E_i}$ , which can also be transmitted off-line to the agents. Moreover, by also transmitting off-line to the agents the initial conditions  $\hat{\vartheta}_O, \hat{d}_O$ , and via the adaptation laws (2.26d), (2.26e), each agent has access to the adaptation signals  $\hat{\vartheta}_O(t), \hat{d}_O(t), \forall t \in \mathbb{R}_{\geq 0}$ . Finally, the structure of the functions  $\delta_i, \delta_O, Y_i, Y_O$ , as well as the constants  $m_i^*, J_i^*$  can be also known by the agents a priori.

The following theorem summarizes the main results of this subsection.

**Theorem 2.1.** Consider  $N$  robotic agents rigidly grasping an object with coupled dynamics described by (2.14) and unknown dynamic parameters. Then, under Assumptions 2.1, 2.4, by applying the control protocol (2.23) with the adaptation laws (2.25), the object pose converges asymptotically to the desired pose trajectory. Moreover, all closed loop signals are bounded.

*Proof.* Consider the nonnegative function

$$\begin{aligned} V := & \frac{1}{2} e_p^\top e_p + 2(1 - e_\varphi) + \frac{1}{2} e_{v_f}^\top \tilde{M} e_{v_f} + \frac{1}{2} e_\vartheta^\top \Gamma^{-1} e_\vartheta + \frac{1}{2\gamma_O} e_{\vartheta_O}^\top e_{\vartheta_O} \\ & + \frac{1}{2} e_d^\top B_g^{-1} e_d + \frac{1}{2\beta_O} e_{d_O}^\top e_{d_O}, \end{aligned} \quad (2.27)$$

By taking the derivative of  $V$  and using (2.22), (2.21), (2.18), and Lemma 2.1, we obtain

$$\begin{aligned} \dot{V} = & -e^\top K_f e + e_{v_f}^\top [G(u - \tilde{Y}_f \vartheta - \tilde{\delta} \bar{d}) + e - \delta_O \bar{d}_O - Y_{f_O} \vartheta_O] - e_\vartheta^\top \Gamma^{-1} \dot{\vartheta} \\ & - \frac{1}{\gamma_O} e_{\vartheta_O}^\top \dot{\vartheta}_O - e_d^\top B_g^{-1} \dot{\bar{d}} - \frac{1}{\beta_O} e_{d_O}^\top \dot{\bar{d}}_O, \end{aligned}$$

and after substituting the adaptive control and adaptation laws (2.26) and using the fact that  $G^\top G_M^+ = I_6$ ,

$$\begin{aligned}\dot{V} &= -e^\top K_f e - e_{v_f}^\top K_v e_{v_f} - e_{v_f}^\top \left[ G \left( \tilde{Y}_f e_\vartheta + \tilde{\delta} e_d \right) + Y_{f_O} e_{\vartheta_O} + \delta_O e_{d_O} \right] \\ &\quad + e_\vartheta^\top \tilde{Y}_f^\top G^\top e_{v_f} + e_d^\top \tilde{\delta}^\top G^\top e_{v_f} + e_{\vartheta_O}^\top Y_{f_O}^\top e_{v_f} + e_{d_O}^\top \delta_O^\top e_{v_f} \\ &= -k_p \|e_p\|^2 - k_\zeta \|e_\epsilon\|^2 - e_{v_f}^\top K_v e_{v_f},\end{aligned}\tag{2.28}$$

which is non-positive. Note, however, that  $\dot{V}$  is not negative definite, and we need to invoke invariance-like properties to conclude the asymptotic stability of  $e_p, e_\epsilon, e_{v_f}$ . Since the closed-loop system is non-autonomous (this can be verified by inspecting (2.17), the derivative of (2.22) and (2.26)), LaSalle's invariance principle is not applicable, and we thus employ Barbalat's lemma (Lemma A.1 of Appendix A). From (2.28) we conclude the boundedness of  $V$  and of  $x$ , which implies the boundedness of the dynamic terms  $\tilde{M}(x), \tilde{C}(x), \tilde{g}(x)$ . Moreover, by invoking the boundedness of  $p_d(t), v_d(t), \omega_d(t), \dot{v}_d(t), \dot{\omega}_d(t)$ , we conclude the boundedness of  $v_f, v_O, v_i, \hat{\vartheta}_O, \hat{\vartheta}, \hat{d}, \hat{d}_O$ . By differentiating (2.17), we also conclude the boundedness of  $\dot{v}_f$  and therefore, the boundedness of the control and adaptation laws (2.23) and (2.25). Thus, we can conclude the boundedness of the second derivative  $\dot{V}$  and by invoking Corollary 8.1 of [?], the uniform continuity of  $\dot{V}$ . Therefore, according to Barbalat's lemma, we deduce that  $\lim_{t \rightarrow \infty} \dot{V}(t) = 0$  and, consequently, that  $\lim_{t \rightarrow \infty} e_p(t) = 0$ ,  $\lim_{t \rightarrow \infty} e_{v_f}(t) = 0$ , and  $\lim_{t \rightarrow \infty} \|e_\epsilon(t)\|^2 = 0$ , which, given that  $e_\zeta$  is a unit quaternion, leads to the configuration  $(e_p, e_{v_f}, e_\varphi, e_\epsilon) = (0, 0, \pm 1, 0)$ .

□

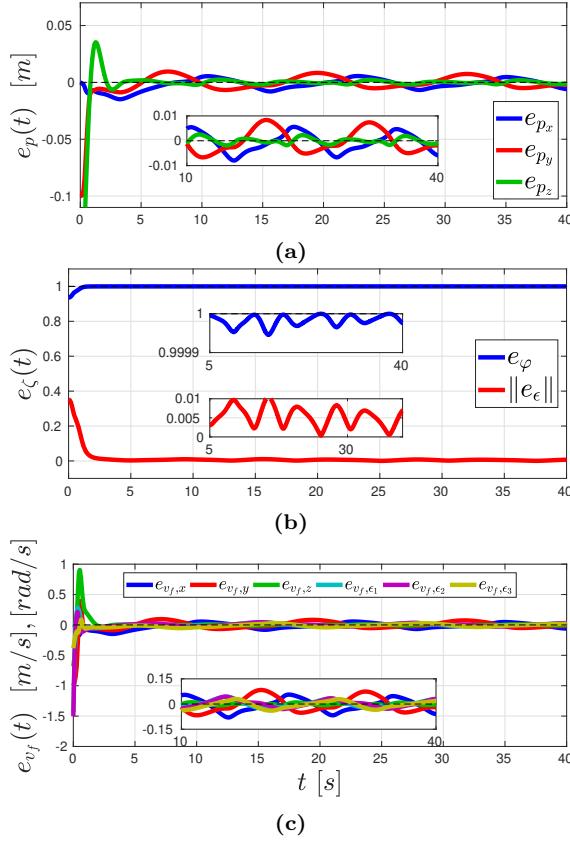
**Remark 2.2 (Unwinding).** Note that the two configurations where  $e_\varphi = 1$  and  $e_\varphi = -1$  represent the same orientation. The closed loop dynamics of  $e_\varphi$ , as given in (2.17b), can be written, in view of (2.21), as  $\dot{e}_\varphi = k_\zeta \frac{1}{2} \|e_\epsilon\|^2 + \frac{1}{2} [0_3^\top, e_\epsilon^\top] e_{v_f}$ . Since the first term is always positive, we conclude that the equilibrium point  $(e_p, e_{v_f}, e_\varphi, e_\epsilon) = (0, 0, -1, 0)$  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  $(e_p, e_{v_f}, e_\varphi, e_\epsilon) = (0, 0, -1, 0)$ , it will stay there, which also corresponds to the desired orientation behavior). This is the so-called *unwinding phenomenon* [?]. Note, however, that the desired equilibrium point  $(e_p, e_{v_f}, e_\varphi, e_\epsilon) = (0, 0, 1, 0)$  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 (2.21) and considering the term  $2(1 + e_\varphi)$  instead of  $2(1 - e_\varphi)$  in the function (2.27).

In order to avoid the unwinding phenomenon, instead of the error  $e = [e_p^\top, -e_\epsilon^\top]^\top$ , we can choose  $e = [e_p^\top, -e_\varphi e_\epsilon^\top]^\top$ . Then by replacing the term  $1 - e_\varphi$  with  $1 - e_\varphi^2$  in (2.27) and using (2.26), we conclude by proceeding with a similar analysis that  $(e_p, \|e_\epsilon\|e_\varphi, e_{v_f}) \rightarrow (0, 0, 0)$ , which implies that the system is asymptotically driven to either the configuration  $(e_p, e_{v_f}, e_\varphi, e_\epsilon) = (0, 0, \pm 1, 0)$ , which is the desired one, or a configuration  $(e_p, e_{v_f}, e_\varphi, e_\epsilon) = (0, 0, 0, \tilde{e}_\epsilon)$ , where  $\tilde{e}_\epsilon \in \mathbb{S}^2$  is a unit vector. The latter represents a set of invariant undesired equilibrium points. The closed loop dynamics are  $\dot{e}_\varphi = \frac{1}{2}e_\varphi\|e_\epsilon\|^2 + \frac{1}{2}[0_3^\top, e_\epsilon^\top]e_{v_f}$ , and  $\frac{\partial}{\partial t}\|e_\epsilon\|^2 = -e_\varphi^2\|e_\epsilon\|^2 - e_\varphi[0_3^\top, e_\epsilon^\top]e_{v_f}$ . We can conclude from the term  $[0_3^\top, e_\epsilon^\top]e_{v_f}$  that there exist trajectories that can bring the system close to the undesired equilibrium, rendering thus the point  $(e_p, e_{v_f}, e_\varphi, e_\epsilon) = (0, 0, \pm 1, 0)$  only locally asymptotically stable. It has been proved that  $e_\varphi = \pm 1$  cannot be globally stabilized with a purely continuous controller [?]. Discontinuous control laws have also been proposed (e.g., [?]), whose combination with adaptation techniques constitutes part of our future research directions.

**Remark 2.3 (Robustness (adaptive controller)).** Notice also that the control protocol compensates the uncertain dynamic parameters and external disturbances through the adaptation laws (2.25), although the errors (2.19), (2.20) do not converge to zero, but remain bounded. Finally, the control gains  $k_p, k_\zeta, K_v$  can be tuned appropriately so that the proposed control inputs do not reach motor saturations in real scenarios.

## Simulation Results

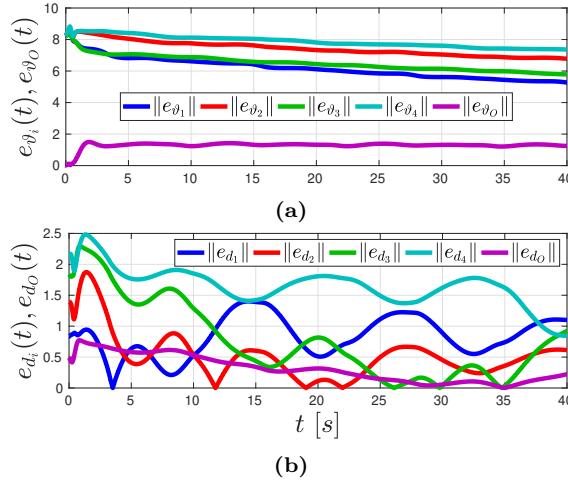
We provide here simulation results for the developed control scheme. The tested scenario consists of four UR5 robotic manipulators rigidly grasping a rectangular object. The object's initial pose is  $x_o(0) = [-0.225, -0.612, 0161, -\pi, \frac{\pi}{3}, 0]^\top$  ([m], [rad]) with respect to a chosen inertial frame and the desired trajectory is set as  $p_d(t) = [-0.225+0.1 \sin(0.5t), -0.612+0.2 \cos(0.5t), 0.25+0.05 \sin(0.5t)]^\top$ ,  $\eta_d(t) = [-\pi+0.25 \cos(0.5t), \frac{\pi}{3}+A_\theta \sin(0.25t), 0.25 \cos(0.5t)]^\top$ , where  $A_\theta = \frac{\pi}{6}$  (note that the desired pitch angle reaches the configuration of  $\frac{\pi}{2}$ , which yields a representation singularity in the Euler-angle formulation). In view of Assumption 2.4, we set  $d_i = (\|q_i\| \sin(\omega_{d_i} t + \phi_{d_i}) + \dot{q}_i)\bar{d}_i$  and  $d_o = (\|\dot{x}_o\| \sin(\omega_{d_O} t + \phi_{d_O}) + v_o)\bar{d}_O$ , where the constants  $\omega_{d_i}, \phi_{d_i}, \omega_{d_O}, \omega_{d_O}$  are randomly chosen in the interval  $(0, 1)$ ,  $\forall i \in \mathcal{N}$ . Regarding the force distribution matrix (2.24), we set  $m_i^* = 1$ ,  $\forall i \in \mathcal{N}$ , and  $J_1^* = 0.6I_3$ ,  $J_2^* = 0.4I_3$ ,  $J_3^* = 0.75I_3$ ,  $J_4^* = 0.25I_3$  to demonstrate a potential difference in the agents' power capabilities. In addition, we set an artificial saturation limit for the joint motors as  $\bar{\tau} = 150$  Nm. We set the control gains appearing in (2.23) and (2.25) as  $k_p = \text{diag}\{[5, 5, 2]\}$ ,  $k_\zeta = 3I_3$ ,  $K_v = 400I_6$ ,  $\gamma_i = \gamma_o = \beta_i = \beta_o = 1$ ,  $\forall i \in \mathcal{N}$ . The simulation results are depicted in Figs. 2.2, 2.4 for  $t \in [0, 40]$ .



**Figure 2.2:** Simulation results for the control scheme of Section 2.2.3. (a): The position errors  $e_p(t)$ ; (b): The quaternion errors  $e_\varphi(t)$ ,  $\|e_\epsilon(t)\|$ ; (c) The velocity errors  $e_{v_f}(t)$ ,  $\forall t \in [0, 40]$ . A zoomed version of the steady-state response has been included in all plots.

seconds. More specifically, Fig. 2.2 shows the evolution of the pose and velocity errors  $e_p(t)$ ,  $e_\zeta(t)$ ,  $e_{v_f}(t)$ , Fig. 2.3 depicts the norms of the adaptation errors  $e_{\vartheta_i}(t)$ ,  $e_{\vartheta_O}(t)$ ,  $e_{d_i}(t)$ ,  $e_{d_O}(t)$ , and Fig. 2.4 shows the resulting joint torques  $\tau_i(t)$ ,  $\forall i \in \{1, \dots, 4\}$ . Note that  $e_p(t)$ ,  $e_\zeta(t)$  and  $e_{v_f}(t)$  converge to the desired values and the adaptation errors are bounded, as predicted by the theoretical analysis.

One can conclude from the aforementioned figures that the simulation results verify the theoretical findings, since asymptotic stability is achieved. Moreover, the joint torques respect the saturation values we set. The simulations were carried out in the MATLAB R2017a environment on a

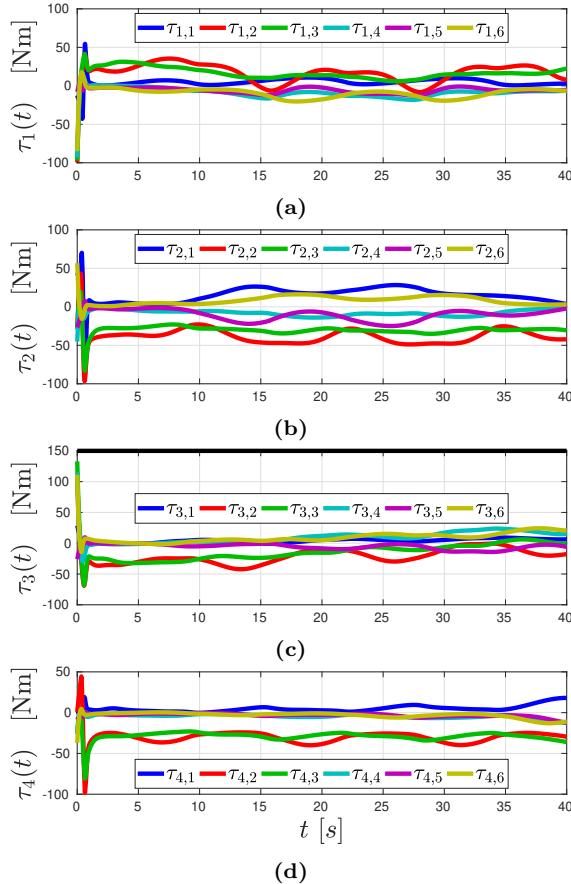


**Figure 2.3:** The adaptation error norms  $\|e_{\vartheta_i}(t)\|$ ,  $i \in \mathcal{N}$ ,  $\|e_{\vartheta_O}(t)\|$  (a),  $\|e_{d_i}(t)\|$ ,  $i \in \mathcal{N}$ ,  $\|e_{d_O}(t)\|$  (b), of the control scheme of Section 2.2.3  $\forall t \in [0, 40]$ .

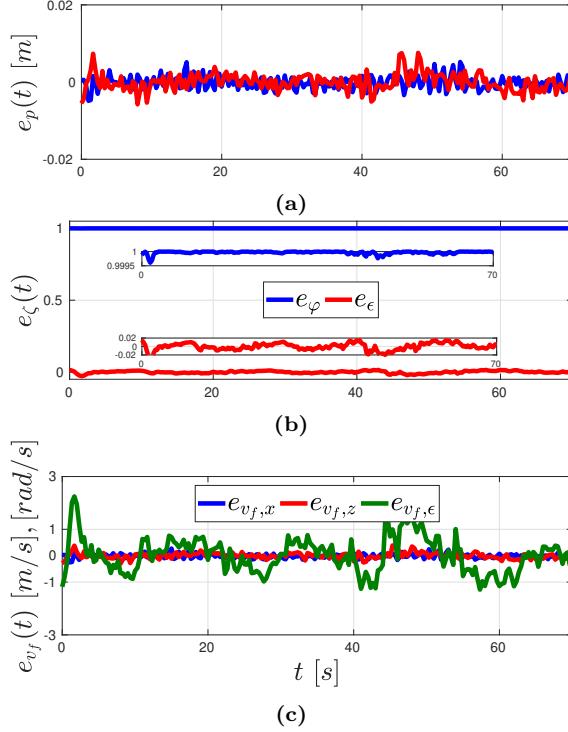
i7-5600 laptop computer at 2.6Hz, with 8GB of RAM.

## Experimental Results

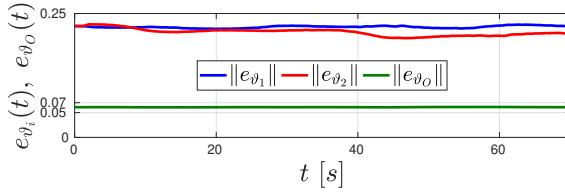
We further validate the developed control scheme through experimental results. The tested scenario for the experimental setup consists of two WidowX Robot Arms rigidly grasping a wooden cuboid object of initial pose  $x_O(0) = [0.3, 0, 0.15, 0, 0, 0]^\top$  ([m], [rad]), which has to track a planar time trajectory  $p_d(t) = [0.3 + 0.05 \sin(\frac{2\pi t}{35}), 0.15 - 0.05 \cos(\frac{2\pi t}{35})]^\top$ ,  $\eta_d(t) = \frac{\pi}{20} \sin(\frac{5\pi t}{35})$ . For that purpose, we employ the three rotational -with respect to the  $y$  axis - joints of the arms. The lower joint 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. 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, 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]. 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. The load distribution coefficients are set as  $m_1^* = m_2^* = 1$ , and  $J_1^* = 0.75I_3$ ,  $J_2^* = 0.25I_3$ . For the adaptive quaternion-feedback control scheme, we set  $\delta_O(x_O, \dot{x}_O, t) = 0$ ,  $\delta_i(q_i, \dot{q}_i, t) = 0$ ,  $\forall i \in \mathcal{N}$ , which essentially means that we do not model any external disturbances. We also



**Figure 2.4:** The agents' joint torques  $\tau_i(t)$ ,  $i \in \mathcal{N}$ , (in (a)-(d), respectively) of the control scheme of Section 2.2.3  $\forall t \in [0, 40]$ , and the motor saturation (with black), which has not been plotted in (a), (b), (d) for better visualization.

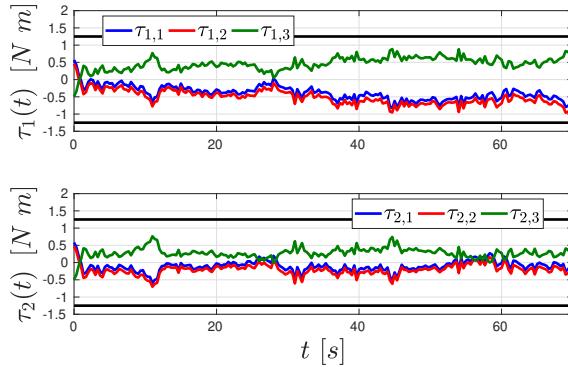


**Figure 2.5:** Experimental results for the control scheme of Section 2.2.3  
(a): The position errors  $e_p(t)$ ; (b): The quaternion errors  $e_\varphi(t)$ ,  $e_\epsilon(t)$ ; (c) The velocity errors  $e_{v_f}(t)$ ,  $\forall t \in [0, 70]$ .



**Figure 2.6:** The norms of the adaptation signals  $e_{\vartheta_i}(t)$ ,  $\forall i \in \{1, 2\}$  (left) and  $e_{\vartheta_O}(t)$ , (right)  $\forall t \in [0, 70]$  of the experiment of the controller in Section 2.2.3

set the control gains appearing in (2.23) and (2.25) as  $k_p = 50$ ,  $k_\zeta = 80$ ,  $K_v = \text{diag}\{3.5, 0.5, 0.5\}$ . The experimental results are depicted in Fig. 2.5–2.7 for  $t \in [0, 70]$  seconds. More specifically, Fig. 2.5 pictures the pose and velocity errors  $e_p(t)$ ,  $e_\zeta(t)$ ,  $e_{v_f}(t)$ , Fig. 2.6 depicts the norms of the adaptation errors  $e_{\vartheta_i}(t)$ ,  $e_{\vartheta_O}(t)$ , and Fig. 2.7 shows the joint torques  $\tau_1(t)$ ,  $\tau_2(t)$  of the agents. Although external disturbances and modeling uncertainties are



**Figure 2.7:** The agents' joint torques of the experiment of the controller in Section 2.2.3, for  $t \in [0, 70]$ , with their respective limits (with black).

not taken into account in the system model, they are indeed present during the experiment run time and one can observe that the errors converge to the desired values and the adaptation errors remain bounded, verifying the theoretical findings. A video illustrating the simulation and experimental results (along with the control scheme of the next section) can be found on <https://youtu.be/jJWeI5ZvQPY>.

## 2.2.4 Prescribed Performance Control

In this section, we adopt the concepts and techniques of prescribed performance control, proposed in [?], 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 Appendix B, 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

$$e_s := [e_{s_x}, e_{s_y}, e_{s_z}, e_{s_\phi}, e_{s_\theta}, e_{s_\psi}]^\top := x_O - x_d \quad (2.29)$$

Similarly to the result of the previous subsection, the Euler angle Euclidean difference here does not represent a valid orientation distance metric. However, as also stated before, the desired equilibrium point will be rendered *eventually attractive*, which stems from stabilization on the unit sphere. A PPC scheme based on a proper distance metric on  $\mathbb{SO}(3)$  is introduced in the next chapter.

We now relax Assumption 2.4 and impose a controllability assumption on  $\theta_d$ , given that Euler angles are used now:

**Assumption 2.5** (Uncertainties/Disturbances bound). The functions  $d_o(x_o, \dot{x}_o, t)$  and  $d_i(q_i, \dot{q}_i, t)$  are continuous in  $(x_o, \dot{x}_o)$  and  $(q_i, \dot{q}_i)$ , respectively, and bounded in  $t$  by unknown positive constants  $\bar{d}_o$  and  $\bar{d}_i$ , respectively,  $\forall i \in \mathcal{N}$ .

**Assumption 2.6.** It holds that  $\theta_d(t) \in [-\bar{\theta}, \bar{\theta}] \subset (-\frac{\pi}{2}, \frac{\pi}{2}), \forall t \in \mathbb{R}_{\geq 0}$ .

More specifically, the requirement  $\theta_d(t) \in [-\bar{\theta}, \bar{\theta}] \subset (-\frac{\pi}{2}, \frac{\pi}{2}), \forall t \in \mathbb{R}_{\geq 0}$  is a necessary condition needed to ensure that tracking of  $\theta_d$  will not result in singular configurations of  $J_o(\eta_o)$ . The constant  $\bar{\theta} \in [0, \frac{\pi}{2})$  can be taken arbitrarily close to  $\frac{\pi}{2}$ .

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

$$-\rho_{s_k}(t) < e_{s_k}(t) < \rho_{s_k}(t), \forall k \in \mathcal{K}, \quad (2.30)$$

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

$$\rho_{s_k} := \rho_{s_k}(t) := (\rho_{s_k,0} - \rho_{s_k,\infty}) \exp(-l_{s_k} t) + \rho_{s_k,\infty}, \quad \forall k \in \mathcal{K}, \quad (2.31)$$

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 (2.30) for all  $t \in \mathbb{R}_{\geq 0}$ . More specifically, given the errors (2.29):

**Step I-a.** Select the functions  $\rho_{s_k}$  as in (2.31) with

- (i)  $\rho_{s_\theta,0} = \rho_{s_\theta}(0) = \theta^*, \rho_{s_k,0} = \rho_{s_k}(0) > |e_{s_k}(0)|, \forall k \in \mathcal{K} \setminus \{\theta\}$ ,
- (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  $\theta^*$  is a positive constant satisfying  $\theta^* + \bar{\theta} < \frac{\pi}{2}$ .

**Step I-b.** Introduce the normalized errors

$$\xi_s := [\xi_{s_x}, \dots, \xi_{s_\psi}]^\top := \rho_s^{-1} e_s, \quad (2.32)$$

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

$$\varepsilon_s := \varepsilon_s(\xi_s) := [\varepsilon_{s_x}, \dots, \varepsilon_{s_\psi}]^\top := \left[ \ln \left( \frac{1+\xi_{s_x}}{1-\xi_{s_x}} \right), \dots, \ln \left( \frac{1+\xi_{s_\psi}}{1-\xi_{s_\psi}} \right) \right]^\top \quad (2.33)$$

$$\begin{aligned} r_s := r_s(\xi_s) &:= \text{diag}\{[r_{s_k}(\xi_{s_k})]_{k \in \mathcal{K}}\} := \text{diag} \left\{ \left[ \frac{\partial \varepsilon_{s_k}}{\partial \xi_{s_k}} \right]_{k \in \mathcal{K}} \right\} \\ &= \text{diag} \left\{ \left[ \frac{2}{1 - \xi_{s_k}^2} \right]_{k \in \mathcal{K}} \right\}, \end{aligned} \quad (2.34)$$

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

$$v_r := v_r(\xi_s, t) := -g_s J_O \left( \eta_d(t) + \rho_{s_\eta}(t) \xi_{s_\eta} \right)^{-1} \rho_s^{-1} r_s \varepsilon_s, \quad (2.35)$$

where  $\rho_{s_\eta} := \rho_{s_\eta}(t) := \text{diag}\{\rho_{s_\phi}, \rho_{s_\theta}, \rho_{s_\psi}\}$ ,  $\xi_{s_\eta} := [\xi_{s_\phi}, \xi_{s_\eta}, \xi_{s_\psi}]^\top$ , and we have further used the relation  $\xi_s = \rho_s^{-1}(x_O - x_d)$  from (2.29) and (2.32).

**Step II-a.** Define the velocity error vector

$$e_v := [e_{v_x}, \dots, e_{v_\psi}]^\top := v_O - v_r, \quad (2.36)$$

and select the corresponding positive performance functions  $\rho_{v_k} := \rho_{v_k}(t) : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{>0}$  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  $\alpha$  is an arbitrary positive constant.

**Step II-b.** Define the normalized velocity error

$$\xi_v := [\xi_{v_x}, \dots, \xi_{v_\psi}]^\top := \rho_v^{-1} e_v, \quad (2.37)$$

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

$$\begin{aligned} \varepsilon_v := \varepsilon_v(\xi_v) &:= [\varepsilon_{v_x}, \dots, \varepsilon_{v_\psi}]^\top := \left[ \ln \left( \frac{1+\xi_{v_x}}{1-\xi_{v_x}} \right), \dots, \ln \left( \frac{1+\xi_{v_\psi}}{1-\xi_{v_\psi}} \right) \right]^\top \\ r_v(\xi_v) &:= \text{diag}\{[r_{v_k}(\xi_{v_k})]_{k \in \mathcal{K}}\} := \text{diag} \left\{ \left[ \frac{\partial \varepsilon_{v_k}}{\partial \xi_{v_k}} \right]_{k \in \mathcal{K}} \right\} \\ &= \text{diag} \left\{ \left[ \frac{2}{1 - \xi_{v_k}^2} \right]_{k \in \mathcal{K}} \right\}, \end{aligned} \quad (2.38)$$

and design the decentralized feedback control protocol for each agent  $i \in \mathcal{N}$  as  $u_i : \mathbb{S}_i \times (-1, 1)^6 \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^6$ , with

$$u_i := u_i(q_i, \xi_v, t) := -g_v J_{M_i}(q_i) \rho_v^{-1} r_v \varepsilon_v, \quad (2.39)$$

where  $g_v$  is a positive constant gain and  $J_{M_i}$  as defined in (2.24). The control laws (2.39) can be written in vector form  $u := [u_1^\top, \dots, u_N^\top]^\top$ , with:

$$u = -g_v G_M^+(q) \rho_v^{-1} r_v \varepsilon_v. \quad (2.40)$$

**Remark 2.4 (Decentralized manner and robustness (PPC)).** Similarly to (2.26), notice from (2.39) that 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. The terms  $l_k$ ,  $\rho_{k,0}$ ,  $\rho_{k,\infty}$ ,  $\alpha$ ,  $l_{v_k}$ , and  $\rho_{v_k,\infty}$ ,  $k \in \mathcal{K}$  needed for the calculation of the performance functions can be transmitted off-line to the agents. Moreover, the Prescribed Performance Control protocol is also robust to uncertainties of model uncertainties and external disturbances. In particular, note that the control laws do not even require the structure of the terms  $\tilde{M}, \tilde{C}, \tilde{g}, \tilde{d}$ , but only the positive definiteness of  $\tilde{M}$ , as will be observed in the subsequent proof of Theorem 2.2. It is worth noting that, in the case that one or more agent failed to participate in the task, then the remaining agents would need to appropriately update their control protocols (e.g., update  $J_{M_i}$ ) to compensate for the failure.

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

**Theorem 2.2.** Consider  $N$  agents rigidly grasping an object with unknown coupled dynamics (2.14). Then, under Assumptions 2.1, 2.3, 2.5, the decentralized control protocol (2.32)-(2.39) 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 initial conditions satisfying  $|\theta_O(0) - \theta_d(0)| < \theta^*$  (from **Step I-a (i)**), with all closed loop signals being bounded.

*Proof.* The proof consists of two main parts. Firstly, we prove that there exists a maximal solution  $(\xi_s(t), \xi_v(t)) \in (-1, 1)^{12}$  for  $t \in [0, \tau_{\max})$ , where  $\tau_{\max} > 0$ . Secondly, we prove that  $(\xi_s(t), \xi_v(t))$  is contained in a compact subset of  $(-1, 1)^{12}$  and consequently, that  $\tau_{\max} = \infty$ . Without loss of generality, we assume that  $v_O(0) = 0$ .

Part A: Consider the combined state  $\sigma := [q, \xi_s, \xi_v] \in \mathsf{S} \times \mathbb{R}^{12}$ . Differentiation of  $\sigma$  yields, in view of (2.9), (2.32) and (2.37)

$$\dot{\sigma} = \begin{bmatrix} \tilde{J}G^\top v_O \\ \rho_s^{-1}(\dot{x}_O - \dot{x}_d - \dot{\rho}_s \xi_s) \\ \rho_v^{-1}(\dot{v}_O - \dot{v}_r - \dot{\rho}_v \xi_v), \end{bmatrix}, \quad (2.41)$$

where  $\tilde{J} := \tilde{J}(q) := \text{diag}\{[J_i(q_i)^\top (J_i(q_i) J_i(q_i)^\top)^{-1}]_{i \in \mathcal{N}}\} \in \mathbb{R}^{n \times 6N}$  is well defined due to Assumption 2.3. Then, by employing (2.5), (2.29), (2.32),

and (2.35)-(2.40) as well as  $GG_M^+ = I_6$ , we can express the right-hand side of (2.41) as a function of  $\sigma$  and  $t$ , i.e.,

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

with

$$\begin{aligned} f_{\text{cl},q}(\sigma, t) &:= \tilde{J}(q)G(q)^\top (\rho_v(t)\xi_v + v_r(\xi_s, t)) \\ f_{\text{cl},s}(\sigma, t) &:= \rho_s(t)^{-1} [J_O(\eta_d(t) + \rho_{s_\eta}(t)\xi_{s_\eta})\rho_v(t)\xi_v - \dot{\rho}_s(t)\xi_s \\ &\quad - g_s\rho_s(t)^{-1}r_s(\xi_s)\varepsilon_s(\xi_s) - \dot{x}_d(t)] \\ f_{\text{cl},v}(\sigma, t) &:= -\rho_v(t)^{-1} \left( \widetilde{M}(x(\sigma, t)) \left[ \widetilde{C}(x(\sigma, t))(\rho_v(t)\xi_v + v_r(\xi_s, t)) \right. \right. \\ &\quad \left. \left. + \tilde{g}(x(\sigma, t)) + \tilde{d}(x(\sigma, t), t) + g_v\rho_v(t)^{-1}r_v(\xi_v)\varepsilon_v(\xi_v) \right] - \dot{\rho}_v(t)\xi_v \right. \\ &\quad \left. + \frac{\partial v_r(\xi_s, t)}{\partial t} + \frac{\partial v_r(\xi_s, t)}{\partial \xi_s}f_{\text{cl},s}(\sigma, t) \right), \end{aligned}$$

and we also express  $x$  as a function of  $\sigma$  and  $t$  via

$$x(\sigma, t) = \begin{bmatrix} q \\ \dot{q} \\ \eta_O \\ \omega_O \end{bmatrix} = \begin{bmatrix} q \\ f_{\text{cl},q}(\sigma, t) \\ \eta_d(t) + \rho_{s_\eta}(t)\xi_{s_\eta} \\ (\rho_v(t)\xi_v + v_r(\xi_s, t))_{3:6} \end{bmatrix}$$

where  $(\cdot)_{3:6}$  denotes the three last components of the vector. Consider now the open and nonempty set  $\Omega := S \times (-1, 1)^{12}$ . 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  $|\theta_O(0) - \theta_d(0)| < \theta^*$  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  $f_{\text{cl}} : \Omega \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^{n+12}$  is locally Lipschitz in  $\sigma$  over the set  $\Omega$  and continuous and locally integrable in  $t$  for each fixed  $\sigma \in \Omega$ . Therefore, the hypotheses of Theorem A.1 in Appendix A hold and the existence of a maximal solution  $\sigma : [0, \tau_{\max}) \rightarrow \Omega$ , for  $\tau_{\max} > 0$ , is ensured. We thus conclude

$$\xi_{s_k}(t), \xi_{v_k}(t) \in (-1, 1) \tag{2.42}$$

$\forall k \in \mathcal{K}, t \in [0, \tau_{\max})$ , which also implies that  $\|\xi_s(t)\| < \sqrt{6}$ , and  $\|\xi_v(t)\| < \sqrt{6}, \forall t \in [0, \tau_{\max})$ . In the following, we show the boundedness of all closed loop signals and  $\tau_{\max} = \infty$ .

Part B: Note first from (2.42), that  $|\theta_O(t) - \theta_d(t)| < \rho_\theta(t) \leq \rho_\theta(0) = \theta^*$ , which, since  $\theta_d(t) \in [-\bar{\theta}, \bar{\theta}], \forall t \in \mathbb{R}_{\geq 0}$ , implies that  $|\theta_O(t)| \leq \bar{\theta} := \bar{\theta} + \theta^* < \frac{\pi}{2}, \forall t \in [0, \tau_{\max}]$ . Therefore, by employing (2.6), one obtains that,  $\forall t \in [0, \tau_{\max}]$ ,

$$\|J_O(\eta_O(t))\| \leq \bar{J}_O := \sqrt{\frac{|\sin(\bar{\theta})| + 1}{1 - \sin^2(\bar{\theta})}} < \infty. \quad (2.43)$$

Consider now the positive definite function  $V_s := \frac{1}{2}\|\varepsilon_s\|^2$ . Differentiating  $V_s$  along the solutions of the closed loop system yields  $\dot{V}_s = \varepsilon_s^\top r_s \rho_s^{-1} \dot{\xi}_s$ , which, in view of (2.41), (2.37), (2.35) and the fact that  $\dot{x}_O = J_O(\eta_O)v_O = J_O(\eta_O)(v_r + e_v)$ , becomes

$$\begin{aligned} \dot{V}_s &= -g_s \|\rho_s^{-1} r_s \varepsilon_s\|^2 - \varepsilon_s^\top r_s \rho_s^{-1} (\dot{x}_d + \dot{\rho}_s \xi_s - J_O e_v) \\ &\leq -g_s \|\rho_s^{-1} r_s \varepsilon_s\|^2 + \|\rho_s^{-1} r_s \varepsilon_s\| (\|\dot{x}_d\| + \|J_O \rho_v \xi_v\| + \|\dot{\rho}_s \xi_s\|). \end{aligned}$$

In view of (2.43), (2.42), 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  $\dot{x}_d$ , the last inequality becomes

$$\dot{V}_s \leq -g_s \|\rho_s^{-1} r_s \varepsilon_s\|^2 + \|\rho_s^{-1} r_s \varepsilon_s\| \bar{B}_s,$$

$\forall t \in [0, \tau_{\max}]$ , with

$$\bar{B}_s := \sqrt{6} \bar{J}_O (\|v_r(0)\| + \alpha) + \sup_{t>0} \|\dot{x}_d(t)\| + \sqrt{6} \max_{k \in \mathcal{K}} \{l_k(\rho_{s_k,0} - \rho_{s_k,\infty})\},$$

independent of  $\tau_{\max}$ . Therefore,  $\dot{V}_s$  is negative when  $\|\rho_s^{-1} r_s \varepsilon_s\| > \frac{\bar{B}_s}{g_s}$ , which, by employing (2.34), the decreasing property of  $\rho_{s_k}, k \in \mathcal{K}$  as well as (2.42), is satisfied when  $\|\varepsilon_s\| > \frac{\max_{k \in \mathcal{K}} \{\rho_{s_k,0}\} \bar{B}_s}{2g_s}$ . Hence, by using Theorem A.5 of Appendix A, we conclude that

$$\|\varepsilon_s(\xi_s(t))\| \leq \bar{\varepsilon}_s := \max \left\{ \|\varepsilon_s(0)\|, \frac{\max_{k \in \mathcal{K}} \{\rho_{s_k,0}\} \bar{B}_s}{2g_s} \right\}, \quad (2.44)$$

$\forall t \in [0, \tau_{\max}]$ . Furthermore, since  $|\varepsilon_{s_k}| \leq \|\varepsilon_s\|, \forall k \in \mathcal{K}$ , taking the inverse logarithm function from (2.33), we obtain

$$-1 < \frac{\exp(-\bar{\varepsilon}_s) - 1}{\exp(-\bar{\varepsilon}_s) + 1} =: -\bar{\xi}_s \leq \xi_{s_k}(t) \leq \bar{\xi}_s := \frac{\exp(\bar{\varepsilon}_s) - 1}{\exp(\bar{\varepsilon}_s) + 1} < 1, \quad (2.45)$$

$\forall t \in [0, \tau_{\max}]$ . Hence, recalling (2.34) and (2.35), we obtain the boundedness of  $r_s(\xi_s(t)), v_r(t), \forall t \in [0, \tau_{\max}]$ , and in view of  $v_O = v_r + e_v$ , (2.36), (2.42),

(2.9) and (2.12), the boundedness of  $v_O(t)$  and  $v_i(t)$  as

$$\begin{aligned} \|r_s(\xi_s(t))\| &\leq \bar{r}_s := \frac{2}{1 - \bar{\xi}_s^2} = \frac{(\exp(\bar{\varepsilon}_s) + 1)^2}{2 \exp(\bar{\varepsilon}_s)}, \\ \|v_r(t)\| &\leq \bar{v}_r := g_s \sqrt{2} \frac{\bar{\varepsilon}_s (\exp(\bar{\varepsilon}_s) + 1)^2}{2 \min_{k \in \mathcal{K}} \{\rho_{s_k, \infty}\} \exp(\bar{\varepsilon}_s)} \\ \|v_O(t)\| &\leq \bar{v}_O := \bar{v}_r + \sqrt{6} \max_{k \in \mathcal{K}} \{\rho_{v_k, 0}\} \\ \|v_i(t)\| &\leq \bar{v}_i := (\|p_{O/E_i}^{E_i}\| + 1) \bar{v}_O, \forall i \in \mathcal{N}, \end{aligned} \quad (2.46)$$

$\forall t \in [0, \tau_{\max})$ , From (2.45), (2.5a), and (2.29) we also conclude the boundedness of  $x_O(t)$ ,  $\dot{x}_O(t)$ , as

$$\begin{aligned} \|x_O(t)\| &\leq \bar{x}_O := \sup_{t>0} \|x_d(t)\| + \sqrt{6} \xi_s \max_{k \in \mathcal{K}} \{\rho_{s_k, 0}\}, \\ \|\dot{x}_O(t)\| &\leq \bar{J}_O \bar{v}_O, \end{aligned}$$

$\forall t \in [0, \tau_{\max})$ . The coupled kinematics (2.8) and Assumption 2.3 imply also the boundedness of  $p_{E_i}(t)$ ,  $q_i(t)$ , and  $\dot{q}_i(t)$ ,  $\forall i \in \mathcal{N}$ , as  $\|q(t)\| \leq \bar{q}$ ,  $\|\dot{q}(t)\| \leq \bar{J} \|v\| \leq \bar{J} \sum_{i \in \mathcal{N}} \bar{v}_i$  for a positive constant  $\bar{J}$ ,  $[0, \tau_{\max})$ . Hence, we conclude that

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

$[0, \tau_{\max})$ . In a similar vein, by differentiating the reference velocity (2.35) and using (2.33), (2.34), and (2.44), we also conclude the boundedness of  $\dot{v}_r(t)$  by a positive constant  $\bar{v}_r$ ,  $\forall t \in [0, \tau_{\max})$ .

Applying the aforementioned line of proof, we consider the positive definite function  $V_v := \frac{1}{2} \|\varepsilon_v\|^2$ . By differentiating  $V_v$  we obtain  $\dot{V}_v = \varepsilon_v^\top r_v \rho_v^{-1} \dot{\xi}_v$ , which, in view of (2.41), (2.36), (2.14), becomes

$$\begin{aligned} \dot{V}_v &= -g_v \varepsilon_v^\top r_v \rho_v^{-1} \widetilde{M} \rho_v^{-1} r_v \varepsilon_v + \varepsilon_v^\top r_v \rho_v^{-1} \left( -\dot{\rho}_v \xi_v - \widetilde{M} \left[ \widetilde{C}(\rho_v \xi_v + v_r) + \widetilde{g} \right. \right. \\ &\quad \left. \left. + \widetilde{d} \right] - \dot{v}_r \right). \end{aligned} \quad (2.47)$$

Invoking Assumption 2.5 and the boundedness of  $q_i(t)$ ,  $\dot{q}_i(t)$ ,  $x_O(t)$ ,  $\dot{x}_O(t)$ ,  $\forall t \in [0, \tau_{\max})$ , we conclude the boundedness of  $d_O(x_O(t), \dot{x}_O(t), t)$  and  $d_i(q_i(t), \dot{q}_i(t), t)$  by positive finite constants  $\underline{d}'_O$ ,  $\underline{d}'_i$ ,  $\forall i \in \mathcal{N}$ , respectively,  $\forall t \in [0, \tau_{\max})$ . Hence, from (2.12) and (2.14), we also obtain the boundedness of  $\widetilde{d}(x(t))$  as

$$\|\widetilde{d}(x(t))\| \leq \underline{d} := \underline{d}'_O + \sum_{i \in \mathcal{N}} \{\|p_{O/E_i}^{E_i}\| + 1\} \underline{d}'_i.$$

In addition, the continuity of  $\tilde{C}(x), \tilde{g}(x)$  implies the existence of positive and finite constant  $\bar{c}, \bar{g}$  such that  $\|\tilde{C}(x(t))\| \leq \bar{c}, \|\tilde{g}(x(t))\| \leq \bar{g}, \forall t \in [0, \tau_{\max}]$ .

Thus, by combining the aforementioned discussion with the boundedness of  $\dot{v}_r$ , the positive definitiveness and boundedness of  $\tilde{M}(x)$ , (2.16) and (2.42), we obtain from (2.47)

$$\dot{V}_v \leq -g_v \underline{m} \|\rho_v^{-1} r_v \varepsilon_v\|^2 + \|\rho_v^{-1} r_v \varepsilon_v\| \bar{B}_v,$$

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

$$\bar{B}_v := \sqrt{6} \max_{k \in \mathcal{K}} \{l_{v_k}(\rho_{v_k,0} - \rho_{v_k,\infty})\} + \bar{v}_r + \bar{m}(\bar{g} + \underline{d} + \bar{c}(\bar{v}_r + \sqrt{6}(\|v_r(0)\| + \alpha)))$$

is a positive and finite constant, independent of  $\tau_{\max}$ .

By proceeding similarly as with  $\dot{V}_s$ , we conclude that

$$\|\varepsilon_v(\xi_v(t))\| \leq \bar{\varepsilon}_v := \max \left\{ \|\varepsilon_v(0)\|, \frac{\max_{k \in \mathcal{K}} \{\rho_{v_k,0}\} \bar{B}_v}{2g_v \underline{m}} \right\}, \quad (2.48)$$

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

$$-1 < \frac{\exp(-\bar{\varepsilon}_v) - 1}{\exp(-\bar{\varepsilon}_v) + 1} =: -\bar{\xi}_v \leq \xi_{v_k}(t) \leq \bar{\xi}_v := \frac{\exp(\bar{\varepsilon}_v) - 1}{\exp(\bar{\varepsilon}_v) + 1} < 1, \quad (2.49)$$

$\forall t \in [0, \tau_{\max}]$ . In view of (2.38), (2.39), this also implies

$$\begin{aligned} \|r_v(\xi_v(t))\| &\leq \bar{r}_v := \frac{2}{1 - \bar{\xi}_v^2} = \frac{(\exp(\bar{\varepsilon}_v) + 1)^2}{2 \exp(\bar{\varepsilon}_v)}, \\ \|u_i(t)\| &\leq \bar{u}_i := g_v \bar{J}_{M_i} \max_{k \in \mathcal{K}} \{\rho_{v_k,\infty}^{-1}\} \bar{r}_v \bar{\varepsilon}_v, \end{aligned} \quad (2.50)$$

$\forall t \in [0, \tau_{\max}]$ , where  $\bar{J}_{M_i}$  is an upper bound of  $\|J_{M_i}(q_i)\|$ , which can be proven to be independent of  $q$ .

What remains to be shown is that  $\tau_{\max} = \infty$ . We can conclude from the aforementioned analysis, Assumption 2.3 and (2.45), (2.49) that the solution  $\sigma(t)$  remains in a compact subset  $\Omega'$  of  $\Omega$ ,  $\forall t \in [0, \tau_{\max}]$ , namely  $\sigma(t) \in \Omega', \forall t \in [0, \tau_{\max}]$ . Hence, according to Theorem A.2 of Appendix A, it holds that  $\tau_{\max} = \infty$ . 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 (2.45) by  $\rho_k(t), k \in \mathcal{K}$ , we obtain

$$-\rho_{s_k}(t) < -\bar{\xi}_s \rho_{s_k}(t) \leq e_{s_k}(t) \leq \bar{\xi}_s \rho_{s_k}(t) < \rho_{s_k}(t), \quad (2.51)$$

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

**Remark 2.5 (Prescribed Performance).** 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  $\varepsilon_s, \varepsilon_v$  arbitrarily small by adopting extreme values of the control gains  $g_s$  and  $g_v$  (see (2.44) and (2.48)). More specifically, notice that (2.45) and (2.49) 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 (2.14) can be compensated, as they affect only the size of  $\varepsilon_v$  through  $\bar{B}_v$ , but leave unaltered the achieved stability properties. Hence, the actual performance given in (2.51), 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.

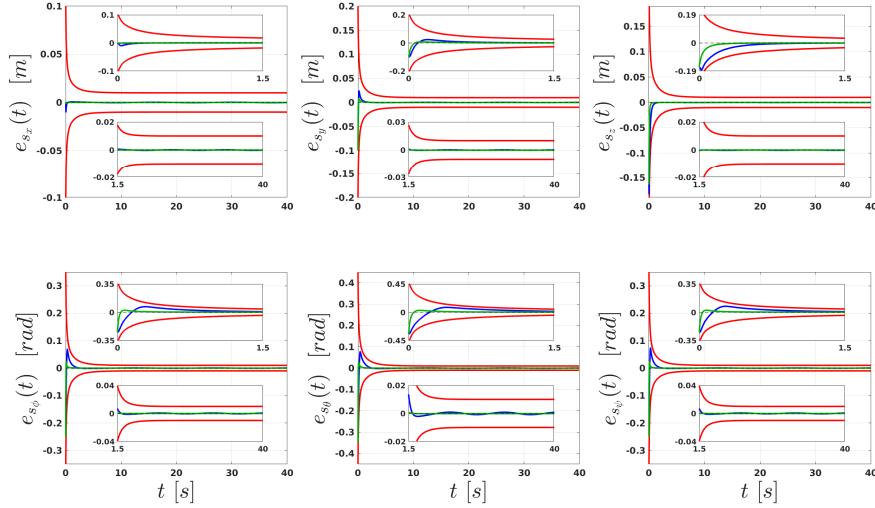
**Remark 2.6 (Control Input Bounds).** The aforementioned analysis of the Prescribed Performance Control methodology reveals the derivation of bounds for the velocity  $v_i$  and control input  $u_i$  of each agent. Note the explicit bounds  $\bar{v}_i$  and  $\bar{u}_i$  for  $v_i$  and  $u_i$  (see (2.46), (2.50)), respectively, which depend on the control gains, the bounds of the dynamic terms, the desired trajectory, and the performance functions. Therefore, given desired bounds for the agents' velocity  $\bar{v}_{i,b}$  and input  $\bar{u}_{i,b}$  (derived from bounds on the joint velocities and torques  $\dot{q}_i, \tau_i$ , respectively) and that the upper bounds of the dynamic terms are known, we can tune appropriately the control gain  $g_s, g_v$  as well as the parameters  $\rho_{s_k,0}, \rho_{v_k,0}, \rho_{s_k,\infty}, \rho_{v_k,\infty}, l_{s_k}, l_{v_k}$  in order to achieve  $\bar{v}_i \leq \bar{v}_{i,b}, \bar{u}_i \leq \bar{u}_{i,b}, \forall i \in \mathcal{N}$ . It is also worth noting that the selection of the control gains  $g_s, g_v$  affects the evolution of the errors  $e_s, e_v$  inside the corresponding performance envelopes.

**Remark 2.7 (Internal forces).** The internal forces were proven, in [?], to be regulated to zero using the distribution matrix  $J_{M_i}$  from (2.24). That result, however, did not take into account the actual dynamic parameters of the robots. In the next chapter we analyze the internal forces in rigid cooperative manipulation and provide conditions that achieve their regulation to zero.

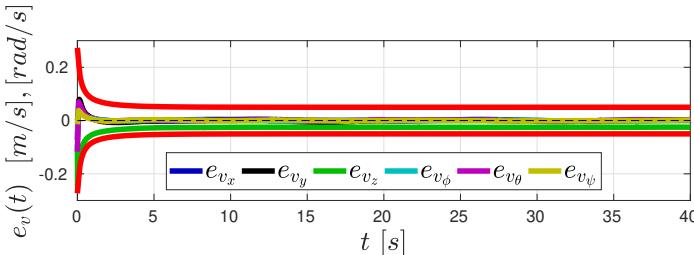
## Simulation Results

We provide here simulation results for the developed control scheme.

The tested scenario is identical to the one used for the adaptive control scheme of Section 2.2.3, with the modification of  $A_\theta = \frac{\pi}{9}$ , in order to avoid  $\theta_d(t) = \pm\frac{\pi}{2}$ . We set the performance functions as  $\rho_{s_k}(t) = (|e_{s_k}(0)| + 0.09) \exp(-0.5t) + 0.01$ ,  $\rho_{v_k}(t) = (|e_{v_k}(0)| + 0.95) \exp(-0.5t) + 0.05$ ,  $\forall k \in \mathcal{K}$ , and the control gains of (2.35), (2.39) as  $g_s = 0.005$ ,  $g_v = 10$ , respectively, by following the bounds derived in the previous section and considering known



**Figure 2.8:** Simulation results for the controller of Section 2.2.4, with (in blue) and without (in green) taking into account input constraints; Top: The position errors  $e_{sx}(t)$ ,  $e_{sy}(t)$ ,  $e_{sz}(t)$  (with blue and green, respectively) along with the respective performance functions (with red); Bottom: The orientation errors  $e_{s\phi}(t)$ ,  $e_{s\theta}(t)$ ,  $e_{s\psi}(t)$  (with blue and green, respectively) along with the respective performance functions (with red),  $\forall t \in [0, 40]$ . Zoomed versions of the transient and steady-state response have been included for all plots.

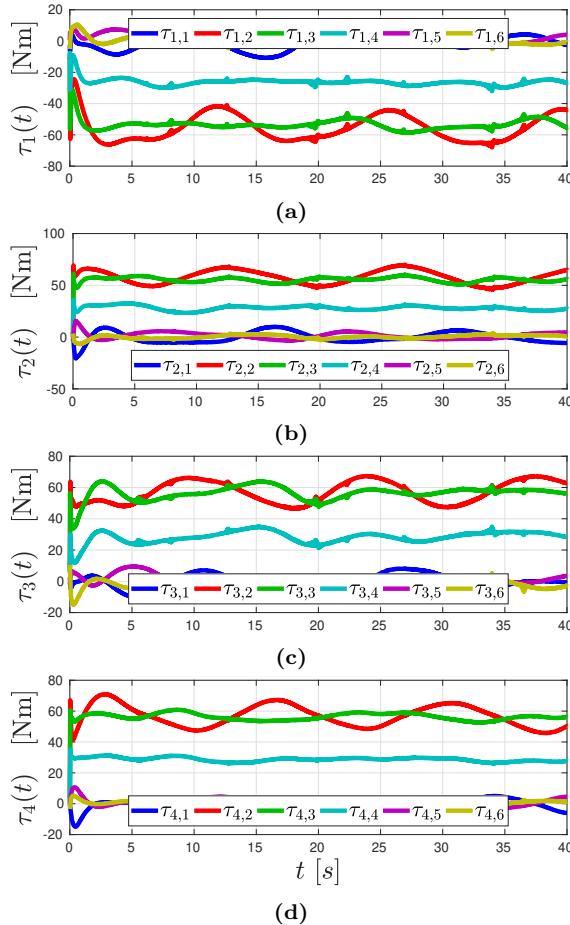


**Figure 2.9:** The velocity errors  $e_v(t)$  along with the respective performance functions (with red) for the controller of Section 2.2.4,  $\forall t \in [0, 40]$ .

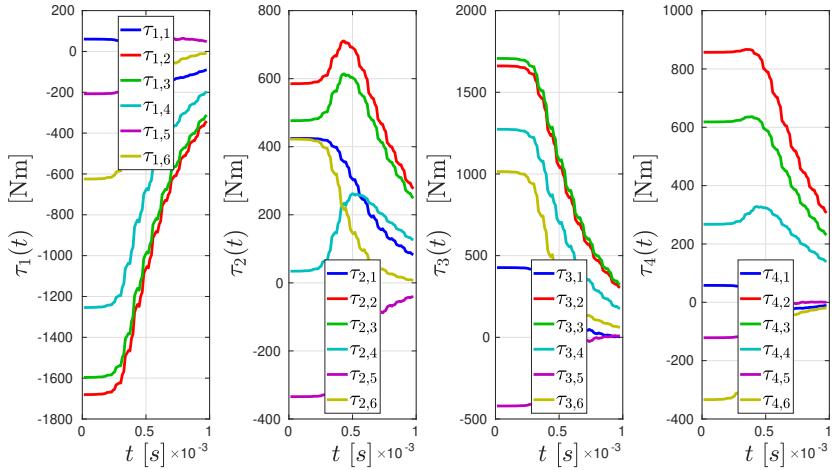
dynamic bounds. The simulation results are depicted in Figs. 2.8[2.10], for  $t \in [0, 40]$  seconds. In particular, Fig. 2.8 depicts the evolution of the pose errors  $e_s(t)$  (in blue), along with the respective performance functions  $\rho_s(t)$  (in red), Fig. 2.9 depicts the evolution of the velocity errors  $e_v(t)$ , along with the respective performance functions  $\rho_v(t)$ , and Fig. 2.10 shows the resulting joint torques  $\tau_i(t)$ ,  $\forall i \in \{1, \dots, 4\}$ . One can conclude from the aforementioned figures that the simulation results verify the theoretical findings, since the errors  $e_s(t)$ ,  $e_v(t)$  stay confined in the performance function funnels. Moreover, the joint torques in respect the saturation values we set. For comparison purposes, we also simulate the same system without taking into account any input constraints. In order to achieve good performance in terms of overshoot, rise, and settling time, we set the control gains as  $g_s = 1$ ,  $g_v = 200$ . The resulting pose errors are depicted in Fig. 2.8 for  $t \in [0, 40]$  seconds (with green) along with the performance functions (with red), and the resulting torques are depicted in Fig. 2.11 for  $t \in [0, 0.001]$  seconds. This small time interval is sufficient to observe the high-value initial peaks of the torque inputs that do not satisfy the desired constraint of  $\bar{\tau} = 150$  Nm, which can be attributed to the lack of gain calibration. Nevertheless, note also the better performance of the pose errors, in terms of overshoot, rise and settling time, as pictured in Fig. 2.8. Finally, note that any Prescribed Performance Control methodology would fail to solve Problem 2.1 with  $\theta(0) = \frac{\pi}{2}$  or  $\theta_d(t) = \frac{\pi}{2}$  for some  $t \in \mathbb{R}_{\geq 0}$ , in contrast to the adaptive quaternion-feedback control scheme of Section 2.2.3. The simulations were carried out in the MATLAB R2017a environment on a i7-5600 laptop computer at 2.6Hz, with 8GB of RAM.

## Experimental Results

We provide here experimental results for the developed Prescribed Performance Control scheme. The scenario here is identical to the one used for Section 2.2.3. We set the performance functions 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], and the control gains of (2.35) and (2.39) as  $g_s = 0.05$  and  $g_v = 10$ , respectively. The experimental results are depicted in Fig. 2.12[2.13] for  $t \in [0, 70]$  seconds. In particular, Fig. 2.12 shows the pose and velocity errors  $e_s(t)$ ,  $e_v(t)$  along with the respective performance functions, and Fig. 2.13 depicts the joint torques  $\tau_1(t)$ ,  $\tau_2(t)$  of the agents. We can conclude that the experimental results verify the theoretical analysis, since the errors evolve strictly within the prespecified performance bounds. Note also that the joint torques respect the saturation limits. A video illustrating the simulation and experimental results (along with the ones of the previous section's control scheme) can be found on <https://youtu.be/jJWeI5ZvQPY>.



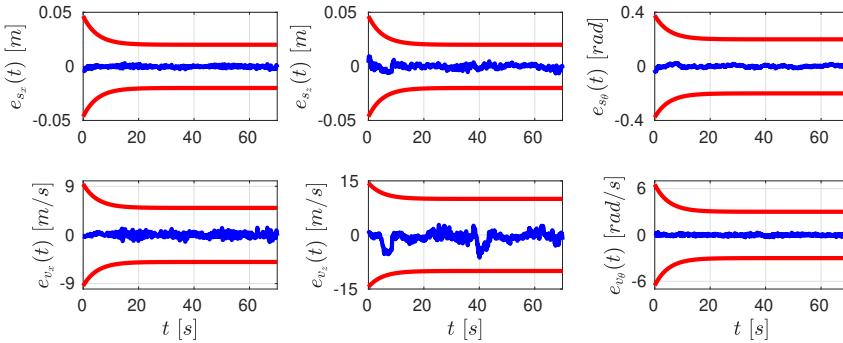
**Figure 2.10:** The agents' joint torques  $\tau_i(t)$ ,  $i \in \mathcal{N}$ , (in (a)-(d), respectively) of the control scheme of Section 2.2.4  $\forall t \in [0, 40]$  by taking into account input constraints.



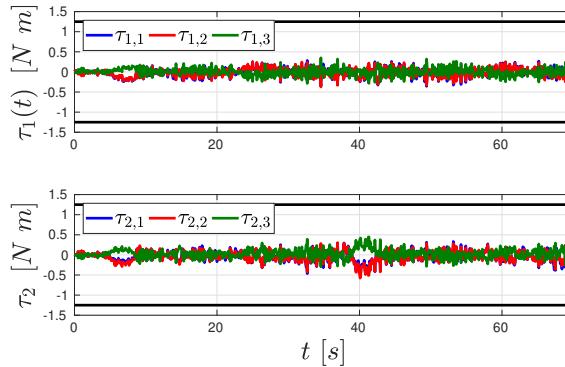
**Figure 2.11:** The agents' joint torques  $\tau_i(t)$ ,  $i \in \mathcal{N}$ , (in (a)-(d), respectively) of the control scheme of Section 2.2.4  $\forall t \in [0, 40]$  without taking into account input constraints,  $\forall t \in [0, 0.001]$ .

## 2.2.5 Discussion

In view of the aforementioned results, we mention some worth-noting differences between the two control schemes. Firstly, note that the PPC methodology allows for *exponential* convergence of the errors to the set defined by the values  $\rho_{s_k, \infty}$ ,  $\rho_{v_k, \infty}$ , achieving *predefined* transient and steady-state performance, without the need to resort to tuning of the control gains. The adaptive quaternion-feedback methodology, however, can only guarantee that the errors converge asymptotically to zero as  $t \rightarrow \infty$ . This is verified by the simulation results, where the error trajectories  $e_p(t)$ ,  $e_\zeta(t)$  and  $e_v(t)$  show an oscillatory behavior. Improvement of such performance (in terms of overshoot, rise, and settling time) would require appropriate gain tuning. Secondly, note that, as shown in the simulations section, the quaternion-feedback methodology allows for trajectories where the pitch angle of the object ( $\theta_O$ ) can be  $\pm 90$  degrees, in contrast to the PPC methodology, where that configuration is ill-posed, since the matrix  $J_O(\eta_O)$  is not defined. Finally, the adaptive quaternion-feedback methodology can be considered less robust to modeling uncertainties in real-time scenarios, since it accounts only for parametric uncertainties (the unknown terms  $\vartheta_i$ ,  $\vartheta_O$ ,  $d_i$ ,  $d_O$ ), assuming a *known* structure of the dynamic terms. The PPC methodology, however, does not require any information of the structure or the parameters of the dynamic model (note that the only requirements are the positive definiteness of the coupled inertia matrix, the locally Lipschitz and continuity properties of the dynamic terms and the boundedness - with respect to time - of the



**Figure 2.12:** Experimental results for the controller of Section 2.2.4: Top: the pose errors  $e_{s_x}(t)$ ,  $e_{s_z}(t)$ ,  $e_{s_\theta}(t)$  (with blue) along with the respective performance functions (with red); Bottom: The velocity errors  $e_{v_x}(t)$ ,  $e_{v_z}(t)$ ,  $e_{v_\theta}(t)$  (with blue) along with the respective performance functions (with red),  $\forall t \in [0, 70]$ .



**Figure 2.13:** The agents' joint torques of the experiment of the controller in Section 2.2.4,  $\forall t \in [0, 70]$ , with their respective limits (with black).

disturbances  $d_i, d_O$ ). In that sense, one would expect the PPC methodology to perform better in real-time experiments, where unmodeled dynamics are involved. The fact, however, that PPC is a control scheme that does not contain any information of the model structure makes it more difficult to tune (in terms of gain tuning) in order to achieve robot velocities and torques that respect specific bounds, especially when the bounds of the dynamic terms are unknown. This has been noticed during both simulations and experiments.

## 2.2.6 Problem Statement - Constrained Transportation

We deal here with a slightly different problem, that is, the problem of cooperatively transporting an object to a desired *constant* pose, while complying to certain constraints. Such constraints consist of inter-robot collision avoidance, collision avoidance with obstacles, singularity avoidance, as well as robot velocity and torque saturation constraints.

Consider  $Z \in \mathbb{N}$  obstacles  $\mathcal{O}_z \subset \mathbb{R}^3$ ,  $z \in \mathcal{Z} := \{1, \dots, Z\}$  and denote by  $\mathcal{A}_i(q_i) \subset \mathbb{R}^3$ ,  $i \in \mathcal{N}$ ,  $\mathcal{A}_o(x_o) \subset \mathbb{R}^3$  the physical volumes occupied by agent  $i$ , at state  $q_i$ ,  $i \in \mathcal{N}$ , and the object, at state  $x_o$ , respectively.

**Remark 2.8.** As mentioned before, since the geometric object parameters  $p_{E_i/o}^{E_i}$  and  $\eta_{E_i/o}$  are known, each agent can compute  $p_o, \eta_o$  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  $q$ .

We can now formulate the problem considered here:

**Problem 2.2.** Consider  $N$  robotic agents rigidly grasping an object, governed by the coupled dynamics (2.14). Given a desired *constant* pose  $x_d := [(p_d)^\top, (\eta_d)^\top]^\top$ ,  $p_d \in \mathbb{R}^3$ ,  $\eta_d := [\varphi_d, \theta_d, \psi_d] \in \mathbb{T}$ , with  $\theta_d \in [-\bar{\theta}, \bar{\theta}] \subset (-\frac{\pi}{2}, \frac{\pi}{2})$ , design the control input  $u \in \mathbb{R}^{6N}$  such that  $\lim_{t \rightarrow \infty} x_o(t) = x_d$ , 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.  $\mathcal{A}_o(x_o(t)) \cap \mathcal{O}_z = \emptyset, \forall z \in \mathcal{Z}$ ,
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.  $q_i(t) \in \mathcal{S}_i$ ,

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

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 2.7.** (Problem feasibility) The set  $\{(q, x_o) \in \mathbb{R}^n \times \mathbb{M} : \mathcal{A}_i(q_i) \cap \mathcal{O}_z = \emptyset, \mathcal{A}_i(q_i) \cap \mathcal{A}_j(q_j) = \emptyset, \mathcal{A}_o(x_o) \cap \mathcal{O}_z = \emptyset, \forall i, j \in \mathcal{N}, i \neq j, z \in \mathcal{Z}\}$ , is connected.

We also define the following sets:

$$\begin{aligned}\mathcal{S}_{i,o} &:= \{q_i \in \mathbb{R}^{n_i} : \mathcal{A}_i(q_i) \cap \mathcal{O}_z = \emptyset, \forall z \in \mathcal{Z}\}, \quad \forall i \in \mathcal{N} \\ \mathcal{S}_A &:= \{q \in \mathbb{R}^n : \mathcal{A}_i(q_i) \cap \mathcal{A}_j(q_j) = \emptyset, \forall i, j \in \mathcal{N}, i \neq j\}, \\ \mathcal{S}_o &:= \{x_o \in \mathbb{M} : \mathcal{A}_o(x_o) \cap \mathcal{O}_z = \emptyset\}.\end{aligned}$$

associated with the desired collision-avoidance properties.

We present next two control schemes, based on Nonlinear Model Predictive Control (NMPC), for the solution of Problem 2.2. The first one is a *centralized scheme*, where a central computer unit (e.g., on one of the robotic agents) has global feedback and computes the control input of the entire team. Secondly, we develop a *decentralized* scheme, where each robotic agent computes its own control signal. The latter is based on a leader-follower coordination as well as inter-agent communication.

We also assume that  $d_i(\cdot) = d_o(\cdot) = 0, \forall i \in \mathcal{N}$  in the dynamics (2.3), (2.5), and that the system model is *accurately known*. Potential uncertainties could be taken into account by using robust variations of NMPC, like, e.g., tube-based NMPC [? ].

### 2.2.7 Centralized NMPC

In this section, a centralized systematic solution to Problem 2.2 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. [? ? ? ? ? ? ? ? ? ?]) has 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:

$$\dot{x}_c = f_c(x_c, u) := \begin{bmatrix} f_{c_1}(x_c, u) \\ f_{c_2}(x_c, u) \\ f_{c_3}(x_c, u) \end{bmatrix}, \quad x_{c0} := x_c(0), \quad (2.52)$$

where  $x_c := [x_o^\top, v_o^\top, q^\top]^\top \in \mathbb{M} \times \mathbb{R}^{n+6}$ ,  $u \in \mathbb{R}^{6N}$  and  $f_c : \mathbb{M} \times \mathcal{S} \times \mathbb{R}^{6N+6} \rightarrow \mathbb{R}^{n+12}$ , with

$$\begin{aligned}f_{c_1}(x_c, u) &:= J_O(\eta_O)v_o, \\ f_{c_2}(x_c, u) &:= \tilde{M}(x)^{-1} \left[ G(q)u - \tilde{C}(x)v_o - \tilde{g}(x) \right], \\ f_{c_3}(x_c, u) &:= \tilde{J}(q)G(q)^\top v_o,\end{aligned}$$

where we have used the first equation of (2.41). Note that  $f_c$  is *locally Lipschitz continuous* in its domain since it is continuously differentiable

there. Next, we define the respective errors:

$$e_c := x_c - x_d = \begin{bmatrix} x_o \\ v_o \\ q \end{bmatrix} - \begin{bmatrix} x_d \\ \dot{x}_d \\ q_d \end{bmatrix} = \begin{bmatrix} x_o - x_d \\ v_o \\ q - q_d \end{bmatrix} \in \mathbb{M} \times \mathbb{R}^6 \times \mathcal{S}, \quad (2.53)$$

where  $q_d := [q_{1,d}^\top, \dots, q_{N,d}^\top]^\top \in \mathbb{R}^n$  is appropriately chosen to comply with the coupled kinematics (2.8) and  $x_d$ . The error dynamics are then  $\dot{e}_c(t) = f_c(x_c(t), u(t))$ , which can be appropriately transformed to:

$$\dot{e}_c = f_e(e_c, u), \quad e_{c0} := e_c(0) = x_c(0) - x_d. \quad (2.54)$$

where  $f_e := f_c(e_c + x_d, u)$ . By ignoring over-actuated input terms, we have that  $\tau_i = J_i^\top u_i$ , which yields

$$\|\tau_i\| \leq \bar{\tau}_i \Leftrightarrow \sigma_{\min}(J_i^\top) \|u_i\| \leq \bar{\tau}_i,$$

where we have employed the property  $\sigma_{\min}(J_i^\top) \|u_i\| \leq \|J_i^\top u_i\|$ , with  $\sigma_{\min}(J_i^\top)$  being positive, if the constraint  $q_i \in \mathcal{S}_i$  is always satisfied. Hence, the constraint  $|\tau_{i,k}| \leq \bar{\tau}_i$  is equivalent to

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

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

$$U_c := \left\{ u \in \mathbb{R}^{6N} : \|u_i\| \leq \frac{\bar{\tau}_i}{\sigma_{\min}(J_i^\top)}, \forall i \in \mathcal{N}, k \in \{1, \dots, n_i\} \right\}, \quad (2.55)$$

as the set that captures the control input constraints of the error dynamics system (2.54). By using (2.52) to express  $\dot{q}$  as a function of  $v_o$ , we define also the set  $X_c \subseteq \mathbb{R}^{n+12}$ :

$$X_c := \left\{ x_c \in \mathbb{R}^{n+12} : \theta_o \in [\bar{\theta}, \bar{\theta}], \|J_i(q_i)^\top (J_i(q_i)J_i(q_i)^\top)^{-1} J_{o_i}(q_i)v_o\| \leq \bar{q}_i, \right. \\ \left. i \in \mathcal{N}, q \in \mathcal{S} \cap \mathcal{S}_A \cap (\mathcal{S}_{1,o} \times \dots \times \mathcal{S}_{N,o}), x_o \in \mathbb{M} \cap \mathcal{S}_o(x_o) \right\}.$$

The set  $X_c$  captures all the state constraint of the system dynamics (2.52). In view of (2.53), we define the set  $E_c \subseteq \mathbb{R}^{n+12}$  as:

$$E_c := \{e_c \in \mathbb{R}^{n+12} : e_c \in X_c \oplus (-x_d)\},$$

as the set that captures all the constraints of the error dynamics system (2.54).

The problem in hand is the design of a control input  $u(t) \in U_c$  such that  $\lim_{t \rightarrow \infty} e_c(t) = 0$  while ensuring  $e_c(t) \in E_c, \forall t \in \mathbb{R}_{\geq 0}$ . The proposed Nonlinear Model Predictive scheme is presented hereafter.

Consider a sequence of sampling times  $\{t_j\}$ ,  $j \in \mathbb{N}$ , with a constant sampling period  $h_s \in (0, T_p)$ , where  $T_p$  is the prediction horizon, such that:

$$t_{j+1} = t_j + h_s, \forall j \geq 0.$$

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

$$\min_{\hat{u}(\cdot)} J_c(e_c(t_j), \hat{u}(\cdot)) := \min_{\hat{u}(\cdot)} \left\{ V_c(\hat{e}_c(t_j + T_p)) + \int_{t_j}^{t_j + T_p} F_c(\hat{e}_c(s), \hat{u}(s)) ds \right\} \quad (2.56a)$$

subject to:

$$\dot{\hat{e}}_c(s) = f_e(\hat{e}_c(s), \hat{u}(s)), \hat{e}_c(t_j) = e_c(t_j), \quad (2.56b)$$

$$\hat{e}_c(s) \in E_c, \hat{u}(s) \in U_c, s \in [t_j, t_j + T_p], \quad (2.56c)$$

$$\hat{e}_c(t_j + T_p) \in \mathcal{E}_{cf}, \quad (2.56d)$$

where the hat  $\hat{\cdot}$  denotes the predicted variables (internal to the controller), i.e.  $\hat{e}_c(\cdot)$  is the solution of (2.56b) driven by the control input  $\hat{u}(\cdot) : [t_j, t_j + T_p] \rightarrow U_c$  with initial condition  $e_c(t_j)$ . Note that the predicted values are not necessarily the same with the actual closed-loop values (see [? ]). The term  $F_c : E_c \times U_c \rightarrow \mathbb{R}_{\geq 0}$ , is the *running cost*, and is chosen as:

$$F_c(e_c, u) := e_c^\top Q_c e_c + u^\top R_c u.$$

The terms  $V_c : E_c \rightarrow \mathbb{R}_{>0}$  and  $\mathcal{E}_{cf}$  are the *terminal penalty cost* and *bounded terminal set*, respectively, and are used to enforce the stability of the system. The terminal cost is given by  $V_c(e_c) := e_c^\top P_c e_c$ ;  $Q_c \in \mathbb{R}^{(n+12) \times (n+12)}$  is chosen as a diagonal positive semi-definite matrix, and  $P_c, R_c \in \mathbb{R}^{(n+12) \times (n+12)}$  as diagonal positive definite matrices.

The solution of the OCP (2.56a)-(2.56d) starting at time  $t_j$  provides an optimal control input denoted by  $\hat{u}^*(s; e_c(t_j))$ , for  $s \in [t_j, t_j + T_p]$ . It defines the open-loop input that is applied to the system until the next sampling instant  $t_{j+1}$ :

$$u(s; e_c(t_j)) = \hat{u}^*(s; e_c(t_j)), s \in [t_j, t_{j+1}]. \quad (2.57)$$

The corresponding *optimal value function* is given by  $J_c^*(e_c(t_j), \hat{u}^*(\cdot; e_c(t_j)))$ , where  $J_c(\cdot)$  as is given in (2.56a). The control input  $u(s; e_c(t_j))$  is a feedback, since it is recalculated at each sampling instant using the new state

information. The solution of (2.54) starting at time  $t_j$  from an initial condition  $e_c(t_j)$ , applying a control input  $u : [t_j, t_{j+1}] \rightarrow U_c$  is denoted by  $e_c(s; u(\cdot), e_c(t_j)), s \in [t_j, t_{j+1}]$ .

Through the following theorem, we guarantee the stability of the system which is the solution to Problem 1 (see also Theorem (D.1) in Appendix D).

**Theorem 2.3.** *Let Assumption 2.7 hold. Suppose also that:*

1. *The OCP (2.56a)-(2.56d) is feasible for the initial time  $t = 0$ .*
2. *The terminal set  $\mathcal{E}_{cf} \subseteq E_c$  is closed, with  $0 \in \mathcal{E}_{cf}$ .*
3. *The terminal set and terminal cost are chosen such that there exists an admissible control input (according to Def. D.1 of Appendix D)  $u_{cf} : [0, h_s] \rightarrow U_c$  such that for all  $e_c(s) \in \mathcal{E}_{cf}$  it holds that:*
  - a)  $e_c(s) \in \mathcal{E}_{cf}, \forall s \in [0, h_s]$ .
  - b)  $\frac{\partial V_c}{\partial e_c} f_e(e_c(s), u_{cf}(s)) + F_c(e_c(s), u_{cf}(s)) \leq 0, \forall s \in [0, h_s]$ .

*Then, the closed loop system (2.54), under the control input (2.57), converges to the origin for  $t \rightarrow \infty$ , i.e.,  $\lim_{t \rightarrow \infty} e_c(t) = 0$ .*

*Proof.* The proof is identical to the proof of Theorem 2.1 in [? ].  $\square$

## Simulation Results

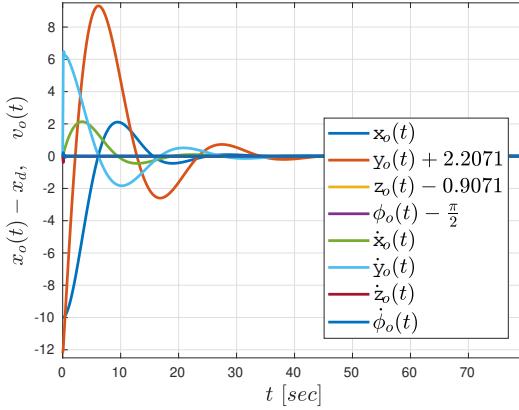
To demonstrate the efficiency of the proposed control protocol, we consider the following simulation scenario.

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 (2.52) we have that  $x = [x_o^\top, v_o^\top, q^\top]^\top \in \mathbb{R}^{16}$ ,  $u \in \mathbb{R}^8$ , with  $x_o = [p_o^\top, \phi_o]^\top \in \mathbb{R}^4$ ,  $v_o = [p_o^\top, \omega_o]^\top \in \mathbb{R}^4$ , where  $\omega_o \in \mathbb{R}$  occurs with respect to only one axis. We also denote  $p_o = [x_o, y_o, z_o]^\top \in \mathbb{R}^3$ ,  $q = [q_1^\top, q_2^\top]^\top \in \mathbb{R}^8$ ,  $q_i = [p_{B_i}^\top, \alpha_i^\top]^\top \in \mathbb{R}^4$ ,  $p_{B_i} = [x_{B_i}, y_{B_i}]^\top \in \mathbb{R}^2$ ,  $\alpha_i = [\alpha_{i1}, \alpha_{i2}]^\top \in \mathbb{R}^2, i \in \{1, 2\}$ , where  $p_{B_i}$  are the vehicles' positions, and  $\alpha_i$  the manipulator angles. The manipulators become singular when  $\sin(\alpha_{i1}) = 0, i \in \{1, 2\}$ , thus the state constraints for the manipulators are set to:

$$\begin{aligned} \epsilon \leq \alpha_{11} &\leq \frac{\pi}{2} - \epsilon, -\frac{\pi}{2} + \epsilon \leq \alpha_{12} \leq \frac{\pi}{2} - \epsilon, \\ -\frac{\pi}{2} + \epsilon \leq \alpha_{21} &\leq -\epsilon, -\frac{\pi}{2} + \epsilon \leq \alpha_{22} \leq \frac{\pi}{2} - \epsilon. \end{aligned}$$

We also consider the input constraints:

$$-10 \leq u_{i,j}(t) \leq 10, i \in \{1, 2\}, j \in \{1, \dots, 4\}.$$



**Figure 2.14:** The errors of the object for  $t \in [0, 80]$  seconds.

The initial conditions are set to:

$$x_o(0) = \left[ 0, -2.2071, 0.9071, \frac{\pi}{2} \right]^\top, v_o(0) = [0, 0, 0, 0]^\top,$$

$$q_1(0) = \left[ 0, 0, \frac{\pi}{4}, \frac{\pi}{4} \right]^\top, q_2(0) = \left[ 0, -4.4142, -\frac{\pi}{4}, -\frac{\pi}{4} \right]^\top,$$

(in (m, rad), (m/s, rad/s), rad, rad/s, respectively). The desired goal states are set to:

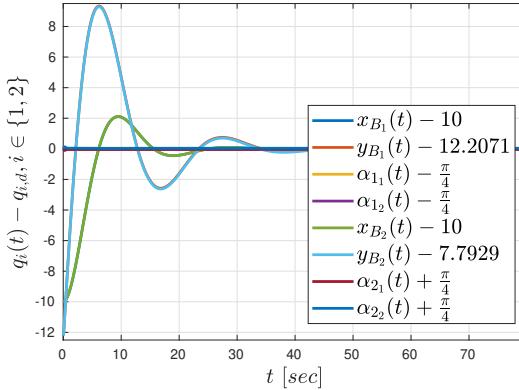
$$x_d = \left[ 10, 10, 0.9071, \frac{\pi}{2} \right]^\top,$$

$$q_{1,d} = \left[ 10, 12.2071, \frac{\pi}{4}, \frac{\pi}{4} \right]^\top, q_{2,d} = \left[ 10, 7.7929, -\frac{\pi}{4}, -\frac{\pi}{4} \right]^\top,$$

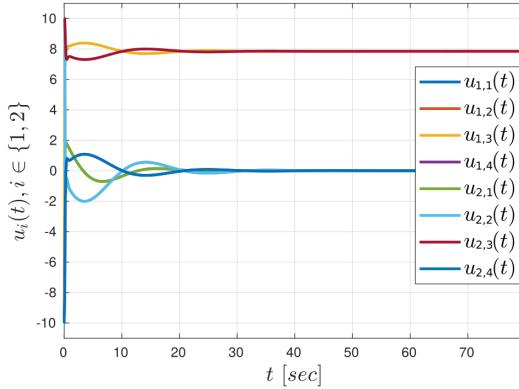
(in (m, rad), rad, respectively). We set a spherical obstacle between the initial and the desired pose of the object, with center (5, 5, 1) m and radius 2 m. The sampling time is  $h = 0.1$  seconds, the horizon is set to  $T_p = 0.5$  seconds, and the total simulation time is 80 seconds; The matrices  $P_c, Q_c, R_c$  are set to:

$$P_c = Q_c = 0.5I_{16}, R_c = 0.5I_8.$$

The terminal set is taken as a ball of radius 0.1 m around 0. The simulation results are depicted in Fig. 2.14|Fig. 2.16, which show 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 [?] and they took 23500 seconds in MATLAB Environment on a desktop computer with 8 cores, 3.60 GHz CPU and 16GB of RAM.



**Figure 2.15:** The errors of robotic agents for  $t \in [0, 80]$  seconds.



**Figure 2.16:** The control inputs of the actuators of the robotic agents  $u_i(t)$ ,  $\forall t \in [0, 80]$  seconds.

### 2.2.8 Decentralized NMPC

In this section, in order to reduce the computational complexity of the NMPC, we develop a decentralized counterpart, where each robotic agent calculates its own control signal.

We first decouple the dynamics (2.14) for each agent's MPC. We define  $x_{O_i} : \mathbb{R}^{n_i} \rightarrow \mathbb{M}$ ,  $v_{O_i} : \mathbb{R}^{2n_i} \rightarrow \mathbb{R}^6$  with  $x_{O_i}(q_i) := [p_{O_i}(q_i)^\top, \eta_{O_i}(q_i)^\top]^\top \in \mathbb{M}$ ,

$$p_{O_i}(q_i) := p_{E_i}(q_i) + R_i(q_i)p_{O/E_i}^{E_i} \quad (2.58a)$$

$$\eta_{O_i}(q_i) := \eta_{E_i}(q_i) + \eta_{O/E_i} \quad (2.58b)$$

$\forall i \in \mathcal{N}$ , as well as

$$v_{O_i}(q_i, \dot{q}_i) := [\dot{p}_{O_i}(q_i)^\top, \omega_{O_i}(q_i, \dot{q}_i)^\top]^\top := J_{i_O}(q_i)v_i(q_i, \dot{q}_i), \quad \forall i \in \mathcal{N}, \quad (2.59)$$

where  $J_{i_O}(q_i) := J_{O_i}(q_i)^{-1}$ ,  $\forall i \in \mathcal{N}$ , which are derived from (2.8) and (2.9), respectively;  $x_{O_i}$  and  $v_{O_i}$  are the pose and velocity of the object as computed by agent  $i \in \mathcal{N}$ .

Consider now the constants  $c_i$ , with  $0 < c_i < 1$  and  $\sum_{i \in \mathcal{N}} c_i = 1$  that play the role of load sharing coefficients for the agents. Then the object dynamics (2.5) can be written as:

$$\begin{aligned} \sum_{i \in \mathcal{N}} c_i \left\{ M_O(\eta_{O_i}(q_i)) \dot{v}_{O_i}(q_i, \dot{q}_i) + C_O(\eta_{O_i}(q_i), \omega_{O_i}(q_i, \dot{q}_i)) v_{O_i}(q_i, \dot{q}_i) + g_O \right\} = \\ \sum_{i \in \mathcal{N}} J_{O_i}(q_i)^\top h_i, \end{aligned}$$

from which, by employing the grasp coupling (see (2.13)), the differential kinematics of the agents, (2.59), and after straightforward algebraic manipulations, we obtain the coupled dynamics

$$\sum_{i \in \mathcal{N}} \left\{ M_{D_i}(q_i) \ddot{q}_i + C_{D_i}(q_i, \dot{q}_i) \dot{q}_i + g_{D_i}(q_i) \right\} = \sum_{i \in \mathcal{N}} J_{O_i}(q_i)^\top u_i, \quad (2.60)$$

where:

$$\begin{aligned} M_{D_i} &:= M_{D_i}(q_i) := c_i M_O J_{i_O} J_i + J_{O_i}^\top M_i J_i, \\ C_{D_i} &:= C_{D_i}(q_i, \dot{q}_i) := J_{O_i}^\top (M_i \dot{J}_i + C_i J_i) + c_i M_O J_{i_O} \dot{J}_i + c_i M_O \dot{J}_{i_O} J_i + c_i C_O, \\ g_{D_i} &:= g_{D_i}(q_i) := c_i g_O + J_{O_i}^\top g_i, \end{aligned}$$

$\forall i \in \mathcal{N}$ . Since the scheme developed here is decentralized, we need the following assumption regarding the agent communication:

**Assumption 2.8.** (Sensing and communication capabilities) Each agent  $i \in \mathcal{N}$  is able to continuously communicate with the other agents  $j \in \mathcal{N} \setminus \{i\}$  and transmit appropriate information.

Along with the sets  $S_{i,O}$ ,  $S_A$  defined in the previous section, we also define

$$\begin{aligned} S_{O_i} &:= \{q_i \in \mathbb{R}^{n_i} : \mathcal{A}_O(x_{O_i}(q_i)) \cap \mathcal{O}_z = \emptyset, \forall z \in \mathcal{Z}\}, \\ S_{i,\mathcal{A}}(q_{-i}) &:= \{q_i \in \mathbb{R}^{n_i} : \mathcal{A}_i(q_i) \cap \mathcal{A}_j(q_j) = \emptyset, \forall j \in \mathcal{N} \setminus \{i\}\}, \end{aligned}$$

where  $q_{-i} := [q_1^\top, \dots, q_{i-1}^\top, q_{i+1}^\top, \dots, q_N^\top]^\top$ ,  $\forall i \in \mathcal{N}$ .

To design a decentralized NMPC control scheme, 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 (2.60) 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:

$$M_{D_1} \ddot{q}_1 + C_{D_1} \dot{q}_1 + g_{D_1} = J_{O_1}^\top u_1, \quad (2.61)$$

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:

$$M_{D_i} \ddot{q}_i + C_{D_i} \dot{q}_i + g_{D_i} = J_{O_i}^\top u_i, \quad (2.62)$$

the state equality constraints:

$$x_{O_i}(q_i) = x_{O_1}(\hat{q}_1), v_{O_i}(q_i, \dot{q}_i) = v_{O_1}(\hat{q}_1, \hat{\dot{q}}_1), \quad (2.63)$$

$i \in \{2, \dots, N\}$  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 (2.61), (2.62) in the following sense. Although the coupled model (2.60) 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 (2.60) is satisfied, since it's the result of the addition of (2.61) and (2.62), 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 (2.63) guarantee that the predicted variables of the agents  $\{2, \dots, N\}$  will comply with the rigidity at the grasping points.

By using the notation  $x_{q_i} := [x_{q_i,1}^\top, x_{q_i,2}^\top]^\top := [q_i^\top, \dot{q}_i^\top]^\top \in \mathbb{R}^{2n_i}$ ,  $i \in \mathcal{N}$ , the nonlinear dynamics of each agent can be written as:

$$\dot{x}_{q_i} = f_{q_i}(x_{q_i}, u_i) := \begin{bmatrix} f_{q_i,1}(x_{q_i}) \\ f_{q_i,2}(x_{q_i}, u_i) \end{bmatrix}, \quad (2.64)$$

where  $f_{q_i} : \mathcal{S}_i \times \mathbb{R}^{n_i+6} \rightarrow \mathbb{R}^{2n_i}$  is the locally Lipschitz function:

$$\begin{aligned} f_{q_i,1}(x_{q_i}) &:= x_{q_i,2}, \\ f_{q_i,2}(x_{q_i}, u_i) &:= \widehat{M}_{D_i}(q_i) \left( J_{O_i}(q_i)^\top u_i - C_{D_i}(q_i, \dot{q}_i) \dot{q}_i - g_{D_i}(q_i) \right), \end{aligned}$$

$\forall i \in \mathcal{N}$ , where  $\widehat{M}_{D_i} : \mathcal{S}_i \rightarrow \mathbb{R}^{n_i \times 6}$ , is the pseudo-inverse

$$\widehat{M}_{D_i}(q_i) := M_{D_i}(q_i)^\top \left( M_{D_i}(q_i) M_{D_i}(q_i)^\top \right)^{-1}.$$

$\forall i \in \mathcal{N}$ . We define now the error vector  $e_{D_1} : \mathbb{R}^{2n_i} \rightarrow \mathbb{R}^{12}$ , as:

$$e_{D_1}(x_{q_1}) := \begin{bmatrix} x_{O_1}(q_1) - x_d \\ v_{O_1}(q_1, \dot{q}_1), \end{bmatrix}$$

which gives us the *error dynamics*:

$$\dot{e}_{D_1} = f_{D_1}(x_{D_1}, u_1), \quad (2.65)$$

with  $f_{D_1} : \mathcal{S}_i \times \mathbb{R}^{n_i+6} \rightarrow \mathbb{R}^{12}$ :

$$\begin{aligned} f_{D_1}(x_{q_1}, u_1) &:= \\ &\left[ \begin{array}{l} J_O(\eta_{O_1}(q_1)) J_{1_O}(q_1) J_1(q_1) \dot{q}_1 \\ J_{1_O}(q_1) J_1(q_1) f_{q_1,2}(x_{q_1}, u_1) + \left( J_{1_O}(q_1) \dot{J}_1(q_1) + j_{1_O}(q_1) J_1(q_1) \right) \dot{q}_1, \end{array} \right] \end{aligned}$$

where we employed (2.65) and the object dynamics. The input constraint sets are defined similarly to (2.55) as

$$U_{D_i} := \left\{ u_i \in \mathbb{R}^6 : \|u_i\| \leq \frac{\bar{\tau}_i}{\sigma_{\min}(J_i^\top)} \right\},$$

Define also the sets

$$\begin{aligned} X_{D_1}(q_{-1}) &:= \left\{ x_{q_1} \in \mathbb{R}^{2n_1} : \theta_{O_1}(q_1) \in [-\bar{\theta}, \bar{\theta}], |\dot{q}_{1_k}| \leq \bar{q}_1, \forall k \in \{1, \dots, n_1\}, \right. \\ &\quad \left. q_1 \in \mathcal{S}_1 \cap \mathcal{S}_{1,\mathcal{A}}(q_{-1}) \cap \mathcal{S}_{1,O} \cap \mathcal{S}_{O_1} \right\} \\ X_{D_i}(q_{-i}) &:= \left\{ x_{q_i} \in \mathbb{R}^{2n_i} : |\dot{q}_{i_k}| \leq \bar{q}_i, \forall k \in \{1, \dots, n_i\}, q_i \in \mathcal{S}_i \cap \mathcal{S}_{i,\mathcal{A}} \cap \mathcal{S}_{i,O} \right\}, \end{aligned}$$

$\forall i \in \{2, \dots, N\}$ . The sets  $X_{D_i}$  capture all the state constraints of the system dynamics (2.64), 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

$$E_{D_1}(q_{-1}) := \{e_{D_1}(x_{q_1}) \in \mathbb{R}^{12} : x_{q_1} \in X_1(q_{-1})\},$$

which now represents the constraints set for the NMPC scheme of the leader.

The main problem at hand is the design of a *feedback control law*  $u_1 \in U_{D_1}$  for agent 1 which guarantees that the error signal  $e_{D_1}$  with dynamics given in (2.65), satisfies  $\lim_{t \rightarrow \infty} \|e_{D_1}(x_{q_1}(t))\| \rightarrow 0$ , while ensuring singularity avoidance, collision avoidance between the leader, the object and the obstacles as well as collision avoidance between the leader and the followers in their current position. The role of the followers  $\{2, \dots, N\}$  is, through the load-sharing coefficients  $c_2, \dots, c_N$  in (2.60), to contribute to the object trajectory execution, as derived by the leader agent 1, while also avoiding collisions. 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 centralized scheme, with  $t_{j+1} = t_j + h_s$ ,  $h_s \in (0, T_p)$ , and  $T_p$  the respective horizon. For agent 1, the open-loop input signal applied in between the sampling instants is given by the solution of the following FHOCP:

$$\begin{aligned} \min_{\hat{u}_1(\cdot)} J_{D_1}(e_{D_1}(x_{q_1}(t_j)), \hat{u}_1(\cdot)) &:= \min_{\hat{u}_1(\cdot)} \left\{ V_{D_1}(e_{D_1}(\hat{x}_{q_1}(t_j + T_p))) \right. \\ &\quad \left. + \int_{t_j}^{t_j + T_p} \left[ F_{D_1}(e_{D_1}(\hat{x}_{q_1}(s)), \hat{u}_1(s)) \right] ds \right\} \end{aligned} \quad (2.66a)$$

subject to:

$$\dot{e}_{D_1}(\hat{x}_{q_1}(s)) = f_{D_1}(\hat{x}_{q_1}(s), \hat{u}_1(s)), \quad e_{D_1}(\hat{x}_{q_1}(t_j)) = f_{D_1}(x_{q_1}(t_j)), \quad (2.66b)$$

$$e_{D_1}(\hat{x}_{q_1}(s)) \in E_{D_1}(q_{-1}(t_j)), \quad s \in [t_j, t_j + T_p], \quad (2.66c)$$

$$\hat{u}_1(s) \in U_{D_1}, \quad s \in [t_j, t_j + T_p], \quad (2.66d)$$

$$e_{D_1}(\hat{x}_{q_1}(t_j + T_p)) \in \mathcal{E}_{D_1}. \quad (2.66e)$$

At a generic time  $t_j$  then, agent 1 solves the aforementioned FHOCP. The functions  $F_{D_1} : E_{D_1}(q_{-1}(t_j)) \times U_{D_1} \rightarrow \mathbb{R}_{\geq 0}$ ,  $V_{D_1} : \mathcal{E}_{D_1}(q_{-1}(t_j)) \rightarrow \mathbb{R}_{\geq 0}$  stand for the *running cost* and the *terminal penalty cost*, respectively, and they are defined as:

$$\begin{aligned} F_{D_1}(e_{D_1}, u_1) &:= e_{D_1}^\top Q_{D_1} e_{D_1} + u_1^\top R_{D_1} u_1 \\ V_{D_1}(e_{D_1}) &:= e_{D_1}^\top P_{D_1} e_{D_1}, \end{aligned}$$

where  $R_{D_1} \in \mathbb{R}^{6 \times 6}$  and  $P_{D_1} \in \mathbb{R}^{2n_1 \times 2n_1}$  are symmetric and positive definite gain matrices;  $Q_{D_1} \in \mathbb{R}^{2n_1 \times 2n_1}$  is a symmetric and positive semi-definite

controller gain matrix. The bounded *terminal* set is defined here as  $\mathcal{E}_{D_1}$ , and we assume that  $\mathcal{E}_{D_1} \subset \bigcap_{j \in \mathbb{N}} \{E_{D_1}(q_{-1}(t_j))\} \neq \emptyset$ .

The solution to FHOCP (2.66a) - (2.66e) starting at time  $t_j$  provides an optimal control input, denoted by  $\hat{u}_1^*(s; e_{D_1}(x_{q_1}(t_j)), x_q(t_j))$ ,  $s \in [t_j, t_j + T_p]$ ,  $x_q := [x_{q_1}^\top, \dots, x_{q_N}^\top]^\top$ . This control input is then applied to the system until the next sampling instant  $t_{j+1}$ :

$$u_1(s; e_{D_1}(x_{q_1}(t_j), x_q(t_j))) = \hat{u}_1^*(s; e_{D_1}(x_{D_1}(t_j)), x_q(t_j)), \quad \forall s \in [t_j, t_{j+1}). \quad (2.67)$$

At time  $t_{j+1} = t_j + h_s$  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 (2.65) starting at time  $t_j$ , from an initial condition  $x_q(t_j), e_{D_1}(x_{q_1}(t_j))$ , by application of the control input  $u_1 : [t_j, t_{j+1}] \rightarrow U_{D_1}$  is denoted by  $e_{D_1}(x_{q_1}(s; u_1(\cdot), x_q(t_j)), e_{D_1}(x_{q_1}(t_j)), s \in [t_j, t_j + T_p]$ .

After the solution of the FHOCP and the calculation of the predicted states  $\hat{x}_{q_1}(s)$ ,  $s \in [t_j, t_{j+1}]$ , at each time instant  $t_j$ , agent 1 transmits the values  $\hat{q}_1(s)$ ,  $\dot{\hat{q}}_1(s)$  as well as  $x_{O_1}(\hat{q}_1(s))$  and  $v_{O_1}(\hat{q}_1(s), \dot{\hat{q}}_1(s))$ , as computed by (2.58), (2.59),  $\forall s \in [t_j, t_j + T_p]$ , to the rest of the agents  $\{2, \dots, N\}$ . The rest of the agents then proceed as follows. Each agent  $i \in \{2, \dots, N\}$ , solves the following FHOCP:

$$\min_{\hat{u}_i(\cdot)} J_{D_i}(x_{q_i}(t_j), \hat{u}_i(\cdot)) \quad (2.68a)$$

subject to:

$$\dot{x}_{q_i} = f_{q_i}(x_{q_i}(s), u_i(s)), \quad (2.68b)$$

$$x_{q_i}(s) \in X_i(\hat{q}_1(s), \dots, \hat{q}_{i-1}(s), q_{i+1}(t_j), \dots, q_N(t_j)), \quad (2.68c)$$

$$x_{O_i}(q_i(s)) = x_{O_1}(\hat{q}_1(s)), s \in [t_j, t_j + T_p] \quad (2.68d)$$

$$v_{O_i}(q_i(s), \dot{q}_i(s)) = v_{O_1}(\hat{q}_1(s), \dot{\hat{q}}_1(s)), s \in [t_j, t_j + T_p] \quad (2.68e)$$

$$u_i(s) \in U_{D_i}, s \in [t_j, t_j + T_p], \quad (2.68f)$$

at every sampling time  $t_j$ , where  $J_{D_i}$  is an associated cost function. The constraint (2.68d) guarantees that agent  $i$  will obtain a trajectory that does not collide with the predicted trajectories of the agents higher in priority, or the agents lower in priority at  $t_j$ . Note that, through the equality constraints (2.68d), (2.68e), the follower agents must comply with the trajectory computed by the leader  $\hat{q}_1(s), \dot{\hat{q}}_1(s)$ . 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, however, is not in the scope of this thesis. We state that with the following assumption:

**Assumption 2.9.** The sets  $\{q \in \mathbb{R}^n : x_{O_i}(q_i(s)) = x_{O_1}(\hat{q}_1(s)), v_{O_i}(q_i(s), \dot{q}_i(s)) = v_{O_1}(\hat{q}_1(s), \dot{\hat{q}}_1(s)), q_i \in X_i(\hat{q}_1(s), \dots, \hat{q}_{i-1}(s), q_{i+1}(t_j), \dots, q_N(t_j))\}$  are nonempty,  $\forall i \in \{2, \dots, N\}$ ,  $\forall s \in [t_j, t_j + T_p]$ ,  $j \in \mathbb{N}$ .

Next, similarly to the leader agent, agent  $i > 1$  calculates the predicted states  $\hat{q}_i(s), \dot{\hat{q}}_i(s)$ ,  $s \in [t_j, t_j + T_p]$ , which then transmits to the agents  $\{i + 1, \dots, N\}$ . In that way, at each time instant  $t_j$ , each agent  $i \in \{2, \dots, N\}$  receives the other agents' states (as stated in Assumption 2.8), incorporates the constraint (2.68c) for the agents  $\{i + 1, \dots, N\}$ , receives the predicted states  $\hat{q}_\ell(s), \dot{\hat{q}}_\ell(s)$  from the agents  $\ell \in \{2, \dots, i - 1\}$  and incorporates the collision avoidance constraint (2.68d) 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  $\hat{q}_\ell(s), \ell \in \{1, \dots, i - 1\}$  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 (2.66a)-(2.66e)), the follower agents contribute in executing this trajectory through the load sharing coefficients  $c_i$  (as indicated in the coupled model (2.60)), and the agents low in priority are responsible for collision avoidance with the agents of higher priority. Moreover, the aforementioned equality constraints (2.68d), (2.68e) as well as the forward dynamics (2.68a) guarantee the compliance of all the followers with the model (2.60).

Therefore, given the constrained FHOCP (2.68a)-(2.68f), the solution of the problem lies in the capability of the leader agent to produce a state trajectory that guarantees  $x_{O_1}(q_1(t)) \rightarrow x_{\text{des}}$ , by solving the FHOCP (2.66a)-(2.66e), which is discussed in Theorem 2.4.

**Theorem 2.4.** Suppose that Assumptions 2.7 - 2.9 hold as well as

- The FHOCP (2.66a)-(2.66e) is feasible for the initial time  $t = 0$
- The terminal set  $\mathcal{E}_{D_1}$  is closed, with  $0 \in \mathcal{E}_{D_1}$
- The terminal set and terminal cost are chosen such that,  $\forall e_{D_1} \in \mathcal{E}_{D_1}$ , there exists an admissible control input  $u_{D_{1F}} : [0, h_s] \rightarrow U_{D_1}$  such that for all  $e_{D_1}(x_{q_1}(s)) \in \mathcal{E}_{D_1}$ ,  $\forall s \in [0, h_s]$  and

$$\frac{\partial V_{D_1}}{\partial e_{D_1}} f_{D_1}(e_{D_1}(x_{q_1}(s)), u_{D_{1F}}(s)) + F_{D_1}(e_{D_1}(x_{q_1}(s)), u_{D_{1F}}(s)) \leq 0$$

Then, the system (2.65), under the control input (2.67), converges to the origin when  $t \rightarrow \infty$ , i.e.  $\lim_{t \rightarrow \infty} e_{D_1}(x_{q_1}(t)) = 0$ .

*Proof.* The proof is identical to the proof of Theorem 2.1 in [? ].  $\square$

## 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:  $q_i = [p_{B_i}^\top, \alpha_i^\top]^\top \in \mathbb{R}^4$ ,  $p_{B_i} = [x_{B_i}, y_{B_i}]^\top \in \mathbb{R}^2$ ,  $\alpha_i = [\alpha_{i1}, \alpha_{i2}]^\top \in \mathbb{R}^2$ ,  $i \in \{1, 2, 3\}$ . The state of the object is  $x_O = [p_O^\top, \phi_O]^\top \in \mathbb{R}^4$  and it is calculated through the states of the agents. The singularity and input constraints are set as in the centralized case. The initial conditions of agents and the object are set to:

$$\begin{aligned} q_1(0) &= [0.5, 0, \frac{\pi}{4}, \frac{\pi}{4}]^\top, q_2(0) = [0, -4.4142, -\frac{\pi}{4}, -\frac{\pi}{4}]^\top, \\ q_3(0) &= [-0.50, -4.4142, -\frac{\pi}{4}, -\frac{\pi}{4}]^\top, \dot{q}_1(0) = \dot{q}_2(0) = \dot{q}_3(0) = [0, 0, 0, 0]^\top, \\ x_O(0) &= [0, -2.2071, 0.9071, \frac{\pi}{2}]^\top, \dot{x}_O(0) = [0, 0, 0, 0]^\top \end{aligned}$$

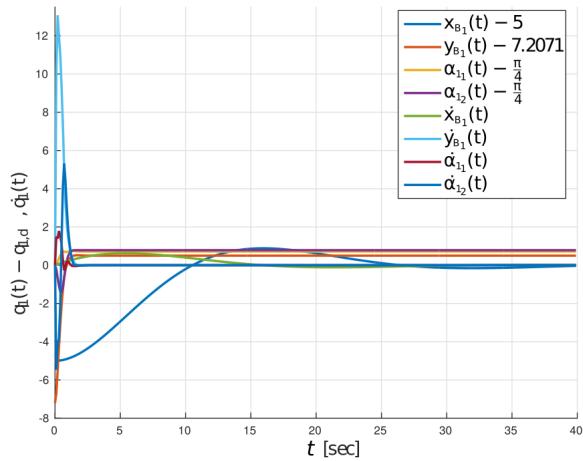
(in rad, rad/s (m, rad), (m/s, rad/s), respectively). The desired goal state the object is set to

$$x_{O,\text{des}} = [5, -2.2071, 0.9071, \frac{\pi}{2}]^\top$$

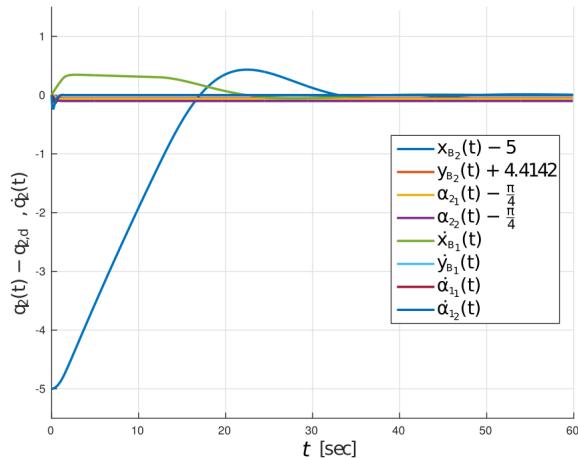
(m, rad), which, due to the structure of the considered robots, corresponds uniquely to

$$\begin{aligned} q_{1,\text{des}} &= [5.5, 0, \frac{\pi}{4}, \frac{\pi}{4}]^\top, q_{2,\text{des}} = [5, -4.4142, -\frac{\pi}{4}, -\frac{\pi}{4}]^\top, \\ q_{3,\text{des}} &= [4.5, 0, -\frac{\pi}{4}, -\frac{\pi}{4}]^\top, \dot{q}_{1,\text{des}} = \dot{q}_{2,\text{des}} = \dot{q}_{3,\text{des}} = [0, 0, 0, 0]^\top \end{aligned}$$

(in rad and rad/s, respectively). We set an obstacle between the initial and the desired pose of the object. The obstacle is spherical with center  $(2.5, -2.2071, 1)$  m and radius  $\sqrt{0.2}$  m. The sampling time is  $h_s = 0.1$  seconds, the horizon is  $T_p = 0.5$  seconds, and the total simulation time is 60 seconds; The matrices  $P_{D_i}$ ,  $Q_{D_i}$ ,  $R_{D_i}$  are set to:  $P_{D_i} = Q_{D_i} = 0.5I_8$ ,  $R_{D_i} = 0.5I_4$ ,  $\forall i \in \{1, 2, 3\}$ , and the load sharing coefficients as  $c_1 = 0.3$ ,  $c_2 = 0.5$ , and  $c_3 = 0.2$ . The functions  $J_{D_i}$  are chosen as simple quadratic functions of their arguments. The control input constraints are taken as in the previous section. The simulation results are depicted in Figs. 2.17 - 2.22. Figs. 2.17, 2.18, 2.19 show the error states of agent 1, 2 and 3, respectively, which converge to 0; Figs. 2.20 - 2.22 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 [?] 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

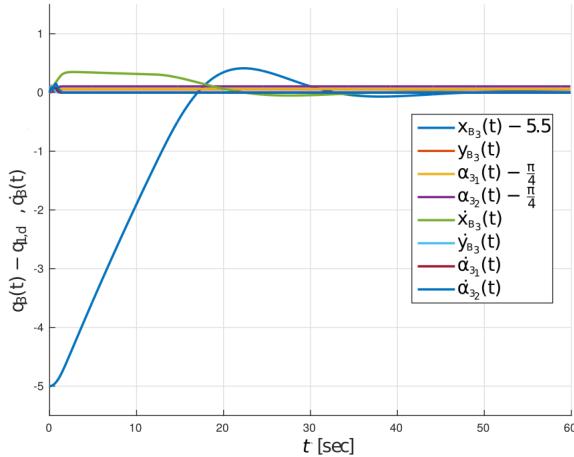


**Figure 2.17:** The error states of agent 1.

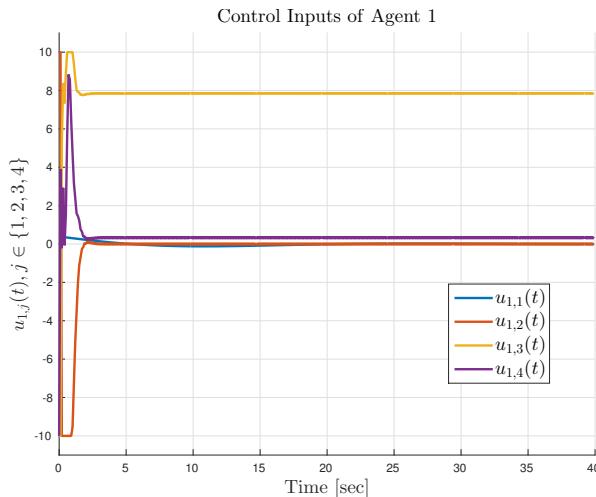


**Figure 2.18:** The error states of agent 2.

with respect to the centralized case of the previous section. Finally, a video illustrating an implementation of the algorithm in real hardware can be found on [https://youtu.be/f\\_95UCAAp6M](https://youtu.be/f_95UCAAp6M).



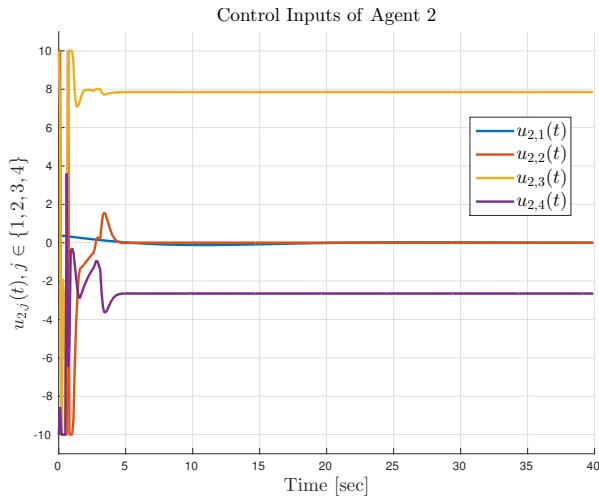
**Figure 2.19:** The error states of agent 3.



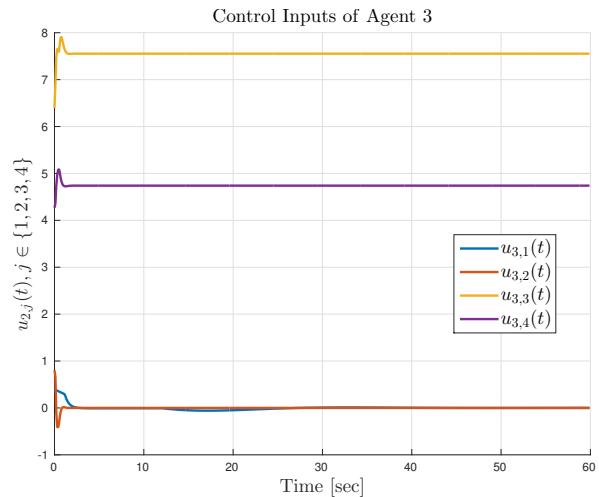
**Figure 2.20:** The control inputs of agent 1 with  $-10 \leq u_{1,j}(t) \leq 10$ ,  $\forall j \in \{1, \dots, 4\}$ .

## 2.3 Rolling Contacts

In this section we relax the assumption on the rigid grasping points. In particular, we assume that the robotic agents are connected to the object in terms of *rolling contacts*. As discussed before, this more natural approach to cooperative manipulation allows for a wider class of objects to be manip-

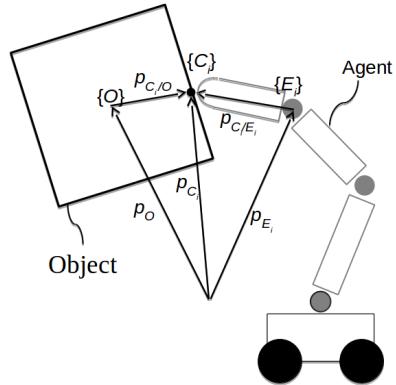


**Figure 2.21:** The control inputs of agent 2 with  $-10 \leq u_{2,j}(t) \leq 10$ ,  $\forall j \in \{1, \dots, 4\}$ .



**Figure 2.22:** The control inputs of agent 3 with  $-10 \leq u_{3,j}(t) \leq 10$ ,  $\forall j \in \{1, \dots, 4\}$ .

ulated, and allows for modular manipulation scenarios where robots can be swapped to adjust the grasp online.



**Figure 2.23:** A robotic agent in contact with a rigid object via a rolling contact.

### 2.3.1 System Model

We provide here the model of the coupled system, which is slightly different with respect to that of Section 2.2.1 to account for the rolling contact constraints. As before, we consider  $N \in \mathbb{R}$  robotic agents grasping a rigid object in 3D space, with generalized variables  $q_i, \dot{q}_i \in \mathbb{R}^{n_i}$ . We relax the assumption of fully actuated agents by requiring  $n_i \geq 3, \forall i \in \mathcal{N}$ . Each agent has a smooth, convex “fingertip” (i.e. passive end-effector) of high stiffness that is in contact with an object via a smooth contact surface. In addition to the end-effector frames  $\mathcal{E}_i$ , we add the contact frames  $\mathcal{C}_i$ , located at  $p_{C_i} \in \mathbb{R}^3$ , with respect to the inertial frame;  $\mathcal{C}_i$  are defined as Gauss frames [?], where one of the axes is defined orthonormal to the contact plane. We further define  $p_{C_i/E_i} := p_{C_i} - p_{E_i}, \forall i \in \mathcal{N}$ . A visual representation of the contact geometry for the  $i$ th agent is shown in Fig. 2.23. The dynamics of the  $i$ th agent is given by [?]

$$B_i \ddot{q}_i + C_{q_i} \dot{q}_i + g_{q_i} = -J_{h_i}^\top f_{C_i} + \tau_i \quad (2.69)$$

with the dynamics terms as in (2.1),  $f_{C_i} \in \mathbb{R}^3$  is the contact force in three dimensions, and  $J_{h_i} := J_{h_i}(q_i, p_{C_i/E_i}) : \mathbb{R}^{n_i+3} \rightarrow \mathbb{R}^{3 \times n_i}$  is the Jacobian matrix to the contact point, defined by

$$J_{h_i}(q_i, p_{C_i/E_i}) := [I_3 \quad -S(p_{C_i/E_i})] J_i(q_i),$$

where  $J_i$  is the manipulator Jacobian that maps  $\dot{q}_i \mapsto v_i$ , defined in the previous section. Note the difference of (2.3) and (2.69) due to the presence of the rolling contacts. Note also that disturbances are not taken into account here for simplicity.

The full hand Jacobian matrix is  $J_h := J_h(q, p_{EC}) := \text{diag}\{[J_{hi}]_{i \in N}\} \in \mathbb{R}^{3N \times n}$ , with  $p_{EC} := [p_{C_1/E_1}^\top, \dots, p_{C_N/E_N}^\top]^\top \in \mathbb{R}^{3N}$ . As before, we consider that the dynamical parameters (masses, moments of inertia) appearing in the terms  $B_i$ ,  $C_{qi}$ ,  $g_{qi}$  are *unknown*,  $\forall i \in \mathcal{N}$ . The dynamics (2.69) can be written in vector form as

$$B\ddot{q} + C_q\dot{q} + g_q = -J_h^\top f_C + \tau, \quad (2.70)$$

where  $B := \text{diag}\{[B_i]_{i \in \mathcal{N}}\}$ ,  $C_q := \text{diag}\{[C_{qi}]_{i \in \mathcal{N}}\} \in \mathbb{R}^{n \times n}$ ,  $g_q := [g_{q_1}^\top, \dots, g_{q_N}^\top]^\top \in \mathbb{R}^n$ ,  $f_C := [f_{C_1}^\top, \dots, f_{C_N}^\top]^\top \in \mathbb{R}^{3N}$ . With a slight abuse of notation, we assume that the set  $S$  ( $S_i$ ) contains the configurations  $q$  ( $q_i$ ) that yield a non-singular  $J_h$  ( $J_{hi}$ ) (in contrast to just  $J$  ( $J_i$ ) of the case of rigid contacts)

A common assumption in the majority of the related literature is that the object center of mass is accurately known, which is typically not the case in practice. We assume here tracking of a traceable point  $p_o$  on the object surface instead of the center of mass  $p_O$ , whose information is considered unknown. Note that appropriate sensor equipment, e.g., cameras and markers, can accurately track such points in practice. Hence, to remove the dependency on an unknown object center of mass, we perform a standard rigid body transformation to the conventional object dynamics as follows. Let  $R_o := R_o(\eta_o) : \mathbb{T} \rightarrow \mathbb{SO}(3)$  be the respective rotation matrix of a frame attached to  $p_o$ , and  $x_o := [p_o^\top, \eta_o^\top]^\top \in \mathbb{M}$ ,  $v_o := [\dot{p}_o^\top, \omega_o^\top]^\top \in \mathbb{R}^6$  denote the pose and generalized velocity of the object frame, with (without loss of generality)  $\eta_o = \eta_O$ ,  $R_o = R_O$ , and  $\omega_o = \omega_O$ . The position vector from  $p_o$  to the respective contact point is  $p_{C_i/o} := p_{C_i} - p_o \in \mathbb{R}^3$ ,  $\forall i \in \mathcal{N}$ . Moreover, define  $p_{C_i/o} := p_{C_i} - p_o$ ,  $\forall i \in \mathcal{N}$ , and  $p_{oC} := [p_{C_1/o}^\top, \dots, p_{C_N/o}^\top]^\top$ ,  $p_{oC} := [p_{C_1/o}^\top, \dots, p_{C_N/o}^\top]^\top$ .

Given the rolling contacts, the conventional object dynamics with respect to the object center of mass are given by the Newton-Euler formulation:

$$M_O \dot{v}_o + C_o v_o + g_o = \bar{G}_{\mathcal{R}} f_C \quad (2.71)$$

with the dynamic terms as in (2.5), and  $\bar{G}_{\mathcal{R}} : \mathbb{R}^{3N} \rightarrow \mathbb{R}^{6 \times 3N}$  is the grasp map, defined by  $\bar{G}_{\mathcal{R}} := \bar{G}_{\mathcal{R}}(p_{oC}) := [\bar{G}_{\mathcal{R}_1}, \dots, \bar{G}_{\mathcal{R}_N}]$  where  $\bar{G}_{\mathcal{R}_i} := \bar{G}_{\mathcal{R}_i}(p_{C_i/o}) : \mathbb{R}^3 \rightarrow \mathbb{R}^{6 \times 3}$ , with

$$\bar{G}_{\mathcal{R}_i}(p_{C_i/o}) := \begin{bmatrix} I_3 \\ S(p_{C_i/o}) \end{bmatrix}.$$

Note the difference of  $\bar{G}_{\mathcal{R}}$  with respect to the rigid contact-case (2.10). We now perform a transformation of the aforementioned dynamics to account for  $p_o$ . Let  $J_a := J_a(\eta_O) : \mathbb{T} \rightarrow \mathbb{R}^{6 \times 6}$  be defined as:

$$J_a(\eta_O) := \begin{bmatrix} I_3 & S(R_O p_{oC}^\mathcal{O}) \\ 0 & I_3 \end{bmatrix} \quad (2.72)$$

where  $p_{O_o}^{\mathcal{O}} := R_O^\top(p_O - p_o)$ , such that  $v_o = J_a v_O$ . Note that  $p_{O_o}^{\mathcal{O}}$  is constant.

Substitution of  $v_o = J_a v_O$  and left multiplication by  $J_a^\top$  in (2.71) yields the adjusted object dynamics with respect to  $p_o$ :

$$M_o \dot{v}_o + C_o v_o + g_o = G_{\mathcal{R}} f_c, \quad (2.73)$$

where

$$\begin{aligned} M_o &:= M_o(\eta_O) := J_a^\top M_O J_a, \\ C_o &:= C_o(\eta_O, \omega_O) := J_a^\top (M_O J_a + C_O J_a), \\ g_o &:= g_o(\eta_O) := J_a^\top g_O, \\ G_{\mathcal{R}} &:= G_{\mathcal{R}}(p_{oC}) := [G_{\mathcal{R}_1}, \dots, G_{\mathcal{R}_N}] := J_a^\top \bar{G}_{\mathcal{R}}, \end{aligned}$$

with  $G_{\mathcal{R}_i} := G_{\mathcal{R}_i}(p_{C_i/o})$ , and

$$G_{\mathcal{R}_i}(p_{C_i/o}) := J_a^\top \bar{G}_{\mathcal{R}_i} = \begin{bmatrix} I_3 \\ -S(R_O p_{O_o}^{\mathcal{O}}) + S(p_{C_i/o}) \end{bmatrix} = \begin{bmatrix} I_3 \\ S(p_{C_i/o}) \end{bmatrix},$$

Note that  $\bar{G}_{\mathcal{R}}$  does not depend on  $p_O$ . Note also by the relation  $p_O = p_o - R_O p_{O_o}^{\mathcal{O}}$ , that  $M_o$ ,  $C_o$ ,  $g_o$  are functions of  $\eta_o = \eta_O$ ,  $\omega_o = \omega_O$  with dependency on the constant but unknown term  $p_{O_o}^{\mathcal{O}}$ . We also note the following relation that will be needed subsequently:

$$\bar{G}_{\mathcal{R}_i}^\top v_O = \begin{bmatrix} I_3 \\ S(p_{C_i} - p_O) \end{bmatrix}^\top \begin{bmatrix} I_3 & S(R_O p_{O_o}^{\mathcal{O}}) \\ 0 & I_3 \end{bmatrix} v_o = G_{\mathcal{R}_i}^\top v_o \quad (2.74)$$

Similarly to the agents, the object dynamic parameters appearing in the terms  $M_o$ ,  $C_o$ ,  $g_o$  are considered to be *unknown*.

The more practical consideration of rolling contacts, as opposed to a rigid grasp, requires no slip to occur between the agents and object by ensuring that each contact force remains inside the friction cone defined by:

$$\mathcal{F}_{C_i}(\mu_f) := \{f_{C_i}^{C_i} \in \mathbb{R}^3 : f_{C,n_i} \mu_f \geq \sqrt{f_{C,x_i}^2 + f_{C,y_i}^2}\} \quad (2.75)$$

where  $f_{C_i}^{C_i} := R_{C_i}^\top f_{C_i} =: (f_{C,x_i}, f_{C,y_i}, f_{C,n_i})$  is the  $i$ th contact force written in frame  $C_i$ , whose orientation is described by  $R_{C_i} := R_{C_i}(\eta_{C_i}) : \mathbb{T} \rightarrow \mathbb{SO}(3)$ ,  $\forall i \in \mathcal{N}$ , ( $\eta_{C_i} \in \mathbb{T}$  being the respective Euler-angle orientation), with tangential force components  $f_{C,x_i}, f_{C,y_i} \in \mathbb{R}$  and normal force component  $f_{C,n_i} \in \mathbb{R}$ ,  $\mu_f \in \mathbb{R}_{>0}$  is the friction coefficient. The full friction cone is the Cartesian product of all the friction cones:  $\mathcal{F}_c := \mathcal{F}_{C_1} \times \dots \times \mathcal{F}_{C_n}$ .

In practice, it is common to approximate the friction cone by an inscribed pyramid with  $l_f \in \mathbb{R}_{>0}$  sides. The set associated with this pyramid is defined as

$$\tilde{\mathcal{F}}_{C_i}(\mu_f) := \{f_{C_i}^{C_i} \in \mathbb{R}^3 : \Lambda_i(\mu_f) f_{C_i}^{C_i} \succeq 0\}, \quad (2.76)$$

where  $\Lambda_i(\mu_f) \in \mathbb{R}^{l_f \times 3}$ . The overall friction pyramid is then  $\tilde{\mathcal{F}}_C(\mu_f) := \{f_C^i \in \mathbb{R}^{3N} : \Lambda(\mu_f)f_C^i \succeq 0\}$ , where  $f_C^i := [(f_{C_1}^{i_1})^\top, \dots, (f_{C_N}^{i_N})^\top]^\top$ , and  $\Lambda := \text{diag}\{\Lambda_i\}_{i \in \mathcal{N}}$ .

When the contact points do not slip, the grasp relation  $J_h \dot{q} = \bar{G}_R^\top v_o$  holds [?], which, after substituting (2.74), becomes:

$$v_C := J_h \dot{q} = G_R^\top v_o, \quad (2.77)$$

where  $v_C := [v_{C_1}^\top, \dots, v_{C_N}^\top]^\top \in \mathbb{R}^{3n}$  is the vector of contact velocities.

As in Section 2.2.3, we use for the object orientation the unit quaternion choice  $\zeta_o := [\varphi_o, \epsilon_o^\top]^\top \in \mathbb{S}^3$ . Let hence now a desired pose trajectory,  $p_d : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^3$ ,  $\zeta_d := [\varphi_d, \epsilon_d^\top]^\top : \mathbb{R}_{\geq 0} \rightarrow \mathbb{S}^3$ , to be tracked by  $x_o$ . To that end, similar to Section 2.2.3, we define the position error  $e_{p_o} := p_o - p_d$  as well as the quaternion product  $e_\zeta := \zeta_d \cdot \zeta_o^+$ . The aim is then to regulate  $e_{p_o}$  to zero and  $e_\zeta$  to  $[\pm 1, 0^\top]^\top$ . Moreover, we aim at ensuring that the agents are always in contact with the object and slipping is avoided. Formally, the problem is defined as follows.

**Problem 2.3.** Given a desired bounded, smooth object pose trajectory defined by  $p_d : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^3$ ,  $\zeta_d : \mathbb{R}_{\geq 0} \rightarrow \mathbb{S}^3$ , with bounded first and second derivatives, as well as uncertain agent and object dynamic parameters involved in (2.69) and (2.73), respectively, determine a control law  $\tau$  in (2.70) such that the following conditions hold:

1.  $\lim_{t \rightarrow \infty} (e_{p_o}(t), e_\zeta(t)) = (0, [\pm 1, 0^\top]^\top)$
2.  $f_{C_i}^{C_i}(t) \in \mathcal{F}_{C_i}, \forall t > 0, i \in \mathcal{N}$ .

In order to solve the aforementioned problem the following assumptions are made for the grasp:

**Assumption 2.10.** The grasp consists of  $N \geq 3$  agents with non-collinear contact points and  $\text{Null}(\bar{G}_R) \cap \text{Int}(\mathcal{F}_C) \neq \emptyset$ .

**Assumption 2.11.** The matrix  $J_h(q)$  is non-singular, and the contact points do not exceed the fingertip surface.

**Remark 2.9.** Note that  $N \geq 3$  agents with non-collinear contact points ensures  $\bar{G}_R$  is full row rank [?]. The condition that  $\text{null}(\bar{G}_R) \cap \text{Int}(\mathcal{F}_C) \neq \emptyset$  ensures the existence of a contact force that lies within the friction cone and yields a desired object wrench, which is called the force-closure condition [?]. Force-closure depends on the initial grasp, and can be ensured by existing high-level grasp planning methods [?]. Moreover, by incorporating optimization techniques, as e.g. in [?], we can enforce prevention of excessive rolling of the contacts and thus relax the respective part of Assumption 2.11. Finally, the non-singular condition of  $J_h$  intuitively implies that tracking

the desired reference trajectory does not force the agents through such singular configurations (such an assumption was also considered in the case of rigid grasps). This can also be achieved by exploiting internal motions of redundant agents ( $n_i > 3$ ).

We also assume that the contact vectors  $R_i^\top p_{C_i/E_i}$  and their derivatives are measured accurately online,  $\forall i \in \mathcal{N}$ . This can be achieved either by the use of appropriate tactile sensors or forward simulation of the contact dynamics [?]. By also assuming the geometry of the fingertips known, we can also compute  $\eta_{C_i}$  and hence  $R_{C_i}$  online,  $\forall i \in \mathcal{N}$ . Finally, note that  $B_i(\cdot)$  are positive definite, and  $\dot{B}_i(\cdot) - 2C_{q_i}(\cdot)$  are skew-symmetric,  $\forall i \in \mathcal{N}$ , similarly to  $M_i(\cdot)$ , and  $\dot{M}_i(\cdot) - 2C_i(\cdot)$ .

In the following, we present two *adaptive* control schemes for the solution of Problem 2.3, a centralized one, where one computer unit (or a “leader” agent) computes the input commands for the entire team, as well as a decentralized one, based on event-triggered inter-agent communication.

### 2.3.2 Centralized Scheme

This section presents the centralized proposed control scheme, which employs adaptive control techniques for the compensation of the dynamic uncertainties of the agent and object present in the problem setup.

Without loss of generality, we assume that  $n_i = 3$ ,  $\forall i \in \mathcal{N}$ , i.e., the agents are not redundant. The proposed solution can be trivially extended to redundant cases, e.g., by following the analysis of [?, Chapter 6]. By combining the agent and object dynamics (2.70), (2.73) as well as (2.77), we can obtain the coupled dynamics

$$\tilde{B}\dot{v}_o + \tilde{C}_qv_o + \tilde{g}_q = G_{\mathcal{R}}J_h^{-T}\tau, \quad (2.78)$$

where

$$\begin{aligned} \tilde{B} &:= \tilde{B}(\ddot{x}) := M_o + G_{\mathcal{R}}J_h^{-T}BJ_h^{-1}G_{\mathcal{R}}^\top, \\ \tilde{C}_q &:= \tilde{C}_q(\ddot{x}, \dot{\ddot{x}}) := C_o + G_{\mathcal{R}}J_h^{-T}(C_qJ_h^{-1}G_{\mathcal{R}}^\top + B\frac{d}{dt}(J_h^{-1}G_{\mathcal{R}}^\top)), \\ \tilde{g}_q &:= \tilde{g}_q(\ddot{x}) := g_o + G_{\mathcal{R}}J_h^{-T}g_q, \end{aligned}$$

and  $\ddot{x} := [\eta_o^\top, q^\top, p_{EC}^\top, p_{oC}^\top]^\top \in \mathbb{T} \times \mathbb{S} \times \mathbb{R}^{6N}$ . The following lemma states useful properties of (2.78):

**Lemma 2.2.** *The matrix  $\tilde{B}$  is symmetric and positive-definite, and the matrix  $\dot{\tilde{B}} - 2\tilde{C}_q$  is skew-symmetric.*

*Proof.* The proof is similar to the one of Lemma 2.1 and is omitted.  $\square$

Next, we proceed to parameterizing the dynamics with respect to constant but unknown dynamic parameters, similarly to the case of rigid contacts. In particular, the left-hand side of the object dynamics (with respect to  $p_o$ ) is parameterized as:

$$M_o(\eta_o)\dot{v}_o + C_o(\eta_o, \omega_o)v_o + g_o = Y_{\mathcal{R}_o}(\eta_o, \omega_o, v_o, \dot{v}_o)\vartheta_{\mathcal{R}_o},$$

where  $\vartheta_{\mathcal{R}_o} \in \mathbb{R}^{\ell_{\mathcal{R}_o}}$ ,  $\ell_{\mathcal{R}_o} \in \mathbb{N}$ , is a vector containing the unknown object dynamic parameters, similarly to  $\vartheta_o$  defined in Section 2.2.3, but also including the term  $p_{o,o}^{\mathcal{O}}$ , introduced in (2.72), and  $Y_{\mathcal{R}_o} : \mathbb{T} \times \mathbb{R}^{15} \rightarrow \mathbb{R}^{6 \times \ell_{\mathcal{R}_o}}$  is the respective (known) regressor matrix. Similarly, the part of (2.78) that concerns the robotic agents can be linearly parameterized as:

$$\begin{aligned} B_i J_{h_i}^{-1} G_{\mathcal{R}_i}^{\top} \dot{v}_o + \left( B_i \frac{\partial}{\partial t} (J_{h_i}^{-1} G_{\mathcal{R}_i}^{\top}) + C_{q_i} J_{h_i}^{-1} G_{\mathcal{R}_i}^{\top} \right) v_o + g_{q_i} = \\ Y_{\mathcal{R}_i}(\ddot{x}_i, \dot{x}_i, v_o, \dot{v}_o) \vartheta_{\mathcal{R}_i}, \end{aligned}$$

with  $\ddot{x}_i := [\eta_o^{\top}, q_i^{\top}, p_{C_i/E_i}^{\top}, p_{C_i/o}^{\top}]^{\top} \in \mathbb{T} \times \mathcal{S}_i \times \mathbb{R}^6$ ,  $Y_{\mathcal{R}_i} : \mathbb{T} \times \mathcal{S}_i \times \mathbb{R}^{30} \rightarrow \mathbb{R}^{3 \times \ell_{\mathcal{R}}}$  being agent  $i$ 's regressor matrix, and  $\vartheta_{\mathcal{R}_i} \in \mathbb{R}^{\ell_{\mathcal{R}}}$ ,  $\ell_{\mathcal{R}} \in \mathbb{N}$  the respective vector of unknown, constant parameters. The aforementioned parameterization is written in vector form:

$$B J_h^{-1} G_{\mathcal{R}}^{\top} \dot{v}_o + \left( B \frac{\partial}{\partial t} (J_h^{-1} G_{\mathcal{R}}^{\top}) + C_q J_h^{-1} G_{\mathcal{R}}^{\top} \right) v_o + g_q = Y_{\mathcal{R}}(\ddot{x}, \dot{x}, v_o, \dot{v}_o) \vartheta_{\mathcal{R}},$$

where  $Y_{\mathcal{R}} := Y_{\mathcal{R}}(\ddot{x}, \dot{x}, v_o, \dot{v}_o) := \text{diag}\{[Y_{\mathcal{R}_i}]_{i \in \mathcal{N}}\} \in \mathbb{R}^{3N \times \ell_{\mathcal{R}}}$ , and  $\vartheta_{\mathcal{R}} := [\vartheta_{\mathcal{R}_1}^{\top}, \dots, \vartheta_{\mathcal{R}_N}^{\top}]^{\top} \in \mathbb{R}^{N \ell_{\mathcal{R}}}$ .

Therefore, the left-hand side of the coupled dynamics (2.78) can be written as

$$\tilde{B} \dot{v}_o + \tilde{C}_q v_o + \tilde{g}_q = Y_{\mathcal{R}_o}(\eta_o, \omega_o, v_o, \dot{v}_o) \vartheta_{\mathcal{R}_o} + G_{\mathcal{R}} J_h^{-T} Y_{\mathcal{R}}(\ddot{x}, \dot{x}, v_o, \dot{v}_o) \vartheta_{\mathcal{R}} \quad (2.79)$$

Let now  $\hat{\vartheta}_{\mathcal{R}} \in \mathbb{R}^{N \ell_{\mathcal{R}}}$ ,  $\hat{\vartheta}_{\mathcal{R}_o} \in \mathbb{R}^{\ell_{\mathcal{R}_o}}$ , be the estimates of  $\vartheta_{\mathcal{R}}$  and  $\vartheta_{\mathcal{R}_o}$ , respectively, by the agents, and the respective errors  $e_{\mathcal{R}_o} := \hat{\vartheta}_{\mathcal{R}_o} - \vartheta_{\mathcal{R}_o}$ , and  $e_{\mathcal{R}_o,o} := \hat{\vartheta}_{\mathcal{R}_o} - \vartheta_{\mathcal{R}_o}$ .

We provide next the proposed control protocol. First, we design the reference velocity signal  $v_{f_o} \in \mathbb{R}^6$  and the associated velocity error  $e_{v_o}$  as

$$v_{f_o} := v_d - K e_{\mathcal{R}} := \begin{bmatrix} \dot{p}_d \\ \omega_d \end{bmatrix} - \begin{bmatrix} k_p e_{p_o} \\ -k_{\eta} \frac{e_{\epsilon}}{e_{\varphi}^3} \end{bmatrix} \quad (2.80a)$$

$$e_{v_o} := v_o - v_{f_o}, \quad (2.80b)$$

where  $K = \text{diag}\{k_p I_3, k_\eta I_3\} \in \mathbb{R}^3$  is the positive definite gain matrix used in (2.21),  $e_{\mathcal{R}} := [e_{p_o}^\top, -\frac{e_\varphi^\top}{e_\varphi^3}]^\top$ , and  $v_d := v_d(t) := [\dot{p}_d^\top, \omega_d^\top]^\top$ . Note the difference in the definition of  $e_{\mathcal{R}}$  and  $e$  from (2.21), which will account to stabilizing the scalar quaternion error  $e_\varphi$  to either 1 or  $-1$ , depending on  $e_\varphi(0)$ , while guaranteeing that  $e_\varphi(t) \neq 0, \forall t \geq 0$  (provided that  $e_\varphi(0) \neq 0$ ), and rendering thus (2.80a) well defined.

We design now the control protocol as  $\tau : \mathcal{T}_{\mathcal{R}} \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^n$ , with

$$\tau := \tau(\chi_{\mathcal{R}}, t) = Y_r \hat{\vartheta}_{\mathcal{R}} + J_h^\top (G_{\mathcal{R}}^\dagger f_d + f_{\text{int}}), \quad (2.81)$$

where  $\chi_{\mathcal{R}} := [\dot{x}^\top, \dot{\dot{x}}^\top, e_{\mathcal{R}}^\top, e_{v_o}^\top, \hat{\vartheta}_{\mathcal{R}}^\top, \hat{\vartheta}_{\mathcal{R}_o}^\top, \eta_c^\top]^\top$ ,  $\mathcal{T}_{\mathcal{R}} := \{\chi_{\mathcal{R}} \in \mathbb{T}^{N+1} \times \mathbb{S} \times \mathbb{R}^{15N+15+N\ell_{\mathcal{R}}+\ell_{\mathcal{R}_o}} : e_\varphi \neq 0\}$ ,  $\eta_c := [\eta_{c_1}^\top, \dots, \eta_{c_N}^\top]^\top$ ,  $G_{\mathcal{R}}^\dagger$  is the Moore-Penrose pseudoinverse of  $G_{\mathcal{R}}$ ,  $f_d := Y_{o_r} \hat{\vartheta}_{\mathcal{R}_o} - e_{\mathcal{R}} - K_v e_{v_o}$  with  $K_v \in \mathbb{R}^{6 \times 6}$  the positive definite gain matrix used in (2.23),  $Y_r := Y_{\mathcal{R}}(\ddot{x}, \dot{x}, v_{f_o}, \dot{v}_{f_o})$ ,  $Y_{o_r} := Y_{\mathcal{R}_o}(\eta_o, \omega_o, v_{f_o}, \dot{v}_{f_o})$ , and  $f_{\text{int}} := f_{\text{int}}(q, \eta_c) : \mathbb{S} \times \mathbb{T}^N \rightarrow \mathbb{R}^{3N}$  is a term in the nullspace of  $G_{\mathcal{R}}$  to prevent contact slip, which will be designed later. Moreover, we design the adaptation signals

$$\dot{\hat{\vartheta}}_{\mathcal{R}} = \text{Proj}(\hat{\vartheta}_{\mathcal{R}}, -\Gamma Y_r^\top J_h^{-1} G_{\mathcal{R}}^\top e_{v_o}), \quad (2.82a)$$

$$\dot{\hat{\vartheta}}_{\mathcal{R}_o} = \text{Proj}(\hat{\vartheta}_{\mathcal{R}_o}, -\Gamma_o Y_{o_r}^\top e_{v_o}), \quad (2.82b)$$

where  $\Gamma \in \mathbb{R}^{N\ell_{\mathcal{R}} \times N\ell_{\mathcal{R}}}$ ,  $\Gamma_o \in \mathbb{R}^{\ell_{\mathcal{R}_o} \times \ell_{\mathcal{R}_o}}$  are constant positive definite gain matrices (as in (2.25)), and  $\text{Proj}()$  is the projection operator, which satisfies [?]:

$$(\hat{y} - y)^\top (W^{-1} \text{Proj}(y, Wz) - z) \leq 0, \quad (2.83)$$

for any symmetric positive definite  $W \in \mathbb{R}^{\ell_z \times \ell_z}$ , and  $\forall \hat{y}, y, z \in \mathbb{R}^{\ell_z}$ , for some  $\ell_z \in \mathbb{N}$ . Moreover, by appropriately choosing the initial conditions of the estimates  $\hat{\vartheta}_{\mathcal{R}}(0)$ ,  $\hat{\vartheta}_{\mathcal{R}_o}(0)$ , we guarantee via the projection operator that  $\hat{\vartheta}_{\mathcal{R}}(t)$ ,  $\hat{\vartheta}_{\mathcal{R}_o}(t)$  will stay uniformly bounded in predefined sets defined by finite constants  $\bar{\vartheta}_{\mathcal{R}}$ ,  $\bar{\vartheta}_{\mathcal{R}_o}$ , i.e.,  $\|\hat{\vartheta}_{\mathcal{R}}(t)\| \leq \bar{\vartheta}_{\mathcal{R}}$ ,  $\|\hat{\vartheta}_{\mathcal{R}_o}(t)\| \leq \bar{\vartheta}_{\mathcal{R}_o}$ ,  $\forall t \geq 0$ . Hence, we can achieve the boundedness of the respective errors as

$$\|e_{\vartheta_{\mathcal{R}}}(t)\| \leq \bar{e}_{\vartheta_{\mathcal{R}}} := \bar{\vartheta}_{\mathcal{R}} + \|\vartheta_{\mathcal{R}}\| \quad (2.84a)$$

$$\|e_{\vartheta_{\mathcal{R},o}}(t)\| \leq \bar{e}_{\vartheta_{\mathcal{R},o}} := \bar{\vartheta}_{\mathcal{R}_o} + \|\vartheta_{\mathcal{R}_o}\|. \quad (2.84b)$$

More details can be found in [? , Chapter 11].

We design next the internal force component  $f_{\text{int}}$  to guarantee slip prevention. Slip is addressed by ensuring the contact forces remain inside the friction cone as specified in (2.75). From (2.76), we have to guarantee that  $\Lambda_i(\mu_f) R_{c_i}^\top f_{c_i} \succeq 0, \forall i \in \mathcal{N}$ , or in vector form,

$$\Lambda(\mu_f) R_c^\top f_c \succeq 0, \quad (2.85)$$

where  $R_C := R_C(\eta_C) := \text{diag}\{[R_{C_i}]_{i \in \mathcal{N}}\}$ .

The design of the internal force component,  $f_{\text{int}}$ , to ensure (2.85) is performed as follows. First,  $f_{\text{int}}$  must be in the nullspace of  $G_R$ , i.e.,  $G_R f_{\text{int}} = 0$ . Second, the internal force must satisfy (2.85). Third, the normal component of the internal force with respect to the contact plane must always be positive (i.e. the manipulators cannot “pull” on the contact point). To enforce this condition we design  $f_{\text{int}_i} = f'_{\text{int}} R_{C_i} \ell_{\text{int},i}$ , where  $\ell_{\text{int},i} := [\ell_{\text{int},i_x}, \ell_{\text{int},i_y}, \ell_{\text{int},i_z}]^\top$  is the internal force direction in the contact frame  $C_i$ ,  $i \in \mathcal{N}$ , and  $f'_{\text{int}} \in \mathbb{R}_{>0}$  is a gain parameter to be designed. Without loss of generality let  $\ell_{\text{int},i_z}$  be aligned with the normal direction of the contact frame such that  $\ell_{\text{int},i_z} > 0$ ,  $i \in \mathcal{N}$ , ensures that only pushing forces are applied at each contact. Satisfaction of the aforementioned conditions is done by solving the following convex quadratic program to define the internal force controller

$$f_{\text{int}} = f'_{\text{int}} R_C \ell_{\text{int}}^* \quad (2.86a)$$

$$\ell_{\text{int}}^* = \underset{\ell_{\text{int}}}{\text{argmin}} \left\{ \sum_{i \in \mathcal{N}} \ell_{\text{int},i_x}^2 + \ell_{\text{int},i_y}^2 + \ell_{\text{int},i_z}^2 \right\} \quad (2.86b)$$

$$\text{s. t.} \quad (2.86c)$$

$$G_R R_C \ell_{\text{int}} = 0, \quad (2.86d)$$

$$\ell_{\text{int},i_z} > 0, \quad \forall i \in \mathcal{N}, \quad (2.86e)$$

$$\Lambda_i(\mu_f) \ell_{\text{int},i} \succ 0, \quad \forall i \in \mathcal{N}, \quad (2.86f)$$

where  $\ell_{\text{int}} := [\ell_{\text{int},1}^\top, \dots, \ell_{\text{int},N}^\top]^\top$ . Note that, since the contact points form a force-closure configuration, (2.86) always has a feasible solution.

Finally, to satisfy (2.85),  $f_{\text{int}}$  must apply sufficient force inside the friction cone to reject perturbations that will arise during the manipulation motion that can push the contact force outside of the friction cone. Rejection of these perturbations is performed by designing the gain  $f'_{\text{int}}$  as follows. For simplicity we define the terms  $k_{\text{int}} := \Lambda(\mu_f) R_C^\top G_R^\dagger f_d$ ,  $l_{\text{int}} = \Lambda(\mu_f) \ell_{\text{int}}^*$ , and we denote by  $k_{\text{int},j}$  and  $l_{\text{int},j}$  the  $j$ th scalar element of  $k_{\text{int}}$  and  $l_{\text{int}}$  respectively for  $j \in \{1, \dots, Nl_f\}$ .

Noting that  $\Lambda(\mu_f) \ell_{\text{int}}^* \succ 0$  from (2.86), we define the *decreasing* function  $\kappa_{\text{int}} : \mathbb{R} \rightarrow \mathbb{R}_{\geq 0}$  as

$$\kappa_{\text{int}}(x) := \begin{cases} -x, & \text{if } x \leq -1, \\ q_{\text{int}}(x), & \text{if } -1 \leq x \leq 0, \\ 0, & \text{if } x \geq 0 \end{cases},$$

---

<sup>1</sup>We use the notation  $f_{\text{int}} = [f_{\text{int}_1}^\top, \dots, f_{\text{int}_N}^\top]^\top$ .

where  $q_{\text{int}}(x) \geq 0, \forall x \in [-1, 0]$ , is an appropriate polynomial that ensures continuous differentiability of  $\kappa_{\text{int}}$ , for instance  $q_{\text{int}}(x) = x^3 + 2x^2$ . Then one can verify that  $\kappa_{\text{int}}(x) + 1 \geq -x, \forall x \in \mathbb{R}$ . We now design the magnitude scaling for the internal forces as

$$f'_{\text{int}} = \frac{\kappa(\min_j \{k_{\text{int},j}\}) + 1 + \epsilon_f}{\min_j \{l_{\text{int},j}\}}, \quad (2.87)$$

where  $\epsilon_f \in \mathbb{R}_{>0}$  is a tuning gain. The intuition behind (2.87) is to upper bound elements of the control and the system dynamics to prevent either from pushing the contact force outside of the friction cone. The term  $\kappa(\min_j \{k_j\}) + 1$  cancels out any effects from  $f_d$ . The term  $\epsilon_f$  handles the system dynamics, which is guaranteed to be bounded in the following theorem.

**Remark 2.10.** The internal force control presented here accounts for the dynamics of the system by appropriately scaling  $f'_{\text{int}}$ , which rejects perturbations from causing slip. However, as opposed to [?], we relax the condition that  $\epsilon_f$  must upper bound all of the dynamics terms by exploiting knowledge of the applied controller via the term  $\kappa_{\text{int}}(\min_j \{k_{\text{int},j}\})$ . This reduces the amount of squeezing force applied to prevent crushing the object.

The stability and slip prevention guarantees of the proposed controller are presented in the following theorem.

**Theorem 2.5.** Consider  $N$  robotic agents in contact with an object, described by the dynamics (2.70), (2.73), and suppose Assumptions 2.10 and 2.11 hold. a the desired object pose  $[p_d^\top, \eta_d^\top]^\top : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^3 \times \mathbb{S}^3$  be bounded with bounded first and second derivatives. Moreover, assume that  $e_\varphi(0) \neq 0$  and  $f_{C_i}^{C_i}(0) \in \text{Int}(\mathcal{F}_{C_i}(\mu_f)), \forall i \in \mathcal{N}$ . Then, the control protocol (2.80a)-(2.87) guarantees that  $\lim_{t \rightarrow \infty} (e_{p_o}(t), e_\eta(t)) = (0, [\pm 1, 0^\top]^\top)$ , as well as boundedness of all closed-loop signals. Moreover, by choosing a sufficiently large  $\epsilon_f$  in (2.87), it holds that  $f_{C_i}^{C_i}(t) \in \mathcal{F}_{C_i}, \forall t > 0, i \in \mathcal{N}$ .

*Proof.* Consider the stack vector state  $\chi := [e_{p_o}^\top, e_\epsilon^\top, e_{v_o}^\top, e_{\vartheta_R}^\top, e_{\vartheta_{R,o}}^\top]^\top \in \mathcal{X} := \mathbb{R}^{12+N\ell_R+\ell_{R,o}}$ . Next, note by (2.70), (2.71), and (2.77) that, when  $f_{C_i}^{C_i} \in \mathcal{F}_{C_i}$ , each  $f_{C_i}^{C_i}$  can be written as a function of the stack state, i.e.,  $f_{C_i}^{C_i} = f_{C_i}^{C_i}(\chi)$ ,  $\forall i \in \mathcal{N}$ . Consider also the set

$$\begin{aligned} \mathcal{U} := \{ \chi \in \mathcal{X} : \|e_\epsilon\| < \bar{e}_\epsilon, \|e_{p_o}\| < \bar{e}_{p_o}, \|e_{v_o}\| < \bar{e}_{v_o}, \|e_{\vartheta_R}\| < \bar{e}_{\vartheta_R}, \\ \|e_{\vartheta_{R,o}}\| < \tilde{e}_{\vartheta_{R,o}}, f_{C_i}^{C_i}(\chi) \in \text{Int}(\mathcal{F}_{C_i}), \forall i \in \mathcal{N} \}, \end{aligned}$$

for some positive constants  $\bar{e}_\epsilon, \bar{e}_{p_o}, \bar{e}_{v_o}$  satisfying  $\|e_\epsilon(0)\| \leq \bar{e}_\epsilon, \|e_{v_o}(0)\| < \bar{e}_{v_o}, \|e_{p_o}(0)\| < \bar{e}_{p_o}$ , and  $\tilde{e}_{\vartheta_R}, \bar{e}_{\vartheta_{R,o}}$  larger than  $\bar{e}_{\vartheta_R}, \bar{e}_{\vartheta_{R,o}}$ , respectively, which were introduced in (2.84). Note that  $\chi(0) \in \mathcal{U}$ . Next, by using (2.81) and (2.82),

one obtains the closed-loop dynamics  $\dot{\chi} = h_\chi(\chi, t)$ , where  $h_\chi : \mathcal{X} \times \mathbb{R}_{\geq 0} \rightarrow \mathcal{X}$  is a function that is continuous in  $t$  and locally Lipschitz in  $\chi$ . Then, according to Theorem A.1 of Appendix A, there exists a positive time constant  $t_{\max} > 0$  and a unique solution  $\chi : [0, t_{\max}) \rightarrow \mathcal{U}$ , i.e., defined for  $[0, t_{\max})$  and satisfying  $\chi(t) \in \mathcal{U}, \forall t \in [0, t_{\max})$ . Hence, slip is prevented and the dynamics (2.78) are well-defined, for  $t \in [0, t_{\max})$ .

Let now the Lyapunov function

$$V_f := \frac{1}{2} e_{p_o}^\top e_{p_o} + \frac{2}{e_\varphi^2} + \frac{1}{2} e_{v_o}^\top \tilde{B} e_{v_o} + \frac{1}{2} e_{\vartheta_{\mathcal{R}}}^\top \Gamma^{-1} e_{\vartheta_{\mathcal{R}}} + \frac{1}{2} e_{\vartheta_{\mathcal{R},o}}^\top \Gamma_o^{-1} e_{\vartheta_{\mathcal{R},o}}. \quad (2.88)$$

Since  $e_\varphi(0) \neq 0$ , it holds that  $V_f(0) \leq \bar{V}_{f0}$  for a finite positive  $\bar{V}_{f0}$ . Differentiation of  $V_f$  results in:

$$\begin{aligned} \dot{V}_f = & e_{\mathcal{R}}^\top (v_o - v_d) + \frac{1}{2} e_{v_o}^\top \dot{\tilde{B}} e_{v_o} + e_{v_o}^\top (-\tilde{C}_q v_o - \tilde{g}_q - \tilde{B} \dot{v}_{f_o} + G_{\mathcal{R}} J_h^{-T} \tau) \\ & + e_{\vartheta_{\mathcal{R}}}^\top \Gamma^{-1} \dot{\vartheta}_{\mathcal{R}} + e_{\vartheta_{\mathcal{R},o}}^\top \Gamma_o^{-1} \dot{\vartheta}_{\mathcal{R},o}. \end{aligned}$$

Exploitation of the skew symmetry of  $\tilde{B} - 2\tilde{C}_q$ , use of  $v_o = e_{v_o} + v_{f_o}$ , use of (2.79), and substitution of the control law (2.81) results in:

$$\begin{aligned} \dot{V}_f = & -e_{\mathcal{R}}^\top K e_{\mathcal{R}} - e_{v_o}^\top K_v e_{v_o} + e_{v_o}^\top (Y_{o_r} e_{\vartheta_{\mathcal{R},o}} + G_{\mathcal{R}} J_h^{-T} Y_r e_{\vartheta_{\mathcal{R}}}) \\ & + e_{\vartheta_{\mathcal{R}}}^\top \Gamma^{-1} \dot{\vartheta}_{\mathcal{R}} + e_{\vartheta_{\mathcal{R},o}}^\top \Gamma_o^{-1} \dot{\vartheta}_{\mathcal{R},o}, \end{aligned}$$

where we used the fact that  $G_{\mathcal{R}} f_{\text{int}} = 0$  through (2.86). Finally, by substituting the adaptation laws (2.82), we obtain

$$\begin{aligned} \dot{V}_f = & -e_{\mathcal{R}}^\top K e_{\mathcal{R}} - e_{v_o}^\top K_v e_{v_o} + e_{\vartheta_{\mathcal{R}}}^\top \left( \Gamma^{-1} \text{Proj}(\vartheta_{\mathcal{R}}, -Y_r^\top J_h^{-1} G_{\mathcal{R}}^\top e_{v_o}) \right. \\ & \left. + \Gamma Y_r^\top J_h^{-1} G_{\mathcal{R}}^\top e_{v_o} \right) + e_{\vartheta_{\mathcal{R},o}}^\top \left( \Gamma_o^{-1} \text{Proj}(\vartheta_{\mathcal{R},o}, -Y_{o_r}^\top e_{v_o}) + \Gamma_o Y_{o_r}^\top e_{v_o} \right) \end{aligned}$$

which, by invoking the projection operator property (2.83) becomes  $\dot{V}_f \leq -e_{\mathcal{R}}^\top K e_{\mathcal{R}} - e_{v_o}^\top K_v e_{v_o}$ . Thus  $\dot{V}_f$  is negative semi-definite, and  $V_f$  is bounded in a compact set as  $V_f(t) \leq V_f(0), \forall t \in [0, t_{\max})$ . In addition,  $e_\varphi(t) \neq 0, \forall t \in [0, t_{\max})$ . Hence, the terms  $e_{p_o}(t)$ ,  $e_\epsilon(t)$ ,  $e_\varphi(t)$  are bounded in a compact set defined by  $V_f(0)$  and not dependent on  $t_{\max}, \forall t \in [0, t_{\max})$ . Therefore, since  $p_d(t)$  and  $\eta_d(t)$  are bounded and have bounded derivatives, one concludes that  $p_o(t)$ ,  $\eta_o(t)$ ,  $v_o(t)$ , and  $v_{f_o}(t)$ ,  $\dot{v}_{f_o}(t)$  are also bounded in compact sets,  $\forall t \in [0, t_{\max})$ . This also implies boundedness of  $\ddot{x}, \dot{x}$ , as introduced in (2.78), which, along with Assumption 2.11 and properties of Euler-Lagrange systems [?], implies that  $Y_{\mathcal{R}}(), Y_r, Y_{\mathcal{R},o}(), Y_{o_r}$  are also

bounded in compact sets that are independent of  $t_{\max}$ ,  $\forall t \in [0, t_{\max}]$ . We prove next the slip prevention using the design of the internal force component  $f_{\text{int}}$ . By using (2.70), (2.73) and (2.77), one obtains the following expression for the interaction forces:

$$\begin{aligned} f_C = & W_h^{-1} \left( J_h B^{-1} \left[ \tau - g_q - \left( C_q J_h^{-1} G_{\mathcal{R}}^{\top} + B \frac{d}{dt} (J_h^{-1} G_{\mathcal{R}}^{\top}) \right) v_o \right] \right. \\ & \left. + G_{\mathcal{R}}^{\top} M_o^{-1} (C_o v_o + g_o) \right), \end{aligned} \quad (2.89)$$

where  $W_h := J_h B^{-1} J_h^{\top} + G_{\mathcal{R}}^{\top} M_o^{-1} G_{\mathcal{R}}$ , which, by replacing  $\tau$ , using  $v_{f_o} = e_{v_o} + v_o$  and (2.79), adding and subtracting  $W_h^{-1} G_{\mathcal{R}}^{\top} M_o^{-1} G_{\mathcal{R}} G_{\mathcal{R}}^{\dagger} f_d$  and adding  $W_h^{-1} G_{\mathcal{R}}^{\top} M_o^{-1} G_{\mathcal{R}} f_{\text{int}} = 0$ , becomes

$$f_C = G_{\mathcal{R}}^{\dagger} f_d + f_{\text{int}} + h_f \quad (2.90)$$

where

$$\begin{aligned} h_f := & W_h^{-1} J_h B^{-1} (g_q - Y_{\mathcal{R}}(\ddot{x}, \dot{x}, e_{v_o}, \dot{e}_{v_o}) \vartheta_{\mathcal{R}} + Y_r e_{\vartheta_{\mathcal{R}}}) + W_h^{-1} G_{\mathcal{R}}^{\top} M_o (e_{\mathcal{R}} \\ & + K_v e_{v_o} + Y_{\mathcal{R}_o}(\eta_o, \omega_o, e_{v_o}, \dot{e}_{v_o}) \vartheta_{\mathcal{R}_o} - Y_{o_r} e_{\vartheta_{\mathcal{R}, o}} - g_o). \end{aligned}$$

By combining the aforementioned expression with (2.85), one obtains the following condition for slip prevention:

$$\Lambda(\mu_f) R_C^{\top} f_{\text{int}} \succeq -\Lambda(\mu_f) R_C^{\top} G^{\dagger} f_d - \Lambda(\mu_f) R_C^{\top} h_f. \quad (2.91)$$

Note that due to the aforementioned Lyapunov analysis, as well as the adaptation laws (2.82) through the projection operator,  $e_{\mathcal{R}}(t)$ ,  $e_{v_o}(t)$ ,  $\dot{e}_{v_o}(t)$ ,  $e_{\vartheta_{\mathcal{R}}}(t)$ ,  $e_{\vartheta_{\mathcal{R}, o}}(t)$  are bounded in compact set independent of  $t_{\max}$ ,  $\forall t \in [0, t_{\max}]$ . By combining this with the aforementioned analysis, we conclude that  $h_f$  is bounded for all  $\forall t \in [0, t_{\max}]$  in a compact set, independent of  $t_{\max}$ . Hence, by denoting  $\varepsilon_h$  the maximum bound of the elements of  $\pm \Lambda(\mu_f) R_C^{\top} h_f$  and using the designed internal force component  $f_{\text{int}} = f'_{\text{int}} R_C \ell_{\text{int}}$ , a sufficient condition for (2.91) to hold is for the  $j$ th element to satisfy

$$l_{\text{int}, j} f'_{\text{int}} \geq -k_{\text{int}, j} + \varepsilon_h,$$

$\forall j \in \{1, \dots, Nl_f\}$ . By substituting (2.87), the left side satisfies

$$\begin{aligned} l_{\text{int}, j} \frac{\kappa(\min_j \{k_{\text{int}, j}\}) + 1 + \epsilon_f + \delta_f}{\min_j \{l_{\text{int}, j}\}} & \geq \kappa(\min_j \{k_{\text{int}, j}\}) + 1 + \epsilon_f + \delta_f \\ & \geq -k_{\text{int}, j} + \epsilon_f, \end{aligned}$$

where we use  $\kappa_{\text{int}}(x) \geq 0$ ,  $\kappa_{\text{int}}(x) + 1 \geq -x$ ,  $\forall x \in \mathbb{R}$ , and  $\kappa_{\text{int}}(\min_j \{k_{\text{int}, j}\}) \geq \kappa_{\text{int}}(k_{\text{int}, j})$ ,  $\forall j \in \{1, \dots, Nl_f\}$ , since  $\kappa_{\text{int}}()$  is decreasing. Hence, by choosing

a large enough  $\epsilon_f$  we guarantee  $\epsilon_f \geq \varepsilon_h$  and hence contact slip is avoided  $\forall t \in [0, t_{\max}]$ . In fact, the internal forces analysis above and the fact that  $\Lambda(\mu_f)$  defines pyramid constraints imply that  $f_{C_i}^{C_i} \in \bar{\mathcal{F}}_{C_i}$ , where  $\bar{\mathcal{F}}_{C_i}$  is a compact subset of  $\text{Int}(\mathcal{F}_{C_i})$ ,  $\forall i \in \mathcal{N}$ . Therefore, since  $e_{\vartheta_{\mathcal{R}}}$  and  $e_{\vartheta_{\mathcal{R},o}}$  are uniformly bounded through the projection operator by  $\bar{e}_{\vartheta_{\mathcal{R}}}$  and  $\bar{e}_{\vartheta_{\mathcal{R},o}}$ , respectively, by choosing large enough  $\bar{e}_{p_o}$ ,  $\bar{e}_\epsilon$ , and  $\bar{e}_{v_o}$  in the definition of  $\mathcal{U}$ ,  $\chi(t)$  belongs to a compact subset  $\bar{\mathcal{U}}$  of  $\mathcal{U}$ ,  $\forall t \in [0, t_{\max}]$ . Thus by invoking Theorem A.2 of Appendix A, it follows that  $t_{\max} = \infty$ .

Note, finally, that  $\tau(\chi_{\mathcal{R}}(t), t)$ , as designed in (2.81), is bounded,  $\forall t \geq 0$ . Therefore, one can conclude that  $\dot{e}_{v_o}(t)$  and thus  $\ddot{q}(t)$  is bounded,  $\forall t \geq 0$ . Hence, it follows that  $\dot{V}_f(t)$  is also bounded,  $\forall t \geq 0$ . Thus by invoking Barbalat's lemma (Lemma A.1 of Appendix A), it follows that  $\lim_{t \rightarrow \infty} \dot{V}_f(t) = 0$  and so  $\lim_{t \rightarrow \infty} e_{\mathcal{R}}(t) \rightarrow 0$  and  $\lim_{t \rightarrow \infty} e_{v_o}(t) \rightarrow 0$ . This implies that  $\lim_{t \rightarrow \infty} e_\epsilon(t) \rightarrow 0$ , which, given that  $e_\eta$  is a unit quaternion and  $e_\varphi(t) \neq 0$ ,  $\forall t \geq 0$ , ensures asymptotic stability of the pose error as  $\lim_{t \rightarrow \infty} (e_{p_o}(t), e_\zeta(t)) = (0, [\text{sgn}(e_\varphi(0)), 0^\top]^\top)$ .

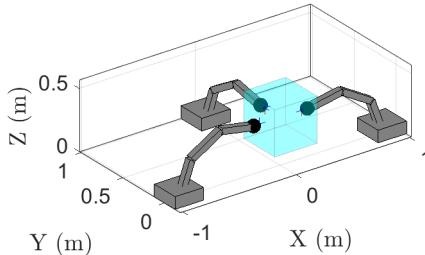
□

**Remark 2.11.** Note that the bound  $\varepsilon_h$  of  $h_f$  in (2.90) can be computed a priori. In practice, the terms  $\vartheta_{\mathcal{R}}$ ,  $\vartheta_{\mathcal{R},o}$ , which concern masses and moments of inertia of the object and the agents, can be known a priori up to a certain accuracy, leading thus to respective bounds. Hence, one can compute upper bounds for  $V_f(0)$  and hence for  $e_{\mathcal{R}}$ ,  $e_{v_o}$ ,  $e_{\vartheta_{\mathcal{R}}}$ , and  $e_{\vartheta_{\mathcal{R},o}}$ . Since the structure of the dynamic terms is known, this can also lead to a bound of the terms  $W_h^{-1}$ ,  $B^{-1}$ ,  $M_o$ ,  $Y_{\mathcal{R}}(\cdot)$ ,  $Y_{\mathcal{R},o}(\cdot)$ ,  $Y_r$ , and  $Y_{o_r}$  that appear in  $h_f$ . Hence, tuning of  $\epsilon_f$  to overcome  $\varepsilon_h$  can be performed off-line.

## Simulation Results

The proposed control algorithm ensures asymptotic stability for cooperative manipulation with rolling contacts, as well as no slip, while being robust to dynamic uncertainties of the object-robot system. In this section, we implement the proposed control scheme on three 6 DoF mobile manipulators consisting of a 3 DoF, 3 kg base (X-Y translation, rotation about Z) and a 3-DoF manipulator with 3 identical links of length 0.3 m and mass of 0.5 kg each, as depicted in Fig. 2.24. The objective is to transport a 2 kg box along the desired reference trajectory defined by  $p_d(t) := [0.1 \sin(0.125t), 0.1 \sin(0.125t), 0.1 \sin(0.125t)]^\top$  m,  $\eta_d(t) := [\cos(0.1 \sin(0.125t)), 0, 0, \sin(0.1 \sin(0.125t))]^\top$ . The control gains used are:  $k_p = 1$ ,  $k_\eta = .5$ ,  $K_v = \text{diag}[5, 5, 5, 2, 2, 2]$ ,  $\epsilon_f = 0.1$ ,  $\Gamma_o = 0.5I_{\ell_{\mathcal{R},o}}$ ,  $\Gamma = 0.5I_{N\ell_{\mathcal{R}}}$ . The control is implemented with 30% error in all uncertain parameter (including the object center of mass), and the projection operator enforces the following bounds on the uncertain terms:  $\hat{\theta}_{\mathcal{R}} = 2.25$ ,  $\hat{\theta}_{\mathcal{R},o} = 1.5$ .

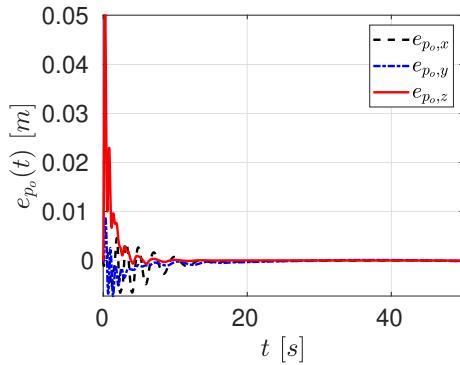
The simulation results are depicted in Figs. 2.25–2.29 for 50 seconds. More specifically, Figs. 2.25 and 2.26 show the resulting error trajectories of the object-agent system, which satisfy  $\lim_{t \rightarrow \infty} e_{p_o}(t) = 0$ ,  $\lim_{t \rightarrow \infty} e_\epsilon(t) = 0$ , and  $\lim_{t \rightarrow \infty} e_\varphi(t) = \text{sgn}(e_\varphi(0)) = 1$  in the presence of rolling effects. Fig. 2.27 illustrates the boundedness of the uncertain parameters,  $\hat{\vartheta}_{\mathcal{R}}, \hat{\vartheta}_{\mathcal{R}_o}$ , that is enforced by the proposed control scheme. Fig. 2.28 shows the required friction,  $\mu_{ri} := \frac{\sqrt{f_{C,x_i}^2 + f_{C,y_i}^2}}{f_{C,n_i}}$ , which denotes the minimum friction coefficient necessary to prevent slip throughout the motion [?]. If the required friction surpasses the true coefficient, then the contact point will slip and the grasp is compromised. As shown in Fig. 2.28, however, the required friction for each contact is below the true coefficient of  $\mu_f = 0.9$ , which indicates that slip is prevented as guaranteed by the proposed method. Finally, Fig. 2.29 depicts the control inputs of the agents. As predicted by the theoretical analysis, asymptotic error stability as well as contact slip prevention are achieved.



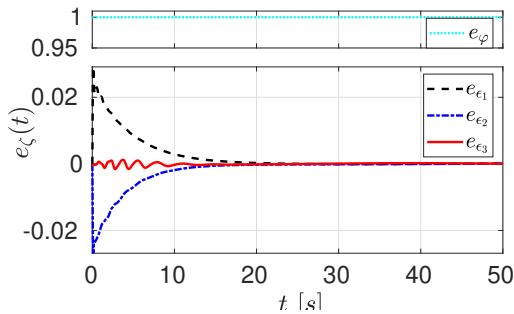
**Figure 2.24:** Initial configuration of the system that consists of three mobile manipulators and a rigid object.

### 2.3.3 Decentralized Scheme

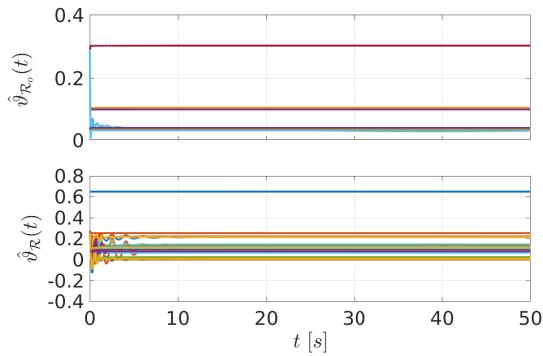
This section presents a decentralized extension of the aforementioned scheme via event-triggered communication among the agents. The event-triggered control requires an update law (to be designed) that updates relevant variables at each time  $t_k \in \mathbb{R}_{>0}$  for  $k \in \mathbb{N}$ . We use the subscript with  $k$  to denote a variable that is held constant over the time interval  $[t_k, t_{k+1})$  and updated at each  $t_k$ . The variables communicated among the agents at time  $t_k$  are  $p_{C_i/o}(t_k)$  and  $p_{C_i/E_i}(t_k)$ , allowing all the agents to reconstruct  $G_{\mathcal{R},k} := [G_{\mathcal{R},k_1}, \dots, G_{\mathcal{R},k_N}]$ , as well as  $R_{C,k} := \text{diag}\{R_{C,k_i}\}_{i \in \mathcal{N}}$ , with  $G_{\mathcal{R},k_i} := G_{\mathcal{R}_i}(p_{C_i/o}(t_k))$ ,  $R_{C,k_i} := R_{C_i}(t_k)$ ,  $\forall i \in \mathcal{N}$ . The event-triggered



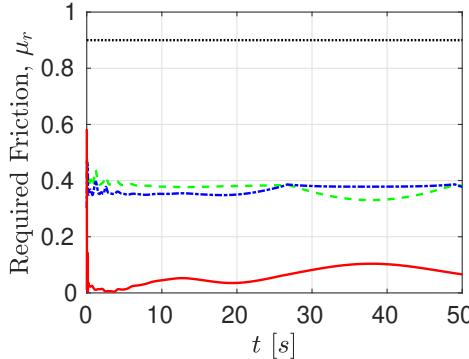
**Figure 2.25:** The evolution of the position error,  $e_{p_o}(t)$ ,  $\forall t \in [0, 50]$ .



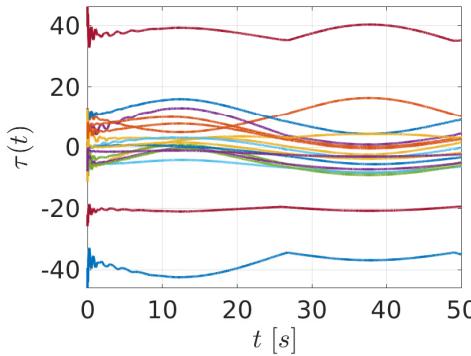
**Figure 2.26:** The evolution of  $e_\varphi(t)$ ,  $e_\zeta(t)$ ,  $\forall t \in [0, 50]$ .



**Figure 2.27:** The evolution of  $\hat{v}_{R_o}(t)$ ,  $\hat{v}_{\mathcal{R}}(t)$ ,  $\forall t \in [0, 50]$ .



**Figure 2.28:** The required friction to prevent slip for the three agents. The black dashed line represents  $\mu_f = 0.9$ .



**Figure 2.29:** The resulting inputs  $\tau(t)$  of the agents,  $\forall t \in [0, 50]$ .

manipulation control law is defined as:

$$\tau_{k_i}(\chi_{\mathcal{R}_i}, t) = Y_{r_i} \hat{\vartheta}_{\mathcal{R}_i} + J_{h_i}^\top (\lambda_{k_i} + f_{\text{int}, k_i}), \quad (2.92a)$$

$$\lambda_{k_i} := -G_{\mathcal{R}_i}^\top K_v e_{v_o} + G_{\mathcal{R}, k_i}^* (Y_{o_r} \hat{\vartheta}_{\mathcal{R}_o} - e_{\mathcal{R}}) \quad (2.92b)$$

with  $\chi_{\mathcal{R}_i} := [\ddot{x}_i^\top, \dot{\ddot{x}}_i^\top, e_{\mathcal{R}}^\top, e_{v_o}^\top, \hat{\vartheta}_{\mathcal{R}_i}^\top, \hat{\vartheta}_{\mathcal{R}_o}^\top, \eta_{C_i}^\top]^\top \in \mathcal{T}_{\mathcal{R}_i} := \{\chi_{\mathcal{R}_i} \in \mathbb{T}^2 \times S_i \times \mathbb{R}^{30+\ell_{\mathcal{R}}+\ell_{\mathcal{R}_o}} \times \mathbb{T} : e_\varphi \neq 0\} \forall i \in \mathcal{N}$ , with the decentralized version of the adaptive update laws defined by (2.82):

$$\dot{\hat{\vartheta}}_{\mathcal{R}_i} = \text{Proj}(\hat{\vartheta}_{\mathcal{R}_i}, -\Gamma_i Y_{r_i}^\top J_{h_i}^{-1} G_{\mathcal{R}_i}^\top e_{v_o}), \quad (2.93a)$$

$$\dot{\hat{\vartheta}}_{\mathcal{R}_o} = \text{Proj}(\hat{\vartheta}_{\mathcal{R}_o}, -\Gamma_o Y_{o_r}^\top e_{v_o}). \quad (2.93b)$$

where  $Y_{r_i} := Y_{\mathcal{R}_i}(\ddot{x}_i, \dot{\ddot{x}}_i, v_{f_o}, \dot{v}_{f_o})$ ,  $\Gamma_i \in \mathbb{R}^{\ell_{\mathcal{R}} \times \ell_{\mathcal{R}}}$  is such that  $\Gamma = \text{diag}\{\Gamma_i\}_{i \in \mathcal{N}}$  from (2.82), and  $f_{\text{int},k_i} \in \mathbb{R}^3$  is the event-triggered internal force control yet to be designed. Similarly,  $G_{\mathcal{R},k}^*$  is a generalized inverse of the grasp map at  $t = t_k$  and we denote  $G_{\mathcal{R},k}^* := [G_{\mathcal{R},k_1}^*, \dots, G_{\mathcal{R},k_N}^*]^\top$ . Let  $\Delta G_{\mathcal{R}} := G_{\mathcal{R}} - G_{\mathcal{R},k}$ ,  $\Delta p_{oC} := [\Delta p_{C_i/o}^\top, \dots, \Delta p_{C_N/o}^\top]^\top := p_{oC} - p_{oC}(t_k)$ ,  $\Delta p_{EC} := [\Delta p_{C_i/EC}^\top, \dots, \Delta p_{C_N/EC}^\top]^\top := p_{EC} - p_{EC}(t_k)$ ,  $\Delta R_C := R_C - R_{C,k}$  denote the triggering errors. Note that (2.92) is only dependent on the full grasp map,  $G_{\mathcal{R},k}$ , in the term  $G_{\mathcal{R},k}^*$ , whereas the adaptation laws (2.93) and remainder of the control depend on  $G_{\mathcal{R},k}$ .

Moreover, in order to ensure no slip, recall that the condition (2.85) must hold. Notice that when there is no communication between agents,  $R_C$  and  $G_{\mathcal{R}}$  are unknown as each agent only has knowledge of  $R_{(de)rolling}C,k$  and  $G_{\mathcal{R},k}$ . Thus if the original internal force control (2.86) is implemented with  $R_{C,k}$ ,  $G_{\mathcal{R},k}$  the errors  $\Delta R_C$  and  $\Delta G_{\mathcal{R}}$  may induce slip. To account for this, we use a conservative  $\mu'_f \in (0, \mu_f)$  that effectively shrinks the friction pyramid. The design of  $\mu'_f$  is dependent on the allowable error that will result from triggering. This presents a trade-off where to reduce communication (i.e reduced triggering), a smaller more conservative  $\mu'_f$  is required, and vice versa (i.e. larger  $\mu'_f$  requires more communication between agents). We introduce the following Lemma to compute  $\mu'_f$ .

**Lemma 2.3.** *Let  $\mu_f > 0$ , and  $W \in \mathbb{R}^{3 \times 3}$  satisfying  $\|W\| \leq \delta_c$ , where  $\delta_c$  is a positive constant satisfying*

$$\delta_c < \frac{\sqrt{\mu_f^2 + 1} - 1}{\sqrt{\mu_f^2 + 1} + 1} < 1, \quad (2.94)$$

and define

$$\mu'_f := \tan \left( \tan^{-1}(\mu_f) - \cos^{-1} \left( \frac{1 - \delta_c}{1 + \delta_c} \right) \right). \quad (2.95)$$

If  $y \in \mathbb{R}^3$  satisfies  $y \in \mathcal{F}_{C_i}(\mu'_f)$ , then  $(I_3 - W)y \in \mathcal{F}_{C_i}(\mu_f)$ .

*Proof.* Denote by  $\theta_W$  the angle defined by  $y$  and  $(I_3 - W)y$ , satisfying

$$\cos(\theta_W) = \frac{y^\top (I_3 - W)y}{\|(I_3 - W)y\| \|y\|} = \frac{y^\top (I_3 - W_{\text{sym}})y}{\|(I_3 - W)y\| \|y\|}, \quad (2.96)$$

where  $W_{\text{sym}} := \frac{W + W^\top}{2}$ . Note that  $\lambda_{\max}(W_{\text{sym}}) \leq \|W_{\text{sym}}\| \leq \|W\| \leq \delta_c < 1$  and hence  $I_3 - W_{\text{sym}}$  has strictly positive eigenvalues, rendering  $\cos(\theta_W)$  positive and  $|\theta_W| < \frac{\pi}{2}$ .

Moreover, it holds  $\|(I_3 - W)y\| \|y\| \leq (1 + \|W\|) \|y\|^2$  as well as  $y^\top(I_3 - W_{\text{sym}})y \geq \lambda_{\min}(I_3 - W_{\text{sym}})\|y\|^2 \geq (1 - \lambda_{\max}(W_{\text{sym}}))\|y\|^2 \geq (1 - \|W\|)\|y\|^2$ . Hence, by taking into account  $\|W\| \leq \delta_c$  and (2.94), (2.96) becomes

$$\cos(\theta_W) \geq \frac{1 - \|W\|}{1 + \|W\|} \geq \frac{1 - \delta_c}{1 + \delta_c} > \frac{1}{\sqrt{\mu_f^2 + 1}} = \cos(\tan^{-1}(\mu_f)),$$

implying

$$|\theta_W| \leq \cos^{-1}\left(\frac{1 - \delta_c}{1 + \delta_c}\right) < \tan^{-1}(\mu_f) \quad (2.97)$$

and rendering hence  $\mu'_f$  positive.

In order for  $(I_3 - W)y$  to belong to  $\mathcal{F}_{C_i}(\mu_f)$ ,  $y$  must lie in a new friction cone  $\mathcal{F}_{C_i}(\mu^*)$ , whose angle  $\tan^{-1}(\mu^*)$  must be reduced by  $|\theta_W|$  from  $\tan^{-1}(\mu_f)$ , leading thus to  $\mu^* := \tan(\tan^{-1}(\mu_f) - |\theta_W|)$ . In view of (2.97), it holds that  $\mu'_f < \mu_f^*$  and hence  $\mathcal{F}_{C_i}(\mu'_f) \subset \mathcal{F}_{C_i}(\mu_f^*)$  and  $y \in \mathcal{F}_{C_i}(\mu'_f) \Rightarrow y \in \mathcal{F}_{C_i}(\mu_f^*) \Rightarrow (I_3 - W)y \in \mathcal{F}_{C_i}(\mu_f)$ .  $\square$

The event-triggered internal force controller is now defined as follows:

$$f_{\text{int},k} = f'_{\text{int},k} R_{C,k} \ell_{\text{int},k}^* \quad (2.98a)$$

$$\ell_{\text{int},k}^* = \underset{\ell}{\operatorname{argmin}} \ell^\top \ell \quad (2.98b)$$

$$\text{s. t.} \quad (2.98c)$$

$$G_{R,k} R_{C,k} \ell = 0 \quad (2.98d)$$

$$\Lambda_i(\mu'_f) \ell_i \succ 0, \forall i \in \mathcal{N}, \quad (2.98e)$$

$$f'_{\text{int},k} := \frac{\kappa(\min_j\{b_{k,j}\}) + 1 + \epsilon_f}{\min_j\{l_{k,j}\} - \varepsilon_d \delta_p \max_j\{\ell_{\text{int},k,j}^*\}}, \quad (2.99)$$

$$b_k := \Lambda(\mu'_f) R_{C,k}^\top \lambda_k, l_k := \Lambda(\mu'_f) \ell_{\text{int},k}^*$$

where  $\lambda_k := [\lambda_{k,1}^\top, \dots, \lambda_{k,N}^\top]^\top$ ,  $b_{k,j}$ ,  $\ell_i \in \mathbb{R}^3$  is the  $i$ th agent's part in vector  $\ell$ ,  $l_{k,j}$  and  $\ell_{\text{int},k,j}$  are the  $j$ th scalar elements of  $b_k$ ,  $l_k$  and  $\ell_{\text{int},k}$ , respectively,  $\varepsilon_d, \epsilon_f \in \mathbb{R}_{>0}$  are design parameters, and  $\mu'_f$  is defined as in (2.95). Note that  $\ell_{\text{int},k}^*$  is constant for  $t \in [t_k, t_{k+1})$  such that it need only be computed at each  $k$  update.

Now that the full control protocol is defined, the final step is to define the event-triggering condition to update  $G_k$  and  $R_{C,k}$  which are:

$$\|\Delta p_{C_i/o}\| = \delta_p, \quad (2.100a)$$

$$\delta_p := \min \left\{ \frac{1}{\sum_i \|G_{\mathcal{R},k_i}^*\|} \min\{k_1 - c_{\mathcal{R}_2}, 2k_2 - c_{\mathcal{R}_2}\}, \frac{\delta_c}{2\varepsilon_c}, \frac{\min_j\{l_{k_j}\}}{\varepsilon_d \max_j\{\ell_{\text{int},k_j}^*\}} \right\} \quad (2.100b)$$

$$\|\Delta p_{C_i/E_i}\| = \delta_r < \frac{\delta_c}{2\varepsilon_c} \quad (2.100c)$$

$$(2.100d)$$

$$e_{v_o}^\top \begin{bmatrix} 0 \\ S(\Delta p_{C_i/o}) \end{bmatrix} \left( f_{\text{int},k_i} + G_{\mathcal{R},k_i}^* Y_{o_r} \hat{v}_{\mathcal{R}_o} \right) - c_{\mathcal{R}_2} \gamma_q = 0, \quad (2.100e)$$

where  $f_{\text{int},k_i} \in \mathbb{R}^3$  is the  $i$ th component of  $f_{\text{int},k}$ ,  $k_1 := \lambda_{\min}(K)$ ,  $k_2 := \lambda_{\min}(G_{\mathcal{R}} G_{\mathcal{R}}^* K_v)$ ,  $c_{\mathcal{R}_2}$ ,  $\gamma_q$ ,  $\delta_e$ ,  $\varepsilon_c$ , are design parameters. Note that  $k_2 > 0$  due to the fact that  $G_{\mathcal{R}}$  is full row rank. The time for which an event is triggered is when (2.100a), (2.100d), or (2.100e) are satisfied, and formally defined as:

$$t(0) = 0, \quad t_{k+1} = \inf\{t \in \mathbb{R} : t > t_k \wedge ((2.100a) \vee (2.100d) \vee (2.100e))\} \quad (2.101)$$

Note that the condition (2.101) can be evaluated by each agent individually. When one agent identifies a triggering condition, the agent then signals to all agents that an update is required and all agents then *only* communicate  $p_{C_i/o}$  and  $p_{C_i/f}$  for all  $i \in \mathcal{N}$ .

The proposed control is decentralized with aperiodic updates of only each agent's contact information. The event-triggered, decentralized control law ensures practical asymptotic stability of the origin as presented in the following theorem:

**Theorem 2.6.** *Consider  $N$  robotic agents in contact with an object, described by the dynamics (2.70), (2.73), and suppose Assumptions 2.10 and 2.11 hold. Let the desired object pose  $[p_d^\top, \eta_d^\top]^\top : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^3 \times \mathbb{S}^3$  be bounded with bounded first and second derivatives. Moreover, assume that  $e_\varphi(0) \neq 0$  and  $f_{C_i}^{C_i}(0) \in \text{Int}(\mathcal{F}_{C_i})$ ,  $\forall i \in \mathcal{N}$ . Then, by choosing sufficiently large control gains  $k_p$ ,  $k_\eta$ ,  $K_v$ , the event-triggered control protocol (2.92), (2.93), (2.98) with event-triggered mechanism (2.101) guarantees ultimate boundedness of  $e_{\mathcal{R}}$ ,  $e_v$  in a set around the origin, and by choosing sufficiently large  $\epsilon_f$ ,  $\varepsilon_c$ ,  $\varepsilon_d$ , it holds that  $f_{C_i}^{C_i}(t) \in \mathcal{F}_{C_i}$ ,  $\forall t > 0$ ,  $i \in \mathcal{N}$ .*

*Proof.* The proof is structured into 2 Cases. Case 1 addresses the system if no event is triggered. Case 2 addresses the triggering conditions and ensuring non-Zeno behavior for the time updates.

Case 1: Here we address the case when no event is triggered such that  $t_k = 0$  and  $t_{k+1} = \infty$ . We note from the proof of Theorem 2.5 that the continuous control law,  $\tau$  from (2.81), ensures asymptotic stability of the system with the Lyapunov candidate function,  $V_f$ , defined in (2.88). We define the following compact set:

$$\Omega_k := \{\chi \in \mathcal{X} : V_f(\chi(t)) \leq V_k(\chi)\}, k \in \mathbb{Z}_{\geq 0}$$

with  $V_f$  as defined in (2.88), and  $V_k := V_f(\chi(t=t_k))$ .

From  $f_{C_i}^{C_i}(0) \in \text{Int}(\mathcal{F}_{C_i}^{C_i})$ , the same analysis from Theorem 2.5 applies here such that there exists a  $t_{\max} \in \mathbb{R}_{>0}$  such that for  $t \in [0, t_{\max})$ , slip does not occur and the solution is unique. In the following we will apply the Lyapunov analysis over the time interval  $[0, t_{\max})$ .

After substitution of (2.92),  $\dot{V}_f$  becomes

$$\begin{aligned} \dot{V}_f = & -e_{\mathcal{R}}^\top K e_{\mathcal{R}} - e_{v_o}^\top G_{\mathcal{R}} G_{\mathcal{R}}^\top K_v e_{v_o} + e_{v_o}^\top G_{\mathcal{R}} G_{\mathcal{R},k}^* Y_{o_r} \hat{\vartheta}_{\mathcal{R}_o} - e_{v_o}^\top Y_{o_r} \vartheta_{\mathcal{R}_o} \\ & - e_{v_o}^\top (G_{\mathcal{R}} G_{\mathcal{R},k}^* - I_6) e_{\mathcal{R}} + e_{v_o}^\top G_{\mathcal{R}} f_{\text{int}_k} + e_{\vartheta_{\mathcal{R},o}}^\top \Gamma_o^{-1} \dot{\hat{\vartheta}}_{\mathcal{R}_o} \end{aligned}$$

From  $\Delta G_{\mathcal{R}} = G_{\mathcal{R}} - G_{\mathcal{R},k}$  it follows that  $G_{\mathcal{R}} G_{\mathcal{R},k}^* - I = \Delta G_{\mathcal{R}} G_{\mathcal{R},k}^*$  which yields, along with (2.93), (2.83), and the fact that  $G_{\mathcal{R},k} f_{\text{int}_k} = 0$ :

$$\begin{aligned} \dot{V}_f \leq & -k_1 \|e_{\mathcal{R}}\|^2 - k_2 \|e_{v_o}\|^2 + e_{v_o}^\top \Delta G_{\mathcal{R}} G_{\mathcal{R},k}^* Y_{o_r} \hat{\vartheta}_{\mathcal{R}_o} - e_{v_o}^\top \Delta G_{\mathcal{R}} G_{\mathcal{R},k}^* e_{\mathcal{R}} \\ & + e_{v_o}^\top \Delta G_{\mathcal{R}} f_{\text{int}_k}. \end{aligned}$$

Note that  $k_2$  can be increased by tuning  $K_v$ . From  $\Delta G_{\mathcal{R}_i} := G_{\mathcal{R}_i} - G_{\mathcal{R},k_i} = \begin{bmatrix} 0 \\ S(\Delta p_{C_i/o}) \end{bmatrix}$  and  $\Delta G_{\mathcal{R}} = [\Delta G_{\mathcal{R}_1}, \dots, \Delta G_{\mathcal{R}_N}]$ , it follows that  $\|\Delta G_{\mathcal{R}} G_{\mathcal{R},k}^*\| \leq \|\Delta G_{\mathcal{R}}\| \|G_{\mathcal{R},k}^*\| \leq \sum_i \|\Delta p_{C_i/o}\| \|G_{\mathcal{R},k_i}^*\|$ . From the triggering condition (2.101), it follows that  $\|\Delta p_{C_i/o}\| \leq \delta_p$  for all  $i \in \mathcal{N}$ . We thus define  $c_{\mathcal{R}_1} := \delta_p \sum_i \|G_{\mathcal{R},k_i}^*\|$ , which is constant between events, such that  $\|\Delta G_{\mathcal{R}} G_{\mathcal{R},k}^*\| \leq c_{\mathcal{R}_1}$ . Note that Assumptions 2.10 and 2.11 as well as the fact that slip does not occur for  $[0, t_{\max})$  imply that  $\|\Delta G_{\mathcal{R}} G_{\mathcal{R},k}^*\|$  is well defined and bounded,  $\forall i \in \mathcal{N}$ . Hence  $\dot{V}_f$  becomes

$$\begin{aligned} \dot{V}_f \leq & -k_1 \|e_{\mathcal{R}}\|^2 - k_2 \|e_{v_o}\|^2 + e_{v_o}^\top \Delta G_{\mathcal{R}} G_{\mathcal{R},k}^* Y_{o_r} \hat{\vartheta}_{\mathcal{R}_o} + c_{\mathcal{R}_1} \|e_{v_o}\| \|e_{\mathcal{R}}\| \\ & + e_{v_o}^\top \Delta G_{\mathcal{R}} f_{\text{int}_k} \end{aligned}$$

We then complete the squares such that  $c_{\mathcal{R}_1} \|e_{v_o}\| \|e_{\mathcal{R}}\| \leq \frac{c_{\mathcal{R}_1}}{2} \|e_{v_o}\|^2 + \frac{c_{\mathcal{R}_1}}{2} \|e_{\mathcal{R}}\|^2$  and hence  $\dot{V}_f$  becomes

$$\begin{aligned} \dot{V}_f \leq & -\left(k_1 - \frac{c_{\mathcal{R}_1}}{2}\right) \|e_{\mathcal{R}}\|^2 - \left(k_2 - \frac{c_{\mathcal{R}_1}}{2}\right) \|e_{v_o}\|^2 + e_{v_o}^\top \Delta G_{\mathcal{R}} G_{\mathcal{R},k}^* Y_{o_r} \hat{\vartheta}_{\mathcal{R}_o} \\ & + e_{v_o}^\top \Delta G_{\mathcal{R}} f_{\text{int}_k} \end{aligned}$$

Now we introduce  $c_{\mathcal{R}_2} \in \mathbb{R}_{>0}$  such that:

$$\begin{aligned}\dot{V}_f &\leq - \left( k_1 - \frac{c_{\mathcal{R}_1}}{2} - c_{\mathcal{R}_2} \right) \|e_{\mathcal{R}}\|^2 - \left( k_2 - \frac{c_{\mathcal{R}_1}}{2} - c_{\mathcal{R}_2} \right) \|e_{v_o}\|^2 \\ &\quad - c_{\mathcal{R}_2} \|e_{\mathcal{R}}\|^2 - c_{\mathcal{R}_2} \|e_{v_o}\|^2 + e_{v_o}^\top \Delta G_{\mathcal{R}} G_{\mathcal{R},k}^* Y_{o_r} \hat{\vartheta}_{\mathcal{R}_o} + e_{v_o} \Delta G_{\mathcal{R}} f_{\text{int}_k} \\ &=: - (k_e - c_{\mathcal{R}_2}) \|e_{\mathcal{R}}\|^2 - (k_{e_v} - c_{\mathcal{R}_2}) \|e_{v_o}\|^2 - c_{\mathcal{R}_2} \|e_{\mathcal{R}}\|^2 - c_{\mathcal{R}_2} \|e_{v_o}\|^2 \\ &\quad + e_{v_o}^\top \Delta G_{\mathcal{R}} G_{\mathcal{R},k}^* Y_{o_r} \hat{\vartheta}_{\mathcal{R}_o} + e_{v_o}^\top \Delta G_{\mathcal{R}} f_{\text{int}_k}\end{aligned}$$

where  $k_e := k_1 - \frac{c_{\mathcal{R}_1}}{2}$  and  $k_{e_v} := k_2 - \frac{c_{\mathcal{R}_1}}{2}$ . By choosing large enough  $k_p$ ,  $k_\eta$ , and  $K_v$ , we can achieve  $k_e > c_{\mathcal{R}_2}$  and  $k_{e_v} > c_{\mathcal{R}_2}$ .

Let now  $\mathcal{Q} := \{\chi \in \mathcal{X} : \|e_{\mathcal{R}}\|^2 + \|e_{v_o}\|^2 \leq \gamma_q\}$ . Note that  $\mathcal{Q}$  is compact since  $e_{\vartheta_{\mathcal{R}}}$ ,  $e_{\vartheta_{\mathcal{R},o}}$  are bounded as per (2.84). Moreover, in  $\mathcal{X} \setminus \mathcal{Q}$  it holds that  $\|e_{\mathcal{R}}\|^2 + \|e_{v_o}\|^2 > \gamma_q$  and hence  $c_{\mathcal{R}_2} \|e_{\mathcal{R}}\|^2 + c_{\mathcal{R}_2} \|e_{v_o}\|^2 > c_{\mathcal{R}_2} \gamma_q$ , and  $\dot{V}_f$  becomes

$$\begin{aligned}\dot{V}_f &\leq - (k_e - c_{\mathcal{R}_2}) \|e_{\mathcal{R}}\|^2 - (k_{e_v} - c_{\mathcal{R}_2}) \|e_{v_o}\|^2 \\ &\quad + e_{v_o}^\top \Delta G_{\mathcal{R}} G_{\mathcal{R},k}^* Y_{o_r} \hat{\vartheta}_{\mathcal{R}_o} + e_{v_o}^\top \Delta G_{\mathcal{R}} f_{\text{int}_k} - c_{\mathcal{R}_2} \gamma_q\end{aligned}$$

According to (2.100e), it holds, between events, that

$$e_{v_o}^\top \begin{bmatrix} 0 \\ S(\Delta p_{C_i/o}) \end{bmatrix} \left( f_{\text{int},k_i} + G_{\mathcal{R},k_i}^* Y_{o_r} \hat{\vartheta}_{\mathcal{R}_o} \right) - c_{\mathcal{R}_2} \gamma_q \leq 0.$$

By summing for all  $i \in \mathcal{N}$ , the latter becomes

$$e_{v_o}^\top \Delta G_{\mathcal{R}} G_{\mathcal{R},k}^* (Y_{o_r} \hat{\vartheta}_{\mathcal{R}_o} + f_{\text{int}_k}) - c_{\mathcal{R}_2} \gamma_q \leq 0,$$

implying that  $\dot{V}_f \leq - (k_e - c_{\mathcal{R}_2}) \|e_{\mathcal{R}}\|^2 - (k_{e_v} - c_{\mathcal{R}_2}) \|e_{v_o}\|^2 \leq 0$ . By following Barbalat's Lemma, it can be shown that  $\chi$  will enter the set  $\mathcal{Q}$  in finite time.

By using (2.84), we now investigate  $\dot{V}_f$  inside  $\mathcal{Q}$  for which it holds  $\|e_{v_o}\| \leq \sqrt{\gamma_q}$ :

$$\begin{aligned}\dot{V}_f &\leq - k_e \|e_{\mathcal{R}}\|^2 - k_{e_v} \|e_{v_o}\|^2 + e_{v_o}^\top \Delta G_{\mathcal{R}} G_{\mathcal{R},k}^* Y_{o_r} \hat{\vartheta}_{\mathcal{R}_o} + e_{v_o} \Delta G_{\mathcal{R}} f_{\text{int}_k} \\ &\leq - k_e \|e_{\mathcal{R}}\|^2 - k_{e_v} \|e_{v_o}\|^2 - \beta_{\vartheta} \|e_{\vartheta_{\mathcal{R}}}\|^2 - \beta_{\vartheta_o} \|e_{\vartheta_{\mathcal{R},o}}\|^2 + \beta_{\vartheta} \bar{e}_{\vartheta_{\mathcal{R}}}^2 \\ &\quad + \beta_{\vartheta_o} \bar{e}_{\vartheta_{\mathcal{R},o}}^2 + e_{v_o}^\top \Delta G_{\mathcal{R}} G_{\mathcal{R},k}^* Y_{o_r} e_{\vartheta_{\mathcal{R},o}} + e_{v_o}^\top \Delta G_{\mathcal{R}} G_{\mathcal{R},k}^* Y_{o_r} \vartheta_{\mathcal{R}_o} \\ &\quad + e_{v_o}^\top \Delta G_{\mathcal{R}} f_{\text{int}_k} \\ &\leq - k_e \|e_{\mathcal{R}}\|^2 - k_{e_v} \|e_{v_o}\|^2 - \beta_{\vartheta} \|e_{\vartheta_{\mathcal{R}}}\|^2 - \beta_{\vartheta_o} \|e_{\vartheta_{\mathcal{R},o}}\|^2 + \beta_{\vartheta} \bar{e}_{\vartheta_{\mathcal{R}}}^2 \\ &\quad + \beta_{\vartheta_o} \bar{e}_{\vartheta_{\mathcal{R},o}}^2 + \sqrt{\gamma_q} \|\Delta G_{\mathcal{R}} G_{\mathcal{R},k}^*\| \|Y_{o_r}\| \|\bar{e}_{\vartheta_{\mathcal{R},o}}\| \\ &\quad + \sqrt{\gamma_q} \|\Delta G_{\mathcal{R}} G_{\mathcal{R},k}^*\| \|Y_{o_r}\| \|\vartheta_{\mathcal{R}_o}\| + \sqrt{\gamma_q} \|\Delta G_{\mathcal{R}}\| \|f_{\text{int}_k}\|,\end{aligned}$$

where  $\beta_{\vartheta}$ ,  $\beta_{\vartheta_o} \in \mathbb{R}_{>0}$  are positive constants. Since  $\|\Delta G_{\mathcal{R}} G_{\mathcal{R},k}^*\| \leq c_{\mathcal{R}_1}$ , it holds that  $\|\Delta G_{\mathcal{R}}\| \leq \frac{c_{\mathcal{R}_1}}{\|G_{\mathcal{R},k}^*\|}$ , which is bounded, since  $\sigma_{\min}(G_{\mathcal{R},k}^*) =$

$\frac{1}{\sigma_{\max}(G_{\mathcal{R},k})}$ , and  $G_{\mathcal{R},k}$  is full row rank. Furthermore  $f_{\text{int}_k}$  is constant between events. Thus in view of (2.84) and since  $\chi$  lies in the compact set  $\mathcal{Q}$ , we can conclude that there exists a  $\bar{\delta}_k$  such that:

$$\begin{aligned}\bar{\delta}_k \geq & +\beta_\vartheta \bar{e}_{\vartheta_{\mathcal{R}}}^2 + \beta_{\vartheta_o} \bar{e}_{\vartheta_{\mathcal{R},o}}^2 + \sqrt{\gamma_q} \|\Delta G_{\mathcal{R}} G_{\mathcal{R},k}^*\| \|Y_{o_r}\| \bar{e}_{\vartheta_o} \\ & + \sqrt{\gamma_q} \|\Delta G_{\mathcal{R}} G_{\mathcal{R},k}^*\| \|Y_{o_r}\| \|\vartheta_{\mathcal{R},o}\| + \sqrt{\gamma_q} \|\Delta G_{\mathcal{R}}\| \|f_{\text{int}_k}\|\end{aligned}$$

Hence  $\dot{V}$  becomes

$$\dot{V} \leq -k_\chi \|\chi\|^2 + \bar{\delta}_k,$$

where  $k_\chi := \min\{k_e, k_{e_v}, \beta_\vartheta, \beta_{\vartheta_o}\}$ . Therefore, by invoking Lemma A.1 of Appendix A, we guarantee that  $\chi$  is ultimately bounded in a compact set defined by  $k_\chi$  and  $\bar{\delta}_k$ , for  $t \in [0, t_{\max}]$ .

Now we investigate the slip prevention properties, similar to that of Theorem 2.5. The same derivation of  $f_C$  yields:

$$\begin{aligned}f_C = & W_h^{-1} \left( J_h B^{-1} \left[ \tau_k - g_q - \left( C_q J_h^{-1} G_{\mathcal{R}}^\top + B \frac{d}{dt} (J_h^{-1} G_{\mathcal{R}}^\top) \right) v_o \right] \right. \\ & \left. + G_{\mathcal{R}}^\top M_o^{-1} (C_o v_o + g_o) \right).\end{aligned}$$

By following a similar procedure as with the previous section, we conclude that

$$f_C = \lambda_k + (I - W_h^{-1} G_{\mathcal{R}}^\top M_o^{-1} \Delta G_{\mathcal{R}}) f_{\text{int}_k} + h_{f_k}$$

$$\begin{aligned}h_{f_k} := & W_h^{-1} J_h B^{-1} (g_q - Y_{\mathcal{R}}(\ddot{x}, \dot{\ddot{x}}, e_{v_o}, \dot{e}_{v_o}) + Y_r e_{\vartheta_{\mathcal{R}}}) + \\ & W_h^{-1} G_{\mathcal{R}}^\top M_o^{-1} \left( G_{\mathcal{R}} G_{\mathcal{R}}^\top K_v e_{v_o} + e_{\mathcal{R}} + Y_{\mathcal{R},o}(\eta_o, \omega_o, e_{v_o}, \dot{e}_{v_o}) \vartheta_{\mathcal{R},o} \right. \\ & \left. - Y_{o_r} e_{\vartheta_{\mathcal{R},o}} + \Delta G_{\mathcal{R}} G_{\mathcal{R},k}^* (e_{\mathcal{R}} - Y_{o_r} \hat{\vartheta}_{\mathcal{R},o}) - g_o \right)\end{aligned}$$

Substitution of  $f_C$  into (2.85), which ensures slip prevention, yields the following condition to be satisfied:

$$\Lambda(\mu_f) R_C^\top (I_{3N} - W_h^{-1} G_{\mathcal{R}}^\top M_o^{-1} \Delta G_{\mathcal{R}}) f_{\text{int}_k} \succeq -\Lambda(\mu_f) R_{C,k}^\top \lambda_k - \Lambda(\mu_f) R_{C,k}^\top h_{f_k}$$

From the boundedness of signals, we conclude that  $h_{f_k}$  is bounded for all  $\forall t \in [0, t_{\max}]$  in a compact set, independent of  $t_{\max}$ . Now let  $\varepsilon_{h_k}$  denote the maximum bound of the elements of  $\pm \Lambda(\mu_f) R_{C,k}^\top h_{f_k}$  and substitute  $f_{\text{int}_k} = f'_{\text{int}_k} R_{C,k} \ell_{\text{int},k}^*$  with  $W := R_C^\top (\Delta R_C + W_h^{-1} G_{\mathcal{R}}^\top M_o^{-1} \Delta G_{\mathcal{R}} R_{C,k})$  to rewrite the sufficient condition for no slip as:

$$f'_{\text{int},k} \Lambda(\mu_f) (I_3 - W) \ell_{\text{int},k}^* \succeq -\Lambda(\mu_f) R_{C,k}^\top \lambda_k + \varepsilon_{h_k} \mathbb{1}$$

or, for each agent separately,

$$\begin{aligned} f'_{\text{int},k} \Lambda_i(\mu_f) (I_3 - W_{ii}) \ell_{\text{int},k,i}^* &\succeq -\Lambda_i(\mu_f) R_{C,k_i}^\top \lambda_{k_i} \\ &+ f'_{\text{int},k} \sum_i \sum_{j \neq i} D_{ij} \ell_{\text{int},k,j}^* + \varepsilon_h \mathbb{1} \end{aligned} \quad (2.102)$$

where  $\ell_{\text{int},k,i}^* \in \mathbb{R}^3$  is the  $i$ th agent's part in  $\ell_{\text{int},k}^*$  (as opposed to the scalar  $\ell_{\text{int},k_i}^*$ ),  $W_{ii}$  is  $i$ th block matrix of  $W$ 's diagonal and  $D_{ij} \in \mathbb{R}^3$  is the  $ij$ -block matrix of  $\Lambda(\mu_f)W$ . Here we show that the triggering conditions (2.100a) and (2.100b) ensure that  $\lambda_{\max}(W) \leq \delta_c < 1$ . By boundedness of the system dynamics, for sufficiently large  $\varepsilon_c \in \mathbb{R}_{>0}$  bounding the terms  $W_h^{-1} G_{\mathcal{R}}^\top M_{o_i}^{-1}$ , it follows that  $\lambda_{\max}(W) \leq \|W\| \leq \varepsilon_c (\delta_r + \delta_p)$ . Since  $\delta_r < \frac{\delta_c}{2\varepsilon_c}$  and  $\delta_p < \frac{\delta_c}{2\varepsilon_c}$  it follows that  $\lambda_{\max}(W_{ii}) \leq \delta_c < \frac{\sqrt{\mu^2+1}-1}{\sqrt{\mu^2+1}+1} < 1$ ,  $\forall i \in \mathcal{N}$ , and hence Lemma 2.3 dictates that (2.102) is satisfied when

$$f'_{\text{int},k} \Lambda_i(\mu'_f) \ell_{\text{int},k,i}^* \succeq -\Lambda_i(\mu_f) R_{C,k_i}^\top \lambda_{k_i} + \sum_i \sum_{j \neq i} D_{ij} \ell_{\text{int},k,j}^* + \varepsilon_{h_k} \mathbb{1},$$

where  $\mu'_f$  as given by (2.95).

From the boundedness of signals, there exists a  $\varepsilon_d \in \mathbb{R}_{>0}$  such that  $\|D_{ij}\| \leq \varepsilon_d \delta_p$  for all  $i, j \in \mathcal{N}$ . Furthermore, we know  $\ell_{\text{int},k}^*$  is constant between updates and known to all agents. Therefore at each update  $\min_j \{l_{k_j}\}$  and  $\max_j \{\ell_{\text{int},k_j}^*\}$  can be computed, and we note that  $\max_j \{\ell_{\text{int},k_j}^*\}$  is bounded by Assumption 2.10. Thus a sufficient condition for the above expression to hold is:

$$f'_{\text{int},k} (\min_j \{l_{k_j}\} - \varepsilon_d \delta_p \max_j \{\ell_{\text{int},k_j}^*\}) \succeq -b_{k_j} + \varepsilon_{h_k}$$

$\forall j \in \{1, \dots, Nn_s\}$ , which is feasible since  $\delta_p < \frac{\min_j \{l_{k_j}\}}{\varepsilon_d \max_j \{\ell_{\text{int},k_j}^*\}}$  from (2.100b).

By substituting (2.99) with the choice of  $\epsilon_f \geq \varepsilon_d$ , the left side satisfies

$$\kappa(\min_j \{b_{k_j}\}) + 1 + \epsilon_f \geq -b_{k_j} + \varepsilon_{h_k},$$

where we use  $\kappa(x) \geq 0$ ,  $\kappa(x) + 1 \geq -x$ ,  $\forall x \in \mathbb{R}$ , and  $\kappa(\min_j \{b_{k_j}\}) > \kappa(b_{k_j})$ ,  $\forall j \in \{1, \dots, Nl_f\}$ , since  $\kappa()$  is decreasing. Hence, by choosing a large enough  $\epsilon_f$  we guarantee  $\epsilon_f \geq \varepsilon_{h_k}$  and hence contact slip is actively prevented  $\forall t \in [0, t_{\max}]$ . Following the proof of 2.5, it follows that  $t_{\max} = \infty$ , and thus slip prevention is ensured for the entirety of the manipulation task.

Furthermore, since  $t_{\max} = \infty$ , the previous Lyapunov analysis ensures that  $\dot{V}_f < 0$  when  $\|\chi\| > \sqrt{\frac{\delta_k}{k_\chi}}$ , guaranteeing thus, in view of Lemma A.1 of

Appendix A] that  $\chi$  will be ultimately bounded in a compact set around the origin, rendering the closed-loop system practically asymptotically stable.

Case 2: In the previous analysis, practical asymptotic stability is ensured when no triggering occurs. Here we show that indeed the event triggering preserves the results from Case 1 and that the system does not exhibit Zeno behavior. For any  $\chi \in \mathcal{X} \setminus \mathcal{Q}$ , it follows that  $\dot{V}_f \leq 0$ . Thus if any event triggers in  $\mathcal{X} \setminus \mathcal{Q}$  at  $t = t_k$ ,  $\Delta G_{\mathcal{R},k} = 0$  and  $\Delta R_{C,k} = 0$ , and it is straightforward to see that  $V_{k+1} \leq V_k$  such that  $\Omega_{k+1} \subseteq \Omega_k$ . Furthermore  $\dot{V}_f \leq 0$  holds after the event occurs and ensures  $\chi$  enters  $\mathcal{Q}$  in finite time.

For  $\chi \in \mathcal{Q}$ , the condition  $\dot{V} \leq k_\chi \|\chi\|^2 + \bar{\delta}_k$  holds although  $\bar{\delta}_k$  will change between events. However, since  $\chi$  and  $f_{\text{int},k}$  are bounded in  $\mathcal{Q}$ , there exists a maximum  $\bar{\delta} \geq \bar{\delta}_k$  for which  $\chi$  is ultimately bounded, and practical stability is preserved.

Now we show there exists a lower bound between each event time instant. Events (2.100a) and (2.100d) are dependent on bounds  $\delta_r$ , and  $\delta_p$ , where  $\delta_r > 0$  is fixed and  $\delta_p > 0$  and will never tend to zero due to boundedness of  $p_{C_i/o}$ . From the continuous differentiability of  $p_{C_i/o}$  and  $p_{C_i/E_i}$ , let  $L_p, L_r \in \mathbb{R}_{>0}$  denote their respective Lipschitz constants. It follows that there exist lower bounds on event times defined by  $\Delta t_p = \delta_p/L_p$ ,  $\Delta t_r = \delta_r/L_r$ , respectively.

Similarly, the event defined by (2.100e) depends on the bound  $c_{\mathcal{R}_2}\gamma_q$ . Denote by  $e_{v_o} = [e_{v_o,p}^\top, e_{v_o,\eta}^\top]^\top \in \mathbb{R}^3 \times \mathbb{R}^3$ . Then (2.100e) occurs when

$$e_{v_o,\eta}(t_{k+1})^\top S(\Delta p_{C_i/o}(t_{k+1}))\tilde{h}_{f,k_i}(t_{k+1}) = c_{\mathcal{R}_2}\gamma_q,$$

where  $\tilde{h}_{f,k_i} := f_{\text{int},k_i} + G_{\mathcal{R},k_i}^* Y_{o_r} \hat{\vartheta}_{\mathcal{R}_o}(t)$ ,  $\forall i \in \mathcal{N}$ ,  $\forall k \in \mathbb{N}$ , with  $t_1 = 0$ . Therefore, since  $\|\Delta p_{C_i/o}\| \leq \delta_p$  from (2.100b) and  $e_{v_o}$ ,  $Y_{o_r}$ ,  $\hat{\vartheta}_{\mathcal{R}_o}$ ,  $f_{\text{int},k}$  are bounded in compact sets for  $t \in [t_k, t_{k+1}]$  from the previous analysis, there exist positive constants  $\underline{e}$  and  $\underline{h}_i$  such that  $\|e_{v_o,\eta}(t_{k+1})\| \geq \underline{e}$  and  $\|\tilde{h}_{f,k_i}(t_{k+1})\| \geq \underline{h}_i$ ,  $\forall i \in \mathcal{N}$ . Hence, by taking into account (2.100a) it holds that  $c_{\mathcal{R}_2}\gamma_q \leq \delta_p \|e_{v_o,\eta}(t_{k+1})\| \|\tilde{h}_{f,k_i}(t_{k+1})\| \Delta t_e$ , with  $\Delta t_e$  being the inter-sampling time between the updates defined by (2.100e). We conclude then that  $\Delta t_e \geq \frac{c_{\mathcal{R}_2}\gamma_q}{\underline{e}\underline{h}_i\delta_p}$ .

Finally, as  $t_k$  is defined by satisfaction of any events from (2.100a), (2.100d), or (2.100e), it follows that  $\Delta t := t_{k+1} - t_k = \min\{\Delta t_p, \Delta t_r, \Delta t_e\}$  where  $\Delta t > 0$  and lower bounded.  $\square$

## 2.4 Conclusion

This chapter presented novel control protocols for the cooperative manipulation of a single object by  $N$  robotics agents without employing force sensing. Firstly, we focused on rigid grasps, by introducing two adaptive decentralized control schemes that used quaternion-feedback and prescribed

performance control, respectively. Next, we incorporated collision avoidance by using nonlinear MPC, in a centralized and a communication-based decentralized scheme. Secondly, we considered the case of rolling contacts. We developed novel adaptive centralized and decentralized control schemes that compensate for the object's and the agents' dynamic uncertainties and guarantee avoidance of contact loss at the contact points.



## Chapter 3

# Formation Control and Rigid Cooperative Manipulation

As discussed in Chapter 1, an important problem associated with multi-agent coordination is formation control. On one hand, formation specifications can be imposed by a higher level planner associated with temporal tasks. Moreover, as we show here, a particular instance of multi-agent formations, namely rigid formation, is tightly associated to rigid cooperative manipulation presented in the previous chapter. More specifically, this chapter addresses the following two topics.

Firstly, it deals with the problem of distance- and orientation-based formation control of a class of second-order nonlinear multi-agent systems in  $\mathbb{SE}(3)$ , under static and undirected communication topologies. More specifically, we design a decentralized model-free control protocol in the sense that each agent uses only local information from its neighbors to calculate its own control signal, without incorporating any knowledge of the model nonlinearities and exogenous disturbances. Moreover, the transient and steady-state response is solely determined by certain designer-specified performance functions and is fully decoupled by the agents' dynamic model, the control gain selection, the underlying graph topology as well as the initial conditions. Additionally, by introducing certain inter-agent distance constraints, we guarantee collision avoidance and connectivity maintenance between neighboring agents.

Secondly, we introduce a new notion of distance rigidity, namely distance- and bearing-rigidity in  $\mathbb{SE}(3)$ , and we connect it with rigid cooperative manipulation. More specifically, the nodes of a general rigid framework are associated to the robotic agents of rigid cooperative manipulation schemes and the object-agent interaction forces are expressed by using the rigidity matrix of the graph formed by the robots, which encodes the infinitesimal rigid body motions of the system. Moreover, we show that the associated cooperative manipulation grasp matrix is related to the rigidity matrix via a range-nullspace relation, based on which we provide novel results on the

relation between the arising interaction and internal forces and consequently on the energy-optimal force distribution on a cooperative manipulation system.

### 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 [? ? ? ? ], and at the same time fulfill certain properties, e.g., network connectivity [? ? ].

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 ([? ]) position-based control, displacement-based control, distance-based control and orientation-based control.

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 position of its neighboring agents. When orientation alignment is considered as a control design goal, the problem is known as orientation-based (or bearing-based) formation control. 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. [? ? ? ? ? ? ? ? ? ? ? ? ? ? ? ? ]). Orientation-based formation control has been addressed in [? ? ? ? ], whereas the authors in [? ? ? ] 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. Other important issues concern the connectivity maintenance, the collision avoidance between the 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 works. 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.

Another special instance of formation control that has practical relevance and numerous applications in robotics is that of *rigid formations*. Two cases of rigid formation control have been widely studied in the literature, namely *distance rigidity* and *bearing rigidity*. The classic distance rigidity theory studies the problem of under what conditions can the geometric pattern of a network be uniquely determined if the length (distance) of each edge in the multi-agent team is fixed. It is a combinatorial theory for characterizing the “stiffness” or “flexibility” of structures formed by rigid bodies connected by flexible linkages or hinges, and it has been applied extensively in distance-based formation control and network localization [? ? ? ? ? ? ? ? ? ? ? ? ]. Bearing rigidity theory studies the fundamental problem of under what conditions can the geometric pattern of a multi-agent system be uniquely determined if the bearing of each edge is fixed [? ], and it has been used for bearing-based control and estimation problems [? ? ? ? ]. Recent works have developed bearing rigidity theory on the manifolds of  $\mathbb{SE}(2)$  [? ] and  $\mathbb{SE}(3)$  [? ]. In this chapter, we introduce the notion of *distance and bearing rigidity*, which studies under what conditions can the geometric pattern of a multi-agent system be uniquely determined if both the *distance* and the *bearing* of each edge is fixed. Moreover, we combine the latter with *rigid* cooperative manipulation, i.e., configurations where a number of robotic agents are attached to a common object by means of rigid contact points.

As shown in the previous chapter, rigid cooperative manipulation by robotic agents (i.e., when the grasps are rigid) is an important and challenging topic, indispensable in cases of difficult maneuvers or heavy payloads. An important property in rigid cooperative manipulation systems that has been studied thoroughly in the related literature and overlooked in the previous chapter, is the regulation of internal forces. Internal forces are forces exerted by the agents at the grasping points that do not contribute to the motion of the object. While a certain amount of such forces is required in many cases (e.g., to avoid contact loss in multi-fingered manipulation), they need to be minimized in order to prevent object damage and unnecessary effort of the agents. Most works in rigid cooperative manipulation assume a certain decomposition of the interaction forces in motion-inducing and internal ones, without explicitly showing that the actual internal forces will be indeed regulated to the desired ones (e.g., [? ? ? ]); [? ? ? ? ] analyze specific load decompositions based on whether they provide internal force-free expressions, whereas [? ] is concerned with the cooperative manipulation interaction dynamics. The decompositions in the aforementioned works, however, are based on the inter-agent distances and do not take into account the actual

dynamics of the agents. The latter, as we show in this chapter, is tightly connected to the internal forces as well as their relation to the total force exerted by the agents at the grasping points.

This chapter deals with the following two topics:

1. Firstly, we address the distance-based formation control problem with orientation alignment for a team of rigid bodies operating in  $\mathbb{SE}(3)$ , 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 neighboring agents and in parallel ensures the satisfaction of prescribed transient and steady state performance. The prescribed performance control framework has been incorporated in multi-agent systems in [? ] and [? ] for minimally rigid formations, where first order dynamics have been considered without taking into account the problem of orientation alignment.
2. We integrate rigid cooperative manipulation with rigidity theory. Motivated by rigid cooperative manipulation systems, where the inter-agent distances *and* bearings are fixed, we introduce the notion of *distance and bearing rigidity* in the special Euclidean group  $\mathbb{SE}(3)$ . Based on recent results, we show next that the interaction forces in a rigid cooperative manipulation system depend on the distance and bearing rigidity matrix, a matrix that encodes the allowed coordinated motions of the multi-agent-object system. Moreover, we prove that the cooperative manipulation grasp matrix, which relates the object and agent velocities, is connected via a range-nullspace relation to the rigidity matrix. Furthermore, we rely on the aforementioned findings to provide new results on the internal force-based rigid cooperative manipulation. We derive novel results on the relation between the arising interaction and internal forces in a cooperative manipulation system. This leads to novel conditions on the internal force-free object-agents force distribution and consequently to optimal, in terms of energy resources, cooperative manipulation.

Finally, we verify all the theoretical findings through simulation results.

## 3.2 Formation Control in $\mathbb{SE}(3)$

### 3.2.1 Problem Formulation

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 by  $R_i \in \mathbb{SO}(3)$  the rotation matrix associated with the orientation of the  $i$ th 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  $\omega_i$  are expressed in 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  $x_i := (p_i, R_i) \in \mathbb{SE}(3)$  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:

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

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

where the matrix  $M_i \in \mathbb{R}^{6 \times 6}$  is the constant positive definite inertia matrix,  $C_i := C_i(v_i) : \mathbb{R}^6 \rightarrow \mathbb{R}^{6 \times 6}$  is the Coriolis matrix,  $g_i := g_i(x_i) : \mathbb{SE}(3) \rightarrow \mathbb{R}^6$  is the body-frame gravity vector,  $w_i := w_i(x_i, v_i, t) : \mathbb{SE}(3) \times \mathbb{R}^6 \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^6$  is a bounded vector representing model uncertainties and external disturbances, and  $\mathbb{T}_{R_i} := \mathbb{R}^3 \times T_R \mathbb{SO}(3)$ , where  $T_R \mathbb{SO}(3)$  is the tangent space to  $\mathbb{SO}(3)$  at  $R$ . Finally,  $u_i \in \mathbb{R}^6$  is the control input vector representing the 6D 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

$$0 < \underline{m}_i < \bar{m}_i < \infty \quad (3.2a)$$

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

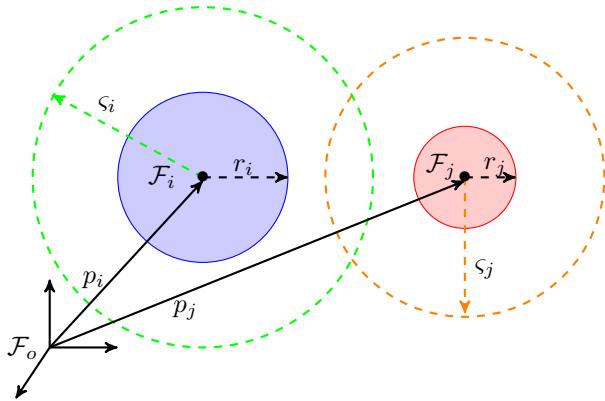
$\forall i \in \mathcal{N}$ , where  $\bar{g}_i$  is a finite *unknown* positive constant and  $\underline{m}_i := \lambda_{\min}(M_i)$ , and  $\bar{m}_i := \lambda_{\max}(M_i)$ , which are also *unknown*,  $\forall i \in \mathcal{N}$ .

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

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

$$\dot{x} = h_x, \quad (3.3a)$$

$$u = M \dot{v} + C v + g + w, \quad (3.3b)$$



**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.

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

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

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  $\mathcal{N}_i : \mathbb{R}^{3N} \Rightarrow \mathbb{N}$ , with  $\mathcal{N}_i(p) := \{j \in \mathcal{N} : p_j \in \mathcal{B}(p_i, s_i)\}$ , and  $p := [p_1^\top, \dots, p_N^\top]^\top \in \mathbb{R}^{3N}$ , 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  $\tilde{v}_i := [\tilde{v}_{i,L}^\top, \tilde{\omega}_i^\top]^\top := v_i + \mathbf{n}_i$ ,  $\forall i \in \mathcal{N}$ , where  $\mathbf{n}_i := \mathbf{n}_i(x_i, t) : \mathbb{SE}(3) \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^6$  are vector fields bounded by *unknown* positive finite constants  $\bar{n}_i$ , i.e.,  $\|\mathbf{n}_i(x_i, t)\| \leq \bar{n}_i$ ,  $\forall x_i \in \mathbb{SE}(3)$ ,  $t \in \mathbb{R}_{\geq 0}$ ,  $i \in \mathcal{N}$ . Moreover, the vector fields  $\mathbf{n}_{i,d} := \mathbf{n}_{i,d}(x_i, \dot{x}_i, t) : \mathbb{SE}(3) \times \mathbb{T}_{R_i} \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^6$  with  $\mathbf{n}_{i,d}(x_i, \dot{x}_i, t) := \dot{\mathbf{n}}_i(x_i, \dot{x}_i) = \frac{\partial \mathbf{n}_i(x_i, t)}{\partial x_i} \dot{x}_i + \frac{\partial \mathbf{n}_i(x_i, t)}{\partial t}$  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  $(x_i, t) \rightarrow \mathbf{n}_{i,d}(x_i, \dot{x}_i, t)$  are assumed to be bounded by *unknown* positive finite constants  $\bar{\mathbf{n}}_{i,d}$ , i.e.,  $\|\mathbf{n}_{i,d}(x_i, \dot{x}_i, t)\| \leq \bar{\mathbf{n}}_{i,d}, \forall x_i \in \mathbb{SE}(3), t \in \mathbb{R}_{\geq 0}, i \in \mathcal{N}$ .

**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  $\varsigma_i$ ), the fact that the initial graph is nonempty, connected and undirected implies that

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

with  $d_{k,\text{con}} := \min\{\varsigma_{k_1}, \varsigma_{k_2}\}, \forall k \in \mathcal{K}$ . 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

$$d_{k,\text{col}} < \|p_{k_2}(0) - p_{k_1}(0)\| < d_{k,\text{con}}, \forall k \in \mathcal{K}. \quad (3.4)$$

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

### 3.2.2 Problem Solution

Let us first introduce the distance and orientation errors:

$$e_k := \|p_{k_2} - p_{k_1}\|^2 - d_{k,\text{des}}^2 \in \mathbb{R}, \quad (3.5a)$$

$$\psi_k := \frac{1}{2} \text{tr} \left[ I_3 - R_{k,\text{des}}^\top R_{k_2}^\top R_{k_1} \right] \in [0, 2], \quad (3.5b)$$

$\forall k \in \mathcal{K}$ , where we have used Proposition G.3 of Appendix G. Regarding  $e_k$ , our goal is to guarantee  $\lim_{t \rightarrow \infty} e_k(t) \rightarrow 0$  from all initial conditions satisfying (3.4), while avoiding inter-agent collisions and connectivity losses among the initially connected agents specified by  $\mathcal{E}$ . Regarding  $\psi_k$ , we aim to guarantee the following:

1.  $\lim_{t \rightarrow \infty} \psi_k(t) \rightarrow 0$ , which, according to Proposition G.3 of Appendix G implies that

$$\lim_{t \rightarrow \infty} R_{k_2}(t)^\top R_{k_1}(t) = R_{k,\text{des}}$$

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<sup>1</sup>.

---

<sup>1</sup>It has been proved that topological obstructions do not allow global stabilization on  $\mathbb{SO}(3)$  with a continuous feedback control law (see [? ? ?]).

By using the properties of skew-symmetric matrices presented Appendix G, we derive the following dynamics of the errors (3.5):

$$\begin{aligned}\dot{e}_k &= 2(p_{k_2} - p_{k_1})^\top (R_{k_2} v_{k_2,L} - R_{k_1} v_{k_1,L}) \\ &= 2(R_{k_1}^\top \tilde{p}_{k_2,k_1})^\top (R_{k_1}^\top R_{k_2} v_{k_2,L} - v_{k_1,L}),\end{aligned}\quad (3.6a)$$

$$\dot{\psi}_k = \frac{1}{2} e_{R_k}^\top (R_{k_1}^\top R_{k_2} \omega_{k_2} - \omega_{k_1}), \quad (3.6b)$$

where  $\tilde{p}_{k_2,k_1} := p_{k_2} - p_{k_1}$  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 G.2 of Appendix G, we obtain  $\|e_{R_k}\|^2 = \|R_{k_2}^\top R_{k_1} - R_{k,\text{des}}\|_F^2(1 - \frac{1}{8}\|R_{k_2}^\top R_{k_1} - R_{k,\text{des}}\|_F^2)$  as well as

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

Hence, it holds that:

$$\|e_{R_k}\|^2 = 2\psi_k(2 - \psi_k), \quad (3.7)$$

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.

The concepts and techniques of prescribed performance control (see Appendix B) are adapted in this work in order to: a) achieve predefined transient and steady-state response for the distance and orientation errors  $e_k$ ,  $\psi_k$ ,  $\forall k \in \mathcal{K}$ , as well as ii) avoid the violation of the collision and connectivity constraints between initially neighboring agents, as presented in Section 3.2.1. The mathematical expressions of prescribed performance are given by the inequality objectives:

$$-C_{k,\text{col}}\rho_{e_k}(t) < e_k(t) < C_{k,\text{con}}\rho_{e_k}(t), \quad (3.8a)$$

$$0 \leq \psi_k(t) < \rho_{\psi_k}(t) < 2, \quad (3.8b)$$

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

$$C_{k,\text{col}} := d_{k,\text{des}}^2 - d_{k,\text{col}}^2, \quad (3.9a)$$

$$C_{k,\text{con}} := d_{k,\text{con}}^2 - d_{k,\text{des}}^2, \quad (3.9b)$$

$\forall k \in \mathcal{K}$ , 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 (2.4), that:

$$-C_{k,\text{col}}\rho_{e_k}(0) < e_k(0) < \rho_{e_k}(0)C_{k,\text{con}}, \quad (3.10a)$$

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

$$\rho_{\psi_k,0} = \rho_{\psi_k}(0) \in (\psi_k(0), 2), \quad (3.10b)$$

it is also guaranteed that:

$$0 \leq \psi_k(0) < \rho_{\psi_k}(0) < 2, \quad (3.10c)$$

$\forall k \in \mathcal{K}$ . Hence, if we guarantee prescribed performance via (3.8), 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:

$$-C_{k,\text{col}} < e_k(t) < C_{k,\text{con}}, \quad (3.11a)$$

$$0 \leq \psi_k(t) < \rho_{\psi_k}(t), \quad (3.11b)$$

$\forall t \in \mathbb{R}_{\geq 0}$ , which, owing to (3.9), 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 (3.11) guarantee that  $\psi_k(t) < 2$ ,  $\forall t \in \mathbb{R}_{\geq 0}$  and the avoidance of the singular equilibrium.

In the sequel, we propose a decentralized control protocol that does not incorporate any information on the agents' dynamic model and guarantees (3.8) for all  $t \in \mathbb{R}_{\geq 0}$ .

Given the errors  $e_k, \psi_k$  defined in the previous section, 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 (3.8), (3.10b), and (3.9),

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,  $\forall k \in \mathcal{K}$ ,

$$\xi_{e_k} := \frac{e_k}{\rho_{e_k}}, \xi_{\psi_k} := \frac{\psi_k}{\rho_{\psi_k}}. \quad (3.12)$$

**Step I-b:** Define the transformations  $T_{e_k} : (-C_{k,\text{col}}, C_{k,\text{con}}) \rightarrow \mathbb{R}$ ,  $k \in \mathcal{K}$ , and  $T_\psi : [0, 1) \rightarrow [0, \infty)$  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  $\varepsilon_{e_k} := \varepsilon_{e_k}(\xi_{e_k}) : (-1, 1) \rightarrow \mathbb{R}$ ,  $\varepsilon_{\psi_k} := \varepsilon_{\psi_k}(\xi_{\psi_k}) : [0, 1) \rightarrow \mathbb{R}_{\geq 0}$ ,  $\forall k \in \mathcal{K}$ ,

$$\varepsilon_{e_k} := T_{e_k}(\xi_{e_k}), \quad (3.13a)$$

$$\varepsilon_{\psi_k} := T_\psi(\xi_{\psi_k}). \quad (3.13b)$$

Next, we design the decentralized reference velocity vector for each agent as

$$v_{i,\text{des}} := \begin{bmatrix} v_{i,L\text{des}} \\ \omega_{i,\text{des}} \end{bmatrix} := -\delta_i \begin{bmatrix} 2 \sum_{k \in \mathcal{M}} \alpha_f \frac{r_{e_k}(\xi_{e_k})}{\rho_{e_k}} \varepsilon_{e_k} R_{k_1}^\top \tilde{p}_{k_2,k_1} \\ \sum_{k \in \mathcal{K}} \alpha_f \frac{r_\psi(\xi_{\psi_k})}{\rho_{\psi_k}} e_{R_k} \end{bmatrix}, \quad (3.14)$$

where  $\delta_i \in \mathbb{R}_{>0}$  are positive gains,  $\forall i \in \mathcal{N}$ ,  $r_{e_k} : (-C_{k,\text{col}}, C_{k,\text{con}}) \rightarrow [1, \infty)$ ,  $r_\psi : [0, 1) \rightarrow [1, \infty)$ , 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  $\alpha_f := \alpha_f(i, k, R_{k_1}, R_{k_2})$  is defined as  $\alpha_f(i, k, R_{k_1}, R_{k_2}) = -I_3$ , if  $i$  is the tail of the  $k$ th edge ( $i = k_1$ ),  $\alpha_f(i, k, R_{k_1}, R_{k_2}) = R_{k_2}^\top R_{k_1}$  if  $i$  is the head of the  $k$ th edge ( $i = k_2$ ), and 0 otherwise (see Appendix E for more details on graph edges). The assignment of the head and tail in each edge can be done off-line according to the specified orientation of the graph.

**Step II-a:** Define for each agent the velocity errors  $e_{v_i} := [e_{v_i,1}^\top, \dots, e_{v_i,6}^\top]^\top := \tilde{v}_i - v_{i,\text{des}}$ ,  $\forall i \in \mathcal{N}$ , and design the decreasing performance functions as  $\rho_{v_{i,\ell}} := \rho_{v_{i,\ell}}(t) : \mathbb{R}_{\geq 0} \rightarrow [\rho_{v_{i,\ell}^0}, \rho_{v_{i,\ell}^\infty}]$ , 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

$$\xi_{v_i} := [\xi_{v_i,1}, \dots, \xi_{v_i,6}]^\top := \rho_{v_i}^{-1} e_{v_i}, \quad (3.15)$$

where  $\rho_{v_i} := \rho_{v_i}(t) := \text{diag}\{\rho_{v_{i,\ell}}\}_{\ell \in \{1, \dots, 6\}}$ ,  $\forall i \in \mathcal{N}$ .

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

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

and the transformed error states  $\varepsilon_{v_i} : (-1, 1)^6 \rightarrow \mathbb{R}^6$  as

$$\varepsilon_{v_i}(\xi_{v_i}) := \varepsilon_{v_i} := \begin{bmatrix} \varepsilon_{v_{i,1}} \\ \vdots \\ \varepsilon_{v_{i,6}} \end{bmatrix} := \begin{bmatrix} T_v(\xi_{v_{i,1}}) \\ \vdots \\ T_v(\xi_{v_{i,6}}) \end{bmatrix}. \quad (3.16)$$

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

$$u_i := u_i(\xi_{v_i}, t) := -\gamma_i \rho_{v_i}(t)^{-1} \bar{r}_v(\xi_{v_i}) \varepsilon_{v_i}(\xi_v), \quad (3.17)$$

where  $\bar{r}_v(\xi_{v_i}) := \text{diag}\{[r_v(\xi_{v_{i,\ell}})]_{\ell \in \{1, \dots, 6\}}\}$  with  $r_v : (-1, 1) \rightarrow [1, \infty)$ ,  $r_v(x) := \frac{\partial T_v(x)}{\partial x}$ , 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  $C_{k,\text{col}}, C_{k,\text{con}}$  according to (3.9) and of  $\rho_{\psi_k}(t), \rho_{v_{i,\ell}}(t)$  such that  $\rho_{\psi_{k,0}} = \rho_{\psi_k}(0) \in (\psi_k(0), 2)$ ,  $\rho_{v_{i,\ell}^0} = \rho_{v_{i,\ell}}(0) > |e_{v_{i,\ell}}(0)|$  along with (3.4), guarantee that  $\xi_{e_k}(0) \in (C_{k,\text{col}}, C_{k,\text{con}})$ ,  $\psi_k(0) \in [0, 2)$ ,  $\xi_{v_{i,\ell}}(0) \in (-1, 1)$ ,  $\forall k \in \mathcal{K}$ ,  $\ell \in \{1, \dots, 6\}$ ,  $i \in \mathcal{N}$ . 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  $(-C_{k,\text{col}}, C_{k,\text{con}})$ ,  $[0, 2)$ , and  $(-1, 1)$ , respectively,  $\forall k \in \mathcal{K}, \ell \in \{1, \dots, 6\}, i \in \mathcal{N}, t \geq 0$ , 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  $\omega_{k_1,\text{des}} = \omega_{k_2,\text{des}} = 0$ ,  $\forall k \in \mathcal{K}$ . 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  $\mathbb{SO}(3)$ , and cannot be resolved with a purely continuous controller [?]. 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 [?]. 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

(3.8b) 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  $l_{\psi_k}$ . It can be observed from the desired angular velocities designed  $\omega_{i,\text{des}}$  in (3.14) 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  $\delta_i$  and  $\gamma_i$ .

**Remark 3.4. (Decentralized manner, relative feedback, and robustness)** Notice by (3.14) and (3.17) 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  $\varsigma_j, \forall j \in \mathcal{N}_i(p(0))$ , 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  $\rho_{v_{i,\ell}^\infty}, l_{v_{i,\ell}}$  can be transmitted off-line to each agent  $i$ , which can also compute  $\rho_{v_{i,\ell}^0}$ , given the initial velocity errors  $e_{v_i}(0)$ . Notice also from (3.14) 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  $R_{k_1}^\top(p_{k_2} - p_{k_1})$ , and the head of edge  $k$ , i.e., agent  $k_2$ , uses feedback of  $R_{k_2}^\top R_{k_1} R_{k_1}^\top(p_{k_2} - p_{k_1}) = R_{k_2}^\top(p_{k_2} - p_{k_1})$ . 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  $R_{k_2}^\top R_{k_1}$ , as well as the signal  $S^{-1}(R_{k_1}^\top R_{k_2} R_{k,\text{des}} - R_{k,\text{des}}^\top R_{k_2}^\top R_{k_1})$ , which is a function of  $R_{k_2}^\top R_{k_1}$ . 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 (3.17) 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.

We provide now the main result of this section0, 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  $R_{k,des}$ ,  $\forall k \in \mathcal{K}$ . Then, the control protocol (3.12)-(3.17) guarantees the prescribed transient and steady-state performance

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

$\forall k \in \mathcal{K}$ ,  $t \in \mathbb{R}_{\geq 0}$ , under all initial conditions satisfying  $\psi_k(0) < 2$ ,  $\forall k \in \mathcal{K}$  and (3.4), providing thus a solution to Problem 3.1.

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

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

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

With the introduced notation, (3.6) can be written in vector form as:

$$\dot{e} = \begin{bmatrix} \dot{e}_1 \\ \vdots \\ \dot{e}_K \end{bmatrix} = \begin{bmatrix} 2(R_{1_1}^\top \tilde{p}_{1_2,1_1})^\top (R_{1_1}^\top R_{1_2} v_{1_2,L} - v_{1_1,L}) \\ \vdots \\ 2(R_{K_1}^\top \tilde{p}_{K_2,K_1})^\top (R_{K_1}^\top R_{K_2} v_{K_2,L} - v_{K_1,L}) \end{bmatrix}$$

$$= 2 \begin{bmatrix} \tilde{p}_{1_2,1_1}^\top & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \tilde{p}_{K_2,K_1}^\top \end{bmatrix} \hat{R} D_R^\top v_L =: \mathbb{F}_p^\top \hat{R} D_R^\top v_L, \quad (3.18a)$$

$$\dot{\psi} = \begin{bmatrix} \dot{\psi}_1 \\ \vdots \\ \dot{\psi}_K \end{bmatrix} = \frac{1}{2} \begin{bmatrix} e_{R_1}^\top (R_{1_1}^\top R_{1_2} \omega_{1_2} - \omega_{1_1}) \\ \vdots \\ e_{R_K}^\top (R_{K_1}^\top R_{K_2} \omega_{1_2} - \omega_{K_1}) \end{bmatrix}$$

$$= \frac{1}{2} \begin{bmatrix} e_{R_1}^\top & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & e_{R_K}^\top \end{bmatrix} D_R^\top \omega =: \mathbb{F}_R^\top D_R^\top \omega, \quad (3.18b)$$

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

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

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

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

$$D_R(R, \mathcal{G}) := \bar{R}^\top [D \otimes I_3] \hat{R}, \quad (3.19)$$

with  $\bar{R} := \text{diag}\{[R_i]_{i \in \mathcal{N}}\} \in \mathbb{R}^{3N \times 3N}$ , and  $D := D(\mathcal{G})$  is the incidence matrix of the graph (see Section E.1 of Appendix E). The terms  $\bar{R}$  and  $\hat{R}$  in  $D_R$  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  $\alpha_f(\cdot)$  in the desired velocities  $v_{i,\text{des}}$  designed in (3.14), since, as shown next, the vector form  $v_{\text{des}}$  yields the orientation incidence matrix  $D_R(R, \mathcal{G})$ .

The desired velocities (3.14) and control inputs (3.17) can be written in vector form as

$$\dot{v}_{L\text{des}} = -\Delta D_R \hat{R}^\top \mathbb{F}_p \Sigma_e \varepsilon_e, \quad (3.20\text{a})$$

$$\omega_{\text{des}} = -\Delta D_R [\Sigma_\psi \otimes I_3] e_R, \quad (3.20\text{b})$$

$$u = -\Gamma \Sigma_v \varepsilon_v, \quad (3.20\text{c})$$

where  $\Delta := \text{diag}\{\delta_i I_3\}_{i \in \mathcal{N}} \in \mathbb{R}^{3N \times 3N}$  and  $\Gamma := \text{diag}\{\gamma_i I_6\}_{i \in \mathcal{N}} \in \mathbb{R}^{6N \times 6N}$ . Note from (3.20c) and (3.12), (3.15), (3.13), (3.16) that  $u$  can be expressed as a function of the states  $x, v, t$ . Hence, the closed loop system can be written as

$$\dot{x} = h_x(x, v)$$

$$\dot{v} = -M^{-1} \left\{ C(v)v + g(x) + w(x, v, t) - u(\cdot) \right\} =: h_v(x, v, t).$$

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

$$\dot{z} = h_z(z, t) := (h_x(z), h_v(z, t)). \quad (3.21)$$

Next, define the set

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

where we abuse the notation and express  $\xi_{e_k}$ ,  $\xi_{\psi_k}$ ,  $\xi_{v_i}$  from (3.12), (3.15) as a function of the states. It can be verified that the set  $\Omega$  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 (3.9). Our goal here is to prove first that (3.21) has a unique and maximal solution  $(z(t), t)$  in  $\Omega$  and then that this solution stays in a compact subset of  $\Omega$ .

It can be verified that the function  $h : \Omega \rightarrow \mathbb{T}_{R_1} \times \cdots \times \mathbb{T}_{R_N} \times \mathbb{R}^{6N}$  is (a) continuous in  $t$  for each fixed  $(x, v) \in \{(x, v) \in \mathbb{SE}(3)^N \times \mathbb{R}^{6N} : (x, v, t) \in \Omega\}$ , and (b) continuous and locally lipschitz in  $(x, v)$  for each fixed  $t \in \mathbb{R}_{\geq 0}$ . Therefore, the conditions of Theorem A.1 of Appendix A are satisfied and hence, we conclude the existence of a unique and maximal solution of (3.21) 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

$$\xi_{e_k}(t) = \frac{e_k(t)}{\rho_{e_k}(t)} \in (-1, 1), \quad (3.22\text{a})$$

$$\xi_{\psi_k}(t) = \frac{\psi_k(t)}{\rho_{\psi_k}(t)} < 1, \quad (3.22\text{b})$$

$$\xi_{v_i}(t) = \rho_{v_i}(t)^{-1} e_{v_i}(t) \in (-1, 1)^6, \quad (3.22\text{c})$$

$\forall k \in \mathcal{K}, i \in \mathcal{N}, t \in [0, t_{\max}]$ . 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  $\Omega$  and hence, by employing Theorem A.2 of Appendix A, that  $t_{\max} = \infty$ .

Consider the positive definite Lyapunov candidate  $V_e := V_e(\varepsilon_e) : (-1, 1)^K \rightarrow \mathbb{R}_{\geq 0}$ , with  $V_e(\varepsilon_e) := \frac{1}{2}\|\varepsilon_e\|^2$ , which is well defined for  $t \in [0, t_{\max}]$ , due to (3.22a). By differentiating  $V_e$  and taking into account the dynamics  $\dot{\xi}_e = \rho_e(t)^{-1} [\dot{e} - \dot{\rho}_e(t)\xi_e]$ , we obtain

$$\dot{V}_e = \left[ \frac{\partial V_e}{\partial \varepsilon} \right] \dot{\varepsilon}_e = \varepsilon_e^\top \Sigma_e \left( \mathbb{F}_p^\top \hat{R} D_R^\top v_L - \dot{\rho}_e \xi_e \right),$$

which, by substituting  $v_L = \tilde{v}_L - \mathbf{n}_p = e_{v_p} + v_{L\text{des}} - \mathbf{n}_p$  and (3.18), becomes

$$\dot{V}_e = -\varepsilon_e^\top \Sigma_e \mathbb{F}_p^\top \tilde{D} \mathbb{F}_p \Sigma_e \varepsilon_e + \varepsilon_e^\top \Sigma_e \left[ \mathbb{F}_p^\top \hat{R} D_R^\top (e_{v_p} - \mathbf{n}_p) - \dot{\rho}_e \xi_e \right], \quad (3.23)$$

where  $\tilde{D} := \tilde{D}(\mathcal{G}) := \hat{R} D_R^\top D_R \hat{R}^\top = D^\top \otimes I_3 \Delta D \otimes I_3 \in \mathbb{R}^{3K \times 3K}$  (by employing (3.19)), and  $e_{v_p}, \mathbf{n}_p$  are the linear parts of  $\bar{e}_v$  and  $\mathbf{n} := [\mathbf{n}_1^\top, \dots, \mathbf{n}_N^\top]^\top$  (i.e., the stack vector of the first three components of every  $e_{v_i}, \mathbf{n}_i$ ), respectively. Note first that, due to (3.22a), the function  $e_{v_p}(t)$  is bounded for all  $t \in [0, t_{\max}]$ . Moreover, note that (3.22a) 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  $\text{rank}(\mathbb{F}_p(\tilde{p}(t))) = K$ ,  $\forall t \in [0, t_{\max}]$ . In addition, since  $\mathcal{G}$  is a connected tree graph and  $\delta_i \in \mathbb{R}_{>0}$ ,  $\forall i \in \mathcal{N}$ ,  $\tilde{D}$  is positive definite (see Lemma E.1 of Appendix E) and hence  $\text{rank}(\tilde{D}) = 3K$ . Hence, we conclude that  $\text{rank}(\mathbb{F}_p(\tilde{p}(t))^\top \tilde{D} \mathbb{F}_p(\tilde{p}(t))) = K$  and the positive definiteness of  $\mathbb{F}_p(\tilde{p}(t))^\top \tilde{D} \mathbb{F}_p(\tilde{p}(t))$ ,  $\forall t \in [0, t_{\max}]$ . In addition, since  $\|p_{k_2}(t) - p_{k_1}(t)\| < d_{k,\text{con}}$ , we also conclude that the term  $\mathbb{F}_p^\top \hat{R} D_R^\top$  is upper bounded,  $\forall t \in [0, t_{\max}]$ . Finally,  $\dot{\rho}_e$  and  $\mathbf{n}_p$  are bounded by definition and assumption, respectively,  $\forall x \in \mathbb{SE}(3)^N, t \in \mathbb{R}_{\geq 0}$ . Note that all the aforementioned bounds are independent of  $t_{\max}$ . We obtain now from (3.23):

$$\dot{V}_e \leq -\lambda_{\tilde{D}} \|\Sigma_e \varepsilon_e\|^2 + \|\Sigma_e \varepsilon_e\| \bar{B}_e = -\lambda_{\tilde{D}} \|\Sigma_e \varepsilon_e\| \left( \|\Sigma_e \varepsilon_e\| - \frac{\bar{B}_e}{\lambda_{\tilde{D}}} \right),$$

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

$$\lambda_{\tilde{D}} := \inf_{p(t), t \in [t_0, t_{\max}]} \left\{ \lambda_{\min} \left( \mathbb{F}_p(\tilde{p}(t))^\top \tilde{D} \mathbb{F}_p(\tilde{p}(t)) \right) \right\} \geq d_{k,\text{col}}^2 \lambda_{\min}(\tilde{D}) > 0,$$

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

Hence, we conclude that  $\dot{V}_e < 0 \Leftrightarrow \|\Sigma_e \varepsilon_e\| > \frac{\bar{B}_e}{\lambda_{\bar{D}}}$ . By noting that

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

$\forall x \in (-C_{k,\text{col}}, C_{k,\text{con}})$ , as well as  $\rho_{ek}(t) \leq 1, \forall t \in \mathbb{R}_{\geq 0}, k \in \mathcal{K}$ , we conclude that  $\|\Sigma_e(\xi_e(t), t)\varepsilon_e(\xi_e(t))\| = \sqrt{\sum_{k \in \mathcal{K}} \frac{r_{ek}(\xi_{ek}(t))^2}{\rho_{ek}(t)^2} \varepsilon_{ek}(\xi_e(t))^2} \geq \bar{C}_e \|\varepsilon_e(\xi_e(t))\|$ ,  $\forall t \in [0, t_{\max})$ , where  $\bar{C}_e := \max_{k \in \mathcal{K}} \left\{ \frac{1}{C_{k,\text{col}}} + \frac{1}{C_{k,\text{con}}} \right\}$ . Hence, we conclude that  $\dot{V}_e < 0, \forall \|\varepsilon_e\| \geq \frac{\bar{B}_e}{\lambda_{\bar{D}} \bar{C}_e}, \forall t \in [0, t_{\max})$  and therefore

$$\|\varepsilon_e(\xi_e(t))\| \leq \bar{\varepsilon}_e := \max \left\{ \varepsilon_e(\xi_e(0)), \frac{\bar{B}_e}{\lambda_{\bar{D}} \bar{C}_e} \right\}, \quad (3.24)$$

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

$$-C_{k,\text{col}} < -\underline{\xi}_e \leq \xi_{ek}(t) \leq \bar{\xi}_e < C_{k,\text{con}}, \quad (3.25)$$

$\forall t \in [0, t_{\max})$ , where  $\bar{\xi}_e := \frac{\exp(\bar{\varepsilon}_e)-1}{\exp(\bar{\varepsilon}_e)+1} C_{k,\text{con}}$ , and  $\underline{\xi}_e := \frac{\exp(-\bar{\varepsilon}_e)-1}{\exp(-\bar{\varepsilon}_e)+1} C_{k,\text{col}}$ . Hence, (3.24) and (3.25) imply the boundedness of  $\varepsilon_{ek}(\xi_{ek}(t))$ ,  $r_{ek}(\xi_{ek}(t))$ ,  $\tilde{p}(t)$ , and  $p(t)$  in compact sets,  $\forall k \in \mathcal{K}$ , and therefore, through (3.14), 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  $V_\psi := V_\psi(\xi_\psi) : [0, 1]^K \rightarrow \mathbb{R}_{\geq 0}$ , with  $V_\psi = 2 \sum_{k \in \mathcal{K}} \varepsilon_{\psi_k}$ . By differentiating  $V_\psi$  and taking into account the dynamics  $\dot{\xi}_{\psi_k} = \rho_{\psi_k}^{-1} [\dot{\psi}_k - \dot{\rho}_{\psi_k} \xi_{\psi_k}]$ , we obtain

$$\dot{V}_\psi := \left[ \frac{\partial V_\psi}{\partial \varepsilon_e} \right] \dot{\varepsilon}_\psi = 2 \sum_{k \in \mathcal{K}} \frac{r_\psi(\xi_{\psi_k})}{\rho_{\psi_k}} (\dot{\psi}_k - \dot{\rho}_{\psi_k} \xi_{\psi_k}),$$

which, after substituting (3.6b), (3.18), becomes

$$\begin{aligned} \dot{V}_\psi &= e_R^\top (\Sigma_\psi \otimes I_3) D_R^\top \omega - 2 \sum_{k \in \mathcal{K}} \frac{r_\psi(\xi_{\psi_k})}{\rho_{\psi_k}} \dot{\rho}_{\psi_k} \xi_{\psi_k} \\ &= e_R^\top (\Sigma_\psi \otimes I_3) D_R^\top (\omega_{\text{des}} + e_{v_R} - \mathbf{n}_R) - 2 \sum_{k \in \mathcal{K}} \frac{r_\psi(\xi_{\psi_k})}{\rho_{\psi_k}} \dot{\rho}_{\psi_k} \xi_{\psi_k}, \end{aligned}$$

where  $e_{v_R}$  and  $\mathbf{n}_R$  are the angular parts of  $\bar{e}_v$  and  $\mathbf{n}$  (i.e., the stack vector of the last three components of every  $e_{v_i}, \mathbf{n}_i$ ), respectively. By substituting

(3.20b) and defining  $\tilde{\Sigma}_\psi := \Sigma_\psi \otimes I_3 \in \mathbb{R}^{3K \times 3K}$ ,  $\tilde{D}_R := D_R^\top \Delta D_R \in \mathbb{R}^{3K \times 3K}$ , we obtain:

$$\dot{V}_\psi = -e_R^\top \tilde{\Sigma}_\psi \tilde{D}_R \tilde{\Sigma}_\psi e_R + e_R^\top \tilde{\Sigma}_\psi D_R^\top (e_{v_R} - \mathbf{n}_R) - 2 \sum_{k \in \mathcal{K}} \frac{r_\psi(\xi_{\psi_k})}{\rho_{\psi_k}} \dot{\rho}_{\psi_k} \xi_{\psi_k}. \quad (3.26)$$

According to (3.19),  $D_R = \bar{R}^\top (D \otimes I_3) \hat{R}$ . Since  $\bar{R}$  and  $\hat{R}$  are rotation (and thus unitary) matrices, the singular values of  $D_R$  are identical to the ones of  $D$ , and hence  $\lambda_{\min}(\tilde{D}_R) = \lambda_{\min}(\tilde{D}) > 0$ . Indeed, let  $D \otimes I_3 = U \Sigma_D V^\top$  be a singular value decomposition of  $D \otimes I_3$ , where  $U, V$  are unitary matrices, and  $\Sigma_D$  is a diagonal matrix containing the singular values of  $D \otimes I_3$ . Then  $D_R = \bar{R}^\top U \Sigma_D V^\top \hat{R} = \tilde{U} \Sigma_D \tilde{V}^\top$  where  $\tilde{U} := \bar{R}^\top U$ , and  $\tilde{V} = \hat{R}^\top V$  are unitary matrices (being products of unitary matrices). Thus,  $\tilde{U} \Sigma_D \tilde{V}^\top$  is the singular value decomposition of  $D_R$ , and hence its singular values are the diagonal values of  $\Sigma_D$ . By further defining  $\beta := [\beta_1^\top, \dots, \beta_K^\top]^\top := D_R^\top (e_{v_R} - \mathbf{n}_R) \in \mathbb{R}^{3M}$ , with  $\beta_k \in \mathbb{R}^3, \forall k \in \mathcal{K}$ , (3.26) becomes

$$\dot{V}_\psi \leq -\lambda_{\min}(\tilde{D}) \|\tilde{\Sigma}_\psi e_R\|^2 + \sum_{k \in \mathcal{K}} \frac{r_\psi(\xi_{\psi_k})}{\rho_{\psi_k}} e_{R_k}^\top \beta_k - 2 \sum_{k \in \mathcal{K}} \frac{r_\psi(\xi_{\psi_k})}{\rho_{\psi_k}} \dot{\rho}_{\psi_k} \xi_{\psi_k}.$$

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 (3.22b), we conclude that  $r_\psi(\xi_{\psi_k}(t)) > 1, \forall t \in [0, t_{\max}]$ . By noting also that  $\dot{\rho}_{\psi_k}(t) < 0, \forall t \in \mathbb{R}_{\geq 0}$ ,  $\dot{V}_\psi$  becomes

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

where  $\bar{B}_{\psi_1}$  is a positive constant, independent of  $t_{\max}$ , satisfying  $\bar{B}_{\psi_1} \geq \max_{k \in \mathcal{K}} \{\|\beta_k(t)\|\}, \forall t \in [0, t_{\max}]$ . Note that  $\bar{B}_{\psi_1}$  is finite,  $\forall t \in [0, t_{\max}]$ , due to (3.22b) and the boundedness of the noise signals. After substituting (3.7), we obtain

$$\begin{aligned} \dot{V}_\psi &\leq -2\lambda_{\min}(\tilde{D}) \sum_{k \in \mathcal{K}} \left[ \frac{r_\psi(\xi_{\psi_k})}{\rho_{\psi_k}} \right]^2 \psi_k (2 - \psi_k) \\ &\quad + \bar{B}_{\psi_1} \sum_{k \in \mathcal{K}} \frac{r_\psi(\xi_{\psi_k})}{\rho_{\psi_k}} \sqrt{2\psi_k(2 - \psi_k)} + 2 \max_{k \in \mathcal{K}} \{l_{\psi_k}(\rho_{\psi_{k,0}} - \rho_{\psi_{k,\infty}})\} \sum_{k \in \mathcal{K}} \frac{r_\psi(\xi_{\psi_k})}{\rho_{\psi_k}} \xi_{\psi_k}. \end{aligned} \quad (3.27)$$

From (3.22b) 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  $\psi_k = \xi_{\psi_k} \rho_{\psi_k}(t)$ ,  $\forall k \in \mathcal{K}$ , (3.27) becomes

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

where

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

From (3.22b), (3.12), 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

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

where:

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

We conclude therefore that  $\dot{V}_{\psi} < 0 \Leftrightarrow \sqrt{\sum_{k \in \mathcal{K}} r_{\psi}(\xi_{\psi_k})^2 \xi_{\psi_k}} > \frac{\bar{B}_{\psi}}{\tilde{\mu}}$ . From (3.13b), given  $y = T_{\psi}(x)$ , we obtain:

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

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

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

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

$$\|\varepsilon_\psi(\xi_\psi(t))\| \leq \bar{\varepsilon}_\psi := \max \left\{ \varepsilon_\psi(\xi_\psi(0)), \frac{\bar{B}_\psi}{\bar{\mu}} \right\}, \quad (3.28)$$

and, by taking the inverse logarithm:

$$0 \leq -\underline{\xi}_\psi \leq \xi_{\psi_k}(t) \leq \bar{\xi}_\psi < 1, \quad (3.29)$$

where  $\bar{\xi}_\psi := \frac{\exp(\bar{\varepsilon}_\psi) - 1}{\exp(\bar{\varepsilon}_\psi)}$  and  $\underline{\xi}_\psi := \frac{\exp(-\bar{\varepsilon}_\psi) - 1}{\exp(-\bar{\varepsilon}_\psi)}$ ,  $\forall k \in \mathcal{K}$ . Therefore, we conclude the boundedness of  $\varepsilon_{\psi_k}(\psi_k(t)), r_{\psi_k}(\xi_{\psi_k}(t)), \bar{e}_v(t)$  in compact sets,  $\forall k \in \mathcal{K}$ , and therefore, through (3.14), 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  $\mathbf{n}(x(t), t)$  and invoking  $\tilde{v} = v + \mathbf{n}(x, t) = \bar{e}_v(t) - v_{\text{des}}(t)$  and (3.22c), the boundedness of  $v(t)$  and  $\dot{x}(t)$ ,  $\forall t \in [0, t_{\max}]$ . Moreover, in view of (3.24), (3.25), (3.21), (3.14), we also conclude the boundedness of  $\dot{v}_{\text{des}}(t)$ .

Proceeding along similar lines, we consider the positive definite Lyapunov candidate  $V_v := V_v(\varepsilon_v) : (-1, 1)^{6N} \rightarrow \mathbb{R}_{\geq 0}$  with  $V_v = \frac{1}{2}\varepsilon_v^\top \Gamma \varepsilon_v$ . By computing  $\dot{V}_v = \left[ \frac{\partial V_v}{\partial \varepsilon_v} \right] \dot{\varepsilon}_v$  and using the dynamics  $\dot{\xi}_v = \rho_v^{-1}(\dot{e}_v - \dot{\rho}_v \xi_v)$ , we obtain

$$\begin{aligned} \dot{V}_v = & \varepsilon_v^\top \Gamma \Sigma_v [\dot{v} + \dot{\mathbf{n}}] - \varepsilon_v^\top \Gamma \Sigma_v \dot{v}_{\text{des}} - \varepsilon_v^\top \Gamma \Sigma_v \dot{\rho}_v \xi_v = -\varepsilon_v^\top \Sigma_v \Gamma M^{-1} \Gamma \Sigma_v \varepsilon_v \\ & - \varepsilon_v^\top \Sigma_v \left\{ \Gamma M^{-1} [Cv + g + w] - \dot{\mathbf{n}} + \dot{v}_{\text{des}} + \dot{\rho}_v \xi_v \right\}. \end{aligned} \quad (3.30)$$

Since we have proved the boundedness of  $v(t)$  and  $\dot{x}(t)$ ,  $\forall t \in [0, t_{\max}]$  the terms  $Cv$ ,  $\dot{\mathbf{n}}$ , and  $w$  are also bounded,  $t \in [0, t_{\max}]$ , due to the continuities of  $C$ ,  $w$ , and  $\dot{\mathbf{n}}$  in  $v$ ,  $\dot{x}$  and the boundedness of  $w$  and  $\dot{\mathbf{n}}$  in  $x, t$ . Moreover,  $g$ ,  $\xi_v$ , and  $\dot{\rho}_v$  are also bounded due to (3.2b), (3.22c), and by construction, respectively. By also using (3.2a), we obtain from (3.30):

$$\dot{V}_v \leq -\lambda_K \|\Sigma_v \varepsilon_v\|^2 + \|\Sigma_v \varepsilon_v\| \bar{B}_v,$$

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

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

$\forall x \in (-1, 1)$ , 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  $\|\Sigma_v \varepsilon_v(\xi_v(t))\| = \sqrt{\sum_{i \in \mathcal{N}} \sum_{\ell \in \{1, \dots, 6\}} \frac{r_v(\xi_{v_i, \ell}(t))^2}{\rho_{v_i, \ell}(t)^2} \varepsilon_{v_i, \ell}(\xi_{v_i, \ell}(t))^2}$

$\geq \frac{1}{\tilde{\rho}} \|\varepsilon_v(\xi_v(t))\|, \forall t \in [0, t_{\max}),$  where  $\tilde{\rho} := \max_{\substack{i \in \mathcal{N} \\ m \in \{1, \dots, 6\}}} \{\rho_{v_i^0, m}\}.$  Hence, we

conclude that  $\dot{V}_v < 0, \forall \|\varepsilon_v\| \geq \frac{\tilde{\rho} \bar{B}_v}{\lambda_K}, \forall t \in [0, t_{\max}),$  and consequently that

$$\|\varepsilon_v(\xi_v(t))\| \leq \bar{\varepsilon}_v := \max \left\{ \varepsilon_v(\xi_v(0)), \frac{\tilde{\rho} \bar{B}_v}{\lambda_K} \frac{\max_{i \in \mathcal{N}} \{\gamma_i\}}{\min_{i \in \mathcal{N}} \{\gamma_i\}} \right\},$$

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

$$-1 < -\bar{\xi}_v \leq \xi_{v_i, \ell}(t) \leq \bar{\xi}_v < 1, \quad (3.31)$$

$\forall \ell \in \{1, \dots, 6\}, t \in [0, t_{\max})$  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  $\varepsilon_v(\xi_v(0))$  is finite due to the choice  $\rho_{v_i^0, \ell} > |e_{v_i, \ell}(0)|, \forall \ell \in \{1, \dots, 6\}, i \in \mathcal{N}.$  Hence, since  $\lambda_K$  is strictly positive, the term  $\bar{\varepsilon}_v$  is also finite. Thus, the term  $\tilde{r}_v(\xi_v(t))$  and hence the control laws (3.17) 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 A.2 of Appendix A, it holds that  $\lim_{t \rightarrow t_{\max}^-} \left( \|z(t)\| + \frac{1}{d_S((z(t), t), \partial\Omega)} \right) = \infty,$  where  $d_S(\cdot)$  is the distance of  $(z(t), t)$  to  $\partial\Omega.$  We first rewrite the condition in a more explicit form, in order to account for the matrix tuple  $R \in \mathbb{SO}(3)^N.$  We define  $z_{p, v} := [p^\top, v^\top]^\top \in \mathbb{R}^{3N} \times \mathbb{R}^{6N},$  the projection sets  $\Omega_R := \{(R, t) \in \mathbb{SO}(3)^N \times \mathbb{R}_{\geq 0} : (x, v, t) \in \Omega\}$  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  $A \subset \mathbb{SO}(3)^N \times \mathbb{R}_{\geq 0}$  as  $d_{S, \mathbb{SO}(3)} : \mathbb{SO}(3)^N \times \mathbb{R}_{\geq 0} \times 2^{\mathbb{SO}(3)^N \times \mathbb{R}_{\geq 0}} \rightarrow \mathbb{R}_{\geq 0}$  with  $d_{S, \mathbb{SO}(3)}((R, t), A) := \inf_{(R_A, t_A) \in A} \{\|R - R_A\|_T + t - t_A\},$

where  $\|\cdot\|_T$  is the induced norm in  $\mathbb{SO}(3)^N$  defined as  $\|R\|_T := \sum_{i \in \mathcal{N}} \|R_i\|_F,$  for  $R = (R_1, \dots, R_N) \in \mathbb{SO}(3)^N.$  Therefore, the condition of Theorem A.2 of Appendix A can now be stated as follows: Since  $t_{\max} < \infty,$  it holds that

$$L := \lim_{t \rightarrow t_{\max}^-} \left( \|p(t)\| + \|v(t)\| + \|R(t)\|_T + \frac{1}{d_S((z_{p, v}(t), t), \partial\Omega_{p, v}) + d_{S, \mathbb{SO}(3)}((R(t), t), \partial\Omega_R)} \right) = \infty \quad (3.32)$$

which we aim to prove that is a contradiction. Firstly, it holds that

$$\|R(t)\|_T = \sum_{i \in \mathcal{N}} \|R_i(t)\|_F \leq N \sup_{t \in [0, t_{\max}]} \left\{ \max_{i \in \mathcal{N}} \{R_i(t)\} \right\}.$$

However, according to Proposition G.3, it holds that  $-1 \leq \text{tr}(R) \leq 3$  for any  $R \in \mathbb{SO}(3).$  Hence,  $\|R(t)\|_T \leq 3N, \forall t \in [0, t_{\max}].$  Moreover, from (3.31) and

(3.15) we obtain  $\|\bar{e}_v(t)\| \leq \sqrt{6}\bar{\xi}_v\tilde{\rho}$ ,  $\forall t \in [0, t_{\max}]$ . By invoking (3.24), (3.28), we can also conclude that there exists a finite  $\bar{v}_{\text{des}}$  such that  $\|v_{\text{des}}(t)\| \leq \bar{v}_{\text{des}}$ ,  $\forall t \in [0, t_{\max}]$ . Hence, since  $\|\mathbf{n}_i(x_i(t), t)\| \leq \bar{\mathbf{n}}_i$ ,  $\forall t \in \mathbb{R}_{\geq 0}$ ,  $i \in \mathcal{N}$ ,  $v = \tilde{v} - \mathbf{n} = \bar{e}_v + v_{\text{des}} - \mathbf{n}$  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  $\partial\Omega_{p,v} = \{(p, v, t) \in \mathbb{R}^{3N} \times \mathbb{R}^{6N} \times \mathbb{R}_{\geq 0} : (\exists k \in \mathcal{K} : \xi_{e_k}(p_{k_1}, p_{k_2}, t) = -C_{k,\text{col}} \text{ or } \xi_{e_k}(p_{k_1}, p_{k_2}, t) = C_{k,\text{con}}) \text{ or } (\exists i \in \mathcal{N}, \ell \in \{1, \dots, 6\} : \xi_{v_{i,\ell}}(x, v_i, t) = -1 \text{ or } \xi_{v_{i,\ell}}(x, v_i, t) = 1)\}$  and  $\partial\Omega_R = \{(R, t) \in \mathbb{SO}(3)^N \times \mathbb{R}_{\geq 0} : \exists k \in \mathcal{K} : \xi_{\psi_k}(R_{k_1}, R_{k_2}, t) = 1\}$ . We have proved, however, from (3.25), (3.29), and (3.31) that the maximal solution satisfies the strict inequalities  $-C_{k,\text{col}} < -\xi_e \leq \xi_{e_k}(p_{k_1}(t), p_{k_2}(t), t) \leq \bar{\xi}_e < C_{k,\text{con}}$ ,  $\xi_{\psi_k}(R_{k_1}(t), R_{k_2}(t), t) \leq \bar{\xi}_{\psi} < 1$ , and  $|\xi_{v_{i,\ell}}(x(t), v_i(t), t)| \leq \bar{\xi}_v < 1$ ,  $\forall k \in \mathcal{K}$ ,  $\ell \in \{1, \dots, 6\}$ ,  $i \in \mathcal{N}$ ,  $t \in [0, t_{\max}]$ . Therefore, we conclude that there exist strictly positive constants  $\epsilon_{p,v}, \epsilon_R \in \mathbb{R}_{>0}$  such that  $d_S((z_{p,v}(t), t), \partial\Omega_{p,v}) \geq \epsilon_{p,v}$  and  $d_{S,\mathbb{SO}(3)}((R(t), t), \partial\Omega_R) \geq \epsilon_R$ ,  $\forall t \in [0, t_{\max}]$ . 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 (3.32) 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  $\mathcal{E}$ .  $\square$

**Remark 3.5.** It is worth noting that the control scheme applies also to the case that the graph is minimally rigid, which implies similar properties regarding  $\tilde{D}$  [?].

### 3.2.3 Simulation Results

We considered  $N = 4$  spherical agents with  $\mathcal{N} = \{1, 2, 3, 4\}$  and dynamics of the form (3.1), with  $r_i = 1\text{m}$  and  $\varsigma_i = 4\text{m}$ ,  $i \in \{1, \dots, 4\}$ . 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:  $p_1(0) = [0, 0, 0]^\top \text{ m}$ ,  $p_2(0) = [2, 2, 2]^\top \text{ m}$ ,  $p_3(0) = [2, 4, 4]^\top \text{ m}$ ,  $p_4(0) = [2, 3, 3]^\top \text{ m}$ ,  $R_1(0) = R_3(0) = R_4(0) = I_3$  and

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

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

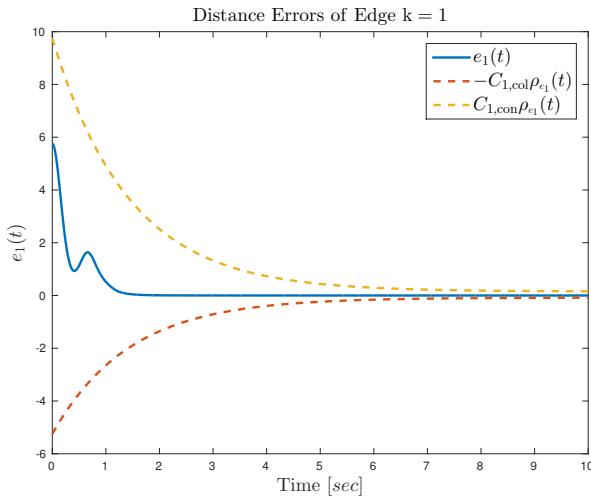
$$D(\mathcal{G}) = \begin{bmatrix} -1 & 0 & 0 \\ 1 & -1 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

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

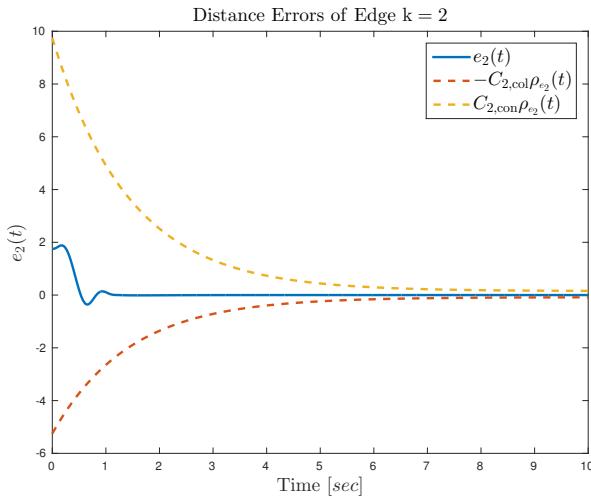
$$R_{k,\text{des}} = \begin{bmatrix} 0.5000 & -0.8660 & 0.0000 \\ 0.6124 & 0.3536 & -0.7071 \\ 0.6124 & 0.3536 & 0.7071 \end{bmatrix}, \forall k \in \{1, 2, 3\}.$$

The definitions of  $d_{k,\text{col}}$ ,  $d_{k,\text{con}}$  yield:  $d_{k,\text{col}} = 2$  and  $d_{k,\text{con}} = 4$ . Invoking (3.9), 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  $\rho_{v_{i,\ell}^0} = 2|e_{v_{i,\ell}}(0)| + 0.5$ ,  $l_{v_{i,\ell}} = 1.55$  and  $\rho_{v_{i,\ell}^\infty} = 0.15$ , for every  $i \in \{1, \dots, 4\}$ ,  $\ell \in \{1, \dots, 6\}$ . Finally, the control gains were set to  $\Gamma = 10I_{24}$  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. A video illustrating the simulation results can be found on <https://www.youtube.com/watch?v=Z4xLyO1twvk>.



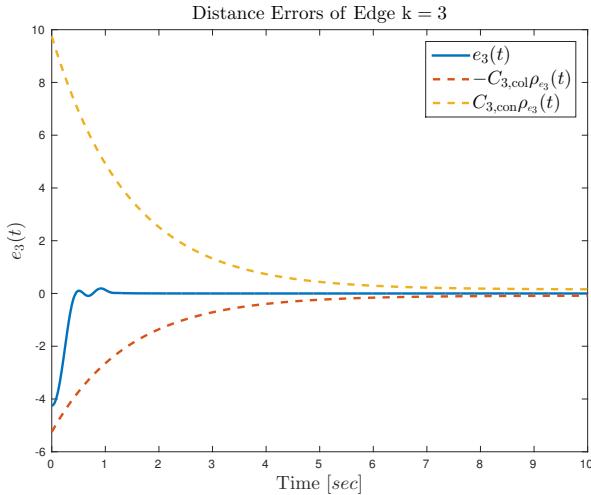
**Figure 3.2:** The distance error signal of the edge (1, 2).



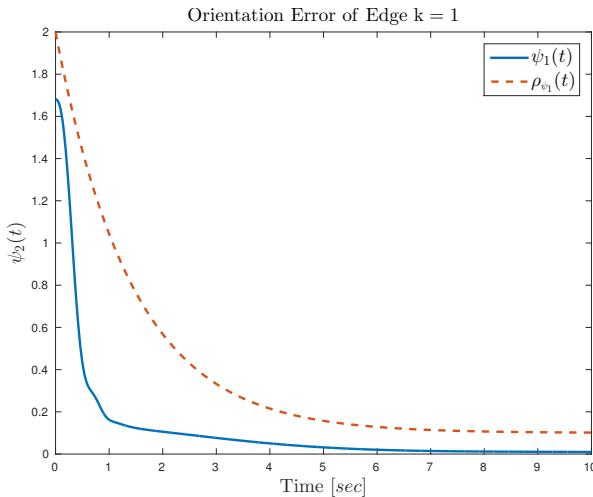
**Figure 3.3:** The distance error signal of the edge (2, 3).

### 3.3 Cooperative Manipulation via Internal Force Regulation: A Rigidity Theory Approach

Except from the more classical formation control framework, connected to the previous section, formation control can be connected to cooperative manipulation, where the nodes of the formation graph are the robotic agents

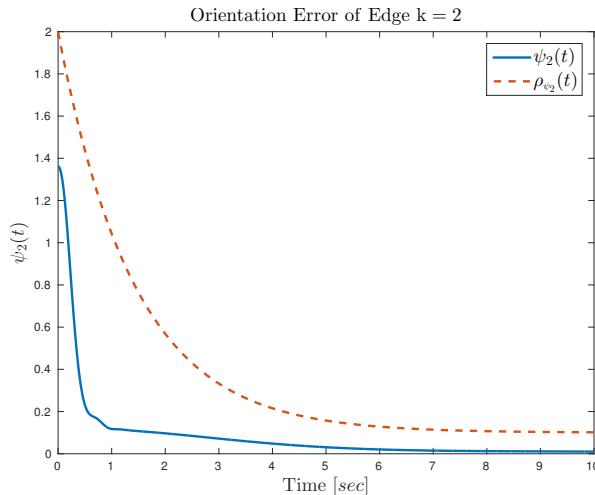


**Figure 3.4:** The distance error signal of the edge (2, 4).

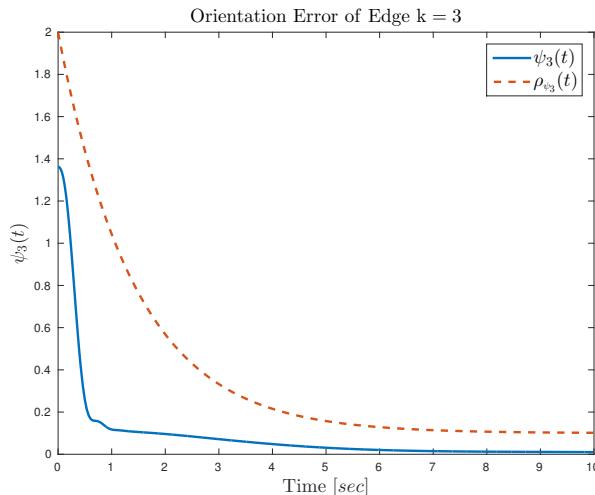


**Figure 3.5:** The orientation error signal of the edge (1, 2).

grasping the object. In this section we associate classical *rigidity* theory with rigid cooperative manipulation. In particular, motivated by the rigid grasps in a cooperative manipulation scheme, we first introduce the notion of distance and bearing rigidity of a graph in  $\mathbb{SE}(3)$ . Next, we associate the nodes of the graph to the robotic agents in a cooperative manipulation scheme, and we provide new results on the interaction and internal forces as



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

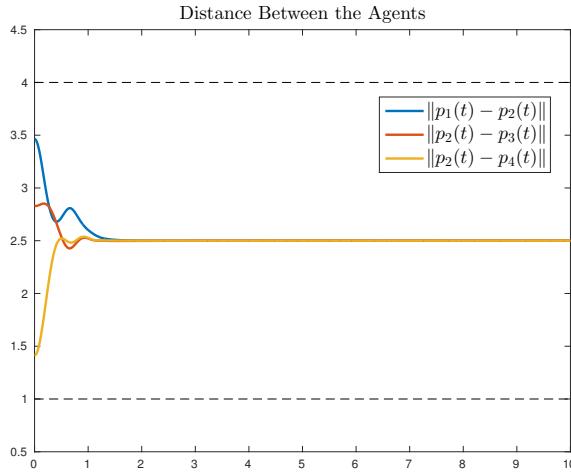


**Figure 3.7:** The orientation error signal of the edge (2, 4).

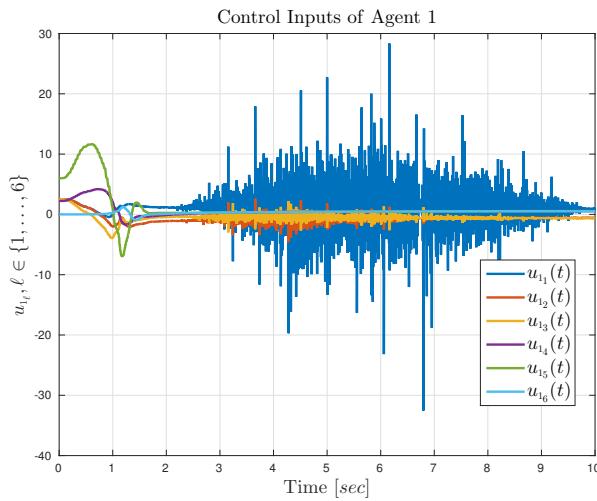
well as optimal cooperative manipulation.

### 3.3.1 Cooperative Manipulation Modeling

Regarding rigid cooperative manipulation, we follow the notation and dynamics of the previous chapter. We provide here a brief recap.

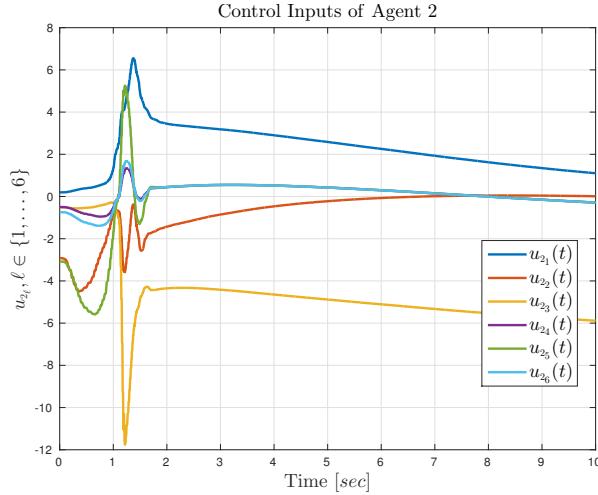


**Figure 3.8:** The distance between the agents.

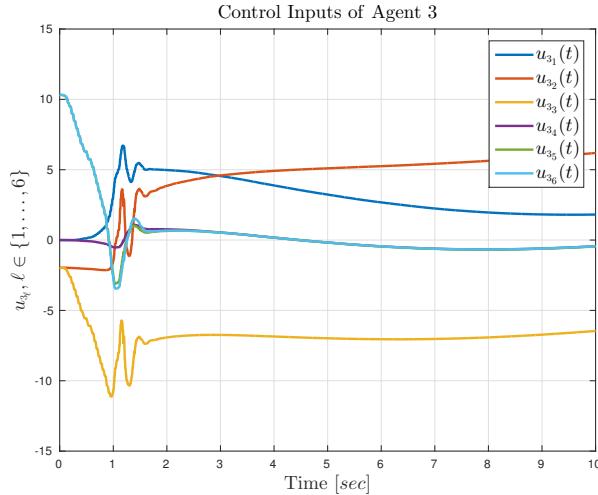


**Figure 3.9:** The control input signals of agent 1.

Let  $N$  robotic agents, with  $\mathcal{N} := \{1, \dots, N\}$ , rigidly grasping an object, with  $q_i \in \mathbb{R}^{n_i}$  their joint configurations,  $q := [q_1^\top, \dots, q_N^\top]^\top \in \mathbb{R}^n$ ,  $n := \sum_{i \in \mathcal{N}} n_i$ ,  $p_{E_i} \in \mathbb{R}^3$ ,  $\eta_{E_i} \in \mathbb{T}$ ,  $R_i(\eta_i) \in \mathbb{SO}(3)$  and  $v_i := [p_{E_i}^\top, \omega_{E_i}^\top]^\top \in \mathbb{R}^6$  the end-effector poses and velocities, and  $x_i := (p_{E_i}, R_i) \in \mathbb{SE}(3)$ ,  $x := (x_1, \dots, x_N) \in \mathbb{SE}(3)^N$ ,  $v := [v_1^\top, \dots, v_N^\top]^\top \in \mathbb{R}^{6N}$ . The stacked agent



**Figure 3.10:** The control input signals of agent 2.



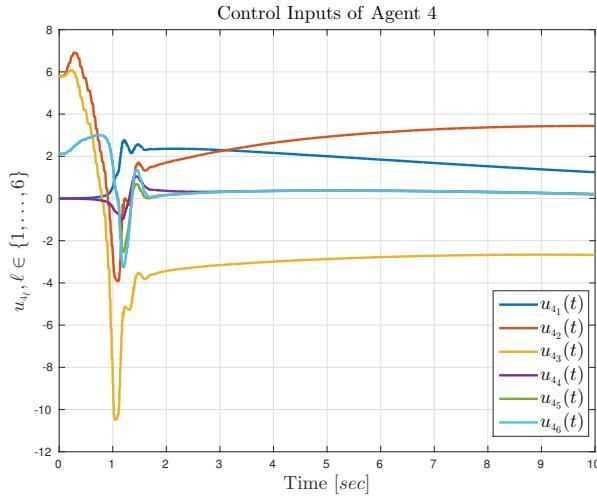
**Figure 3.11:** The control input signals of agent 3.

dynamics in joint- and task-space are (see (2.1) and (2.4))

$$B(q)\ddot{q} + C_q(q, \dot{q})\dot{q} + g_q(q) = \tau - J(q)^\top h, \quad (3.33a)$$

$$M(q)\dot{v} + C(q, \dot{q})v + g(q) = u - h, \quad (3.33b)$$

where  $J := \text{diag}\{[J_i]_{i \in \mathcal{N}}\}$ , and we have removed the disturbance vector for simplicity. We remind the reader that the task-space terms are defined in



**Figure 3.12:** The control input signals of agent 4.

$S_i = \{q_i \in \mathbb{R}^n : \det(J_i(q_i)J_i(q_i)^\top) > 0\}$ . The object pose and velocity are denoted by  $x_o := [p_o^\top, \eta_o^\top]^\top \in \mathbb{M}$ ,  $R_o(\eta_o) \in \mathbb{SO}(3)$ ,  $v_o := [\dot{p}_o^\top, \omega_o^\top]^\top \in \mathbb{R}^6$ , and dynamics

$$\dot{R}_o = S(\omega_o)R_o, \quad (3.34a)$$

$$M_o(\eta_o)\dot{v}_o + C_o(\eta_o, \omega_o)v_o + g_o = h_o. \quad (3.34b)$$

In view of Fig. 2.1, one obtains the coupled kinematics

$$v_i = J_{O_i}v_o, \quad (3.35)$$

where  $J_{O_i} := J_{O_i}(x_i) : \mathbb{SE}(3) \rightarrow \mathbb{R}^{6 \times 6}$  is the object-to-agent Jacobian introduced in (2.9), redefined here as a function of  $x_i$  instead of  $q_i$ , i.e.,

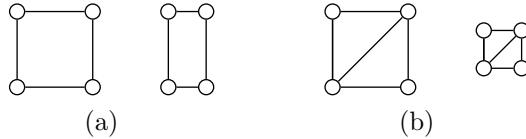
$$J_{O_i}(x_i) := \begin{bmatrix} I_3 & -S(R_i^\top p_{E_i/O}^{E_i}) \\ 0 & I_3 \end{bmatrix},$$

which forms the respective grasp matrix

$$G := G(x) := [J_{O_1}(x_1)^\top, \dots, J_{O_N}(x_N)^\top] \in \mathbb{R}^{6 \times 6N}, \quad (3.36)$$

and has full column-rank due to the rigidity of the grasping contacts; Note that (3.35) can now be written in stack vector form as

$$v = G^\top v_o. \quad (3.37)$$



**Figure 3.13:** Illustration of bearing rigidity. The networks in (a) are not bearing rigid because the same inter-neighbor bearings may lead to different geometric patterns of the networks, for example, a square on the left and a rectangle on the right. The networks in (b) are bearing rigid because the same inter-neighbor bearings imply the same geometric pattern though the networks may differ in terms of translation and scale.

Next, we associate  $h$  and  $h_O$  via  $G$  (as in (2.13)) to obtain

$$h_O = Gh, \quad (3.38)$$

which leads to the coupled dynamics (see (2.14))

$$\widetilde{M}(\bar{x})\dot{v}_O + \widetilde{C}(\bar{x})v_O + \widetilde{g}(\bar{x}) = Gu, \quad (3.39)$$

where we slightly change the notation with respect to the previous chapter as  $\bar{x} := [q^\top, \dot{q}^\top, \eta_O^\top, \omega_O^\top]^\top \in \mathbb{X} = \mathbb{S} \times \mathbb{R}^{n+6} \times \mathbb{T}$  (instead of  $x$ ).

The vector of interaction forces  $h$  among the agents and the object can be decoupled into motion-induced and internal forces

$$h = h_m + h_{int}. \quad (3.40)$$

The internal forces  $h_{int}$  are squeezing forces that the agents exert to the object and belong to the nullspace of  $G(x)$  (i.e.,  $Gh_{int} = 0$ ). Hence, they do not contribute to the acceleration of the coupled system and result in internal stresses that might damage the object. A closed form analytic expression for  $h_m$  and  $h_{int}$  will be given in the next section.

Note from (3.37) that the agent velocities  $v$  belong to the range space of  $G^\top$ . Therefore, since  $G$  is a matrix that encodes rigidity constraints, this motivates the association of  $G$  to the *rigidity matrix* used in formation rigidity theory, and of the rigid cooperative manipulation scheme to a multi-agent rigid formation scheme. To this end, we introduce now the notion of Distance and Bearing Rigidity in  $\mathbb{SE}(3)$ .

### 3.3.2 Distance and Bearing Rigidity in $\mathbb{SE}(3)$

We begin by recalling that the range space of the grasp matrix  $G^T$  corresponds to the rigid body translations and rotations of the system. While this matrix appears naturally in the context of dynamic modeling of rigid bodies, it is

also indirectly related to the notion of structural rigidity in discrete geometry, which is a combinatorial theory for determining the flexibility of ensembles formed by rigid bodies connected by flexible linkages or hinges.

In the classical structural rigidity theory, one considers a collection of rigid bars connected by joints allowing free rotations around the joint axis - this is known as a *bar-and-joint* framework. One is then interested in understanding what are the allowable motions of the framework, i.e., those motions that preserve the lengths of the bars and their connections to the joints. The so-called *trivial motions* for these frameworks are precisely the rigid body translations and rotations of the system. For some frameworks, there may be additional motions, known as *flexes*, that also preserve the constraints. This is captured by the notion of *infinitesimal motions* of the framework and is characterized by the *rigidity matrix* of the framework [? ].

Here we can consider frameworks that also encode the pose of the joints in addition to the lengths of the rigid bars connecting them, leading to a *distance and bearing*-type framework. Bearing rigidity has been recently explored in the context of formation control and studies the problem of under what conditions the geometric pattern of a network can be uniquely determined if the bearing of each edge in the network is fixed [? ] (see Fig. 3.13). The bearing rigidity has also been extended to frameworks embedded in  $\mathbb{SE}(2)$  and  $\mathbb{SE}(3)$ [? ? ]. Both the bearing and distance rigidity theories have found many applications for multi-agent systems, in particular for formation control and localization [? ? ? ? ].

In this chapter we introduce and formalize the concept of distance and bearing rigidity (abbreviated as D&B Rigidity in the following). This is motivated by the fusion of both distance and pose constraints in the cooperative grasping problem. D&B Rigidity in  $\mathbb{SE}(3)$  aims at studying the problem of under what conditions the geometric pattern of a network can be uniquely determined if *both* the bearing and the distance of each edge in the network is fixed. In this direction, we focus on the notion of infinitesimal rigidity for D&B frameworks. We first formally define a D&B framework in  $\mathbb{SE}(3)$ , similarly to Section E.2 of Appendix E:

**Definition 3.1.** A framework in  $\mathbb{SE}(3)$  is a triple  $(\mathcal{G}, p_{\mathcal{G}}, R_{\mathcal{G}})$ , where  $\mathcal{G} := (\mathcal{N}, \mathcal{E})$  is a graph,  $p_{\mathcal{G}} : \mathcal{N} \rightarrow \mathbb{R}^3$  is a function mapping each node to a position in  $\mathbb{R}^3$ , and  $R_{\mathcal{G}} : \mathcal{N} \rightarrow \mathbb{SO}(3)$  is a function associating each node with an orientation element of  $\mathbb{SO}(3)$  (both with respect to an inertial frame).

As in the previous section concerning formation control, we employ the Special Orthogonal Group (rotation matrices)  $\mathbb{SO}(3)$  to express the orientation of the agents. Moreover, we use the shorthand notation  $p_i := p_{\mathcal{G}}(i)$ ,  $R_i := R_{\mathcal{G}}(i)$ ,  $p := [p_1^\top, \dots, p_N^\top]^\top \in \mathbb{R}^{3N}$ ,  $R := (R_1, \dots, R_N) \in \mathbb{SO}(3)^N$ ,

$x_i := (p_i, R_i) \in \mathbb{SE}(3)$ , and  $x := (x_1, \dots, x_N) \in \mathbb{SE}(3)^N$ . The distances and bearings in a framework can be summarized through the following *SE(3) D&B rigidity function*,  $\gamma_{\mathcal{G}}$ , that encodes the rigidity constraints in the framework. Consider a directed graph  $\mathcal{G}$ , where  $\mathcal{E} \subseteq \{(i, j) \in \mathcal{N}^2 : i \neq j\}$ , as well as its *undirected part*  $\mathcal{E} \supseteq \mathcal{E}_u := \{(i, j) \in \mathcal{E} : i < j\}$ . Then  $\gamma_{\mathcal{G}}$  can be formed by considering the distance and bearing functions  $\gamma_{e,d} : \mathbb{R}^3 \times \mathbb{R}^3 \rightarrow \mathbb{R}_{\geq 0}$ ,  $\gamma_{e,b} : \mathbb{SE}(3)^2 \rightarrow \mathbb{S}^2$ , with

$$\gamma_{e,d}(p_i, p_j) := \frac{1}{2} \|p_i - p_j\|^2, \forall e = (i, j) \in \mathcal{E}_u, \quad (3.41a)$$

$$\gamma_{e,b}(x_i, x_j) := R_i^\top \frac{p_j - p_i}{\|p_i - p_j\|}, \forall e = (i, j) \in \mathcal{E}, \quad (3.41b)$$

which encodes the distance  $\|p_i - p_j\|$  between two agents as well as the local bearing vector  $R_i^\top \frac{p_j - p_i}{\|p_i - p_j\|}$ , expressed in the frame of agent  $i$ . Note that the distance functions are considered only for the undirected part of  $\mathcal{G}$ , since  $\gamma_{(i,j),d} = \gamma_{(j,i),d}$ . Now  $\gamma_{\mathcal{G}}$  is formed by stacking the aforementioned distance and bearing functions, i.e.,  $\gamma_{\mathcal{G}} := \gamma_{\mathcal{G}}(x) : \mathbb{SE}(3)^N \rightarrow \mathbb{R}^{|\mathcal{E}_u|} \times \mathbb{S}^{2|\mathcal{E}|}$ , with

$$\gamma_{\mathcal{G}} := \begin{bmatrix} \gamma_d(p) \\ \gamma_b(x) \end{bmatrix} := \begin{bmatrix} \gamma_{1,d} \\ \vdots \\ \gamma_{|\mathcal{E}_u|,d} \\ \gamma_{1,b} \\ \vdots \\ \gamma_{|\mathcal{E}|,b} \end{bmatrix}. \quad (3.42)$$

Note that the aforementioned expressions for  $\gamma_{e,d}$ ,  $\gamma_{e,b}$  are not unique and other choices that capture the rigidity constraints can also be made. We also mention our slight abuse of notation, where the index  $k$  in  $\gamma_{k,d}$  and  $\gamma_{k,b}$  refers to a labeled edge in  $\mathcal{E}_u$  and  $\mathcal{E}_b$ .

In this section, we are interested in the set of D&B *infinitesimal* motions of a framework in  $\mathbb{SE}(3)$ . These can be thought as perturbations to a framework in  $\mathbb{SE}(3)$  that leave  $\gamma_{\mathcal{G}}$  unchanged. More information about the separate distance and bearing infinitesimal motions can be found in Section E.2 of Appendix E. The set of D&B *infinitesimal* motions is characterized by the nullspace of the Jacobian of the  $\mathbb{SE}(3)$ -D&B rigidity function arising from the Taylor series expansion of  $\gamma_{\mathcal{G}}$ . That is, the nullspace of the matrix  $\nabla_{(p,R)} \gamma_{\mathcal{G}}$ , that we term the  $\mathbb{SE}(3)$ -D&B *rigidity matrix*. This matrix is

denoted as  $\mathcal{R}_{\mathcal{G}} : \mathbb{SE}(3)^N \rightarrow \mathbb{R}^{(|\mathcal{E}_u|+3|\mathcal{E}|) \times 6N} := \nabla_{(p,R)} \gamma_{\mathcal{G}}$ , i.e.,

$$\mathcal{R}_{\mathcal{G}}(x) = \begin{bmatrix} \frac{\partial \gamma_{1,d}}{\partial p_1} & \frac{\partial \gamma_{1,d}}{\partial R_1} & \cdots & \frac{\partial \gamma_{1,d}}{\partial p_N} & \frac{\partial \gamma_{1,d}}{\partial R_N} \\ \vdots & \ddots & & & \vdots \\ \frac{\partial \gamma_{|\mathcal{E}_u|,d}}{\partial p_1} & \frac{\partial \gamma_{|\mathcal{E}_u|,d}}{\partial R_1} & \cdots & \frac{\partial \gamma_{|\mathcal{E}_u|,d}}{\partial p_N} & \frac{\partial \gamma_{|\mathcal{E}_u|,d}}{\partial R_N} \\ \frac{\partial \gamma_{1,b}}{\partial p_1} & \frac{\partial \gamma_{1,b}}{\partial R_1} & \cdots & \frac{\partial \gamma_{1,b}}{\partial p_N} & \frac{\partial \gamma_{1,b}}{\partial R_N} \\ \vdots & \ddots & & & \vdots \\ \frac{\partial \gamma_{|\mathcal{E}|,b}}{\partial p_1} & \frac{\partial \gamma_{|\mathcal{E}|,b}}{\partial R_1} & \cdots & \frac{\partial \gamma_{|\mathcal{E}|,b}}{\partial p_N} & \frac{\partial \gamma_{|\mathcal{E}|,b}}{\partial R_N} \end{bmatrix}, \quad (3.43)$$

with

$$\begin{aligned} \frac{\partial \gamma_{e,d}}{\partial x_i} &= \begin{bmatrix} \frac{\partial \gamma_{e,d}}{\partial p_i} & \frac{\partial \gamma_{e,d}}{\partial R_i} \end{bmatrix} = \begin{bmatrix} (p_i - p_j)^\top & 0_{1 \times 3} \end{bmatrix}, \\ \frac{\partial \gamma_{e,d}}{\partial x_j} &= \begin{bmatrix} \frac{\partial \gamma_{e,d}}{\partial p_j} & \frac{\partial \gamma_{e,d}}{\partial R_j} \end{bmatrix} = \begin{bmatrix} (p_j - p_i)^\top & 0_{1 \times 3} \end{bmatrix}, \\ \frac{\partial \gamma_{e,b}}{\partial x_i} &= \begin{bmatrix} \frac{\partial \gamma_{e,b}}{\partial p_i} & \frac{\partial \gamma_{e,b}}{\partial R_i} \end{bmatrix} = \begin{bmatrix} -\frac{P_r(\gamma_{e,b})}{\|p_j - p_i\|} R_i^\top & S(\gamma_{e,b}) R_i^\top \end{bmatrix}, \\ \frac{\partial \gamma_{e,b}}{\partial x_j} &= \begin{bmatrix} \frac{\partial \gamma_{e,b}}{\partial p_j} & \frac{\partial \gamma_{e,b}}{\partial R_j} \end{bmatrix} = \begin{bmatrix} \frac{P_r(\gamma_{e,b})}{\|p_j - p_i\|} R_i^\top & 0_{3 \times 3} \end{bmatrix}. \end{aligned}$$

Here,  $P_r(\gamma_{e,b})$  is defined as

$$P_r(\gamma_{e,b}) := I_3 - \frac{(p_j - p_i)(p_j - p_i)^T}{\|p_j - p_i\|^2},$$

and projects vectors onto the orthogonal complement of  $(p_j - p_i)$ . See [?] for more discussion on this projection matrix and its use in the bearing rigidity theory. Infinitesimal motions, therefore, are motions  $x(t)$  produced by velocities  $v(t)$  that lie in the nullspace of  $\mathcal{R}_{\mathcal{G}}$ , for which it holds that  $\dot{\gamma}_{\mathcal{G}} = \mathcal{R}_{\mathcal{G}}(x(t))v(t) = 0$ , where  $v := [\dot{p}_1^\top, \omega_1^\top, \dots, \dot{p}_N^\top, \omega_N^\top]^\top$ , as defined before. The infinitesimal motions therefore depend on the number of motion degrees of freedom the entire framework possesses. This directly relates to the structure of the underlying graph. Motions that preserve the distances and bearings of the framework for *any* underlying graph are called *D&B trivial motions*. This leads to the definition of *infinitesimal rigidity*, stated below.

**Definition 3.2.** A framework  $(\mathcal{G}, p_{\mathcal{G}}, R_{\mathcal{G}})$  is D&B infinitesimally rigid in  $\mathbb{SE}(3)$  if every D&B infinitesimal motion is a D&B trivial motion.

We now aim to identify precisely what the trivial motions of a D&B framework are, and to determine conditions for a framework to be infinitesimally rigid based on properties of the D&B rigidity matrix. Before we

proceed, we first note that the D&B rigidity function in  $\mathbb{SE}(3)$  can be seen as a superposition of the rigidity functions associated with the classic distance rigidity theory [?] and the  $\mathbb{SE}(3)$  bearing rigidity theory [?]. In particular, we note that  $\mathcal{R}_{\mathcal{G},d} : \mathbb{R}^{3N} \rightarrow \mathbb{R}^{|\mathcal{E}_u| \times 3N} := \nabla_p \gamma_d$  is the well-studied (distance) rigidity matrix, while  $\mathcal{R}_{\mathcal{G},b} : \mathbb{SE}^{3N} \rightarrow \mathbb{R}^{3\mathcal{E} \times 6N} := \nabla_{(p,R)} \gamma_{\mathcal{G},b}$  is the  $\mathbb{SE}(3)$  bearing rigidity matrix. Note that the distance rigidity matrix is associated with the framework  $(\mathcal{G}, p_{\mathcal{G}})$ , which is the projection of  $(\mathcal{G}, p_{\mathcal{G}}, R_{\mathcal{G}})$  to  $\mathbb{R}^3$ . With an appropriate permutation,  $P_R$ , of the columns of  $\mathcal{R}_{\mathcal{G}}$ , we have that

$$\begin{aligned} \tilde{\mathcal{R}}_{\mathcal{G}} &:= \mathcal{R}_{\mathcal{G}} P_R \\ &= \begin{bmatrix} \frac{\partial \gamma_{1,d}}{\partial p_1} & \cdots & \frac{\partial \gamma_{1,d}}{\partial p_N} & \frac{\partial \gamma_{1,d}}{\partial R_1} & \cdots & \frac{\partial \gamma_{1,d}}{\partial R_N} \\ \vdots & & \ddots & & & \vdots \\ \frac{\partial \gamma_{M_{\mathcal{G}},d}}{\partial p_1} & \cdots & \frac{\partial \gamma_{M_{\mathcal{G}},d}}{\partial p_N} & \frac{\partial \gamma_{M_{\mathcal{G}},d}}{\partial R_1} & \cdots & \frac{\partial \gamma_{M_{\mathcal{G}},d}}{\partial R_N} \\ \frac{\partial \gamma_{1,b}}{\partial p_1} & \cdots & \frac{\partial \gamma_{1,b}}{\partial p_N} & \frac{\partial \gamma_{1,b}}{\partial R_1} & \cdots & \frac{\partial \gamma_{1,b}}{\partial R_N} \\ \vdots & & \ddots & & & \vdots \\ \frac{\partial \gamma_{M_{\mathcal{G}},b}}{\partial p_1} & \cdots & \frac{\partial \gamma_{M_{\mathcal{G}},b}}{\partial p_N} & \frac{\partial \gamma_{M_{\mathcal{G}},b}}{\partial R_1} & \cdots & \frac{\partial \gamma_{M_{\mathcal{G}},b}}{\partial R_N} \end{bmatrix}, \end{aligned} \quad (3.44)$$

which is equal to

$$\tilde{\mathcal{R}}_{\mathcal{G}} = \begin{bmatrix} [\mathcal{R}_{\mathcal{G},d} & 0_{|\mathcal{E}_u| \times 3N}] \\ \mathcal{R}_{\mathcal{G},b} \end{bmatrix} =: \begin{bmatrix} \bar{\mathcal{R}}_{\mathcal{G},d} \\ \mathcal{R}_{\mathcal{G},b} \end{bmatrix}.$$

The nullspace of  $\tilde{\mathcal{R}}_{\mathcal{G}}$ , therefore, is the intersection of the nullspaces of  $\bar{\mathcal{R}}_{\mathcal{G},d}$  and  $\mathcal{R}_{\mathcal{G},b}$ .

With the above interpretation, we can now understand the trivial motions to be the intersection of trivial motions associated to distance rigidity with those associated to  $\mathbb{SE}(3)$  bearing rigidity. In particular, let

$$\mathcal{S}_d := \text{span} \{ \mathbb{1}_N \otimes I_3, \mathcal{L}_{\mathbb{R}^3}^{\circlearrowleft}(\mathcal{G}) \},$$

denote the trivial motions associated to a distance framework [?]. That is,  $\mathbb{1}_N \otimes I_3$  represents translations of the entire framework, and  $\mathcal{L}_{\mathbb{R}^3}^{\circlearrowleft}(\mathcal{G})$  is the rotational subspace induced by the graph  $\mathcal{G}$  in  $\mathbb{R}^3$ , i.e.,

$$\mathcal{L}_{\mathbb{R}^3}^{\circlearrowleft}(\mathcal{G}) = \text{span} \{ (I_3 \otimes S(\mathbf{e}_h)) p_{\mathcal{G}}, h = 1, 2, 3 \}.$$

These motions can be produced by the *linear* velocities of the agents. It is known that  $\mathcal{S}_d \subseteq \text{null}(\mathcal{R}_{\mathcal{G},d})$  for any underlying graph  $\mathcal{G}$  [?]. For the matrix  $\bar{\mathcal{R}}_{\mathcal{G},d}$ , we can define the corresponding set

$$\bar{\mathcal{S}}_d := \text{span} \left\{ \begin{bmatrix} \mathbb{1}_N \otimes I_3 \\ * \end{bmatrix}, \begin{bmatrix} \mathcal{L}_{\mathbb{R}^3}^{\circlearrowleft}(\mathcal{G}) \\ * \end{bmatrix} \right\} \subseteq \text{null}(\bar{\mathcal{R}}_{\mathcal{G},d}).$$

Note that the distance rigidity does not explicitly depend on the orientation of the nodes when expressed as a point in  $\mathbb{SE}(3)$ . This accounts for the free  $\star$  entry in the subspace  $\bar{S}_d$  corresponding to the rotations. Thus, the set of trivial motions in  $\mathbb{R}^3$  can be seen as the projection of  $\bar{S}_d$  in  $\mathbb{R}^3$ .

Similarly, for an  $\mathbb{SE}(3)$  bearing framework one can define the subspace [? ]

$$\mathcal{S}_b := \text{span} \left\{ \begin{bmatrix} \mathbb{1}_N \otimes I_3 \\ 0_{3N \times 3} \end{bmatrix}, \begin{bmatrix} p_{\mathcal{G}} \\ 0_{3N} \end{bmatrix}, \mathcal{L}_{\mathbb{SE}(3)}^{\circlearrowleft}(\mathcal{G}) \right\},$$

where the vector  $[p_{\mathcal{G}}^T, 0_{3N}^T]^T$  represents a scaling of the framework. The space  $\mathcal{L}_{\mathbb{SE}(3)}^{\circlearrowleft}(\mathcal{G})$  is the rotational subspace induced by  $\mathcal{G}$ , in  $\mathbb{SE}(3)$ ,

$$\mathcal{L}_{\mathbb{SE}(3)}^{\circlearrowleft}(\mathcal{G}) = \text{span} \left\{ \begin{bmatrix} (I_3 \otimes S(\mathbf{e}_h)) p_{\mathcal{G}} \\ \mathbb{1}_n \otimes \mathbf{e}_h \end{bmatrix}, h = 1, 2, 3 \right\}. \quad (3.45)$$

It is also known that  $\mathcal{S}_b \subseteq \text{null}(\mathcal{R}_{\mathcal{G}, b})$ . Thus  $\mathcal{S}_b$  describes the trivial motions of an  $\mathbb{SE}(3)$  bearing framework [? ].

The above discussion immediately leads to the following proposition.

**Proposition 3.1.** The trivial motions of a D&B framework are characterized by the set

$$\mathcal{S}_{db} := \bar{S}_d \cap \mathcal{S}_b = \text{span} \left\{ \begin{bmatrix} \mathbb{1}_N \otimes I_3 \\ 0_{3N \times 3} \end{bmatrix}, \mathcal{L}_{\mathbb{SE}(3)}^{\circlearrowleft}(\mathcal{G}) \right\}.$$

Furthermore, it follows that  $\mathcal{S}_{db} \subseteq \text{null}(\tilde{\mathcal{R}}_{\mathcal{G}})$ .

Having characterized the trivial motions, it now follows from Definition 3.2 that for infinitesimal rigidity, we require that  $\text{null}(\tilde{\mathcal{R}}_{\mathcal{G}}) = \mathcal{S}_{db}$ . This is summarized in the following proposition.

**Proposition 3.2.** The framework  $(\mathcal{G}, p_{\mathcal{G}}, R_{\mathcal{G}})$  is D&B infinitesimally rigid in  $\mathbb{SE}(3)$  if and only if

$$\begin{aligned} \text{null}(\tilde{\mathcal{R}}_{\mathcal{G}}) &= \text{null}(\bar{\mathcal{R}}_{\mathcal{G}, d}) \cap \text{null}(\mathcal{R}_{\mathcal{G}, b}) \\ &= \text{span} \left\{ \begin{bmatrix} \mathbb{1}_N \otimes I_3 \\ 0_{3N \times 3} \end{bmatrix}, \mathcal{L}_{\mathbb{SE}(3)}^{\circlearrowleft}(\mathcal{G}) \right\} = \mathcal{S}_{db}. \end{aligned}$$

Equivalently, the D&B framework is infinitesimally rigid in  $\mathbb{SE}(3)$  if and only if

$$\text{rank}(\tilde{\mathcal{R}}_{\mathcal{G}}) = \dim(\tilde{\mathcal{R}}_{\mathcal{G}}) - \dim(\text{null}(\tilde{\mathcal{R}}_{\mathcal{G}})) = 6N - 6.$$

Hence, all the motions produced by the nullspace of  $\tilde{\mathcal{R}}_{\mathcal{G}}$  for an infinitesimally rigid framework must correspond to trivial motions, i.e., coordinated

translations and rotations. Moreover, given (3.44), it follows that  $(\mathcal{G}, p_{\mathcal{G}}, R_{\mathcal{G}})$  is D&B infinitesimally rigid in  $\mathbb{SE}(3)$  if and only if

$$\text{null}(\mathcal{R}_{\mathcal{G}}) = \{x = P_R y \in \mathbb{SE}(3)^N : y \in \text{null}(\tilde{\mathcal{R}}_{\mathcal{G}})\}, \quad (3.46)$$

i.e., the nullspace of  $\mathcal{R}_{\mathcal{G}}$  consists of the vectors of  $\text{null}(\tilde{\mathcal{R}}_{\mathcal{G}})$  whose elements are permuted by  $P_R$ .

It is worth noting that the aforementioned results are not valid if the rigidity matrix loses rank, i.e.,  $\text{rank}(\mathcal{R}_{\mathcal{G}}) < \max\{\text{rank}(\mathcal{R}_{\mathcal{G}}(x)), x \in \mathbb{SE}(3)\}$ . These are degenerate cases that correspond, for example, to when all agents are aligned along a direction  $\mathbf{v} \in \mathbb{S}^2$ . For more discussion on these degenerate cases, the reader is referred to [?].

As a last remark, we observe that frameworks over the complete graph,  $(\mathcal{K}_N, p_{\mathcal{K}_N}, R_{\mathcal{K}_N})$ , are (except for the degenerate configurations), infinitesimally rigid. That is,  $\text{rank}(\tilde{\mathcal{R}}_{\mathcal{K}_N}) = 6N - 6$ . This leads to the following corollary.

**Corollary 3.1.** *Consider the D&B frameworks  $(\mathcal{G}, p_{\mathcal{G}}, R_{\mathcal{G}})$  and  $(\mathcal{K}_N, p_{\mathcal{K}_N}, R_{\mathcal{K}_N})$  for nondegenerate configurations  $(p_{\mathcal{G}}, R_{\mathcal{G}})$ . Then  $(\mathcal{G}, p_{\mathcal{G}}, R_{\mathcal{G}})$  is D&B infinitesimally rigid if and only if*

$$\text{rank}(\tilde{\mathcal{R}}_{\mathcal{G}}) = \text{rank}(\tilde{\mathcal{R}}_{\mathcal{K}_N}) = 6N - 6.$$

In the next section, we use the aforementioned results to link the D&B rigidity matrix of a complete graph to the forces  $h_m$  and  $h_{\text{int}}$  of (3.40).

### 3.3.3 Interaction Forces based on the D&B Rigidity Matrix and Internal-Force based Optimal Cooperative Manipulation

We provide here the main results of this section. Firstly, we give a closed form expression for the interaction and internal forces of the coupled system object-robots. Next, we connect these forces with the D&B rigidity matrix introduced before. After that, we use these results to provide a novel relation between the arising interaction and internal forces and we give conditions on the agent force distribution for cooperative manipulation free from internal forces. For the rest of the chapter, we use the following notation for the cooperative object-manipulation system:

$$\begin{aligned} \tilde{x} &:= [\dot{q}^\top, v_O^\top]^\top & \bar{B} &:= \bar{B}(\bar{x}) := \text{diag}\{B(q), M_O(\eta_O)\} \\ \bar{\tau} &:= [\tau^\top, 0_{1 \times 6}]^\top & \bar{C}_q &:= \bar{C}_q(\bar{x}) := \text{diag}\{C_q(q, \dot{q}), C_O(\eta_O, \omega_O)\} \\ \bar{g}_q &:= \bar{g}_q(\bar{x}) := [g_q(q)^\top, g_O^\top]^\top & \bar{M} &:= \bar{M}(\bar{x}) := \text{diag}\{M(q), M_O(\eta_O)\} \\ \bar{v} &:= [v^\top, v_O^\top]^\top & \bar{C} &:= \bar{C}(\bar{x}) := \text{diag}\{C(q, \dot{q}), C_O(\eta_O, \omega_O)\} \\ \bar{g} &:= \bar{g}(\bar{x}) := [g(q)^\top, g_O^\top]^\top & \bar{u} &:= [u^\top, 0_{1 \times 6}]^\top \\ \bar{J} &:= \bar{J}(q) := \text{diag}\{J(q), I_6\} \end{aligned}$$

### Interaction Forces Based on the D&B Rigidity Matrix

We provide here closed form expressions for the interaction forces of the coupled object-agents system and link them to the D&B rigidity matrix notion introduced in the previous section. In particular, we consider that the robotic agents and the object form a graph that will be defined in the sequel. Note that, due to the rigidity of the grasping points, the forces exerted by an agent influence, not only the object, but all the other agents as well. Hence, since there exists interaction among all the pairs of agents as well as the agents and the object, we model their connection as a complete graph, as described rigorously below. Moreover, as will be clarified later, the rigidity matrix of this graph encodes the constraints of the agents-object system, imposed by the rigidity of the grasping points, and plays an important role in the expression of the agents-object interaction forces.

Let the robotic agents form a framework  $(\mathcal{G}, p_{\mathcal{G}}, R_{\mathcal{G}})$  in  $\mathbb{SE}(3)$ , where  $\mathcal{G} := (\mathcal{N}, \mathcal{E})$  is the complete graph, i.e.,  $\mathcal{E} = \{(i, j) \in \mathcal{N}^2 : i \neq j\}$ , and  $p_{\mathcal{G}} := [p_1^\top, \dots, p_N^\top]^\top$ ,  $R_{\mathcal{G}} := (R_1, \dots, R_N)$ . Consider also the undirected part  $\mathcal{E}_u = \{(i, j) \in \mathcal{E} : i < j\}$  of  $\mathcal{E}$ , as also described in the previous section. Since the graph is complete, we conclude that  $|\mathcal{E}| = N(N - 1)$  and  $|\mathcal{E}_u| = \frac{N(N-1)}{2}$ . Moreover, consider the extended framework  $(\bar{\mathcal{G}}, p_{\bar{\mathcal{G}}}, R_{\bar{\mathcal{G}}})$  of the robotic agents and the object, i.e., where the object is considered as the  $(N+1)$ th agent;  $\bar{\mathcal{G}}$  is the complete graph  $\bar{\mathcal{G}} := (\bar{\mathcal{N}}, \bar{\mathcal{E}})$ , where  $\bar{\mathcal{N}} := \{1, \dots, \bar{N}\}$ ,  $\bar{N} := N + 1$ , and  $\bar{\mathcal{E}} := \{(i, j) \in \bar{\mathcal{N}}^2 : i \neq j\}$ , with  $|\bar{\mathcal{E}}| = \bar{N}(\bar{N} - 1)$ . Let also  $\bar{\mathcal{E}}_u := \{(i, j) \in \bar{\mathcal{N}} : i < j\}$  be the undirected edge part, with  $|\bar{\mathcal{E}}_u| = \frac{\bar{N}(\bar{N}-1)}{2}$ .

Consider now the rigidity functions  $\gamma_{e,d} : \mathbb{R}^3 \times \mathbb{R}^3 \rightarrow \mathbb{R}_{\geq 0}$ ,  $\forall e \in \bar{\mathcal{E}}_u$  and  $\gamma_{e,b} : \mathbb{SE}(3)^2 \rightarrow \mathbb{S}^2$ ,  $\forall e \in \bar{\mathcal{E}}$ , as given in (3.41), as well as the stack vector  $\gamma_{\bar{\mathcal{G}}} : \mathbb{SE}(3)^{\bar{N}} \rightarrow \mathbb{R}^{\frac{\bar{N}(\bar{N}-1)}{2}} \times \mathbb{S}^{2\bar{N}(\bar{N}-1)}$  as given in (3.42). The rigidity constraints of the framework are encoded in the constraint  $\gamma_{\bar{\mathcal{G}}} = \text{const.}$  Since the rigidity of the framework stems from the rigidity of the grasping points, these constraints encode also the rigidity constraints of the object-agent cooperative manipulation. By differentiating  $\gamma_{\bar{\mathcal{G}}} = \text{const.}$ , one obtains

$$\mathcal{R}_{\bar{\mathcal{G}}}\bar{v} = 0 \Leftrightarrow \mathcal{R}_{\bar{\mathcal{G}}}\bar{J}\tilde{x} = 0 \Rightarrow \mathcal{R}_{\bar{\mathcal{G}}}\bar{J}\dot{\tilde{x}} = -(\dot{\mathcal{R}}_{\bar{\mathcal{G}}}\bar{J} + \mathcal{R}_{\bar{\mathcal{G}}}\dot{\bar{J}})\tilde{x},$$

where  $\mathcal{R}_{\bar{\mathcal{G}}} := \mathcal{R}_{\bar{\mathcal{G}}}(x, x_O) : \mathbb{SE}(3)^{\bar{N}} \rightarrow \mathbb{R}^{\frac{7\bar{N}(\bar{N}-1)}{2} \times (6\bar{N})}$  is the rigidity matrix associated to  $\bar{\mathcal{G}}$  and has the form (3.43). We now write the aforementioned equations as

$$A\dot{\tilde{x}} = b,$$

where

$$A := A(\bar{x}, t) := \mathcal{R}_{\bar{\mathcal{G}}}(x, x_O)\bar{J}(q) \quad (3.47a)$$

$$b := b(\bar{x}, t) := -(\dot{\mathcal{R}}_{\bar{\mathcal{G}}}(x, x_O)\bar{J}(q) + \mathcal{R}_{\bar{\mathcal{G}}}(x, x_O)\dot{\bar{J}}(q))\tilde{x}. \quad (3.47b)$$

One can verify that the motion of the cooperative object-agents manipulation system that is enforced by the aforementioned constraints corresponds to rigid body motions (coordinated translations and rotations of the system). Hence, since  $\bar{\mathcal{G}}$  is complete, the analysis of the previous section dictates that these motions are the infinitesimal motions of the framework and are the ones produced by the nullspace of  $\mathcal{R}_{\bar{\mathcal{G}}}(x, x_o)$ .

Next, we turn to the main focus of our results, which is the case of internal forces and we consider the framework comprising only of the robotic agents ( $\mathcal{G}, p_{\mathcal{G}}, R_{\mathcal{G}}$ ). The inter-agent rigidity constraints are expressed by the D&B rigidity functions  $\gamma_{e,d} : \mathbb{R}^3 \times \mathbb{R}^3 \rightarrow \mathbb{R}_{\geq 0}$ ,  $\forall e \in \mathcal{E}_u$  and  $\gamma_{e,b} : \mathbb{SE}(3)^2 \rightarrow \mathbb{S}^2$ ,  $\forall e \in \mathcal{E}$ , as given in (3.41), as well as the stack vector  $\gamma_{\mathcal{G}} : \mathbb{SE}(3)^N \rightarrow \mathbb{R}^{\frac{N(N-1)}{2}} \times \mathbb{S}^{2N(N-1)}$  as given in (3.42). Differentiation of  $\gamma_{\mathcal{G}}(x(q)) = \text{const.}$ , which encodes the rigidity constraints of the system comprised by the robotic agents, yields

$$\mathcal{R}_{\mathcal{G}}v = 0 \Leftrightarrow \mathcal{R}_{\mathcal{G}}J\dot{q} = 0 \Rightarrow \mathcal{R}_{\mathcal{G}}J\ddot{q} = -(\mathcal{R}_{\mathcal{G}}\dot{J} + \dot{\mathcal{R}}_{\mathcal{G}}J)\dot{q},$$

written more compactly as

$$A_{\text{int}}\ddot{q} = b_{\text{int}},$$

where

$$A_{\text{int}} := A_{\text{int}}(q, \dot{q}, t) := \mathcal{R}_{\mathcal{G}}(x(q))J(q), \quad (3.48a)$$

$$b_{\text{int}} := b_{\text{int}}(q, \dot{q}, t) := -(\mathcal{R}_{\mathcal{G}}(x(q))\dot{J}(q) + \dot{\mathcal{R}}_{\mathcal{G}}(x(q))J(q))\dot{q}. \quad (3.48b)$$

Similarly to the case of  $\bar{\mathcal{G}}$ , we conclude that the agent motions produced by the aforementioned constraints correspond to rigid body motions, which are the infinitesimal motions produced by the nullspace of  $\mathcal{R}_{\mathcal{G}}$ .

After giving the rigidity constraints in the cooperative manipulation system, we are now ready to derive the expressions for the interaction forces,  $h$ , in terms of the aforementioned rigidity matrices. We follow the same methodology as in [? ]. Consider first (3.33a) and (3.34b) written in vector form as

$$\bar{B}\dot{\tilde{x}} + \bar{C}_q\tilde{x} + \bar{g} = \bar{\tau} + \begin{bmatrix} -J^\top h \\ h_o \end{bmatrix},$$

with the barred terms as introduced in the beginning of this section. We use Gauss' principle [? ] to derive closed form expressions for  $J^\top h$  and  $h_o$ . Let the *unconstrained* coupled object-robots system be

$$\bar{B}\alpha := \bar{\tau} - \bar{C}_q\tilde{x} - \bar{g},$$

where  $\alpha := \alpha(\bar{x}) : \mathbb{R}^{2n+6} \times \mathbb{T} \rightarrow \mathbb{R}^{n+6}$  is the unconstrained acceleration, i.e., the acceleration the system would have if the agents did not grasp the object.

According to Gauss's principle [? ], the actual accelerations  $\dot{\tilde{x}}$  of the system are the closest ones to  $\alpha(\bar{x})$ , while satisfying the rigidity constraints. More rigorously,  $\dot{\tilde{x}}$  is the solution of the constrained minimization problem

$$\begin{aligned} \min_z \quad & (z - \alpha(\bar{x}))^\top \bar{B}(\bar{x})(z - \alpha(\bar{x})) \\ \text{s.t.} \quad & A(\bar{x}, t)z = b(\bar{x}, t). \end{aligned}$$

The solution to this problem is obtained by using the Karush-Kuhn-Tucker conditions [? ] and has a closed-form expression. It can be shown that it satisfies

$$\bar{B}z = \alpha + A^\top (A\bar{B}^{-1}A^\top)^\dagger (b - A\alpha),$$

where  $^\dagger$  denotes the Moore-Penrose inverse. The aforementioned expression is compliant with the one in [? ],

$$\bar{B}z = \alpha + \bar{B}^{\frac{1}{2}}(A\bar{B}^{-\frac{1}{2}})^\dagger (b - A\alpha),$$

since it holds that  $A^\top (A\bar{B}^{-1}A^\top)^\dagger = \bar{B}^{\frac{1}{2}}(A\bar{B}^{-\frac{1}{2}})^\dagger$ . Indeed, according to Theorem 3.8 of [? ], it holds that  $H^\dagger = H^\top (HH^\top)^\dagger$ , for any  $H \in \mathbb{R}^{x \times y}$ . Then the aforementioned equality is obtained by setting  $H = A\bar{B}^{-\frac{1}{2}}$ .

Therefore, the forces, projected onto the joint-space of the agents, have the form

$$\begin{bmatrix} -J^\top h \\ h_O \end{bmatrix} = A^\top (A\bar{B}^{-1}A^\top)^\dagger (b - A\alpha) \quad (3.49a)$$

$$= \bar{B}^{\frac{1}{2}}(A\bar{B}^{-\frac{1}{2}})^\dagger (b - A\alpha). \quad (3.49b)$$

Consider now that  $h_O = h_m = 0_6 \Leftrightarrow h = h_{int}$ , i.e., the agents produce only internal forces, without inducing object acceleration. Then, the agent dynamics are

$$B\ddot{q} + C_q\dot{q} + g_q = \tau - J^\top h_{int},$$

and the respective *unconstrained* acceleration  $\alpha_{int} := \alpha_{int}(q, \dot{q}) : \mathbb{R}^{2n} \rightarrow \mathbb{R}^n$  is given by

$$B\alpha_{int} := \tau - C_q\dot{q} - g_q.$$

Hence, by proceeding in a similar fashion as for  $\dot{\tilde{x}}$ , we derive an expression for the *internal* forces as

$$-J^\top h_{int} = A_{int}^\top (A_{int}B^{-1}A_{int}^\top)^\dagger (b_{int} - A_{int}\alpha_{int}) \quad (3.50a)$$

$$= B^{\frac{1}{2}}(A_{int}B^{-\frac{1}{2}})^\dagger (b_{int} - A_{int}\alpha_{int}). \quad (3.50b)$$



**Figure 3.14:** Two agents rigidly grasping an object in a 1D scenario.

with  $A_{\text{int}}$ ,  $b_{\text{int}}$  as defined in (3.48).

Therefore, one concludes that when the unconstrained motion of the system does not satisfy the constraints (i.e., when  $b_{\text{int}} \neq A_{\text{int}}\alpha_{\text{int}}$ ), then the actual accelerations of the system are modified in a manner directly proportional to the extent to which these constraints are violated. Moreover, it is evident from the aforementioned expression that the internal forces depend, not only on the relative distances  $p_i - p_j$ , but also on the closed loop dynamics and the inertia of the unconstrained system (see the dependence on  $\alpha_{\text{int}}$  and  $B$ ). Therefore, given a desired force  $h_{O,\text{d}}$  to be applied to the object, an internal force-free distribution to agent forces  $h_{i,\text{d}}$  at the grasping points cannot be independent of the system dynamics. This is clearly illustrated in the following example.

**Example 3.1.** Consider a simplified 1D scenario, with two agents rigidly grasping an object (see Fig. 3.14) subject to the dynamics

$$\begin{aligned} m_i \ddot{p}_i &= u_i - h_i, \quad i \in \{1, 2\} \\ m_o \ddot{p}_o &= h_o = h_1 + h_2, \end{aligned}$$

with the Jacobian matrices being  $J_1 = J_2 = 1$ . The inter-agent constraints here are simply  $\dot{p}_1 = \dot{p}_2 \Rightarrow [1 \ -1] [\ddot{p}_1^\top \ \ddot{p}_2^\top]^\top = 0$ , which gives  $A = [1 \ -1]$ , and  $b = 0$ . In view of (3.50), one can conclude that in this simplified scenario internal forces appear when

$$\frac{u_1}{m_1} \neq \frac{u_2}{m_2},$$

which depends on the masses of the agents.

Note that, as dictated in the previous section, the rigidity matrix  $\mathcal{R}_{\mathcal{G}}$  is not unique, since different choices of  $\gamma_{\mathcal{G}}$  that encode the rigidity constraints can be made. Hence, one might think that different expressions of  $\mathcal{R}_{\mathcal{G}}$  will result in different rigidity constraints of the form (3.48) and hence different interaction and internal forces - which is unreasonable. Nevertheless, note that all different expressions of the rigidity matrix  $\mathcal{R}_{\mathcal{G}}$  have the same nullspace (the coordinated translations and rotations of the framework), and that suffices to prove that this is not the case, as illustrated in Corollary 3.2.

**Corollary 3.2.** Let  $\mathcal{R}_{\mathcal{G},1}$  and  $\mathcal{R}_{\mathcal{G},2}$  such that  $\text{null}(\mathcal{R}_{\mathcal{G},1}) = \text{null}(\mathcal{R}_{\mathcal{G},2})$  and let

$$J^\top h_{\text{int},i} := B^{\frac{1}{2}} \left( \mathcal{R}_{\mathcal{G},i} J B^{-\frac{1}{2}} \right)^\dagger \left( (\mathcal{R}_{\mathcal{G},i} J + \dot{\mathcal{R}}_{\mathcal{G},i} J) \dot{q} + \mathcal{R}_{\mathcal{G},i} J B^{-1} \alpha_{\text{int}} \right),$$

$\forall i \in \{1, 2\}$ , where we have used (3.50) and (3.48). Then  $h_{\text{int},1} = h_{\text{int},2}$ .

*Proof.* The poses and velocities in the terms  $(\mathcal{R}_{\mathcal{G},i}\dot{J} + \dot{\mathcal{R}}_{\mathcal{G},i}J)\dot{q}$  are the actual ones resulting from the coupled system dynamics and hence they respect the rigidity constraints imposed by  $R_{\mathcal{G},i}J\ddot{q} = (\mathcal{R}_{\mathcal{G},i}\dot{J} + \dot{\mathcal{R}}_{\mathcal{G},i}J)\dot{q}$ ,  $\forall i \in \{1, 2\}$ . Therefore, exploiting the positive definiteness of  $B$ , we need to prove that  $(\mathcal{R}_{\mathcal{G},1}JB^{-\frac{1}{2}})^{\dagger}\mathcal{R}_{\mathcal{G},1}J = (\mathcal{R}_{\mathcal{G},2}JB^{-\frac{1}{2}})^{\dagger}\mathcal{R}_{\mathcal{G},2}J$ . In view of Definition G.1 and Proposition G.4 in Appendix G, since  $\mathcal{R}_{\mathcal{G},1}$  and  $\mathcal{R}_{\mathcal{G},2}$  have the same nullspace, they are left equivalent matrices and there exists an invertible matrix  $P$  such that  $\mathcal{R}_{\mathcal{G},1} = P\mathcal{R}_{\mathcal{G},2}$ . Hence, it holds that

$$\begin{aligned} & \left(\mathcal{R}_{\mathcal{G},2}JB^{-\frac{1}{2}}\right)^{\dagger}\mathcal{R}_{\mathcal{G},2}J - \left(\mathcal{R}_{\mathcal{G},1}JB^{-\frac{1}{2}}\right)^{\dagger}\mathcal{R}_{\mathcal{G},1}J = \\ & \left( \left(\mathcal{R}_{\mathcal{G},2}JB^{-\frac{1}{2}}\right)^{\dagger}\mathcal{R}_{\mathcal{G},2}JB^{-\frac{1}{2}} - \left(P\mathcal{R}_{\mathcal{G},2}JB^{-\frac{1}{2}}\right)^{\dagger}P\mathcal{R}_{\mathcal{G},2}JB^{-\frac{1}{2}} \right)B^{\frac{1}{2}}, \end{aligned}$$

which is equal to 0, according to Proposition G.5 of Appendix G and the positive definiteness of  $B$ .  $\square$

One can verify that a similar argument holds for the interaction forces  $\begin{bmatrix} -J^{\top}h \\ h_O \end{bmatrix}$  and  $\mathcal{R}_{\bar{\mathcal{G}}}$  as well.

The aforementioned expressions concern the forces in the joint-space of the robotic agents. The next Corollary gives the expression of the forces in task-space:

**Corollary 3.3.** *The internal forces  $h_{\text{int}}$  are given by*

$$h_{\text{int}} = \mathcal{R}_{\mathcal{G}}^{\top} (\mathcal{R}_{\mathcal{G}}M^{-1}\mathcal{R}_{\mathcal{G}}^{\top})^{\dagger} (\dot{\mathcal{R}}_{\mathcal{G}}v + \mathcal{R}_{\mathcal{G}}\alpha_{\text{int}}^{\text{ts}}), \quad (3.51)$$

where  $\alpha_{\text{int}}^{\text{ts}} := \alpha_{\text{int}}^{\text{ts}}(q, \dot{q}) : \mathbb{S} \times \mathbb{R}^n \rightarrow \mathbb{R}^{6N}$  is the acceleration vector of the task-space unconstrained system

$$M\alpha_{\text{int}}^{\text{ts}} := u - Cv - g,$$

and the forces  $h, h_O$  are given by

$$\begin{bmatrix} -h \\ h_O \end{bmatrix} = -\mathcal{R}_{\bar{\mathcal{G}}}^{\top} (\mathcal{R}_{\bar{\mathcal{G}}}\bar{M}^{-1}\mathcal{R}_{\bar{\mathcal{G}}}^{\top})^{\dagger} (\dot{\mathcal{R}}_{\bar{\mathcal{G}}}\bar{v} + \mathcal{R}_{\bar{\mathcal{G}}}\alpha^{\text{ts}}), \quad (3.52)$$

where  $\alpha^{\text{ts}} := \alpha^{\text{ts}}(\bar{x}) : \mathbb{X} \rightarrow \mathbb{R}^{6N+6}$  is the acceleration vector of the task-space unconstrained system

$$\bar{M}\alpha^{\text{ts}} := \bar{u} - \bar{C}\bar{v} - \bar{g}.$$

*Proof.* By using the expressions of  $M(q), C(q, \dot{q})v, g(q)$  from (3.33) to expand (3.51), one can conclude that  $J^\top h_{\text{int}}$ , with  $h_{\text{int}}$  given by (3.51) and in view of (3.48), is equal to (3.50a). Similarly, by expanding the dynamic terms of (3.52) and using (3.47), one can verify that the vector  $[-(J^\top h)^\top, h_O^\top]^\top$ , with  $[-h^\top, h_O^\top]^\top$  given by (3.52), is equal to (3.49).  $\square$

We also show later that the derived forces (3.52) are consistent with the relation  $h_O = G(x)h$  (see (3.38)).

We now give a more explicit expression for  $h$ . One can verify that, by appropriately arranging the rows of  $\gamma_G$ , it holds that

$$\mathcal{R}_{\bar{\mathcal{G}}} := \begin{bmatrix} \mathcal{R}_G & 0_{\frac{7N(N-1)}{2} \times 6} \\ \mathcal{R}_{O_1} & \mathcal{R}_{O_2} \end{bmatrix} \in \mathbb{R}^{\frac{7N(\bar{N}-1)}{2} \times (6N+6)}, \quad (3.53)$$

where  $\mathcal{R}_{O_1} \in \mathbb{R}^{7N \times 6N}$  and  $\mathcal{R}_{O_2} \in \mathbb{R}^{7N \times 6}$  are the matrices

$$\mathcal{R}_{O_1} := \begin{bmatrix} (p_1 - p_O)^\top & 0_{1 \times 3} & \dots & 0_{1 \times 3} & 0_{1 \times 3} \\ \vdots & \vdots & \dots & \ddots & \vdots \\ 0_{1 \times 3} & 0_{1 \times 3} & \dots & (p_N - p_O)^\top & 0_{1 \times 3} \\ \frac{\partial \gamma_{e_{1O}, b}}{\partial p_1} & \frac{\partial \gamma_{e_{1O}, b}}{\partial R_1} & \dots & 0_{3 \times 3} & 0_{3 \times 3} \\ \frac{\partial \gamma_{e_{O1}, b}}{\partial p_1} & \frac{\partial \gamma_{e_{O1}, b}}{\partial R_1} & \dots & 0_{3 \times 3} & 0_{3 \times 3} \\ \vdots & \dots & \ddots & \vdots & \vdots \\ 0_{3 \times 3} & 0_{3 \times 3} & \dots & \frac{\partial \gamma_{e_{NO}, b}}{\partial p_N} & \frac{\partial \gamma_{e_{NO}, b}}{\partial R_N} \\ 0_{3 \times 3} & 0_{3 \times 3} & \dots & \frac{\partial \gamma_{e_{ON}, b}}{\partial p_N} & \frac{\partial \gamma_{e_{ON}, b}}{\partial R_N} \end{bmatrix}$$

$$\mathcal{R}_{O_2} := \begin{bmatrix} -(p_1 - p_O)^\top & 0_{1 \times 3} \\ \vdots & \vdots \\ -(p_N - p_O)^\top & 0_{1 \times 3} \\ \frac{\partial \gamma_{e_{1O}, b}}{\partial p_O} & \frac{\partial \gamma_{e_{1O}, b}}{\partial R_O} \\ \frac{\partial \gamma_{e_{O1}, b}}{\partial p_O} & \frac{\partial \gamma_{e_{O1}, b}}{\partial R_O} \\ \vdots & \vdots \\ \frac{\partial \gamma_{e_{NO}, b}}{\partial p_O} & \frac{\partial \gamma_{e_{NO}, b}}{\partial R_O} \\ \frac{\partial \gamma_{e_{ON}, b}}{\partial p_O} & \frac{\partial \gamma_{e_{ON}, b}}{\partial R_O} \end{bmatrix},$$

where  $e_{io} := (i, \bar{N})$ ,  $e_{oi} := (\bar{N}, i) \in \bar{\mathcal{E}}$  corresponding to the edge among the  $i$ th agent and the object,  $\forall i \in \mathcal{N}$ . Therefore, (3.52) can be written as

$$h = [\mathcal{R}_{\bar{G}}^\top \quad \mathcal{R}_{o_1}^\top] (\mathcal{R}_{\bar{G}} \bar{M}^{-1} \mathcal{R}_{\bar{G}}^\top)^\dagger (\dot{\mathcal{R}}_{\bar{G}} \bar{v} + \mathcal{R}_{\bar{G}} \alpha^{\text{ts}}) \quad (3.54a)$$

$$h_o = -[0 \quad \mathcal{R}_{o_2}^\top] (\mathcal{R}_{\bar{G}} \bar{M}^{-1} \mathcal{R}_{\bar{G}}^\top)^\dagger (\dot{\mathcal{R}}_{\bar{G}} \bar{v} + \mathcal{R}_{\bar{G}} \alpha^{\text{ts}}). \quad (3.54b)$$

Note also that

$$G \mathcal{R}_{o_1}^\top = -\mathcal{R}_{o_2}^\top, \quad (3.55)$$

which will be used in the analysis to follow.

Another expression for the interaction forces  $h$  can be obtained by differentiating (3.37), which, after using (3.33) and (3.34) yields after straightforward manipulations (similarly to (2.89))

$$h = (M^{-1} + G^\top M_o^{-1} G)^{-1} \left( M^{-1}(u - g - Cv) - \dot{G}^\top v_o + G^\top M_o^{-1}(C_o v_o + g_o) \right). \quad (3.56)$$

In order to show the consistency of our results, we prove next that (3.54a) and (3.56) are identical.

**Corollary 3.4.** *Let  $h_a$  be given by (3.54a) and  $h_b$  be given by (3.56). Then  $h_a = h_b$ .*

*Proof.* By using (3.53), (3.54a) is expanded as

$$\begin{aligned} h_a &= \mathcal{R}_{o_1}^\top (\mathcal{R}_{o_1} M^{-1} \mathcal{R}_{o_1}^\top + \mathcal{R}_{o_2} M_o^{-1} \mathcal{R}_{o_2}^\top)^\dagger (\dot{\mathcal{R}}_{o_1} v + \dot{\mathcal{R}}_{o_2} v_o \\ &\quad + \mathcal{R}_{o_1} M^{-1}(u - g - Cv) - \mathcal{R}_{o_2} M_o^{-1}(g_o + C_o v_o)) \end{aligned}$$

which, after using (3.55) and  $v = G^\top v_o$ , becomes

$$\begin{aligned} h_a &= \mathcal{R}_{o_1}^\top (\mathcal{R}_{o_1} (M^{-1} + G^\top M_o^{-1} G) \mathcal{R}_{o_1}^\top)^\dagger (\dot{\mathcal{R}}_{o_1} v - \dot{\mathcal{R}}_{o_1} G^\top v_o - \mathcal{R}_{o_1} \dot{G}^\top v_o \\ &\quad + \mathcal{R}_{o_2} M^{-1}(u - g - Cv) + \mathcal{R}_{o_1} G^\top M_o^{-1}(g_o + C_o v_o)) \\ &= \mathcal{R}_{o_1}^\top (\mathcal{R}_{o_1} (M^{-1} + G^\top M_o^{-1} G) \mathcal{R}_{o_1}^\top)^\dagger \mathcal{R}_{o_1} \left( -\dot{G}^\top v_o + M^{-1}(u - g - Cv) \right. \\ &\quad \left. + M_o^{-1}(g_o + C_o v_o) \right). \end{aligned}$$

Denote now for convenience  $M_G := M^{-1} + G^\top M_o^{-1} G$ . According to Theorem 3.8 of [?], it holds that  $\mathcal{R}_{o_1}^\top (\mathcal{R}_{o_1} M_G \mathcal{R}_{o_1}^\top)^\dagger \mathcal{R}_{o_1} = M_G^{-\frac{1}{2}} (\mathcal{R}_{o_1} M_G^{\frac{1}{2}})^\dagger \mathcal{R}_{o_1}$ . Next, note that  $\mathcal{R}_{o_1}$  has linearly independent columns and hence

$$(\mathcal{R}_{o_1} M_G^{\frac{1}{2}})^\dagger = (M_G^{\frac{1}{2}} \mathcal{R}_{o_1}^\top \mathcal{R}_{o_1} M_G^{\frac{1}{2}})^{-1} M_G^{\frac{1}{2}} \mathcal{R}_{o_1}^\top = M_G^{-\frac{1}{2}} (\mathcal{R}_{o_1}^\top \mathcal{R}_{o_1})^{-1} \mathcal{R}_{o_1}^\top,$$

since  $M_G$  is symmetric and positive definite. Therefore, we conclude that  $M_G^{-\frac{1}{2}} \left( \mathcal{R}_{O_1} M_G^{\frac{1}{2}} \right)^\dagger \mathcal{R}_{O_1} = M_G^{-1}$ , and hence  $h_a = h_b$ .  $\square$

**Remark 3.6.** According to Theorem 3.8 of [?], the task-space internal forces can also be written as

$$h_{\text{int}} = M^{\frac{1}{2}} \left( \mathcal{R}_G M^{-\frac{1}{2}} \right)^\dagger \left( \dot{\mathcal{R}}_G v + \mathcal{R}_G \alpha_{\text{int}}^{\text{ts}} \right), \quad (3.57)$$

which is compliant with the result in [?].

One concludes, therefore, that in order to obtain internal force-free trajectories, the term  $\dot{\mathcal{R}}_G v + \mathcal{R}_G \alpha_{\text{int}}^{\text{ts}} = \dot{\mathcal{R}}_G v + \mathcal{R}_G M^{-1}(u - C\dot{v} - g)$  must belong to the nullspace of  $M^{\frac{1}{2}} \left( \mathcal{R}_G M^{-\frac{1}{2}} \right)^\dagger$ . The latter, however, is identical to the nullspace of  $\mathcal{R}_G$ , since it holds that  $\text{null}(\mathcal{R}_G M^{-1/2})^\dagger = \text{null}(M^{-\frac{1}{2}} \mathcal{R}_G^\top)$  and  $M$  is positive definite. This result is summarized in the following corollary.

**Corollary 3.5.** *The cooperative manipulation system is free of internal forces, i.e.,  $h_{\text{int}} = 0$ , if and only if*

$$\dot{\mathcal{R}}_G v + \mathcal{R}_G M^{-1}(u - C\dot{v} - g) \in \text{null}(\mathcal{R}_G^\top)$$

In cooperative manipulation schemes, the most energy-efficient way of transporting an object is to exploit the full potential of the cooperating robotic agents, i.e., each agent does not exert less effort at the expense of other agents, which might then potentially exert more effort than necessary. For instance, consider a rigid cooperative manipulation scheme, with only one agent (a leader) working towards bringing the object to a desired location, whereas the other agents have zero inputs. Since the grasps are rigid, if the leader has sufficient power, it will achieve the task by “dragging” the rest of the agents, compensating for their dynamics, and creating non-negligible internal forces. In such cases, when the cooperative manipulation system is rigid (i.e., the grasps are considered to be rigid), the optimal strategy of transporting an object is achieved by regulating the internal forces to zero. Therefore, from a control perspective, the goal of a rigid cooperative manipulation system is to design a control protocol that achieves a desired cooperative manipulation task, while guaranteeing that the internal forces remain zero.

### Cooperative Manipulation via Internal Force Regulation

We derive here a new relation between the interaction and internal forces  $h$  and  $h_{\text{int}}$ , respectively. Moreover, we derive novel sufficient and necessary

conditions on the agent force distribution for the provable regulation of the internal forces to zero, according to (3.51), and we show its application in a standard inverse-dynamics control law that guarantees trajectory tracking of the object's center of mass. This is based on the following main theorem, which links the complete agent graph rigidity matrix  $\mathcal{R}_G$  to the grasp matrix  $G$ :

**Theorem 3.2.** *Let  $N$  robotic agents, with configuration  $x = (p, R) \in \mathbb{SE}(3)^N$ , rigidly grasping an object and associated with a grasp matrix  $G(x)$ , as in (3.36). Let also the agents be modeled by a framework on the complete graph  $(\mathcal{K}_N, p_{\mathcal{K}_N}, R_{\mathcal{K}_N}) = (\mathcal{K}_N, p, R)$  in  $\mathbb{SE}(3)$ , which is associated with a rigidity matrix  $\mathcal{R}_{\mathcal{K}_N}$ . Let also  $x$  be such that  $\text{rank}(\mathcal{R}_{\mathcal{K}_N}(x)) = \max_{y \in \mathbb{SE}(3)^N} \{\text{rank}(\mathcal{R}_{\mathcal{K}_N}(y))\}$ . Then it holds that*

$$\text{null}(G(x)) = \text{range}(\mathcal{R}_{\mathcal{K}_N}(x)^\top).$$

*Proof.* Since  $\mathcal{R}_{\mathcal{K}_N}$  is associated to the complete graph and  $\text{rank}(\mathcal{R}_{\mathcal{K}_N}(x)) = \max_{y \in \mathbb{SE}(3)^N} \{\text{rank}(\mathcal{R}_{\mathcal{K}_N}(y))\}$ , the framework  $(\mathcal{K}_N, p, R)$  is infinitesimally rigid. Hence, the nullspace of  $\mathcal{R}_{\mathcal{K}_N}$  consists only of the infinitesimal motions of the framework, i.e., coordinated translations and rotations, as defined in Proposition 3.1. In particular, in view of (3.46), Proposition 3.2, and (3.45), one concludes that  $\text{null}(\mathcal{R}_{\mathcal{K}_N})$  is the linear span of  $1_N \otimes \begin{bmatrix} I_3 \\ 0_{3 \times 3} \end{bmatrix}$  and the vector space  $[\chi_1^\top, \dots, \chi_N^\top]^\top \in \mathbb{SE}(3)^N$ , with  $\chi_i := [\chi_{i,p}^\top, \chi_{i,R}^\top]^\top \in \mathbb{SE}(3)$ , satisfying

$$\chi_{i,p} - \chi_{j,p} = -S(p_i - p_j)\chi_{i,R} \quad (3.58a)$$

$$\chi_{i,R} = \chi_{j,R}, \quad (3.58b)$$

where  $p_i := p_{\mathcal{K}_N}(i)$ ,  $p_j := p_{\mathcal{K}_N}(j)$ ,  $\forall i, j \in \mathcal{N}$ , with  $i \neq j$ . In view of (3.37), one obtains  $v = G^\top v_o$ , where

$$G^\top = \begin{bmatrix} I_3 & -S(p_{1o}) \\ 0 & I_3 \\ \vdots & \vdots \\ I_3 & -S(p_{No}) \\ 0 & I_3 \end{bmatrix}.$$

The first 3 columns of  $G^\top$  form the space  $1_N \otimes \begin{bmatrix} I_3 \\ 0_{3 \times 3} \end{bmatrix}$  whereas the last 3 columns  $G^\top$  span the aforementioned rotation vector space. Indeed, for any

$\dot{p}_o, \omega_o \in \mathbb{R}^6$  the range of these columns is

$$\begin{bmatrix} -S(p_{1o})\dot{p}_o \\ \omega_o \\ \vdots \\ -S(p_{No})\dot{p}_o \\ \omega_o \end{bmatrix},$$

for which it is straightforward to verify that (3.58) holds. Hence,  $\text{null}(\mathcal{R}_{\mathcal{K}_N}) = \text{range}(G^\top)$  and by using the rank-nullity theorem the result follows.  $\square$

Hence, since the internal forces belong to  $\text{null}(G)$ , one concludes that they are comprised of all the vectors  $z$  for which there exists a  $y$  such that  $z = \mathcal{R}_G^\top y$ . This can also be verified by inspecting (3.57); one can prove that  $\text{range}(M^{\frac{1}{2}}(\mathcal{R}_G M^{-\frac{1}{2}})^\dagger) = \text{range}(\mathcal{R}_G^\top)$ . The aforementioned result provides significant insight regarding the control of the motion of the coupled cooperative manipulation system. In particular, by using (3.57) and Theorem 3.2, we provide next new conditions on the agent force distribution for provable avoidance of internal forces. We first derive a novel relation between the agent forces  $h$  and the internal forces  $h_{\text{int}}$ .

In many works in the related literature, the force  $h$  is decomposed as

$$h = h_m + h_{\text{int}} = G^* G h + (I - G^* G)h, \quad (3.59)$$

where  $G^*$  is a right inverse of  $G$ . The term  $G^* G h$  is a projection of  $h$  on the range space of  $G^\top$ , whereas the term  $(I - G^* G)h$  is a projection of  $h$  on the null space of  $G$ . A common choice is the Moore-Penrose inverse  $G^* = G^\dagger$ , which equals to  $G^\top (GG^\top)^{-1}$ . This specific choice yields the vector  $G^* G h = G^\dagger G h \in \text{range}(G^\top)$  that is closest to  $h$ , i.e.,  $\|h - G^\dagger G h\| \leq \|h - y\|$ ,  $\forall y \in \text{range}(G^\top)$ . However, as the next theorem states, if the second term of (3.59) must equal  $h_{\text{int}}$ , as this is defined in (3.57),  $G^*$  must actually be the weighted pseudo inverse  $MG^\top(GMG^\top)^{-1}$ .

**Theorem 3.3.** Consider  $N$  robotic agents rigidly grasping an object with coupled dynamics (3.39). Let  $h \in \mathbb{R}^{6N}$  be the stacked vector of agent forces exerted at the grasping points. Then the agent forces  $h$  and the internal forces  $h_{\text{int}}$  are related as:

$$h_{\text{int}} = (I_{6N} - MG^\top(GMG^\top)^{-1}G)h.$$

In order to prove Theorem 3.3, we first need the following preliminary result.

**Proposition 3.3.** Consider the grasp and rigidity matrices  $G, \mathcal{R}_G$ , respectively, of the cooperative manipulation system. Then it holds that

$$MG^\top (GMG^\top)^{-1} G + M^{\frac{1}{2}} \left( \mathcal{R}_G M^{-\frac{1}{2}} \right)^\dagger \mathcal{R}_G M^{-1} = I.$$

*Proof.* Let  $A_f := \mathcal{R}_G M^{-\frac{1}{2}}$  and  $B_f := GM^{\frac{1}{2}}$ . Then  $\text{range}(A_f^\top) = \text{null}(B_f)$ . Indeed, according to Theorem 3.2, it holds that if  $z = \mathcal{R}_G^\top y$ , for some  $y \in \mathbb{R}^6$ , then  $Gz = 0$ . By multiplying by  $M^{-\frac{1}{2}}$ , we obtain  $M^{-\frac{1}{2}}z = M^{-\frac{1}{2}}\mathcal{R}_G^\top y$ , which implies that  $\hat{z} := M^{-\frac{1}{2}}z \in \text{range}((\mathcal{R}_G M^{\frac{1}{2}})^\top)$ . It also holds that  $B_f \hat{z} = GM^{\frac{1}{2}}\hat{z} = Gz = 0$ , and hence  $\hat{z} \in \text{null}(B_f)$ . Therefore, in view of Proposition G.6 of Appendix G, Theorem 3.8 of [?], according to which  $G^\top (GMG^\top)^\dagger = M^{-\frac{1}{2}}(GM^{\frac{1}{2}})^\dagger$ , and the fact that  $GMG^\top$  is invertible, we conclude that

$$\begin{aligned} \left( GM^{\frac{1}{2}} \right)^\dagger GM^{\frac{1}{2}} + \left( \mathcal{R}_G M^{-\frac{1}{2}} \right)^\dagger \mathcal{R}_G M^{-\frac{1}{2}} &= I \Leftrightarrow \\ M^{\frac{1}{2}} G^\top (GMG^\top)^\dagger GM^{\frac{1}{2}} + \left( \mathcal{R}_G M^{-\frac{1}{2}} \right)^\dagger \mathcal{R}_G M^{-\frac{1}{2}} &= I, \end{aligned}$$

and by left and right multiplication by  $M^{\frac{1}{2}}$  and  $M^{-\frac{1}{2}}$ , respectively, the result follows.  $\square$

We are now ready to prove Theorem 3.3.

*Proof.* We first show that

$$(I - MG^\top (GMG^\top)^{-1} G) \left( M^{-1} + G^\top M_O^{-1} G \right)^{-1} = M^{\frac{1}{2}} \left( \mathcal{R}_G M^{-\frac{1}{2}} \right)^\dagger \mathcal{R}_G.$$

Indeed, since  $(M^{-1} + G^\top M_O^{-1} G)^{-1}$  has full rank, it suffices to show that

$$(I - MG^\top (GMG^\top)^{-1} G) = M^{\frac{1}{2}} \left( \mathcal{R}_G M^{-\frac{1}{2}} \right)^\dagger \mathcal{R}_G \left( M^{-1} + G^\top M_O^{-1} G \right),$$

which can be concluded from the fact that  $\mathcal{R}_G G^\top = 0$  (due to Theorem 3.2) and Proposition 3.3. Therefore, in view of (3.56), it holds that

$$\begin{aligned} (I - MG^\top (GMG^\top)^{-1} G) h &= (I - MG^\top (GMG^\top)^{-1} G)(M^{-1} + \\ &G^\top M_O^{-1} G)^{-1} \left( -\dot{G}^\top v_O + M^{-1}(u - g - Cv) + G^\top M_O^{-1}(C_O v_O + g_O) \right) = \\ &M^{\frac{1}{2}} \left( \mathcal{R}_G M^{-\frac{1}{2}} \right)^\dagger \mathcal{R}_G (M^{-1}(u - g - Cv) + G^\top M_O^{-1}(C_O v_O + g_O) - \dot{G}^\top v_O), \end{aligned}$$

which, in view of the facts that  $\mathcal{R}_G G^\top = 0$ , and hence by differentiation  $-\mathcal{R}_G G^\top = \dot{\mathcal{R}}_G G^\top$ , as well as  $G^\top v_o = v$ , becomes

$$M^{\frac{1}{2}} \left( \mathcal{R}_G M^{-\frac{1}{2}} \right)^\dagger (\dot{\mathcal{R}}_G v + \mathcal{R}_G M^{-1} (u - g - Cv)) = h_{\text{int}}.$$

□

Based on Theorem 3.3, we provide in the next theorem new results on the optimal distribution of a force to the robotic agents, i.e., a distribution that provably yields zero internal forces.

**Theorem 3.4.** *Consider  $N$  robotic agents rigidly grasping an object, with coupled dynamics (3.39). Let a desired force to be applied to the object  $h_{o,d} \in \mathbb{R}^6$ , which is distributed to the agents' desired forces as  $h_d = G^* h_{o,d}$ , and where  $G^*$  is a right inverse of  $G$ , i.e.,  $GG^* = I_6$ . Then there are no internal forces, i.e.,  $h_{\text{int}} = 0$ , if and only if*

$$G^* = MG^\top (GMG^\top)^{-1}.$$

*Proof.* According to Theorem 3.3, the derivation of  $h_d$  that yields zero internal forces can be formulated as a quadratic minimization problem:

$$\begin{aligned} \text{QP : } \min_{h_d} \quad & \|h_{\text{int}}\|^2 = h_d^\top H h_d \\ \text{s.t.} \quad & Gh_d = h_{o,d}, \end{aligned}$$

where  $H := (I_{6N} - MG^\top (GMG^\top)^{-1}G)^\top (I_{6N} - MG^\top (GMG^\top)^{-1}G)$ . Firstly, note that the choice  $G^* = MG^\top (GMG^\top)^{-1}h_{o,d}$  is a minimizer of QP, since  $GG^* = I_6$ , and  $HG^*h_{o,d} = 0$ , and therefore sufficiency is proved.

In order to prove necessity, we prove next that  $G^*$  is a strict minimizer, i.e., there is no other right inverse of  $G$  that is a solution of QP. Note first that  $G \in \mathbb{R}^{6 \times 6N}$  has full row rank, which implies that the dimension of its nullspace is  $6N - 6$ . Let  $Z := [z_1, \dots, z_{6N-6}] \in \mathbb{R}^{6 \times (6N-6)}$  be the matrix formed by the vectors  $z_1, \dots, z_{6N-6} \in \mathbb{R}^{6N}$  that span the nullspace of  $G$ . It follows that  $\text{rank}(Z) = 6N - 6$  and  $GZ = 0$ . Let now the matrix  $H' := Z^\top H Z \in \mathbb{R}^{(6N-6) \times (6N-6)}$ . Since  $GZ = 0 \Rightarrow Z^\top G^\top = 0$ , it follows that  $H' = Z^\top Z$ . Hence,  $\text{rank}(H') = \text{rank}(Z) = 6N - 6$ , which implies that  $H'$  is positive definite. Therefore, according to [?, Theorem 1.1], QP has a strong minimizer. □

The aforementioned theorem provides novel necessary and sufficient conditions for provable minimization of internal forces in a cooperative manipulation scheme. As discussed before, this is crucial for achieving energy-optimal cooperative manipulation, where the agents do not have to

“waste” control input and hence energy resources that do not contribute to object motion. Related works that focus on deriving internal force-free distributions  $G^*$ , e.g., [? ? ? ?], are solely based on the inter-agent distances, neglecting the actual dynamics of the agents and the object. The expression (3.51), however, gives new insight on the topic and suggests that the dynamic terms of the system play a significant role in the arising internal forces, as also indicated by Corollary 3.5. This is further exploited by Theorem 3.4 to derive a right-inverse that depends on the inertia of the system. Note also that, as explained in [?] and illustrated in Example 3.1, the internal forces depend on the acceleration of the robotic agents and hence the incorporation of  $M$  in  $G^*$  is something to be expected.

The forces  $h$ , however, are not the actual control input of the robotic agents, and hence we cannot simply set  $h = h_d = MG^\top(GMG^\top)^{-1}Gh_{o,a}$  for a given  $h_{o,d}$ . Therefore, we design next a standard inverse-dynamics control algorithm controller that guarantees tracking of a desired trajectory by the object center of mass while provably achieving regulation of the internal forces to zero.

Let a desired position trajectory for the object center of mass be  $p_d : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^3$ , and  $e_p := p_o - p_d$ . Let also a desired object orientation be expressed in terms of a desired rotation matrix  $R_d : \mathbb{R}_{\geq 0} \rightarrow \mathbb{SO}(3)$ , with  $\dot{R}_d = S(\omega_d)R_d$ , where  $\omega_d : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^3$  is the desired angular velocity. Then an orientation error metric that was also used in the previous formation-control section is

$$e_o := \frac{1}{2}\text{tr}(I_3 - R_d^\top R_o) \in [0, 2], \quad (3.60)$$

which, after differentiation and by using (3.34a) becomes (see also (3.6b))

$$\dot{e}_o = \frac{1}{2}e_R^\top R_o^\top (\omega_o - \omega_d), \quad (3.61)$$

where  $e_R := S^{-1}(R_d^\top R_o - R_o^\top R_d) \in \mathbb{R}^3$ . It holds that

$$e_R = 0 \Leftrightarrow \begin{cases} e_o = 0 \Leftrightarrow \text{tr}(R_d^\top R_o) = 3 \Leftrightarrow R_o = R_d \\ e_o = 2 \Leftrightarrow \text{tr}(R_d^\top R_o) = -1 \Leftrightarrow R_o \neq R_d \end{cases}.$$

The second case represents an undesired equilibrium, where the desired and the actual orientation differ by 180 degrees. This issue is caused by topological obstructions on  $\mathbb{SO}(3)$  and it has been proven that no continuous controller can achieve *global* stabilization [?]. The following control design guarantees that  $e_o(t) < 2$ ,  $\forall t \in \mathbb{R}_{\geq 0}$ , from all initial conditions satisfying  $e_o(0) < 2$ .

The next corollary shows that a standard inverse-dynamics control protocol guarantees convergence of  $p(t) - p_d(t)$ ,  $e_o(t)$  to zero while avoiding

internal forces, provided that the right inverse  $G^* = MG^\top(GMG^\top)^{-1}$  is used.

**Corollary 3.6.** Consider  $N$  robotic agents rigidly grasping an object with coupled dynamics (3.39). Let a desired trajectory be defined by  $p_d : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^3$ ,  $R_d : \mathbb{R}_{\geq 0} \rightarrow \mathbb{SO}(3)$ ,  $\dot{p}_d, \omega_d \in \mathbb{R}^3$ , and assume that  $e_o(0) < 2$ , with  $e_o$  as defined in (3.60). Consider the inverse-dynamics control law

$$u = g + \left( CG^\top + M\dot{G}^\top \right) v_o + G^* (g_o + C_o v_o) \\ + (MG^\top + G^* M_o) (\dot{v}_d - K_d e_v - K_p e_x), \quad (3.62)$$

where  $e_v := v_o - v_d$ ,  $v_d := [\dot{p}_d^\top, \omega_d^\top]^\top \in \mathbb{R}^6$ ,  $e_x := [e_p^\top, \frac{1}{2(2-e_o)^2} e_R^\top R_o^\top]^\top$ ,  $K_p := \text{diag}\{K_{p_1}, k_{p_2} I_3\}$ , where  $K_{p_1} \in \mathbb{R}^{3 \times 3}$ ,  $K_d \in \mathbb{R}^{6 \times 6}$  are positive definite matrices, and  $k_{p_2} \in \mathbb{R}_{>0}$  is a positive constant. Then the solution of the closed-loop coupled system satisfies the following:

1.  $e_o(t) < 2, \forall t \in \mathbb{R}_{\geq 0}$
2.  $\lim_{t \rightarrow \infty} (p_o(t) - p_d(t)) = 0, \lim_{t \rightarrow \infty} R_d(t)^\top R_o(t) = I_3$
3. There are no internal forces, i.e.,  $h_{\text{int}}(t) = 0, \forall t \in \mathbb{R}_{\geq 0}$ , if and only if

$$G^* = MG^\top(GMG^\top)^{-1}.$$

*Proof.* 1. By substituting (3.62) in (3.39) and using  $GG^* = I_6$ , we obtain, in view of (2.15a)-(2.15c) and the positive definiteness of  $\widetilde{M}$ :

$$\widetilde{M}(\dot{e}_v + K_d e_v + K_p e_x) = 0_6 \Rightarrow \dot{e}_v = -K_d e_v - K_p e_x. \quad (3.63)$$

Consider now the function

$$V := \frac{1}{2} e_p^\top K_{p_1} e_p + \frac{k_{p_2}}{2 - e_o} + \frac{1}{2} e_v^\top e_v,$$

for which it holds  $V(0) < \infty$ , since  $e_o(0) < 2$ . By differentiating  $V$ , and using (3.61) and (3.63), one obtains

$$\dot{V} = \left[ e_p^\top K_{p_1} - \frac{k_{p_2}}{2(2-e_o)^2} e_R^\top R_o^\top \right] e_v - e_v^\top (K_d e_v + K_p e_x) = -e_v^\top K_d e_v \leq 0$$

Hence, it holds that  $V(t) \leq V(0) < \infty$ , which implies that  $\frac{k_{p_2}}{2 - e_o(t)}$  is bounded and consequently  $e_o(t) < 2$ .

2. Since  $V(t) \leq V(0) < \infty$ , the errors  $e_p, e_v$  are bounded, which, given the boundedness of the desired trajectories  $p_d, R_d$  and their derivatives,

implies the boundedness of the control law  $u$ . Hence, it can be proved that  $\ddot{V}$  is bounded which implies the uniform continuity of  $\dot{V}$ . Therefore, according to Barbalat's lemma (Lemma A.1 of Appendix A), we deduce that  $\lim_{t \rightarrow \infty} \dot{V}(t) = 0 \Rightarrow \lim_{t \rightarrow \infty} e_v(t) = 0$ . Since  $e_x(t)$  is also bounded, it can be proved by using the same arguments that  $\lim_{t \rightarrow \infty} \dot{e}_v(t) = 0$  and hence (3.63) implies that  $\lim_{t \rightarrow \infty} e_x(t) = 0$ .

3. Let the desired object force be

$$h_{O,d} = C_O v_O + g_O + M_O \alpha_d, \quad (3.64)$$

where  $\alpha_d := \dot{v}_d - K_d e_v - K_p e_x$ , which implies that (3.62) becomes

$$u = g + (CG^\top + M\dot{G}^\top)v_O + MG^\top\alpha_d + G^*h_{O,d}$$

In view of Theorem 3.4, it suffices to prove  $h = h_d = G^*h_{O,d}$ . By substituting (3.62) in the expression (3.56) and canceling terms, we obtain

$$h = (M^{-1} + G^\top M_O^{-1}G)^{-1}(M^{-1}G^*h_{O,d} + G^\top\alpha_d + G^\top M_O^{-1}(C_O v_O + g_O)).$$

Next, we add and subtract the term  $G^\top M_O GG^*h_{O,d}$  to obtain

$$\begin{aligned} h &= (M^{-1} + G^\top M_O^{-1}G)^{-1}(M^{-1} + G^\top M_O^{-1}G)G^*h_{O,d} + \\ &\quad (M^{-1} + G^\top M_O^{-1}G)^{-1}(G^\top M_O^{-1}(M_O \alpha_d + C_O v_O + g_O - G^\top M_O h_{O,d})), \end{aligned}$$

which, in view of (3.64), becomes  $h = G^*h_{O,d}$ . □

**Remark 3.7 (Uncertain dynamics and force sensing).** Note that the employed inverse dynamics controller requires knowledge of the agent and object dynamics. In case of dynamic parameter uncertainty, standard adaptive control schemes that attempt to estimate potential uncertainties in the model (see, e.g., [? ] or the previous chapter) would intrinsically create internal forces, since the dynamics of the system would not be accurately compensated. The same holds for schemes that employ force/torque sensors that provide the respective measurements at the grasp points (e.g., [? ? ]) in periodic time instants. Since the interaction forces depend explicitly on the control input, such measurements will unavoidably correspond to the interaction forces of the previous time instants due to causality reasons, creating thus small disturbances in the dynamic model.

**Remark 3.8 (Load-sharing).** Finally, note that  $G^* = MG^\top(GMG^\top)^{-1}$  induces an *implicit* and natural load-sharing scheme via the incorporation of  $M$ . More specifically, note that the force distribution to the robotic agents via  $G^*h_{O,d}$  yields for each agent  $M_i J_{O_i} (\sum_{i \in \mathcal{N}} J_{O_i}^\top M_i J_{O_i})^{-1}$ ,  $\forall i \in \mathcal{N}$ . Hence,

larger values of  $M_i$  will produce larger inputs for agent  $i$ , implying that agents with larger inertia characteristics will take on a larger share of the object load. Note that this is also a *desired* load-sharing scheme, since larger dynamic values usually imply more powerful robotic agents.

In case it is required to achieve a *desired* internal force  $h_{\text{int,d}}$ , one can add in (3.62) a term of the form described in the next corollary.

**Corollary 3.7.** *Let  $h_{\text{int,d}} \in \text{null}(G)$  be a desired internal force to be achieved. Then adding the extra term  $u_{\text{int,d}} := (I_{6N} - MG^\top(GMG^\top)^{-1})h_{\text{int,d}}$  in (3.62) achieves  $h_{\text{int}} = h_{\text{int,d}}$ .*

*Proof.* Since  $h_{\text{int,d}} \in \text{null}(G) = \text{range}(\mathcal{R}_G^\top)$ , it holds that  $M^{-\frac{1}{2}}h_{\text{int,d}} \in \text{range}(M^{-\frac{1}{2}}\mathcal{R}_G^\top) = \text{range}(\mathcal{R}_G M^{-\frac{1}{2}})^\dagger$ . Therefore, it holds that

$$(\mathcal{R}_G M^{-\frac{1}{2}})^\dagger \mathcal{R}_G M^{-1} h_{\text{int,d}} = (\mathcal{R}_G M^{-\frac{1}{2}})^\dagger \mathcal{R}_G M^{-\frac{1}{2}} (M^{-\frac{1}{2}}h_{\text{int,d}}) = M^{-\frac{1}{2}}h_{\text{int,d}}. \quad (3.65)$$

Hence, (3.57) yields the resulting internal forces

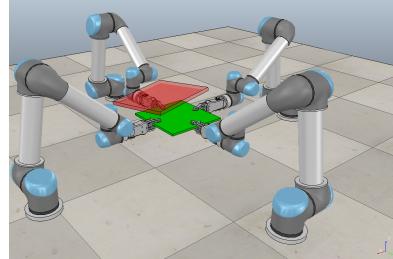
$$\begin{aligned} h_{\text{int}} &= M^{\frac{1}{2}}(\mathcal{R}_G M^{-\frac{1}{2}})^\dagger \mathcal{R}_G M^{-1} (I - MG^\top(GMG^\top)^{-1})h_{\text{int,d}} \\ &= M^{\frac{1}{2}}(\mathcal{R}_G M^{-\frac{1}{2}})^\dagger \mathcal{R}_G M^{-1} h_{\text{int,d}} \\ &= M^{\frac{1}{2}}M^{-\frac{1}{2}}h_{\text{int,d}} = h_{\text{int,d}}, \end{aligned}$$

where we have used (3.65) and the fact that  $\mathcal{R}_G G^\top = 0$  from Theorem 3.2.  $\square$

Finally, in view of Theorem 3.2, one can also verify the consistency of the expressions of  $h, h_O$  in (3.52) with the grasp-matrix rigidity constraint  $h_O = G(x)h$  (see (3.38)). Indeed, Theorem 3.2 dictates that  $G\mathcal{R}_G^\top = 0$ . Therefore, by combining (3.55) and (3.54) we conclude that  $h_O = G(x)h$ . Note also that, in view of Corollary 3.2, the result is still valid if different  $\gamma_{\bar{G}}$  and  $\mathcal{R}_{\bar{G}}$  are chosen.

### 3.3.4 Simulation Results

This section provides simulation results using 4 identical UR5 robotic manipulators in the realistic dynamic environment V-REP [?]. The 4 agents are rigidly grasping an object of 40 kg in an initial configuration as shown in Fig. 3.15. In order to verify the theoretical findings of the previous sections, we apply the controller (3.62) to achieve tracking of a desired trajectory by the object's center of mass. We simulate the closed loop system for two cases of  $G^*$ , namely the proposed one  $G_1^* = MG^\top(GMG^\top)^{-1}$  as well as the



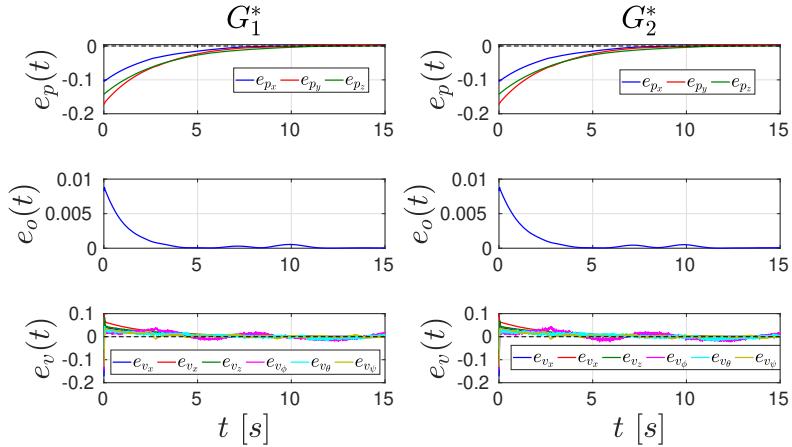
**Figure 3.15:** Four UR5 robotic arms rigidly grasping an object. The red counterpart represents a desired object pose at  $t = 0$ .

more standard choice  $G_2^* = G^\top (GG^\top)^{-1}$ . Moreover, we show for  $G_1^*$  the validity of Theorems 3.3 and 3.4 by plotting the arising internal forces, and we also illustrate the achievement of a desired nonzero internal force.

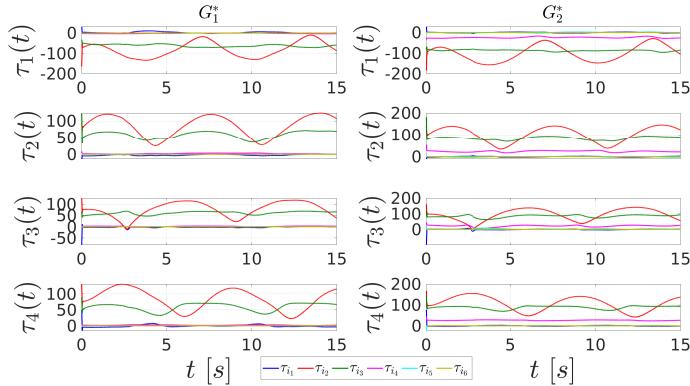
The initial pose of the object is set as  $p_o(0) = [-0.225, -0.612, 0.161]^\top$ ,  $\eta_o(0) = [0, 0, 0]^\top$  and the desired trajectory as  $p_d(t) = p_o(0) + [0.2 \sin(w_p t + \varphi_d), 0.2 \cos(w_p t + \varphi_d), 0.09 + 0.1 \sin(w_\theta t + \varphi_d)]^\top$ ,  $\eta_d(t) = [0.15 \sin(w_\phi t + \varphi_d), 0.15 \sin(w_\psi t + \varphi_d), 0.15 \sin(w_\theta t + \varphi_d)]^\top$  (in m and rad, respectively), where  $\varphi_d = \frac{\pi}{6}$ ,  $w_p = w_\phi = w_\psi = 1$ ,  $w_\theta = 0.5$ , and  $\eta_d(t)$  is transformed to the respective  $R_d(t)$ . The control gains are set as  $K_{p_1} = 15$ ,  $k_{p_2} = 75$ , and  $K_d = 40I_6$ .

The results are given in Figs. 3.16–3.19 for 15 seconds. Fig. 3.16 depicts the pose and velocity errors  $e_p(t)$ ,  $e_o(t)$ ,  $e_v(t)$ , which are shown to converge to zero for both choices of  $G^*$ , as expected. The control inputs  $\tau_i(t)$  of the agents are shown in Fig. 3.17. Moreover, the norm of the internal forces,  $\|h_{\text{int}}(t)\|$ , is computed via (3.51) and shown in Fig. 3.18. It is clear that  $G_2^*$  yields significantly large internal forces, whereas  $G_1^*$  keeps them very close to zero, as proven in the theoretical analysis. The larger internal forces in the case of  $G_2^*$  are associated with the larger control inputs  $\tau_i$ . This can be concluded from Fig. 3.17 and is also more clearly visualized in Fig. 3.19, which depicts the norms  $\|\tau_i(t)\|$  for the two choices of  $G^*$ ,  $\forall i \in \{1, \dots, 4\}$ . It is clear that inputs of larger magnitude occur in the case of  $G_2^*$ , which create internal forces (in the nullspace of  $G$ ). A video illustrating the aforementioned simulations can be found on <https://youtu.be/a31LTBBkE-Q>.

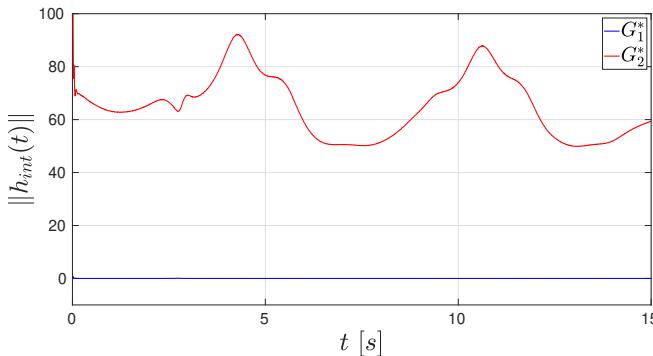
Finally, we set a random force vector  $h_{\text{int},d}$  in the nullspace of  $G$  and we simulate the control law (3.62) with the extra component  $u_{\text{int},d}$  (see Corollary 3.7). Fig. 3.20 illustrates the error norm  $\|e_{\text{int}}(t)\| := \|h_{\text{int},d}(t) - h_{\text{int}}(t)\|$ , which evolves close to zero. The minor observed deviations can be attributed to model uncertainties and hence the imperfect cancellation of the respective dynamics via (3.62).



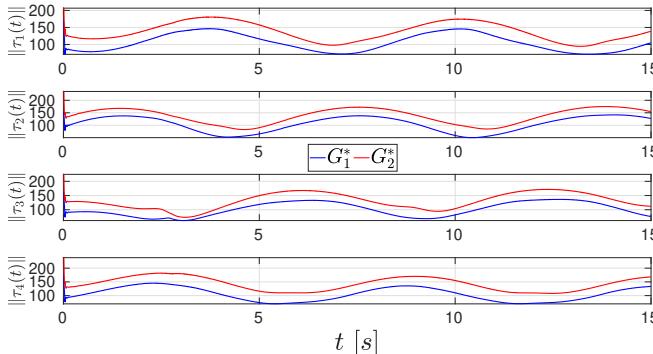
**Figure 3.16:** The error metrics  $e_p(t)$ ,  $e_o(t)$ ,  $e_v(t)$ , respectively, top to bottom, for the two choices  $G_1^*$  and  $G_2^*$  and  $t \in [0, 15]$  seconds.



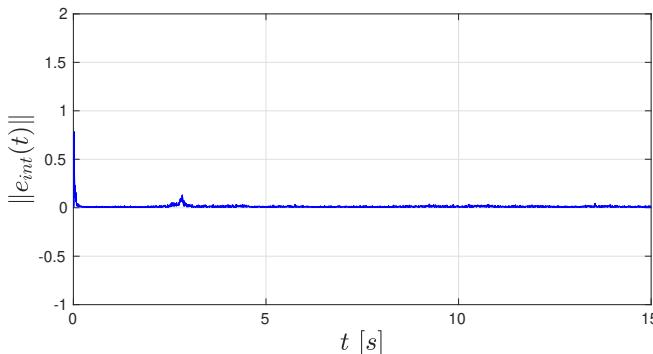
**Figure 3.17:** The resulting control inputs  $\tau_i(t)$  for  $G_1^*$  (left) and  $G_2^*$  (right),  $\forall i \in \{1, \dots, 4\}$  and  $t \in [0, 15]$  seconds.



**Figure 3.18:** The norm of the internal forces  $\|h_{int}(t)\|$  (as computed via (3.51)) for the two cases of  $G^*$  and  $t \in [0, 15]$  seconds.



**Figure 3.19:** The norms of the resulting control inputs,  $\|\tau_i(t)\|$  for  $G_1^*$  (with blue) and  $G_2^*$  (with red),  $\forall i \in \{1, \dots, 4\}$ , and  $t \in [0, 15]$  seconds.



**Figure 3.20:** The norm of the internal force error  $\|e_{int}(t)\|$ , when using  $G_1^*$  and for  $t \in [0, 15]$  seconds.

### 3.4 Conclusion

This chapter focused on multi-agent formation control design as well as its connection to rigid cooperative manipulation. Firstly, we developed a model-free decentralized control protocol for distance- and orientation-based formation control for a class of multi-agent systems modeled by Newton-Euler dynamics. Collision avoidance as well as connectivity maintenance was guaranteed to be satisfied by the proposed feedback control scheme. Secondly, we linked rigidity theory to rigid cooperative manipulation, by relating the former's rigidity matrix to the latter's grasp matrix. Moreover, we provided novel conditions for the internal force-free rigid cooperative manipulation.



## Chapter 4

# Continuous Coordination of Multi-Agent Systems

As discussed in Chapter 1, in order to be able to express complex tasks as temporal logic formulas, we need to have well-defined discrete representations of the continuous multi-agent system. Intuitively, this implies an appropriate discretization of the multi-agent state space, as well as the design of control schemes to navigate the agents among the points of this discretization. At the same time and since we are mainly interested in physical robotic agents, we need to guarantee safe multi-agent behavior, i.e., guarantee collision avoidance among the robotic agents and with potential workspace obstacles. On the same vein, we are interested in developing decentralized schemes, where the agents have local feedback only with respect to the neighbors. Therefore, multi-agent connectivity maintenance is another critical property we impose. Finally, as mentioned before, real robotic agents' equations of motion cannot be accurately known (model uncertainties) and are also subject to external disturbances. Hence, the control design needs to be robust and compensate appropriately for this partial model information (see the previous chapters, where adaptive control and PPC were used).

Motivated by the above, this chapter presents continuous control schemes for the coordination of multi-agent robotic systems. More specifically, we address the following three problems. Firstly, we develop an algorithm for the single- as well as multi-agent go-to-goal and collision-avoidance problem for robotic agents with uncertain dynamics. Secondly, we develop a novel leader-follower scheme for the navigation of a leader to a predefined point subject to model uncertainties and collision avoidance and connectivity maintenance constraints. Since the aforementioned algorithms consider mainly spherical robotic agents, we finally introduce a control scheme that guarantees collision avoidance between robotic agents of ellipsoidal shape.

## 4.1 Introduction

As mentioned in the previous chapters, multi-agent systems have received a large amount of attention lately, due to the advantages they bring with respect to single-agent setups. Apart from cooperative robotic manipulation and formation control, important multi-agent tasks, applicable to real robotic systems and studied in this chapter, consist of multi-robot navigation and leader-follower coordination. Moreover, we impose certain transient properties on the multi-agent system, such as collision avoidance [? ? ? ? ? ], and/or connectivity maintenance [? ? ? ? ? ? ? ? ? ], both crucial properties for real robotic systems. At the same time, we aim at developing control schemes that compensate for potentially uncertain dynamics of the robotic agents.

Multi-robot navigation with collision avoidance, possibly also with workspace obstacles, is a special instance of the motion planning problem [? ? ]. Several techniques have been developed in the related literature for robot motion planning with obstacle avoidance, such as discretization of the continuous space and employment of discrete algorithms (e.g., Dijkstra,  $A^*$ ), probabilistic roadmaps, sampling-based motion planning, and feedback-based motion planning [? ]. The latter offers closed-form analytic solutions by usually evaluating appropriately designed artificial potential fields, avoiding thus the potential complexity of employing discrete algorithms or discretizing the robot workspace. At the same time, feedback-based methods provide a solution to the control aspect of the motion planning problem, i.e., the correctness based on the solution of the closed-loop differential equation that describes the robot model.

Feedback-based motion planning has been receiving attention for more than two decades. Early works established the Koditschek-Rimon navigation function (KRNF) [? ? ], where the robot successfully converges to its goal while avoiding all obstacles from almost all initial conditions (in the sense of a measure-zero set), if the control gain of the goal term is chosen greater than a predefined constant. At the same time, an artificial potential fields based on harmonic functions and the panel method was proposed in [? ]. KRNFs were extended to more general workspaces and adaptive gain controllers [? ? ], to multi-robot systems [? ? ? ], and more recently, to convex potentials and obstacles [? ]. The idea of gain tuning has been also employed to an alternative KRNF in [? ].

Tuning-free constructions of artificial potential fields have also been developed in the related literature; [? ] considers dynamic obstacles and non-smooth controllers, [? ] tackles nonholonomic multi-robot systems, and in [? ? ? ] harmonic functions are combined with adaptive controllers for the goal gain to achieve almost global safe navigation. Harmonic functions are also used in [? ? ]. A transformation of arbitrarily shaped worlds

to points worlds, which facilitates the motion planning problem, is also considered in [? ?] and in [?] for multi-robot systems. The recent works of [?] and [?] guarantee also safe navigation in a predefined *time*.

Barrier functions for multi-robot collision avoidance are employed in [?] and optimization-based techniques via model predictive control (MPC) can be found in [? ?]; [?] and [?] propose reciprocal collision obstacle by local decision making for the desired velocity of the robot(s). Sensing uncertainties are taken into account in [? ]. A recent prescribed performance methodology for ellipsoidal obstacles is proposed in [?] and [?] extends a given potential field to 2nd-order systems. A similar idea is used in [? ], where the effects of an unknown drift term in the dynamics are examined. Workspace decomposition methodologies with hybrid controllers are employed in [?] for single- and [?] for multi-robot systems, respectively; A hybrid controller is also designed in the recent work [? ]; [?] employs reactive collision avoidance using admissible gaps, and [?] employs a contraction-based methodology that can also tackle the case of moving obstacles.

A common assumption that most of the aforementioned works consider is the simplified robot dynamics, i.e., single integrators/unicycle kinematics, without taking into account any robot dynamic parameters and where the control input is the robot velocity. Hence, indirectly, the schemes depend on an embedded internal system that converts the desired signal to the actual robot actuation command. The above imply that the actual robot trajectory might deviate from the desired one, jeopardizing its safety and possibly resulting in collisions.

Second-order realistic robot models are considered in MPC-schemes, like [? ? ]. Such optimization techniques, however, might result in computationally expensive solutions for large horizons. Moreover, regarding model uncertainties, a global upper bound is required, which is used to enlarge the obstacle boundaries and might yield infeasible solutions. A 2nd-order model is considered in [? ], without, however, considering any unknown dynamic terms. The same holds for [? ], where an already given potential function is extended to 2nd-order systems. The works [? ? ? ? ] consider simplified 2nd-order systems with *known* dynamic terms (and in particular, inertia and gravitational terms that are assumed to be successfully compensated); [?] guarantees the asymptotic stability of 2nd-order systems with a class of unknown drift terms to the critical points of a given potential function. However, there is no characterization of the region of attraction of the goal by analyzing the equilibrium points of the whole closed-loop system.

Another important feature of multi-agent systems is their coordination under leader-follower architectures, where an assigned leader aims at executing a task, and the rest of the team is concerned with secondary tasks, such as staying connected with the leader, forming a desired formation, or performing consensus protocols [? ? ? ? ? ? ? ]. When robotic teams

are concerned, such schemes resemble cases where a leader agent contains information regarding a task, and the followers need to comply with certain specifications to aid the leader.

Most leader-follower schemes in the related literature consider the follower consensus problem with fixed or time varying communication graphs, where the followers' states converge to the leader's one, which is assumed to have bounded velocity/acceleration [? ? ? ? ? ? ]. Moreover, connectivity maintenance in the transient state is also taken into account in a variety of leader-follower works (e.g., [? ? ? ]). Such schemes cannot be extended to multi-robot systems though, since collision avoidance is of uttermost importance and it is unreasonable to consider the convergence of the agents' states (e.g., positions) to the same value. Vehicular platoons are special cases of leader-follower structures where collision avoidance is taken into account [? ? ? ], restricted, however, to the longitudinal platoon-type sensing/communication graph.

Moreover, as discussed before, many of the multi-agent works in the related literature consider simplified/known dynamics ([? ]), which can have crucial effects on the actual behavior of real robotic systems, whose dynamics are described accurately by Lagrangian models, jeopardizing their performance/safety. More complex/uncertain dynamics are taken into account in [? ? ? ], without considering collision specifications; [? ] integrates collision avoidance with finite boundedness of the inter-agent distances, and [? ? ? ] deal with the multi-robot collision avoidance problem, without, however, providing theoretical guarantees with respect to the robot dynamics. Gain tuning is also performed in several works to cancel unknown nonlinearities, which are assumed to be uniformly bounded. An MPC methodology is developed in [? ], which can be computationally infeasible in real-time when complex dynamics are considered.

As discussed before, collision avoidance is considered to be a crucial property in real robotic systems, and is tackled in a large variety of multi-robot works. The majority of the related works, however, considers spherical agents, which provide a straightforward metric for the inter-agent or the agent-to-obstacle distances. However, since the shapes of real robotic vehicles can be far from spherical (e.g., robotic manipulators), that approach can be too conservative and may prevent the agents from fulfilling their primary objectives. Ellipsoids, on the other hand, can approximate more accurately the volume of autonomous agents.

The authors in [? ? ? ] employ diffeomorphisms to transform arbitrarily-shaped obstacles, including ellipsoids, to points. This methodology, however, is not straightforwardly extendable to the case of moving obstacles (i.e., multiple autonomous agents). A point-world transformation of multi-agent systems was taken into account in [? ]. As described in [? ] though, each

agent's transformation deforms the other agents into shapes whose implicit closed-form equation (and hence a suitable distance metric) is not trivial to obtain. The methodology of [?] provides useful insight, where the volume of each agent is “absorbed” to the other agents via Minkowski sums. The closed-form implicit equation of the resulting shapes, however, although possible to obtain [?], cannot be used to derive an appropriate distance metric in a straightforward way; [?] derives a conservative inter-ellipsoid distance by employing ellipsoid-to-sphere transformations and eigenvalue computations. An arithmetic algorithm that produces velocities for inter-agent elliptical agents is derived in [?], without, however, theoretical guarantees. Optimization-based techniques (e.g., Model Predictive Control), which can be employed for collision avoidance of convex-shaped agents (like e.g., in Chapter 2), can be too complex to solve, especially in cases where the control must be decentralized and/or complex dynamics are considered. The latter property constitutes another important issue regarding the related literature. In particular, most related works consider simplified single- or double-integrator models, which deviate from the actual dynamics and can lead to performance decline and safety jeopardy.

Barrier functions constitute a suitable tool for expressing objectives like collision avoidance. Originated in optimization, they are continuous functions that diverge to infinity as their argument approaches the boundary of a desired/feasible region. Barrier Lyapunov-like functions for general control systems can be found in [? ?], and in [? ? ?] for multi-agent systems, for obstacle avoidance with spherical obstacles/agents and time-dependent tasks.

This chapter deals with the following three problems. Firstly, we consider the robot navigation in an obstacle-cluttered environment under 2nd-order uncertain robot dynamics. The considered uncertainties consist of (i) unknown friction/drag terms, which are hard to model accurately, and (ii) unknown mass and unmodeled dynamics, motivated by transportation of objects of unknown mass or fuel consumption along a robot task, or cases where the robot is enhanced with other parts (e.g., robotic manipulators), whose dynamics are not known. We design a novel 2nd-order smooth navigation function which is integrated with adaptive control laws that compensate for the uncertain terms. Extensive analysis of the equilibrium points of the closed-loop system shows convergence to the desired goal from almost all initial conditions while avoiding obstacle collisions with the workspace boundary and spherical obstacles. The proposed scheme is then extended to star-worlds, i.e., workspaces with star-shaped obstacles [?]. Finally, using the single-robot methodology, we propose a *decentralized* hybrid coordination algorithm for the navigation of a multi-robot system in an environment cluttered with spherical obstacles.

Secondly, we propose a decentralized control protocol for the coordination

of a multi-agent system with 2nd order uncertain Lagrangian dynamics, subject to collision avoidance and connectivity maintenance. In particular, we consider that a leader agent has to navigate to a desired pose, inter-agent collisions must be avoided, and some of the initially connected agents have to remain connected. We are mainly motivated by cases where a cooperative task (e.g., cooperative pick-and-place tasks) is assigned to a multi-agent system, but the details are given only to a leader agent, which has to lead the entire team along the desired task. By using certain properties of the incidence matrix, we avoid issues of local minima and we relax the assumptions on the connectivity of the graph (as opposed to, e.g., [? ? ]) as well as the access of the leader's velocity by the followers. Moreover, we consider uncertain terms and unknown external disturbances in the dynamic model, which we cope with by using adaptive and discontinuous control laws.

Finally, we design smooth closed-form barrier functions for the collision avoidance of ellipsoidal agents. By employing results from the computer graphics field, we derive a novel closed-form expression that represents a distance metric of two ellipsoids in 3D space. Moreover, we use the latter to design a control protocol that guarantees the collision avoidance of a multi-agent system that aims to achieve a primary objective, subject to uncertain 2nd-order Lagrangian dynamics. The derived control law is (i) decentralized, in the sense that each agent calculates its control signal based on local information, (ii) discontinuous and adaptive, in order to compensate for the uncertainties and external disturbances.

## 4.2 Adaptive Robot Navigation with Collision Avoidance Subject to 2nd-order Uncertain Dynamics

We first consider the problem of single-robot navigation in a workspace cluttered with obstacles, subject to 2nd-order dynamics, whose analysis is necessary for the extension to the more general multi-robot problem.

### 4.2.1 Problem Statement

Consider a spherical robot operating in a bounded workspace  $\mathcal{W}$ , characterized by its position vector  $x \in \mathbb{R}^n$ ,  $n \in \{2, 3\}$  and radius  $r > 0$ , and subject to the dynamics:

$$\dot{x} = v \tag{4.1a}$$

$$m\dot{v} + f(x, v) + mg = u, \tag{4.1b}$$

where  $m > 0$  is the *unknown* mass,  $g \in \mathbb{R}^n$  is the constant gravity vector,  $u \in \mathbb{R}^n$  is the input vector, and  $f : \mathbb{R}^{2n} \rightarrow \mathbb{R}^n$  is a friction-like function, satisfying the following assumption:

**Assumption 4.1.** The function  $f : \mathbb{R}^{2n} \rightarrow \mathbb{R}^n$  is analytic and satisfies

$$\|f(x, v)\| \leq \alpha \|v\|,$$

$\forall x, v \in \mathbb{R}^{2n}$ , where  $\alpha \in \mathbb{R}_{\geq 0}$  is an unknown positive constant.

The aforementioned assumption is a standard condition concerning friction-like terms, which are bounded by the robot velocity [? ? ]. Constant unknown friction terms could be also included in the dynamics (e.g., incorporated in the gravity vector). Note also that  $\|f(x, v)\| \leq \alpha \|v\|$  implies  $f(x, 0) = 0$ , and  $\frac{\partial f(x, v)}{\partial x} \Big|_{v=0} = 0$ . The workspace is assumed to be an open ball centered at the origin

$$\mathcal{W} := \mathcal{B}(0, r_{\mathcal{W}}) = \{z \in \mathbb{R}^n : \|z\| < r_{\mathcal{W}}\}, \quad (4.2)$$

where  $r_{\mathcal{W}} > 0$  is the workspace radius. The workspace contains  $M \in \mathbb{N}$  closed sets  $\mathcal{O}_j$ ,  $j \in \mathcal{J} := \{1, \dots, M\}$ , corresponding to obstacles. Each obstacle is a closed ball centered at  $c_j \in \mathbb{R}^3$ , with radius  $r_{o_j} > 0$ :

$$\mathcal{O}_j := \bar{\mathcal{B}}(c_j, r_{o_j}) = \{z \in \mathcal{W} : \|z - c_j\| \leq r_{o_j}\}, \quad \forall j \in \mathcal{J}.$$

The analysis that follows will be based on the transformed workspace:

$$\bar{\mathcal{W}} := \{z \in \mathbb{R}^n : \|z\| < \bar{r}_{\mathcal{W}} := r_{\mathcal{W}} - r\}, \quad (4.3)$$

and the set of obstacles

$$\bar{\mathcal{O}}_j := \{z \in \mathcal{W} : \|z - c_j\| \leq \bar{r}_{o_j} := r_{o_j} + r\}, \quad \forall j \in \mathcal{J}.$$

and the robot is reduced to the point  $x$ . The free space is defined as

$$\mathcal{F} := \bar{\mathcal{W}} \setminus \bigcup_{j \in \mathcal{J}} \bar{\mathcal{O}}_j, \quad (4.4)$$

also known as a *sphere world* [? ]. We consider the following common feasibility assumption [? ] for  $\mathcal{F}$ :

**Assumption 4.2.** The workspace  $\mathcal{W}$  and the obstacles  $\mathcal{O}_j$  satisfy:

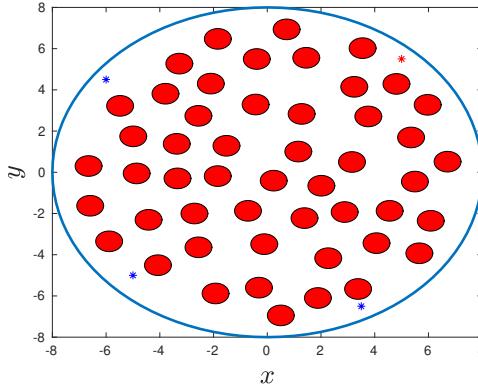
$$\begin{aligned} \|c_i - c_j\| &> r_{o_i} + r_{o_j} + 2r \\ r_{\mathcal{W}} - \|c_j\| &> r_{o_j} + 2r. \end{aligned}$$

The aforementioned assumption implies essentially that there is enough space among the obstacles and the workspace boundary and the obstacles for the robot to navigate, or equivalently,  $\bar{\mathcal{O}}_j \subset \bar{\mathcal{W}}$  and  $\bar{\mathcal{O}}_i \cap \bar{\mathcal{O}}_j = \emptyset, \forall i, j \in \mathcal{J}$ , with  $i \neq j$ .

Moreover, Assumption 4.2 implies that we can find some  $\bar{r} > 0$  such that

$$\|c_i - c_j\| > r_{o_i} + r_{o_j} + 2r + 2\bar{r}, \quad \forall i, j \in \mathcal{J}, i \neq j, \quad (4.5a)$$

$$r_{\mathcal{W}} - \|c_j\| > r_{o_j} + 2r + 2\bar{r}, \quad \forall j \in \mathcal{J} \quad (4.5b)$$



**Figure 4.1:** A 2D example of the workspace  $\bar{\mathcal{W}}$  with 50 obstacles  $\bar{\mathcal{O}}_j, j \in \{1, \dots, 50\}$ . The blue asterisks indicate potential initial configurations of the robot and the obstacles have been enlarged with the robot radius  $r$ . The red asterisk indicates a potential goal robot position.

This section treats the problem of navigating the robot to a destination  $x_d$  while avoiding the obstacles and the workspace boundary, formally stated as follows:

**Problem 4.1.** Consider a robot subject to the *uncertain* dynamics (4.1), operating in the aforementioned sphere world, with  $(x(t_0), v(t_0)) \in \mathcal{F} \times \mathbb{R}^n$ . Given a destination  $x_d \in \mathcal{F}$ , design a control protocol  $u$  such that

$$\begin{aligned} x(t) &\in \mathcal{F}, \quad t \geq t_0 \\ \lim_{t \rightarrow \infty} (x(t), v(t)) &= (x_d, 0) \end{aligned}$$

An illustration of the considered workspace is provided in Fig. 4.1

### 4.2.2 Single-Agent Solution

We provide in this section our methodology for solving Problem 4.1. Define first the set  $\bar{\mathcal{J}} := \{0\} \cup \mathcal{J}$  as well as the distances  $d_j := d_j(x) : \mathcal{F} \rightarrow \mathbb{R}_{\geq 0}$ ,

$j \in \bar{\mathcal{J}}$ , with  $d_j(x) := \|x - c_j\|^2 - \bar{r}_{\sigma_j}^2$ ,  $\forall j \in \mathcal{J}$ , and  $d_0 := d_0(x) := \bar{r}_{\mathcal{W}}^2 - \|x\|^2$ . Note that, by keeping  $d_j(x) > 0$ ,  $d_0(x) > 0$ , we guarantee that  $x \in \mathcal{F}$ <sup>1</sup>.

We introduce first the notion of the *2nd-order navigation function*:

**Definition 4.1.** A *2nd-order navigation function* is a function  $\phi := \phi(x) : \mathcal{F} \rightarrow \mathbb{R}_{\geq 0}$  of the form

$$\phi(x) := k_1 \|x - x_d\|^2 + k_2 \sum_{j \in \bar{\mathcal{J}}} \beta(d_j(x)), \quad (4.6)$$

where  $\beta : \mathbb{R}_{>0} \rightarrow \mathbb{R}_{\geq 0}$  is a (at least) twice contin. differentiable function and  $k_1, k_2$  are positive constants, with the followings properties:

1.  $\beta((0, \tau])$  is strictly decreasing,  $\lim_{z \rightarrow 0} \beta(z) = \infty$ , and  $\beta(z) = \beta(\tau)$ ,  $\forall z \geq \tau$ ,  $j \in \bar{\mathcal{J}}$ , for some  $\tau > 0$ ,
2.  $\phi(x)$  has a global minimum at  $x = x_d \in \text{int}(\mathcal{F})$  where  $\phi(x_d) = 0$ ,
3. if  $\beta'(d_k(x)) \neq 0$  and  $\beta''(d_k(x)) \neq 0$  for some  $k \in \bar{\mathcal{J}}$ , then  $\beta'(d_j(x)) = \beta''(d_j(x)) = 0$ , for all  $j \in \bar{\mathcal{J}} \setminus \{k\}$ , where ' and '' denote function derivatives.
4. The function  $\tilde{\beta} : (0, \tau) \rightarrow \mathbb{R}_{\geq 0}$ , with

$$\tilde{\beta}(z) := \beta''(z)z\sqrt{z}$$

is strictly decreasing.

By using the first property we will guarantee that, by keeping  $\beta(d_j(x))$  bounded, there are no collisions with the obstacles or the free space boundary. Property 2 will be used for the asymptotic stability of the desired point  $x = x_d$ . Property 3 places the rest of the critical points of  $\phi$  (which are proven to be saddle points) close to the obstacles, and the last property is used to guarantee that these are non-degenerate.

Examples for the function  $\beta$  that satisfy properties 1) and 4) are

$$\beta(z) := \begin{cases} \bar{\beta} \frac{\exp(-\frac{1}{z}) + \exp(-\frac{1}{\tau-z})}{\exp(-\frac{1}{z})}, & z \leq \tau \\ \bar{\beta}, & z \geq \tau, \end{cases}$$

---

<sup>1</sup>A safety margin can also be included, which needs, however, to be incorporated in the constant  $\bar{r}$  of (4.5).

for any positive  $\bar{\beta}$  and sufficiently small  $\tau$ , or the functions

$$\begin{aligned}\beta(z) &:= \begin{cases} \frac{1}{6z^5 - 15z^4 + 10z^3}, & z \leq 1 \\ 1, & z \geq 1, \end{cases} \\ \beta(z) &:= \begin{cases} \ln^4\left(\frac{z}{\tau}\right), & z \leq \tau \\ 0, & z \geq \tau. \end{cases}\end{aligned}\quad (4.7)$$

Note that  $\beta'(z) = \beta''(z) = 0$ , for  $z \geq \tau$ . We define also the constant

$$\bar{r}_d := \min \left\{ \bar{r}_{\mathcal{W}}^2 - \|x_d\|^2, \min_{j \in \mathcal{J}} \left\{ \|x_d - c_j\|^2 - \bar{r}_{o_j}^2 \right\} \right\} \quad (4.8)$$

as the minimum distance of the goal to the obstacles/workspace boundary.

We prove next that, by appropriately choosing  $\tau$ , only one  $\beta(d_j(x))$ ,  $j \in \bar{\mathcal{J}}$  affects the robotic agent for each  $x \in \mathcal{F}$ , and furthermore that  $\beta'(d_j(x_d)) = \beta''(d_j(x_d)) = 0$ . Hence, properties 2) and 3) of Def. 4.1 are satisfied.

**Proposition 4.1.** By choosing  $\tau$  as

$$\tau \in (0, \min\{\bar{r}^2, \bar{r}_d\}), \quad (4.9)$$

where  $\bar{r}, \bar{r}_d$  were introduced in (4.5) and (4.8), respectively, we guarantee that at each  $x \in \mathcal{F}$  there is not more than one  $j \in \bar{\mathcal{J}}$  such that  $d_j \leq \tau$ , implying that  $\beta'(d_j(x))$  and  $\beta''(d_j(x))$  are non-zero.

*Proof.* Assume that  $d_j(x) \leq \tau$  for some  $j \in \mathcal{J}$ ,  $x \in \mathcal{F}$ . Then, in view of (4.5), it holds that

$$\begin{aligned}\|x - c_j\|^2 &< \bar{r}^2 + \bar{r}_{o_j}^2 \Rightarrow \\ \|x - c_j\| &< \bar{r} + \bar{r}_{o_j} = \bar{r} + r + r_{o_j} < \|c_j - c_k\|\end{aligned}$$

$\forall k \in \mathcal{J} \setminus \{j\}$ , and hence,

$$\begin{aligned}\|x - c_k\| &= \|x - c_j + c_j - c_k\| \\ &\geq \|c_j - c_k\| - \|x - c_j\| > r_{o_k} + r + \bar{r} \Rightarrow \\ \|x - c_k\|^2 &> (r_{o_k} + r + \bar{r})^2 > (r_{o_k} + r)^2 + \bar{r}^2,\end{aligned}$$

and hence  $d_k(x) > \bar{r}^2 > \tau$ ,  $\forall k \in \mathcal{J} \setminus \{j\}$ . Moreover, in view of (4.5), it holds that

$$\begin{aligned}\|x\| &\leq \|x - c_j\| + \|c_j\| \pm r_{\mathcal{W}} \Rightarrow \\ \|x\| &< r_{\mathcal{W}} - r - \bar{r} \Rightarrow (r_{\mathcal{W}} - r)^2 \geq (\|x\| + \bar{r})^2 \Rightarrow \\ \bar{r}_{\mathcal{W}}^2 &\geq \|x\|^2 + \bar{r}^2 \Rightarrow \bar{r}_{\mathcal{W}}^2 - \|x\|^2 > \bar{r}^2,\end{aligned}$$

and hence  $d_o(x) > \tau$ . Similarly, we conclude by contradiction that  $d_o(x) \leq \tau \Rightarrow d_j > \tau, \forall j \in \mathcal{J}$ .  $\square$

Moreover, it holds for the desired equilibrium that

$$x = x_d \Leftrightarrow d_j(x) = \|x_d - c_j\|^2 - \bar{r}_j^2 \geq \bar{r}_d^2 > \tau,$$

and

$$x = x_d \Leftrightarrow d_0(x) = \bar{r}_{\mathcal{W}}^2 - \|x_d\|^2 \geq \bar{r}_d^2 > \tau,$$

and hence  $\beta'(d_j(x_d)) = \beta''(d_j(x_d)) = 0, \forall j \in \bar{\mathcal{J}}$ .

Intuitively, the obstacles and the workspace boundary have a local region of influence defined by the constant  $\tau$ , which will play a significant role in determining the stability of the overall scheme later. Moreover, it encompasses also the potential *local* sensing capabilities of the robot, since it takes into account the presence of the obstacles and the workspace boundary only when it is “ $\tau$ -close” to them. Similar techniques have been used in the literature, e.g., [? ?]. The expressions for the gradient and the Hessian of  $\phi$ , which will be needed later, are the following:

$$\nabla_x \phi(x) = 2k_1(x - x_d) + 2k_2 \sum_{j \in \mathcal{J}} \beta'(d_j)(x - c_j) - 2k_2 \beta'(d_0)x \quad (4.10a)$$

$$\begin{aligned} \nabla_x^2 \phi(x) = & 2 \left( k_1 - k_2 \beta'(d_0) + k_2 \sum_{j \in \mathcal{J}} \beta'(d_j) \right) I_n - 2k_2 \beta''(d_0) x x^\top \\ & + 2k_2 \sum_{j \in \mathcal{J}} \beta''(d_j)(x - c_j)(x - c_j)^\top. \end{aligned} \quad (4.10b)$$

Given the aforementioned definitions, we design a reference signal  $v_d := v_d(x) : \mathcal{F} \rightarrow \mathbb{R}^n$  for the robot velocity  $v$  as

$$v_d(x) = -\nabla_x \phi(x). \quad (4.11)$$

Next, we will design the control input  $u$  to guarantee tracking of the aforementioned reference velocity as well as compensation of the unknown terms  $m$  and  $f(x, v)$ . More specifically, we define the signals  $\hat{m} \in \mathbb{R}$  and  $\hat{\alpha} \in \mathbb{R}$  as the estimation terms of  $m$  and  $\alpha$  (see Assumption 4.1), respectively, and the respective errors  $\tilde{m} := \hat{m} - m$ ,  $\tilde{\alpha} := \hat{\alpha} - \alpha$ . We design now the control law  $u : \mathcal{F} \times \mathbb{R}^{n+2} \rightarrow \mathbb{R}^n$  as

$$u := u(x, v, \hat{m}, \hat{\alpha}) := -k_\phi \nabla_x \phi(x) + \hat{m}(\dot{v}_d + g) - \left( k_v + \frac{3}{2} \hat{\alpha} \right) e_v, \quad (4.12)$$

where  $e_v := v - v_d$ , and  $k_v, k_\phi$  are positive gain constants. Moreover, we design the adaptation laws for the estimation signals as

$$\dot{\hat{m}} := -k_m e_v^\top (\dot{v}_d + g) \quad (4.13a)$$

$$\dot{\hat{\alpha}} := k_\alpha \|e_v\|^2, \quad (4.13b)$$

with  $k_m, k_\alpha$  positive gain constants,  $\hat{\alpha}(t_0) \geq 0$ , and arbitrary finite initial condition  $\hat{m}(t_0)$ . The correctness of the proposed control protocol is established in the following theorem:

**Theorem 4.1.** Consider a robot operating in  $\mathcal{W}$ , subject to the uncertain 2nd-order dynamics (4.1). Given  $x_d \in \mathcal{F}$ , the control protocol (4.11)-(4.13) guarantees the collision-free navigation to  $x_d$  from almost all initial conditions  $(x(t_0), v(t_0), \hat{m}(t_0), \hat{\alpha}(t_0)) \in \mathcal{F} \times \mathbb{R}^{n+1} \times \mathbb{R}_{\geq 0}$ , given a sufficiently small  $\tau$  and that  $k_\phi > \frac{\alpha}{2}$ . Moreover, all closed loop signals remain bounded,  $\forall t \geq t_0$ .

**Remark 4.1.** Note that the proposed potential function (4.6) is, in a sense, equivalent to the one designed in [?], since the critical points are “pushed” arbitrarily close to the obstacles and, as shown in the proof of Theorem 4.1, they are also non-degenerate by choosing  $\tau$  small enough. In contrast to [?], however, as well as other related works (e.g., [? ? ?]), we do not require large goal gains (the gain  $k_1$  here) in order to establish the correctness of the propose scheme.

**Remark 4.2.** The proposed scheme can be also extended to *unknown* environments, where the amount and location of the spherical obstacles is unknown a priori, and these are sensed locally on-line. In particular, by having a large enough sensing neighborhood, the location (and possibly the radius) of each obstacle  $j \in \mathcal{J}$  can be sensed when  $d_j > \tau$ , and hence the respective term (which will be zero, since  $\beta'(d_j) = 0$ , for  $d_j > \tau$ ) can be smoothly incorporated in  $\nabla_x \phi(x)$ .

*Proof of Theorem 4.1.* Consider the Lyapunov candidate function

$$V := k_\phi \phi + \frac{m}{2} \|e_v\|^2 + \frac{3}{4k_\alpha} \tilde{\alpha}^2 + \frac{1}{2k_m} \tilde{m}^2. \quad (4.14)$$

Since  $x(t_0) \in \mathcal{F}$ , there exists a constant  $\bar{d}_j$  such that  $d_j(x(t_0)) \geq \bar{d}_j > 0$ ,  $j \in \bar{\mathcal{J}}$ , which implies the existence of a finite positive constant  $\bar{V}_0$  such that  $V(t_0) \leq \bar{V}_0$ . By considering the time derivative of  $V$  and using  $v = e_v + v_d$

and Assumption 4.1, we obtain after substituting (4.13):

$$\begin{aligned}\dot{V} &= k_\phi \nabla_x \phi(x)^\top (e_v + v_d) + e_v^\top (u - mg - f(x, v) - m\dot{v}_d) + \frac{3}{2} \tilde{\alpha} \|e_v\|^2 \\ &\quad - \tilde{m} e_v^\top (\dot{v}_d + g) \\ &\leq -k_\phi \|\nabla_x \phi(x)\|^2 + e_v^\top (k_\phi \nabla_x \phi(x) + u - m(g + \dot{v}_d)) + \alpha \|e_v\| \|v\| \\ &\quad + \frac{3\tilde{\alpha}}{2} \|e_v\|^2 - \tilde{m} e_v^\top (\dot{v}_d + g),\end{aligned}$$

which, by substituting (4.12) and using  $\alpha \|e_v\| \|v\| \leq \alpha \|e_v\|^2 + \frac{\alpha}{2} \|\nabla_x \phi(x)\|^2 + \frac{\alpha}{2} \|e_v\|^2$ , becomes

$$\begin{aligned}\dot{V} &\leq -\left(k_\phi - \frac{\alpha}{2}\right) \|\nabla_x \phi(x)\|^2 - k_v \|e_v\|^2 - \frac{3}{2} \hat{\alpha} \|e_v\|^2 + \frac{3}{2} \alpha \|e_v\|^2 + \tilde{m} e_v^\top (g + \dot{v}_d) + \frac{3\tilde{\alpha}}{2} \|e_v\|^2 - \tilde{m} e_v^\top (\dot{v}_d + g) \\ &= -\left(k_\phi - \frac{\alpha}{2}\right) \|\nabla_x \phi(x)\|^2 - k_v \|e_v\|^2 \leq 0.\end{aligned}$$

Hence, we conclude that  $V(t)$  is non-increasing, and hence  $\beta(d_j(x(t))) \leq V(t) \leq V(t_0) \leq \bar{V}_0$ ,  $\forall t \geq t_0$ , which implies that collisions with the obstacles and the workspace boundary are avoided, i.e.,  $x(t) \in \bar{\mathcal{F}} := \{x \in \mathcal{F} : \beta(d_j(x)) \leq \bar{V}_0, \forall j \in \bar{\mathcal{J}}\}$ ,  $\forall t \geq t_0$ . Moreover, (4.10) implies also the boundedness of  $\nabla_x \phi(x)|_{x(t)}$ ,  $\forall t \geq t_0$ . In addition, the boundedness of  $V(t)$  implies also the boundedness of  $x(t)$ ,  $e_v(t)$ ,  $\tilde{m}(t)$ ,  $\tilde{\alpha}(t)$ ,  $\tilde{g}(t)$  and hence of  $v(t)$ ,  $\dot{m}(t)$ ,  $\hat{\alpha}(t)$ ,  $\forall t \geq t_0$ . More specifically, by letting  $s := [x^\top, v^\top, \tilde{\alpha}, \tilde{m}]^\top$ , we conclude that  $s(t) \in \bar{S}$ ,  $\forall t \geq t_0$ , with

$$\begin{aligned}\bar{S} := \left\{ s \in \bar{\mathcal{F}} \times \mathbb{R}^{n+2} : |\tilde{\alpha}| \leq \sqrt{\frac{4}{3} k_\alpha \bar{V}_0}, |\tilde{m}| \leq \sqrt{2 k_m \bar{V}_0}, \right. \\ \left. \|v\| \leq \sqrt{2m\bar{V}_0} + \sup_{x \in \bar{\mathcal{F}}} \|\nabla_x \phi(x)\| \right\}\end{aligned}$$

Therefore, by invoking LaSalle's invariance principle (Theorem A.4 of Appendix A), we conclude that the solution  $s(t)$  will converge to the largest invariant set in  $S := \{s \in \bar{S} : \dot{V} = 0\}$ , which, in view of (4.11), becomes  $S := \{s \in \bar{S} : \nabla_x \phi(x) = 0, v = 0\}$ . Consider now the closed-loop dynamics

for  $s$ :

$$\dot{x} = v \quad (4.15a)$$

$$\dot{v} = \frac{1}{m}(\tilde{m}g + \hat{m}\dot{v}_d - k_\phi \nabla_x \phi(x) - \left(k_v + \frac{3}{2}\hat{\alpha}\right)(v + \nabla_x \phi(x)) - f(x, v)) \quad (4.15b)$$

$$\dot{\tilde{m}} = -k_m(v + \nabla_x \phi(x))^\top (\dot{v}_d + g) \quad (4.15c)$$

$$\dot{\hat{\alpha}} = k_\alpha \|v + \nabla_x \phi(x)\|^2. \quad (4.15d)$$

Note that, in view of the aforementioned discussion and the continuous differentiability of  $f(x, v)$ , the right-hand side of (4.15b) is bounded in  $\bar{S}$ . Note also that (4.10) implies the boundedness of  $\nabla_x^2 \phi(x)$  in  $\bar{\mathcal{F}}$ . Moreover, by differentiating  $\dot{v}$ , using the closed loop dynamics (4.15) and (4.10), we conclude the boundedness of  $\ddot{v}$  and the uniform continuity of  $\dot{v}(t)$  in  $\bar{S}$ . Hence, since  $\lim_{t \rightarrow \infty} v(t) = 0$ , we invoke Barbalat's Lemma (Lemma A.1 in Appendix A) to conclude that  $\lim_{t \rightarrow \infty} \dot{v}(t) = 0$ .

Therefore, the set  $S$  consists of the points where  $\dot{v} = v = \nabla_x \phi(x) = 0$ ,  $\dot{v}_d = \nabla_x^2 \phi(x)v = 0$ , and by also using the property  $f(x, 0) = 0$  we obtain  $\lim_{t \rightarrow \infty} \tilde{m}(t) = 0$  and  $\lim_{t \rightarrow \infty} \dot{s}(t) = 0$ . Note also that  $\hat{\alpha} : [t_0, \infty) \rightarrow \mathbb{R}_{\geq 0}$  is a monotonically increasing function and it converges thus to some constant positive value  $\hat{\alpha}^* > 0$ , since  $\hat{\alpha}(t_0) \geq 0$ , and  $\lim_{t \rightarrow \infty} \dot{\hat{\alpha}}(t) = \lim_{t \rightarrow \infty} \ddot{\hat{\alpha}}(t) = 0$ . Therefore, we conclude that the system will converge to an equilibrium  $s^* := [(x^*)^\top, 0^\top, 0, \hat{\alpha}^*]$  satisfying  $\nabla_x \phi(x)|_{x^*} = 0$ .

Since  $\lim_{t \rightarrow \infty} \nabla_x \phi(x)|_{x(t)} = \lim_{t \rightarrow \infty} v(t) = 0$ , the system converges to the critical points of  $\phi(x)$ , i.e., we obtain from (4.10) that at steady-state:

$$2k_1(x^* - x_d) = -k_2 \sum_{j \in \bar{\mathcal{J}}} \beta'(d_j^*)(x^* - c_j), \quad (4.16)$$

where  $d_j^* := d_j(x^*)$ ,  $\forall j \in \bar{\mathcal{J}}$ . According to the choice of  $\tau$  in (4.9),  $x^* = x_d$  implies that  $\beta'(d_j^*) = 0$ ,  $\forall j \in \bar{\mathcal{J}}$ , and hence the desired equilibrium  $x^* = x_d$  satisfies (4.16). Other *undesired* critical points of  $\phi(x)$  consist of cases where the two sides of (4.16) cancel each other out. However, as already proved, only one  $\beta'_j$  can be nonzero for each  $x \in \mathcal{F}$ . Hence, the undesired critical points satisfy one of the following expressions:

$$k_1(x^* - x_d) = -k_2 \beta'(d_k^*)(x^* - c_k), \quad (4.17a)$$

$$k_1(x^* - x_d) = k_2 \beta'(d_0^*)x^*, \quad (4.17b)$$

for some  $k \in \bar{\mathcal{J}}$ . In the case of (4.17b),  $x^*$  is collinear with the origin and  $x_d$ . However, the choice of  $\tau < \bar{r}_{\mathcal{W}}^2 - \|x_d\|^2$  in (4.9) implies that

$$d_0^* = \bar{r}_{\mathcal{W}}^2 - \|x^*\|^2 \leq \tau < \bar{r}_{\mathcal{W}}^2 - \|x_d\|^2 \Leftrightarrow \|x^*\| \geq \|x_d\|,$$

and hence  $x^* - x_d$  and  $x^*$  have the same direction. Therefore, since  $\beta'(d_j) < 0$ , for  $d_j < \tau$ ,  $\forall j \in \bar{\mathcal{J}}$ , (4.17b) is not feasible.

Moreover, in the case of (4.17a), since  $\beta'(d_k^*) \leq 0$ ,  $x^* - x_d$  and  $x^* - c_k$  point to the same direction. Hence, the respective critical points  $x^*$  are on the 1D line connecting  $x_d$  and  $c_k$ . Moreover, since  $\tau < \bar{r}_d \leq \|x_d - c_k\|^2 - \bar{r}_{o_k}^2$ , as chosen in (4.9), it holds that

$$d_k^* = \|x^* - c_k\|^2 - \bar{r}_{o_k}^2 < \|x_d - c_k\|^2 - \bar{r}_{o_k}^2 \Leftrightarrow \|x^* - x_d\| > \|x^* - c_k\|.$$

We proceed now by showing that the critical points satisfying (4.17a) are saddle points, which have a lower dimension stable manifold. Consider, therefore, the error  $e_x = x - x^*$ , where  $x^* \neq x_d$  represents the potential *undesired* equilibrium point that satisfies (4.17a). Let also  $s_e := [s_x^\top, \hat{\alpha}^\top]^\top$ , where  $s_x := [e_x^\top, v^\top, \tilde{m}]^\top$ , whose linearization around zero yields, after using (4.15) and  $\frac{\partial f(x, v)}{\partial x} \Big|_{v=0} = 0$ ,

$$\dot{s}_e = \bar{A}_s s_e, \quad (4.18)$$

where

$$\begin{aligned} \bar{A}_s &:= \begin{bmatrix} A_s & 0 \\ 0^\top & 0 \end{bmatrix} \\ A_s &:= \begin{bmatrix} 0_{n \times n} & I_n & 0 \\ A_{s,21} & A_{s,22} & g \\ -k_m g^\top (\nabla_x^2 \phi(x))^\top|_{x^*} & -k_m g^\top & 0 \end{bmatrix}, \end{aligned}$$

and

$$\begin{aligned} A_{s,21} &:= -\frac{1}{m} \left( k_\phi + k_v + \frac{3}{2} \hat{\alpha}^* \right) \nabla_x^2 \phi(x)|_{x^*} \\ A_{s,22} &:= -\nabla_x^2 \phi(x)|_{x^*} - \left( k_v + \frac{3}{2} \hat{\alpha}^* \right) I_n - \frac{3}{2} \frac{\partial f(x, v)}{\partial v} \Big|_{s^*}. \end{aligned}$$

We aim to prove that the equilibrium  $s_x^* := [0^\top, 0^\top, 0]^\top$  has at least one positive eigenvalue. To this end, consider a vector  $\bar{\nu} := [\mu\nu^\top, \nu^\top, 0]^\top$ , where  $\mu > 0$  is a positive constant, and  $\nu \in \mathbb{R}^n$  is an orthogonal vector to  $(x^* - c_k)$ , i.e.  $\nu^\top (x^* - c_k) = 0$ . Then the respective quadratic form yields

$$\begin{aligned} \bar{\nu}^\top A_s \bar{\nu} &= [\nu^\top A_{s,21} \quad \mu\nu^\top + \nu^\top A_{s,22} \quad \nu^\top g] \begin{bmatrix} \mu\nu \\ \nu \\ 0 \end{bmatrix} = \mu\nu^\top A_{s,21} \nu + \mu \|\nu\|^2 \\ &\quad + \nu^\top A_{s,22} \nu, \end{aligned}$$

which, after employing (4.10) with  $\beta'(d_j^*) = 0, \forall j \in \mathcal{J} \setminus \{k\}$  and  $\nu^\top(x^* - c_k) = 0$ , becomes

$$\begin{aligned}\bar{\nu}^\top A_s \bar{\nu} = & -\frac{2\mu k_1}{m}(k_\phi + k_v + \frac{3}{2}\hat{\alpha}^*) \left(1 + \frac{k_2}{k_1}\beta'(d_k^*)\right) \|\nu\|^2 + \mu \|\nu\|^2 \\ & - 2k_1 \left(1 + \frac{k_2}{k_1}\beta'(d_k^*)\right) \|\nu\|^2 - \left(k_v + \frac{3}{2}\hat{\alpha}^*\right) \|\nu\|^2 - \nu^\top \frac{\partial f(x, v)}{\partial v} \Big|_{s^*} \nu.\end{aligned}$$

From (4.17a), by recalling that  $\beta'(d_k) \leq 0$ , we obtain that

$$\frac{k_2}{k_1}\beta'(d_k^*) = -\frac{\|x^* - x_d\|}{\|x^* - c_k\|} < -1. \quad (4.19)$$

Therefore by defining  $c^* := -\frac{k_2}{k_1}\beta'(d_k^*) - 1 > 0$ , we obtain

$$\begin{aligned}\bar{\nu}^\top A_s \bar{\nu} = & \left(\frac{2\mu k_1}{m}k_\phi c^* + \left(\frac{2\mu k_1}{m}c^* - 1\right)\left(k_v + \frac{3}{2}\hat{\alpha}^*\right) + \mu + 2k_1 c^*\right) \|\nu\|^2 \\ & - \nu^\top \frac{\partial f(x, v)}{\partial v} \Big|_{s^*} \nu,\end{aligned}$$

which is rendered positive by choosing a sufficiently large  $\mu$ . Hence,  $A_s$  has at least one positive eigenvalue. Next, we prove that  $A_s$  has no zero eigenvalues by proving that its determinant is nonzero. For the determinant of  $\nabla_x^2 \phi(x)|_{x^*}$ , it holds in view of (4.10) that

$$\det(\nabla_x^2 \phi(x)|_{x^*}) = \det \left( 2(k_1 + k_2\beta'(d_k^*)) I_n + 2k_2\beta''(d_k^*)(x^* - c_k)(x^* - c_k)^\top \right).$$

By using the property  $\det(A+uv^\top) = (1+v^\top A^{-1}u) \det(A)$ , for any invertible matrix  $A$  and vectors  $u, v$ , we obtain

$$\det(\nabla_x^2 \phi(x)|_{x^*}) = 2^n (k_1 + k_2\beta'(d_k^*))^n \left(1 + \frac{k_2}{k_1(1 + \frac{k_2}{k_1}\beta'(d_k^*))} \beta''(d_k^*) \|x^* - c_k\|^2\right). \quad (4.20)$$

In view of (4.19) and by using  $\|x^* - x_d\| - \|x^* - c_k\| = \|x_d - c_k\|$  since  $x^*, c_k$  and  $x_d$  are collinear, (4.20) becomes

$$\det(\nabla_x^2 \phi(x)|_{x^*}) = 2^n (k_1 + k_2\beta'(d_k^*))^n \left(1 - \frac{k_2}{k_1 \|x_d - c_k\|} \beta''(d_k^*) \|x^* - c_k\|^3\right).$$

Note that, since  $\lim_{d_j \rightarrow 0} \beta(d_j) = \infty$  and  $\beta(d_j)$  decreases to  $\beta(d_j) = \beta(\tau)$ ,  $\forall d_j \geq \tau$ , the derivatives  $\beta'(d_j)$  satisfy  $\lim_{d_j \rightarrow 0} \beta'(d_j) = -\infty$  and increase

to  $\beta'(d_j) = 0$ ,  $\forall d_j \geq \tau$ . Hence, we conclude that  $\beta''(d_j) > 0$ ,  $\forall d_j \in (0, \tau)$ . Therefore, in order for the critical point to be non-degenerate, we must guarantee that

$$\frac{k_2}{k_1 \|x_d - c_k\|} \beta''(d_k^*) \|x^* - c_k\|^3 > 1. \quad (4.21)$$

By expressing  $\|x^* - c_k\|^3 = (d_k^* + \bar{r}_{o_k}) \sqrt{d_k^* + \bar{r}_{o_k}^2}$ , considering that  $\|x_d - c_k\| \leq 2\bar{r}_W$  and setting  $\underline{r} := \min_{j \in \mathcal{J}} \{\bar{r}_{o_j}\}$ , a lower bound for the left-hand side of (4.21) is

$$f_\ell(d_k^*) := \frac{k_2}{2k_1 \bar{r}_W} \beta''(d_k^*) (d_k(x^*) + \underline{r}^2) \sqrt{d_k(x^*) + \underline{r}^2}. \quad (4.22)$$

According to Property 4 of Definition 4.1, (4.22) is a decreasing function of  $d_k^*$ , for  $d_k^* \in (0, \tau)$ , with  $f_\ell(\tau) = 0$  and  $\lim_{d_k^* \rightarrow 0} f_\ell(d_k^*) = \infty$ . Therefore, there exists a positive  $d_k^{**} > 0$ , such that  $f_\ell(d_k^*) > 1$ ,  $\forall d_k^* < d_k^{**}$ . Hence, by setting  $\tau < d_k^{**}$ , we achieve  $d_k^* < \tau < d_k^{**}$  and guarantee that  $f_\ell(d_k^*) > 1$ .

Next, by defining  $A_{2ns} := \begin{bmatrix} 0_{n \times n} & I_n \\ A_{s,21} & A_{s,22} \end{bmatrix}$ , it holds that

$$\det(A_{2ns}) = \det(A_{s,21}) = (-1)^n \frac{1}{m^n} \left( k_\phi + k_v + \frac{3}{2} \hat{\alpha}^* \right)^n \det(\nabla_x^2 \phi(x)|_{x^*}) \neq 0,$$

and

$$A_{2ns}^{-1} = \begin{bmatrix} * & A_{s,21}^{-1} \\ * & 0_{n \times n} \end{bmatrix}$$

and therefore we obtain that

$$\begin{aligned} \det(A_s) &= \det(A_{s,21}) [k_m g^\top (\nabla_x^2 \phi(x))^\top|_{x^*} \quad k_m g^\top] A_{2ns}^{-1} \begin{bmatrix} 0 \\ g \end{bmatrix} = \\ &= \det(A_{s,21}) [k_m g^\top (\nabla_x^2 \phi(x))^\top|_{x^*} \quad k_m g^\top] \begin{bmatrix} A_{s,21}^{-1} g \\ 0 \end{bmatrix} = \\ &= \det(A_{s,21}) k_m g^\top (\nabla_x^2 \phi(x))^\top|_{x^*} A_{s,21}^{-1} g = \\ &= k_m g^\top (\nabla_x^2 \phi(x))^\top|_{x^*} \text{adj}(A_{s,21}) g, \end{aligned}$$

which is non-zero, since  $g \neq 0$ ,

$$\det \left( (\nabla_x^2 (\phi(x)))^\top|_{x^*} \text{adj}(A_{s,21}) \right) = \det(\nabla_x^2 (\phi(x))^\top|_{x^*}) \det(A_{s,21})^{n-1} \neq 0$$

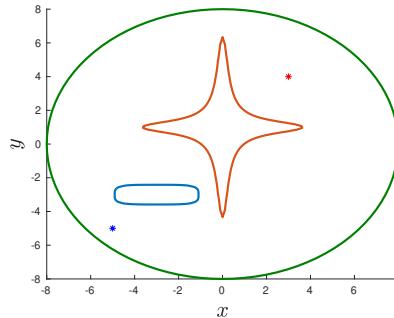
and hence the matrix that forms the latter quadratic form is nonsingular.

Therefore, we conclude that  $A_s$  is non-degenerate and has at least one positive eigenvalue. Note that  $\bar{A}_s$  has the same eigenvalues as  $A_s$  and an extra zero eigenvalue. According to the Reduction Principle (Theorem A.8)

of Appendix A), (4.18) is locally topologically equivalent near the origin to the system

$$\begin{aligned}\dot{\hat{\alpha}} &= k_\alpha \|v_\alpha(\hat{\alpha}) + \nabla_x \phi(x)|_{x_\alpha(\hat{\alpha})}\|^2 \\ \dot{s}_x &= A_s s_x,\end{aligned}$$

where  $v_\alpha(\hat{\alpha})$ ,  $\nabla_x \phi(x)|_{x_\alpha(\hat{\alpha})}$  are the restrictions of  $v$  and  $\nabla_x \phi(x)$  to the center manifold of  $\hat{\alpha}$  (see Theorem A.8 of Appendix A). Regarding the trajectories of  $s_x$ , since  $A_s$  is a non-degenerate saddle (it has at least one positive eigenvalue) its stable manifold has dimension lower than  $2n + 1$  and is thus a set of zero measure. Therefore, all the initial conditions  $(x(t_0), v(t_0), \tilde{m}(t_0)) \in \mathcal{F} \times \mathbb{R}^{n+1}$ , except for the aforementioned lower-dimensional manifold, converge to the desired equilibrium  $(x_d, 0, 0)$ .  $\square$



**Figure 4.2:** A workspace with two star-shaped obstacles. The blue asterisk indicates the center of the robot and the obstacles have been enlarged with the robot radius  $r$  through the Minkowski sum. The red asterisk indicates a potential goal robot position.

**Remark 4.3.** The proof can be trivially extended to the 2D case on the horizontal plane where there is no gravity, i.e.,  $g = 0$ . In addition, it is worth noting that in that case, the estimation parameter  $\tilde{m}(t)$  will converge to a constant value different than the mass  $m$ , as revealed by a careful inspection of the closed-loop system. Moreover, the condition  $k_\phi > \frac{\alpha}{2}$  of Theorem 4.1 is only sufficient and not necessary, as will be shown in the simulation results.

### 4.2.3 Dynamic Disturbance Addition

Except for the already considered dynamic uncertainties, we can add to the right-hand side of (4.1) an unknown disturbance vector  $d := d(x, v, t) :$

$\mathbb{R}^{2n} \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^n$ , i.e.,

$$\begin{aligned}\dot{x} &= v \\ m\dot{v} + f(x, v) + mg + d(x, v, t) &= u,\end{aligned}$$

subject to a uniform boundedness condition  $\|d(x, v, t)\| \leq \bar{d}, \forall x, v, t \in \mathbb{R}^{2n} \times \mathbb{R}_{\geq 0}$ . In this case, by slightly modifying the control scheme, we still guarantee collision avoidance with the workspace obstacles and boundary. In addition, we achieve uniform ultimate boundedness of the error signals as well as the gradient of  $\phi$ , as the analysis in this section shows.

The control scheme of the previous section is appropriately enhanced to incorporate the  $\sigma$ - modification [?], a common technique in adaptive control. More specifically, the adaptation laws (4.13) are modified according to

$$\begin{aligned}\dot{m} &:= -k_m e_v^\top (\dot{v}_d + g) - \sigma_m \hat{m} \\ \dot{\alpha} &:= k_\alpha \|e_v\|^2 - \sigma_\alpha \hat{\alpha},\end{aligned}$$

where  $\sigma_m, \sigma_\alpha$  are positive gain constants, to be appropriately tuned as per the analysis below.

Consider now the function  $V$  as defined (4.14). In view of the analysis of the previous section, the incorporation of  $d(x, v, t)$ , as well as the modification of the adaptation laws, the derivative of  $V$  becomes

$$\dot{V} \leq -\left(k_\phi - \frac{\alpha}{2}\right) \|\nabla_x \phi(x)\|^2 - k_v \|e_v\|^2 + \|e_v\| \bar{d} - \frac{3}{2} \sigma_\alpha \tilde{\alpha} \hat{\alpha} - \sigma_m \tilde{m} \hat{m},$$

which, by using  $\hat{\alpha} = \tilde{\alpha} + \alpha$ ,  $\hat{m} = \tilde{m} + m$ , as well as the properties  $-ab = -\frac{1}{2}(a+b)^2 + \frac{a^2}{2} + \frac{b^2}{2}$ ,  $ab = -\frac{1}{2}(a-b)^2 + \frac{a^2}{2} + \frac{b^2}{2}$ ,  $\forall a, b \in \mathbb{R}$ , becomes

$$\begin{aligned}\dot{V} &\leq -\left(k_\phi - \frac{\alpha}{2}\right) \|\nabla_x \phi(x)\|^2 - \frac{k_v}{2} \|e_v\|^2 + \frac{\bar{d}^2}{2k_v} - 3\sigma_\alpha \frac{\tilde{\alpha}^2}{4} - \sigma_m \frac{\tilde{m}^2}{2} \\ &\quad + 3\sigma_\alpha \frac{\alpha^2}{4} + \sigma_m \frac{m^2}{2} \\ &\leq -k_\xi \|\xi\|^2 + d_\xi,\end{aligned}$$

where  $\xi := [\nabla_x \phi(x)^\top, e_v^\top, \tilde{m}, \tilde{\alpha}]^\top \in \mathbb{R}^{2n+2}$ ,  $k_\xi := \min \left\{ k_\phi - \frac{\alpha}{2}, k_v, \frac{\sigma_m}{2}, \frac{3\sigma_\alpha}{4} \right\}$ , and  $d_\xi := \frac{\bar{d}^2}{2k_v} + 3\sigma_\alpha \frac{\alpha^2}{4} + \sigma_m \frac{m^2}{2}$ . Therefore,  $\dot{V}$  is negative when  $\|\xi\| > \sqrt{\frac{d_\xi}{k_\xi}}$ , which implies uniform ultimate boundedness of  $\|\xi(t)\|$  in a set around zero, whose size is proportional to  $d_\xi$ , which can be shrunk by gain tuning (see Theorem A.5 of Appendix A). In addition,  $\dot{V}$  is sign indefinite only in the set defined by  $\|\xi\| < \sqrt{\frac{d_\xi}{k_\xi}}$  and negative otherwise, which implies that  $V(t)$

remains bounded,  $\forall t \geq 0$ , and hence collisions with the workspace obstacles and boundary are avoided.

Note that the aforementioned analysis guarantees that  $\nabla_x \phi(x)|_{x(t)}$  will be ultimately bounded in a set close to zero. This point, however, might be a critical point of  $\phi$  and it is not guaranteed that  $x(t)$  will be bounded close to the goal configuration  $x_d$ . Nevertheless, intuition suggests that if the disturbance vector  $d(x, v, t)$  does not behave adversarially, the agent will converge close to the goal configuration. This is also verified by the simulation results of Section 4.2.6.

#### 4.2.4 Extension to Star Worlds

Before moving to the multi-robot case, we discuss in this section how the proposed control scheme can be extended to generalized sphere worlds, a wider class of configuration spaces than the spheres worlds described in the previous section. In particular, we consider *star worlds*, which are configuration spaces diffeomorphic to sphere worlds. In particular, a star world is a set of the form

$$\mathcal{T} := \bar{\mathcal{W}} \setminus \bigcup_{j \in \mathcal{J}} \bar{O}_{\mathcal{T}_j},$$

where  $\bar{\mathcal{W}}$  is a workspace of the form (4.3) and  $\bar{O}_{\mathcal{T}_j}$  are  $M$  disjoint star-shaped obstacles (indexed by  $\mathcal{J} = \{1, \dots, M\}$ ). The latter are sets characterized by the property that all rays emanating from a center point cross their boundary only once [? ] (see Fig. 4.2).

One can design a diffeomorphic mapping  $H : \mathcal{T} \rightarrow \mathcal{F}$ , where  $\mathcal{F}$  is a sphere world of the type (4.4). More specifically,  $H$  maps the boundary of  $\mathcal{T}$  to the boundary of  $\mathcal{F}$ .

The control scheme of the previous section is modified now to account for the transformation  $H$  as follows. The desired robot velocity is set to  $v_d : \mathcal{T} \rightarrow \mathbb{R}^n$ , with

$$v_d(x) := -J_H(x)^{-1} \nabla_{H(x)} \phi(H(x)), \quad (4.23)$$

where  $J_H(x) := \frac{\partial H(x)}{\partial x}$  is the nonsingular Jacobian matrix of  $H$ . Next, by letting  $e_v := v - v_d$ , the control law is designed as  $u : \mathcal{T} \times \mathbb{R}^{n+2} \rightarrow \mathbb{R}^n$ , with

$$u := u(x, v, \hat{m}, \hat{\alpha}) := -k_\phi J_h(x)^\top \nabla_{H(x)} \phi(H(x)) + \hat{m}(\dot{v}_d + g) - \left( k_v + \frac{3}{2} \hat{\alpha} \right) e_v, \quad (4.24)$$

where  $\hat{m}$  and  $\hat{\alpha}$  evolve according to the respective expressions in (4.13). The next theorem gives the main result of this section.

**Theorem 4.2.** Consider a robot operating in  $\mathcal{W}$ , subject to the uncertain 2nd-order dynamics (4.1). Given  $x_d \in \mathcal{T}$ , the control protocol (4.13), (4.23), (4.24) guarantees the collision-free navigation to  $x_d$  from almost all initial conditions  $(x(t_0), v(t_0), \hat{m}(t_0), \hat{\alpha}(t_0)) \in \mathcal{T} \times \mathbb{R}^{n_1} \times \mathbb{R}_{\geq 0}$ , given a sufficiently small  $\tau$  and that  $k_\phi > \frac{\alpha}{2}$ . Moreover, all closed loop signals remain bounded,  $\forall t \geq t_0$ .

*Proof.* Following similar steps as in the proof of Theorem 4.1, we consider the Lyapunov candidate function

$$V := k_\phi \phi(H(x)) + \frac{m}{2} \|e_v\|^2 + \frac{1}{2k_\alpha} \tilde{\alpha}^2 + \frac{3}{4k_m} \tilde{m}^2,$$

whose derivative along the solutions of the closed loop system can be proven to satisfy

$$\dot{V} \leq - \left( k_\phi - \frac{\alpha}{2} \right) \|\nabla_{H(x)} \phi(H(x))\|^2 - k_v \|e_v\|^2 \leq 0,$$

which proves the boundedness of the obstacle functions  $\beta(d_j(H(x(t))))$ ,  $\forall j \in \mathcal{J}, t \geq t_0$ . Since the boundaries  $\partial \bar{\mathcal{O}}_j$  are mapped to  $\partial \bar{\mathcal{O}}_{\mathcal{T}_j}$  through  $H(x)$ , we conclude that  $x(t) \in \mathcal{T}$ ,  $t \geq t_0$  and no collisions occur. Next, by following similar arguments as in the proof of Theorem 4.1, we conclude that the solution will converge to a critical point of  $\phi(H(x))$ . By choosing a sufficiently small  $\tau$  for the obstacle functions  $\beta(d_j(H(x(t))))$ , the critical points consist of the desired equilibrium, where  $\beta'(d_j(H(x_d))) = 0$ ,  $\forall j \in \mathcal{J}$ , or undesired critical points  $x^*$  satisfying

$$k_1(H(x^*) - H(x_d)) = -k_2 \beta'(d_{H_k}^*)(H(x^*) - H(c_k)), \quad (4.25)$$

for some  $k \in \mathcal{J}$ , where we define  $d_{H_j}^* := d_j(H(x^*))$ ,  $\forall j \in \mathcal{J}$ . The respective linearization matrix  $\bar{A}_s$  from (4.18) becomes now

$$\begin{aligned} \bar{A}_s &:= \begin{bmatrix} A_s & 0 \\ 0^\top & 0 \end{bmatrix} \\ A_s &:= \begin{bmatrix} 0_{n \times n} & I_n & 0 \\ A_{s,21} & A_{s,22} & g \\ A_{s,31} & -k_m g^\top & 0 \end{bmatrix}, \end{aligned}$$

with

$$\begin{aligned} A_{s,21} &:= -\frac{1}{m} \left( k_\phi J_H(x^*)^\top + \left( k_v + \frac{3}{2} \hat{\alpha}^* \right) J_H(x^*)^{-1} \right) \nabla^2 \phi^* J_H(x^*) \\ A_{s,22} &:= -J_H(x^*)^{-1} \nabla^2 \phi^* J_H(x^*) - \left( k_v + \frac{3}{2} \hat{\alpha}^* \right) I_n - \left. \frac{\partial f(x, v)}{\partial v} \right|_{s^*}, \\ A_{s,31} &:= -k_m g^\top (J_H(x^*)^{-1} \nabla^2 \phi^* J_H(x^*))^\top \end{aligned}$$

and  $\nabla^2\phi^* := \nabla_{H(x)}^2\phi(H(x))|_{x^*}$ , around  $x = x^*, v = 0, \tilde{m} = 0, \tilde{\alpha} = \tilde{\alpha}^*$ . Next, similarly to the proof of Theorem 4.2, we prove that  $\bar{\nu}^\top A_s \bar{\nu} > 0$ , for  $\bar{\nu} := [\mu\nu^\top, \nu^\top, 0]^\top$ , where  $\mu > 0$  is a positive constant and  $\nu := J_H(x^*)^{-1}\hat{\nu}$ , with  $\hat{\nu} \in \mathbb{R}^n$  a vector orthogonal to  $(H(x^*) - H(c_k))$ . The respective quadratic form yields, after employing (4.25) and defining  $c^* := -\left(1 + \frac{k_2}{k_1}\beta'(d_{H_k}^*)\right) > 0$ :

$$\begin{aligned} \bar{\nu}^\top A_s \bar{\nu} &= \hat{\nu}^\top \left[ \frac{2k_1 k_\phi \mu c^*}{m} I_n + J_H(x)^{-\top} \left( \left( \frac{2k_1 c^* (k_v + \frac{3}{2}\hat{\alpha}^*)}{m} + \right. \right. \right. \\ &\quad \left. \left. \left. \mu - \left( k_v + \frac{3}{2}\hat{\alpha}^* \right) + 2k_1 c^* \right) I_n - \frac{\partial f(x, v)}{\partial v} \Big|_{s^*} \right) J_H(x)^{-1} \right] \hat{\nu}, \end{aligned}$$

which can be rendered positive for sufficiently large  $\mu$ .

Moreover, at a critical point  $x^{*,1}$  of  $\phi(H(x))$ , it holds that (see the proof of Prop. 2.6 in [?]),

$$\nabla_{H(x)}^2\phi(H(x))|_{x^{*,1}} = J_H(x^{*,1})^\top \nabla_x^2\phi(x)|_{x^{*,2}} J_H(x^{*,1}),$$

where  $x^{*,2} := H(x^{*,1})$  is a critical point of  $\phi(x)$ . Since  $J_H(x)$  is nonsingular, it holds that  $x^{*,1}$  is non-degenerate if and only if  $x^{*,2}$  is non-degenerate. As already shown in the proof of Theorem 4.1, by choosing  $\tau$  sufficiently small, we render the critical points of  $\phi(x)$  that are close to the obstacles non-degenerate. Hence, we conclude that the respective critical points of  $\phi(H(x))$  are also non-degenerate and  $\det(\nabla^2\phi^*) \neq 0$ .

Next, in order to prove that the critical point  $(x^*, 0, 0)$  is non-degenerate, we calculate the determinant of  $A_s$ . Following the proof of Theorem 4.1, we obtain that

$$\begin{aligned} \det(A_s) &= \det(A_{s,21}) k_m g^\top (J_H(x^*)^{-1} \nabla^2\phi^* J_H(x^*))^\top A_{s,21}^{-1} g \\ &= k_m g^\top (J_H(x^*)^{-1} \nabla^2\phi^* J_H(x^*))^\top \text{adj}(A_{s,21}) g \end{aligned}$$

where

$$\begin{aligned} \det(A_{s,21}) &= (-1)^n \left( \frac{k_\phi^n}{m^n} \det(J_H(x^*)) + \right. \\ &\quad \left. \left( k_v + \frac{3}{2}\hat{\alpha}^* \right)^n \frac{1}{\det(J_H(x^*))} \right) \det(\nabla^2\phi^*) \det(J_H(x^*)), \end{aligned}$$

which is not zero, since  $\det(\nabla^2\phi^*) \neq 0$  and  $J_H(x^*) \neq 0$ . Hence, we conclude that the aforementioned quadratic form is also not zero and hence the non-degeneracy of the critical points under consideration. Hence, by following similar arguments as in the proof of Theorem 4.1, we conclude that the initial conditions that converge to these critical saddle points form a set of measure zero.  $\square$

### 4.2.5 Extension to Multi-Robot Systems

This section is devoted to extending the results of Section 4.2.2 to Multi-Robot systems. Consider, therefore,  $N \in \mathbb{N}$  spherical robots operating in a workspace  $\mathcal{W}$  of the form (4.2), characterized by their position vectors  $x_i \in \mathbb{R}^n$ , as well as their radii  $r_i > 0$ ,  $i \in \mathcal{N} := \{1, \dots, N\}$ , and obeying the second-order uncertain dynamics (4.1), i.e.,

$$\dot{x}_i = v_i \quad (4.26a)$$

$$m_i \ddot{v}_i + f_i(x_i, v_i) + m_i g = u_i, \quad (4.26b)$$

with  $f_i(\cdot)$  satisfying  $\|f_i(x_i, v_i)\| \leq \alpha_i \|v_i\|$ , for unknown positive constants  $\alpha_i$ ,  $\forall i \in \mathcal{N}$ . We also denote  $x := [x_1^\top, \dots, x_N^\top]^\top$ ,  $v := [v_1^\top, \dots, v_N^\top]^\top \in \mathbb{R}^{Nn}$ . The robots desire to navigate to their destination configurations  $x_{d_i}$ ,  $i \in \mathcal{N}$ . The proposed multi-robot scheme is based on a prioritized leader-follower coordination. Prioritization in multi-agent systems for navigation-type objectives has been employed in [?] and [?], where KRNF gain tuning-type methodologies for single integrator agents are developed. Moreover, [?] does not consider inter-agent collision avoidance and [?] does not assume static obstacles.

Intuitively, in the proposed prioritized leader-follower methodology, the leader robot, by appropriately choosing the offset  $\tau$ , “sees” the other robots as static obstacles and hence the overall scheme reduces to the one of Section 4.2.2. The workspace is assumed to satisfy Assumption 4.2 and we further impose extra conditions on the initial states and destinations:

**Assumption 4.3.** The workspace  $\mathcal{W}$ , obstacles  $\mathcal{O}_j$ ,  $j \in \mathcal{J}$ , and destinations  $x_{d_i}$ ,  $i \in \mathcal{N}$ , satisfy:

$$\begin{aligned} \|c_j - x_{d_i}\| &> r_{o_j} + r_i + 2r_M + \varepsilon, \forall i, j \in \mathcal{N} \times \mathcal{J} \\ \|x_{d_i} - x_{d_j}\| &> r_i + r_j + 2r_M + 2\varepsilon, \forall i, j \in \mathcal{N}, i \neq j \\ r_{\mathcal{W}} - \|x_{d_i}\| &> r_i + 2r_M + \varepsilon, \forall i \in \mathcal{N} \end{aligned}$$

whereas the initial positions satisfy:

$$\begin{aligned} \|c_j - x_i(t_0)\| &> r_{o_j} + r_i + 2r_M, \forall i, j \in \mathcal{N} \times \mathcal{J} \\ r_{\mathcal{W}} - \|x_i(t_0)\| &> r_i + 2r_M, \forall i \in \mathcal{N} \\ \|x_{d_i} - x_j(t_0)\| &> r_i + r_j + 2r_M + \varepsilon, \forall i, j \in \mathcal{N}, i \neq j, \end{aligned}$$

for an arbitrarily small positive constant  $\varepsilon$ ,  $\forall i \in \mathcal{N}, j \in \mathcal{J}$ , where  $r_M := \max_{i \in \mathcal{N}} \{r_i\}$ .

Loosely speaking, the aforementioned assumption states that the pairwise distances among obstacles, workspace boundary, initial conditions and final

destinations are large enough so that one robot can always navigate between them. Since the convergence of the agents to the their destinations is asymptotic, we incorporate the threshold  $\varepsilon$ , which is the desired proximity we want to achieve to the destination, as will be clarified in the sequel. Intuitively, since we cannot achieve  $x_i = x_{d_i}$  in finite time, the high-priority agents will stop once  $\|x_i - x_{d_i}\| = \varepsilon$ , which is included in the aforementioned conditions to guarantee the feasibility of the collision-free navigation for the lower-priority agents.

Similarly to the single-agent case, we can find a positive constant  $\bar{r}$  such that (4.5) hold as well as

$$\|c_j - x_i(t_0)\| > r_{o_j} + r_i + 2r_M + 2\bar{r}, \forall i, j \in \mathcal{N} \times \mathcal{J} \quad (4.27a)$$

$$r_W - \|x_i(t_0)\| > r_i + 2r_M + 2\bar{r}, \forall i \in \mathcal{N} \quad (4.27b)$$

$$\|c_j - x_{d_i}\| > r_{o_j} + r_i + 2r_M + \varepsilon + 2\bar{r}, \forall i, j \in \mathcal{N} \times \mathcal{J} \quad (4.27c)$$

$$\|x_{d_i} - x_{d_j}\| > r_i + r_j + 2r_M + 2\varepsilon + 2\bar{r}, \forall i, j \in \mathcal{N}, i \neq j \quad (4.27d)$$

$$\|x_{d_i} - x_j(t_0)\| > r_i + r_j + 2r_M + \varepsilon + 2\bar{r}, \forall i, j \in \mathcal{N}, i \neq j, \quad (4.27e)$$

$$r_W - \|x_{d_i}\| > r_i + 2r_M + \varepsilon + 2\bar{r}, \forall i \in \mathcal{N} \quad (4.27f)$$

We consider that the agents have a limited sensing range, defined by a radius  $\varsigma_i > 0$ ,  $i \in \mathcal{N}$ , and we assume that each agent  $i$  can sense the state of its neighbors, as stated next.

**Assumption 4.4.** Each agent  $i \in \mathcal{N}$  has a limited sensing radius  $\varsigma_i$ , satisfying  $\varsigma_i > \sqrt{\min(\bar{r}^2, \bar{r}_d)} + r_i + r_j + 2r_M + 2\bar{r}$ , with  $\bar{r}_d$  as defined in (4.8), and has access to  $(x_j, v_j)$ ,  $\forall j \in \{j \in \mathcal{N} : \|x_i - x_j\| \leq \varsigma_i\}$ .

Moreover, we consider that the destinations,  $x_{d_i}$ ,  $i \in \mathcal{N}$ , as well as the radii,  $r_i$ , are transmitted off-line to all the agents.<sup>2</sup> Consider now a prioritization of the agents, possibly based on some desired metric (e.g., distance to their destinations), which can be performed off-line and transmitted to all the agents. Our proposed scheme is based on the following algorithm. The agent with the highest priority is designated as the leader of the multi-agent system, indexed by  $i_L$ , whereas the rest of the agents are considered as the followers, defined by the index set  $\mathcal{N}_F := \mathcal{N} \setminus \{i_L\}$ . The followers and leader employ a control protocol that has the same structure as the one of Section 4.2.2. The key difference here lies in the definition of the free space for followers and leaders. We define first the sets

$$\bar{\mathcal{W}}_{i_L} := \{z = [z_1^\top, \dots, z_N^\top]^\top \in \mathbb{R}^{Nn} : \|z_{i_L}\| < r_W - r_{i_L}\},$$

$$\bar{\mathcal{O}}_{i_L, j} := \{z = [z_1^\top, \dots, z_N^\top]^\top \in \bar{\mathcal{W}}_{i_L} : \|z_i - c_j\| \leq r_{o_j} + r_i\}, \forall j \in \mathcal{J}$$

$$\mathcal{C}_{i_L} := \{z = [z_1^\top, \dots, z_N^\top]^\top \in \bar{\mathcal{W}}_{i_L} : \|z_{i_L} - z_j\| \leq r_{i_L} + r_j, \forall j \in \mathcal{N} \setminus \{i_L\}\},$$

---

<sup>2</sup>This implies that the agents can compute  $r_M$  offline.

which correspond to the leader agent, as well as the follower sets

$$\begin{aligned}\bar{\mathcal{W}}_i &:= \{z = [z_1^\top, \dots, z_N^\top]^\top \in \mathbb{R}^{Nn} : \|z_i\| < r_{\mathcal{W}} - r_i - 2r_M - 2\bar{r}\} \\ \bar{\mathcal{O}}_{i,j} &:= \{z = [z_1^\top, \dots, z_N^\top]^\top \in \bar{\mathcal{W}}_i : \|z_i - c_j\| \leq r_{o_j} + r_i + 2r_M + 2\bar{r}\}, \forall j \in \mathcal{J} \\ \mathcal{C}_i &:= \{z = [z_1^\top, \dots, z_N^\top]^\top \in \bar{\mathcal{W}}_i : \|z_i - z_{i_L}\| \leq r_i + r_{i_L}, \\ &\quad \|z_i - z_j\| \leq r_i + r_j + 2r_M + 2\bar{r}, \forall j \in \mathcal{N} \setminus \{i_L, i\}, \\ &\quad \|z_i - x_{d_j}\| \leq r_i + r_j + 2r_M + 2\bar{r} + \varepsilon, \forall j \in \mathcal{N} \setminus \{i\}\},\end{aligned}$$

$\forall i \in \mathcal{N}_{\mathcal{F}}$ . The free space for the agents is defined then as

$$\mathcal{F}_i := \bar{\mathcal{W}}_i \setminus \left\{ \left( \bigcup_{j \in \mathcal{J}} \bar{\mathcal{O}}_{i,j} \right) \bigcup \mathcal{C}_i \right\}, \forall i \in \mathcal{N}.$$

It can be verified that, in view of (4.27), the sets  $\mathcal{F}_i$  are nonempty and  $x(t_0) \in \mathcal{F}_M := \bigcap_{i \in \mathcal{N}} \mathcal{F}_i$ .

The main difference lies in the fact that the follower agents aim to keep a larger distance from each other, the obstacles, and the workspace boundary than the leader agent, and in particular, a distance enhanced by  $2r_M + 2\bar{r}$ . In that way, the leader agent will be able to choose an appropriate constant  $\tau$  (as in the single-agent case of Section 4.2.2) so that it is influenced at each time instant only by one of the obstacles/followers, and will be also able to navigate among the obstacles/followers. Note that the followers are required to stay away also from other agents' destinations, since a potential local minimum in such configurations can prevent the leader agent from reaching its goal. We provide next the mathematical details of the aforementioned reasoning.

Consider the leader distances  $d_{i_L, o_k}, d_{i_L, j}, d_{i_L, o_0} : \mathcal{F}_{i_L} \rightarrow \mathbb{R}_{\geq 0}$  as

$$\begin{aligned}d_{i_L, o_k} &:= d_{i_L, o_k}(x) := \|x_{i_L} - c_k\|^2 - (r_{i_L} + r_{o_k})^2, \forall k \in \mathcal{J} \\ d_{i_L, j} &:= d_{i_L, j}(x) := \|x_{i_L} - x_j\|^2 - (r_{i_L} + r_j)^2, \forall j \in \mathcal{N}_{\mathcal{F}} \\ d_{i_L, o_0} &:= d_{i_L, o_0}(x) := (r_{\mathcal{W}} + r_{i_L})^2 - \|x_{i_L}\|^2\end{aligned}$$

and the follower distances  $d_{i, o_k}, d_{i, i_L}, d_{i, j}, d_{i, d_j}, d_{i, o_0} : \mathcal{F}_i \rightarrow \mathbb{R}_{\geq 0}$  as

$$\begin{aligned}d_{i, o_k} &:= d_{i, o_k}(x) := \|x_i - c_k\|^2 - (r_i + r_{o_k} + 2r_M + 2\bar{r})^2, \forall k \in \mathcal{J} \\ d_{i, i_L} &:= d_{i, i_L}(x) := \|x_i - x_{i_L}\|^2 - (r_i + r_{i_L})^2 = d_{i_L, i}(x) \\ d_{i, j} &:= d_{i, j}(x) := \|x_i - x_j\|^2 - (r_i + r_j + 2r_M + 2\bar{r})^2, \forall j \in \mathcal{N}_{\mathcal{F}} \\ d_{i, d_j} &:= d_{i, d_j}(x) := \|x_i - x_{d_j}\|^2 - (r_i + r_j + 2r_M + 2\bar{r} + \varepsilon)^2, \forall j \in \mathcal{N} \setminus \{i\} \\ d_{i, o_0} &:= d_{i, o_0}(x) := (r_{\mathcal{W}} - r_i - 2r_M - 2\bar{r})^2 - \|x_i\|^2,\end{aligned}$$

$\forall i \in \mathcal{N}_F$ . Note that  $d_{i,j}(x) = d_{j,i}(x)$ ,  $\forall i, j \in \mathcal{N}_F$ , with  $i \neq j$  and also that  $x \in \mathcal{F}_M$  is equivalent to all the aforementioned distances being positive.

Let now functions  $\beta, \beta_i, i \in \mathcal{N}$ , that satisfy the properties of Definition 4.1, as well as the respective constants  $\tau, \tau_i$ , such that  $\beta'(z) = \beta''(z) = 0$ ,  $\forall z \geq \tau$ ,  $\beta'_i(z) = \beta''_i(z) = 0$ ,  $\forall z \geq \tau_i$ ,  $i \in \mathcal{N}$ . The 2nd-order navigation functions for the agents are now defined as  $\phi_i := \phi_i(x) : \mathcal{F}_i \rightarrow \mathbb{R}_{\geq 0}$ ,  $\forall i \in \mathcal{N}$ , with

$$\begin{aligned}\phi_i(x) &:= k_{1i} \|x_i - x_{d_i}\|^2 + k_{2i} \left( b_{1i}(x) + b_{2i}(x) + k_{f_i} b_{3i}(x) \right) \\ b_{1i} &:= b_{1i}(x) := \sum_{j \in \bar{\mathcal{J}}} \beta_i(d_{i,o_j}(x)) \\ b_{2i} &:= b_{2i}(x) := \sum_{j \in \mathcal{N} \setminus \{i\}} \beta(d_{i,j}(x)) \\ b_{3i} &:= b_{3i}(x) := \sum_{j \in \mathcal{N} \setminus \{i\}} \beta_i(d_{i,d_j}(x)),\end{aligned}$$

and  $k_{f_i} = 0$ ,  $k_{f_i} = 1$ ,  $\forall i \in \mathcal{N}_F$ . Note that the robotic agents can choose independently their  $\tau_i$ ,  $i \in \mathcal{N}$ , that concerns the collision avoidance with the obstacles and the workspace boundary. The pair-wise inter-agent distances, however, are required to be the same and hence the same  $\beta$  (and hence  $\tau$ ) is chosen (see the terms  $b_{2i}(x)$  in  $\phi_i(x)$ ), which can, nevertheless, be done off-line. To achieve convergence of the leader to its destination, we choose  $\tau$  and  $\tau_{i_L}$  as in Section 4.2.2, i.e.,  $\tau, \tau_{i_L} \in (0, \min\{\bar{r}^2, \bar{r}_d\})$ . Regarding the ability of the agents to sense each other when  $d_{i,j}(x) < \tau$ , it holds that

$$\begin{aligned}d_{i,j}(x) < \tau &\Leftrightarrow \|x_i - x_j\|^2 \leq \tau + (r_i + r_j + 2r_M + 2\bar{r})^2 \Rightarrow \\ \|x_i - x_j\| &\leq \sqrt{\tau} + r_i + r_j + 2r_M + 2\bar{r} \Rightarrow \\ \|x_i - x_j\| &\leq \sqrt{\min\{\bar{r}^2, \bar{r}_d\}} + r_i + r_j + 2r_M + 2\bar{r} < \varsigma_i,\end{aligned}$$

$\forall i, j \in \mathcal{N}$ ,  $i \neq j$ , as dictated by Assumption 4.4.

The control protocol follows the same structure as the single-agent case presented in Section 4.2.2. In particular, we define the reference velocity for each agent as  $v_{d_i} : \mathcal{F}_i \rightarrow \mathbb{R}^n$ , with

$$v_{d_i} := v_{d_i}(x) := -\nabla_{x_i} \tilde{\phi}_i(x), \quad (4.28)$$

where  $\tilde{\phi}_i : \mathcal{F}_i \rightarrow \mathbb{R}_{\geq 0}$  is the slightly modified function:

$$\tilde{\phi}_i(x) := k_{1i} \|x_i - x_{d_i}\|^2 + k_{2i} \left( b_{1i}(x) + 2b_{2i}(x) + k_{f_i} b_{3i}(x) \right).$$

The need for modification of  $\phi_i$  to  $\tilde{\phi}_i$  stems from the differentiation of the terms  $b_{2i}$ , as will be clarified in the subsequent analysis.

The control law is now designed as  $u_i : \mathcal{F}_i \times \mathbb{R}^{Nn+2} \rightarrow \mathbb{R}^n$ , with

$$u_i := u_i(x, v, \hat{m}_i, \hat{\alpha}_i) := -k_{\phi_i} \nabla_{x_i} \tilde{\phi}_i(x) + \hat{m}_i(\dot{v}_{d_i} + g) - \left( k_{v_i} + \frac{3}{2} \hat{\alpha}_i \right) e_{v_i}, \quad (4.29)$$

$\forall i \in \mathcal{N}$ ;  $k_{\phi_i}$ ,  $k_{v_i}$  are positive constants,  $e_{v_i}$  are the velocity errors  $e_{v_i} := v_i - \dot{v}_{d_i}$ , and  $\hat{m}_i$ ,  $\hat{\alpha}_i$  denote the estimates of  $m_i$  and  $\alpha_i$ , respectively, by agent  $i$ , evolving according to

$$\dot{\hat{m}}_i := -k_{m_i} e_{v_i}^\top (\dot{v}_{d_i} + g) \quad (4.30a)$$

$$\dot{\hat{\alpha}}_i := k_{\alpha_i} \|e_{v_i}\|^2, \quad (4.30b)$$

with  $k_{m_i}$ ,  $k_{\alpha_i}$  positive gain constants,  $\hat{\alpha}_i(t_0) \geq 0$ , and arbitrary initial conditions  $\hat{m}_i(t_0)$ ,  $\forall i \in \mathcal{N}$ . We further denote  $\hat{m} := [\hat{m}_1, \dots, \hat{m}_N]^\top$ ,  $\hat{\alpha} := [\hat{\alpha}_1, \dots, \hat{\alpha}_N]^\top \in \mathbb{R}^N$ .

As presented below, the leader agent will converge to its destination from almost all initial conditions that satisfy Assumption 4.3, whereas the followers might get stuck in local minima. Once the leader reaches  $\varepsilon$ -close to its destination (where  $\varepsilon$  was introduced in Assumption 4.3), it switches off its control and the next robotic agent in the priority list becomes the leader. We assume that once an agent reaches its goal, it can broadcast this information for the next agent in priority. This occurs iteratively until all the robotic agents reach their destinations. The following theorem considers the convergence of a leader to its destination.

**Theorem 4.3.** Consider  $N$  robots operating in  $\mathcal{W}$ , subject to the uncertain 2nd-order dynamics (4.26), and a leader  $i_L$ . Under Assumptions 4.1-4.4, the control protocol (4.28)-(4.30) guarantees collision avoidance between the agents and the agents and obstacles/workspace boundary as well as convergence of  $x_{i_L}$  to  $x_{d_{i_L}}$  from almost all initial conditions  $(x(t_0), v(t_0), \hat{m}(t_0), \hat{\alpha}(t_0)) \in \mathcal{F}_M \times \mathbb{R}^{N(n+1)} \times \mathbb{R}_{\geq 0}^N$ , given sufficiently small  $\tau$ ,  $\tau_{i_L}$ , and that  $k_{\phi_i} > \frac{\alpha_i}{2}$ ,  $i \in \mathcal{N}$ . Moreover, all closed loop signals remain bounded,  $\forall t \geq t_0$ .

*Proof.* We prove first the avoidance of collisions by considering the function

$$V_M := \sum_{i \in \mathcal{N}} \left\{ k_{\phi_i} \phi_i + \frac{m_i}{2} \|e_{v_i}\|^2 + \frac{3}{4k_{\alpha_i}} \tilde{\alpha}_i^2 + \frac{1}{2k_{m_i}} \tilde{m}_i^2 \right\}.$$

Since  $x(t_0) \in \mathcal{F}_M$ ,  $V_M(t_0)$  is bounded. Differentiation of  $V_M$  yields, after using the property

$$\sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{N} \setminus \{i\}} (x_i - x_j)^\top (v_i - v_j) = 2 \sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{N} \setminus \{i\}} (x_i - x_j)^\top v_i,$$

$$\begin{aligned}
\dot{V}_M &= \sum_{i \in \mathcal{N}} \left\{ 2k_{\phi_i} k_{1i} (x_i - x_{d_i}) - 2k_{\phi_i} k_{2i} \left( \beta'_i(d_{i,o_0})x - \sum_{k \in \mathcal{J}} \beta'_i(d_{i,o_k})(x_i - c_k) \right. \right. \\
&\quad \left. \left. - 2 \sum_{j \in \mathcal{N} \setminus \{i\}} \beta'(d_{i,j})(x_i - x_j) - \sum_{j \in \mathcal{N} \setminus \{i\}} k_{f_i} \beta'_i(d_{i,d_j})(x_i - x_{d_j}) \right)^\top v_i \right. \\
&\quad \left. + e_{v_i}^\top (u_i - f_i(x_i, v_i) - m_i g - m_i \dot{v}_{d_i}) + \frac{3}{2k_{\alpha_i}} \tilde{\alpha}_i \dot{\alpha}_i + \frac{1}{2k_{m_i}} \tilde{m}_i \dot{m}_i \right\} \\
&\leq \sum_{i \in \mathcal{N}} \left\{ k_{\phi_i} \nabla_{x_i} \tilde{\phi}_i(x)^\top v_i + e_{v_i}^\top (u_i - m_i(g + \dot{v}_{d_i})) + \frac{3}{2} \alpha_i \|e_{v_i}\| \|v_i\| \right. \\
&\quad \left. + \frac{3}{2} \tilde{\alpha}_i \|e_{v_i}\|^2 - \tilde{m}_i e_{v_i}^\top (\dot{v}_{d_i} + g) \right\},
\end{aligned}$$

which, by using  $v_i = e_{v_i} + v_{d_i}$  and substituting the control and adaptation laws (4.12), (4.30), becomes

$$\dot{V}_M \leq - \sum_{i \in \mathcal{N}} \left\{ \left( k_{\phi_i} - \frac{\alpha_i}{2} \right) \|\nabla_{x_i} \tilde{\phi}_i(x)\|^2 + k_{v_i} \|e_{v_i}\|^2 \right\} \leq 0,$$

and hence,  $V_M(t) \leq V(t_0)$ , which implies the boundedness of all closed-loop signals as well as that collisions between the agents and the agents and obstacles/workspace boundary are avoided  $\forall t \geq t_0$ . Moreover, following similar arguments as in the proof of Theorem 4.1, we conclude that  $\lim_{t \rightarrow \infty} \|\nabla_{x_i} \phi_i(x(t))\| = \lim_{t \rightarrow \infty} \|e_{v_i}(t)\| = \lim_{t \rightarrow \infty} \|v_i(t)\| = \lim_{t \rightarrow \infty} \|\dot{v}_i(t)\| = 0$ ,  $\forall i \in \mathcal{N}$ . For the followers  $\mathcal{N}_F$ , depending on the choice of  $\tau_i$ ,  $i \in \mathcal{N}_F$ , the critical point  $\nabla_{x_i} \tilde{\phi}_i(x(t)) = 0$  might either correspond to their destination  $x_{d_i}$  or a local minimum. In any case, it holds that  $x(t) \in \mathcal{F}_M$ ,  $\forall t \geq t_0$ , and hence, for all the followers  $i \in \mathcal{N}_F$ ,

$$\|x_i(t) - c_k\| > r_i + r_{o_k} + 2r_M + 2\bar{r}, \forall k \in \mathcal{J} \quad (4.31a)$$

$$\|x_i(t) - x_j(t)\| > r_i + r_j + 2r_M + 2\bar{r}, \forall j \in \mathcal{N}_F \setminus \{i\} \quad (4.31b)$$

$$r_{\mathcal{W}} - \|x_i\| > r_i + 2r_M + 2\bar{r}, \quad (4.31c)$$

$$\|x_i(t) - x_{d_i}\| > r_i + r_j + 2r_M + 2\bar{r} + \varepsilon, \forall j \in \mathcal{N} \setminus \{i\}, \quad (4.31d)$$

$\forall t > t_0$ . Therefore, since  $\lim_{t \rightarrow \infty} \|v_i(t)\| = \lim_{t \rightarrow \infty} \|\dot{v}_i(t)\| = 0$ ,  $\forall i \in \mathcal{N}$ , the multi-robot case reduces to the single-robot case of Section 4.2.2, where the followers resemble static obstacles. Note that the obstacle constraints (4.5) are always satisfied by the followers (see (4.31a)-(4.31c)); (4.31d) implies that the configuration that corresponds to the leader destination, i.e.,  $[x_1^\top, \dots, x_{i_L-1}^\top, x_{d_{i_L}}^\top, x_{i_L+1}^\top, \dots, x_N^\top]^\top$ , belongs always in its free space  $\mathcal{F}_{i_L}$ . Hence, by choosing sufficiently small  $\tau, \tau_{i_L}$  in the interval  $(0, \min(\bar{r}^2, \bar{r}_d))$ ,

with  $\bar{r}_d$  as defined in (4.8), we guarantee the safe navigation of  $x_{i_L}$  to  $x_{d_{i_L}}$  from almost all initial conditions, as in Section 4.2.2.  $\square$

When the current leader  $i_L$  reaches  $\varepsilon$ -close to its goal, at a time instant  $t_{i_L}$ <sup>3</sup>, it broadcasts this information to the other agents, switches off its control and remains immobilized, considered hence as a static obstacle with center  $c_{M+1} := x_{i_L}(t_{i_L})$  and radius  $r_{M+1}$  by the rest of the team. Note that  $\|c_{M+1} - x_{d_{i_L}}\| \leq \varepsilon$  and hence, in view of (4.27),  $\|c_j - c_{M+1}\| > r_{o_j} + r_{i_L} + 2r_M + 2\bar{r}$ ,  $\forall j \in \mathcal{J}$ , and  $r_W - \|c_{M+1}\| > r_{i_L} + 2r_M + 2\bar{r}$ , satisfying the obstacle spacing properties (4.5). The next agent  $i'_L \in \tilde{\mathcal{N}} := \mathcal{N} \setminus \{i_L\}$  in priority is then assigned as a leader for navigation, and we redefine the sets

$$\begin{aligned}\tilde{\mathcal{O}}_{i'_L, j} &:= \{q \in \bar{\mathcal{W}}_{i'_L} : \|q_i - c_j\| \leq r_{o_j} + r_i\}, \forall j \in \tilde{\mathcal{J}} \\ \tilde{\mathcal{C}}_{i'_L} &:= \{q \in \bar{\mathcal{W}}_{i'_L} : \|q_{i'_L} - q_j\| \leq r_{i'_L} + r_j, \forall j \in \tilde{\mathcal{N}} \setminus \{i'_L\}\}, \\ \tilde{\mathcal{O}}_{i, j} &:= \{q \in \bar{\mathcal{W}}_i : \|q_i - c_j\| \leq r_{o_j} + r_i + 2r_M + 2\bar{r}\}, \forall j \in \tilde{\mathcal{J}} \\ \tilde{\mathcal{C}}_i &:= \{q \in \bar{\mathcal{W}}_i : \|q_i - q_{i'_L}\| \leq r_i + r_{i'_L}, \\ &\quad \|q_i - q_j\| \leq r_i + r_j + 2r_M + 2\bar{r}, \forall j \in \tilde{\mathcal{N}} \setminus \{i'_L, i\}, \\ &\quad \|q_i - x_{d_j}\| \leq r_i + r_j + 2r_M + 2\bar{r} + \varepsilon, \forall j \in \tilde{\mathcal{N}} \setminus \{i\}\},\end{aligned}$$

$\forall i \in \tilde{\mathcal{N}} \setminus \{i'_L\}$ , where  $\tilde{\mathcal{J}} := \mathcal{J} \cup \{M+1\}$ , to account for the new obstacle  $M+1$ . The new free space is

$$\tilde{\mathcal{F}}_i := \bar{\mathcal{W}}_i \setminus \left\{ \left( \bigcup_{j \in \tilde{\mathcal{J}}} \tilde{\mathcal{O}}_{i, j} \right) \cup \tilde{\mathcal{C}}_i \right\}, \forall i \in \tilde{\mathcal{N}}$$

and, in view of (4.31), one can conclude that  $x_{i'_L}(t_{i_L}) \in \tilde{\mathcal{F}}_{i'_L}$ ,  $x_i(t_{i_L}) \in \tilde{\mathcal{F}}_i \forall i \in \tilde{\mathcal{N}} \setminus \{i'_L\}$ . Therefore, the application of Theorem 4.3 with  $t_{i_L}$  as  $t_0$  and agent  $i'_L$  as leader guarantees its navigation  $\varepsilon$ -close to  $x_{d_{i'_L}}$ . Applying iteratively the aforementioned reasoning, we guarantee the successful navigation of all the agents. More specifically, we initially set off-line the priorities of the agents based on a desired metric (e.g., distance to the goal), and set  $i_L$  as the top priority agent. Then the following procedure is iterated until all agents have reached their goals. The agents apply the control protocol (4.28)-(4.30). When the leader agent satisfies  $\|x_{i_L} - x_{d_{i_L}}\| \leq \varepsilon$ , it switches off its control, broadcasts this information to all other agents, and the next leader is chosen as the next agent in priority. Therefore, in view of Theorem 4.3, all agents will eventually reach  $\varepsilon$ -close to their destinations. This is

<sup>3</sup>Note that the proven asymptotic stability of Theorem 4.3 guarantees that this will occur in finite time.

illustrated in Algorithm 1, which is run by each agent separately. The algorithm receives as input the agent index and destination  $i$ ,  $x_{d_i}$ , respectively, as well as the priority vector  $\text{Pr}$ , which have been set a priori. Next, depending on the priority (lines 3, 4), agent  $i$  applies the control algorithm (4.28)-(4.30) (line 8). In case agent  $i$  has the top priority and reaches its goal, it broadcasts to the other agents that it has arrived and exits the loop (lines 5-7). Finally, the agents are equipped with a callback function `Receive` that continuously checks whether some agent  $j \in \mathcal{N} \setminus \{i\}$  broadcasts the arrival to its destination, so that they update accordingly the priority vector  $\text{Pr}$  (lines 9, 10). Note that the latter is a synchronous procedure and the priority variable  $\text{Pr}$  is always the same for all agents.

---

**Algorithm 1** Hybrid Control Strategy for Agent  $i$ 


---

```

1: Input:  $i$ ,  $x_{d_i}$ ,  $\text{Pr}$ 
2: while True do
3:   if  $\text{Pr}[i] > \text{Pr}[j], \forall j \in \mathcal{N} \setminus \{i\}$  then
4:      $i_L \leftarrow i$ ;
5:     if  $\|x_i - x_{d_i}\| \leq \varepsilon$  then
6:       Broadcast("arrived");
7:       break;
8:     Apply (4.28)-(4.30)
9:     if Receive("Arrived",  $j$ ) then
10:     $\text{Pr} \leftarrow \text{Update}(\text{Pr}, j)$ ;

```

---

As a final remark, note that  $\varepsilon$  can be arbitrarily small, achieving thus practical convergence of the agents to their destinations  $x_{d_i}$ ,  $i \in \mathcal{N}$ .

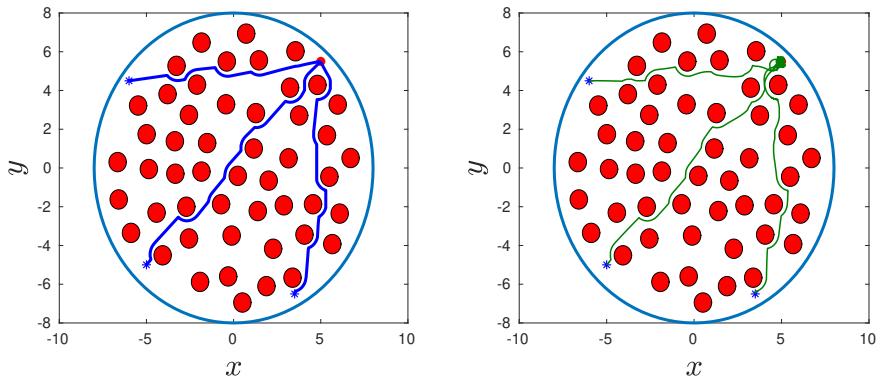
#### 4.2.6 Simulation Results

This section verifies the theoretical findings of Sections 4.2.2-4.2.5 via computer simulations.

##### Sphere worlds

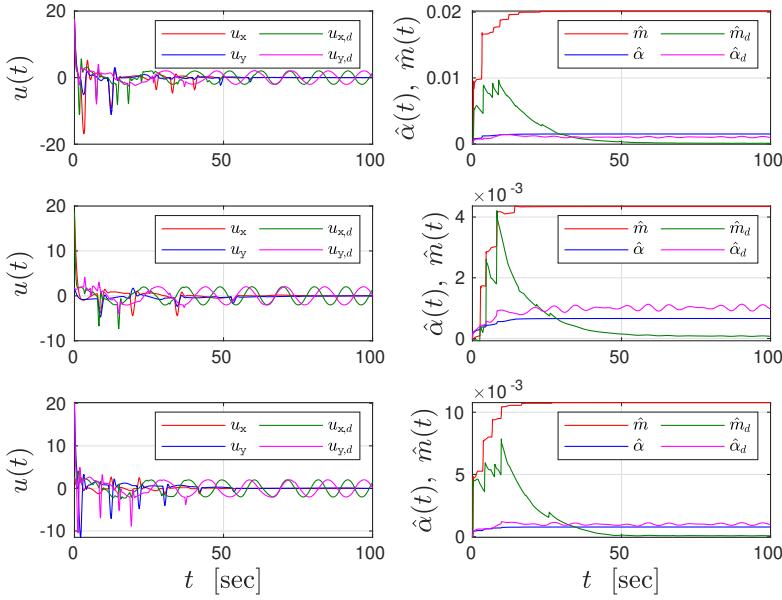
We consider first a 2D workspace on the horizontal plane with  $r_W = 8$ , populated with  $M = 50$  randomly placed obstacles, whose radius, enlarged by the robot radius, is  $\bar{r}_{o_j} = 0.5$ ,  $\forall j \in \mathcal{J}$ , as depicted in Fig. 4.1. The mass, and function  $f(x, v)$ , both *unknown* to the robotic agent, are taken as  $m = 1$ , and  $f(x, v) = \frac{\alpha}{16} \sin(0.5(x_x + x_y)) F(v)v$ , with  $F(v) = \text{diag}\{\exp(-\text{sgn}(v_i)v_i) + 1\}_{i \in \{x, y\}}$ , and  $\alpha = 10$ , where we denote  $(x_x, x_y) = x$ ,  $(v_x, v_y) = v$ . We choose the goal position as  $x_d = (5, 5)$ , which the robot aims to converge to from 3 different initial positions, namely  $x(0) = -(5, 5), (-6, 4.5)$ , and

(3.5, -7). We choose a variation of (4.7) for  $\beta$  with  $\tau = \bar{r}^2$  and  $\bar{\beta} = 100$ . The control gains are chosen as  $k_1 = 0.04$ ,  $k_2 = 5$ ,  $k_v = 20$ ,  $k_\phi = 1$ , and  $k_m = k_\alpha = 0.01$ . The results for  $t \in [0, 100]$  seconds are depicted in Figs. 4.3, 4.4. Fig. 4.3 (left) shows that the robot navigates to its destination without any collisions, and Fig. 4.4 depicts the input and adaptation signals  $u(t)$ ,  $\hat{a}(t)$ ,  $\hat{m}(t)$ . In addition, note that the fact that  $\alpha > 2$  does not affect the performance of the proposed control protocol and hence we can verify that the condition  $k_\phi > \frac{\alpha}{2}$  is only sufficient and not necessary. Moreover, in order to verify the results of Section 4.2.2, we add a bounded time-varying disturbance vector  $d(x, v, t) = d(t) := 2 [\sin(0.5t + \frac{pi}{3}), \cos(0.4t - \frac{\pi}{4})]^\top \in \mathbb{R}^2$  and we choose the extra control gains as  $\sigma_m = \sigma_\alpha = 0.1$ . The results are depicted in Fig. 4.3 (right), which shows the collision-free navigation of the agent to a set close to  $x_d$ , and Fig. 4.4, which shows the input and adaptation signals  $u(t)$ ,  $\hat{a}(t)$ ,  $\hat{m}(t)$ .



**Figure 4.3:** The resulting trajectories  $x(t)$ ,  $t \in [0, 100]$  seconds, from the initial points  $-(5, 5)$ ,  $(-6, 4.5)$ , and  $(3.5, -7)$  to the destination  $(5, 5)$ . Left: without any disturbances. Right: with bounded disturbance  $d(x, v, t)$ .

Next, we consider a 3D workspace with  $r_{\mathcal{W}} = 8$ , populated with  $M = 150$  randomly placed obstacles, whose radius, enlarged by the robot radius, is  $\bar{r}_{o_j} = 0.5$ ,  $\forall j \in \mathcal{J}$ ;  $f(x, v)$  and  $m$  as well as the  $\beta$  functions and control gains are chosen as in the 2D scenario. We choose the goal position as  $x_d = (4, 4, 4)$ , which the robot aims to converge to from 3 different initial positions, namely  $x(0) = -(4, 4, 4)$ ,  $(-4, 4, -4)$ , and  $(-4, -4, 4)$ . The parameter  $\bar{r}$  is chosen as  $\bar{r} = 0.75$ . The robot navigation as well as the input and adaptation signals  $u(t)$ ,  $\hat{a}(t)$ ,  $\hat{m}(t)$  are depicted in Figs. 4.5, and 4.6 for  $t \in [0, 100]$  seconds. Note that the robot navigates to its destination without any collisions and that  $\hat{m}$  converges to  $m$ , as predicted by the theoretical results.

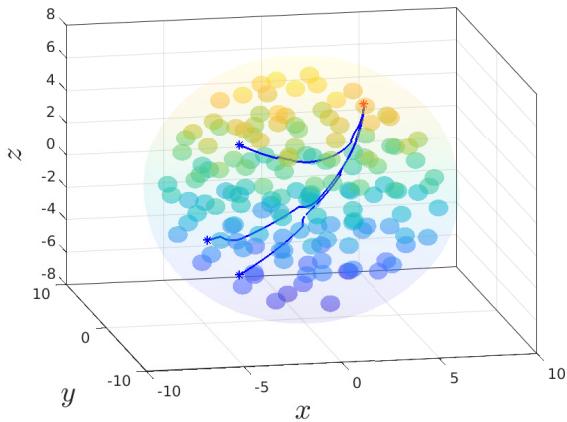


**Figure 4.4:** Left: The resulting input signals  $u(t) = (u_x(t), u_y(t))$ ,  $t \in [0, 100]$  seconds, for the 2D trajectories of Fig. 4.3. Right: The resulting adaptation signals  $\hat{\alpha}(t)$ ,  $\hat{m}(t)$ ,  $t \in [0, 100]$  seconds, for the 2D trajectories of Fig. 4.3. The extra subscript  $d$  corresponds to the model where a bounded disturbance vector  $d(x, v, t)$  was included.

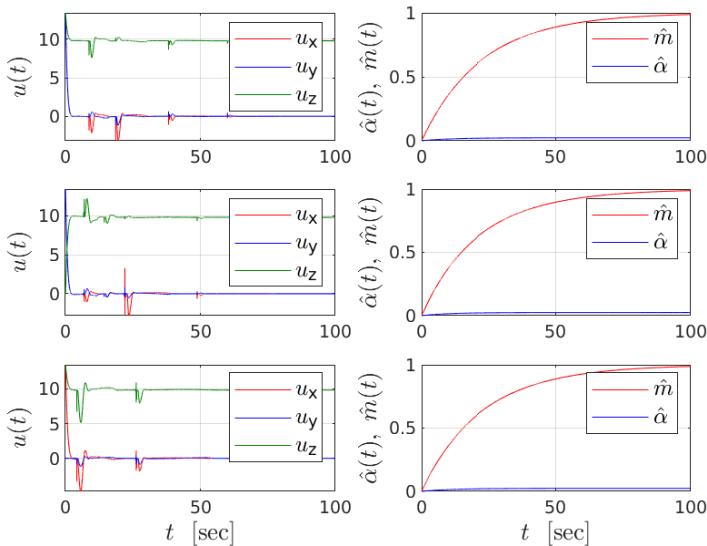
## Star worlds

Next, we illustrate the performance of the control protocol of Section 4.2.4 in a 2D and a 3D star-world. We first consider the 2D workspace shown in Fig. 4.2, with  $r_W = 8$ , which contains 2 star-shaped obstacles, centered at  $(-3, -3)$  and  $(0, 1)$ , respectively. The mass  $m$  and function  $f(x, v)$  are given as in the sphere-world case, with  $\alpha = 1$ . In order to transform the workspace to a sphere world, we employ the transformation proposed in [? ]. In the transformed sphere world, we choose  $\bar{r} = 4$  and  $\bar{r}_{o_j} = 0.5$ , whereas the function  $\beta$  is chosen as in the sphere-world case. The initial and goal position are selected as  $x(0) = (-5, -5)$  and  $x_d = (3, 4)$ , respectively, and the control gains as  $k_1 = 0.04$ ,  $k_2 = .2$ ,  $k_v = 20$ ,  $k_\phi = 1$ , and  $k_m = k_\alpha = 0.01$ . The results are depicted in Figs. 4.7 and 4.8a, for  $t \in [0, 500]$  seconds. More specifically, 4.7 shows the resulting trajectory, both in the original star world as well as in the transformed sphere world, and Fig. 4.8a depicts the resulting control input  $u(t)$  and the adaptation signals  $\hat{\alpha}(t)$ ,  $\hat{m}(t)$ .

Next, we consider a 3D workspace, with 2 star-shaped obstacles, centered at  $(-4, -4, -2)$ ,  $(1, 2, 2)$ , similar to the previous 2D star-shaped



**Figure 4.5:** The resulting trajectories  $x(t)$ ,  $t \in [0, 100]$  seconds, from the initial points  $-(5, 5)$ ,  $(-6, 4.5)$ , and  $(3.5, -7)$  to the destination  $(5, 5)$ .



**Figure 4.6:** Left: The resulting input signals  $u(t) = (u_x(t), u_y(t), u_z(t))$ ,  $t \in [0, 100]$  seconds, for the 3D trajectories of Fig. 4.5. Right: The resulting adaptation signals  $\hat{\alpha}(t)$ ,  $\hat{m}(t)$ ,  $t \in [0, 100]$  seconds, for the 3D trajectories of Fig. 4.5.

workspace, and  $r_{\mathcal{W}} = 12$ . By setting the initial and goal configurations at  $(-5.1, -5.2, -5)$ , and  $(3, 4, 4)$ , respectively, and all the parameters and control gains as in the 2D counterpart, we obtain the results shown in Figs. 4.9 and 4.8b, for 200 seconds; 4.9 shows the safe robot navigation to the goal and Fig. 4.8b depicts the evolution of the control and adaptation signals  $u(t)$ ,  $\hat{a}(t)$ , and  $\hat{m}(t)$ .

### Multi-Agent case

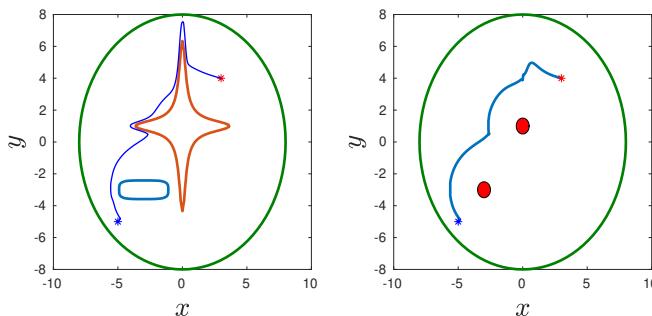
Finally, we use the control scheme of Section 4.2.5 in a multi-agent scenario. We consider 20 agents in a 2D workspace of  $r_{\mathcal{W}} = 120$ , populated with 70 obstacles. The agents and obstacles are randomly initialized to satisfy the conditions of the free space of Section 4.2.5 (see Fig. 4.10). The radius of the agents and the obstacles is chosen as  $r_i = r_{o_j} = 2$ ,  $\forall i \in \mathcal{N}, j \in \mathcal{J}$ , and the sensing radius of the agents is taken as  $\varsigma_i = 20$ ,  $\forall i \in \mathcal{N}$ . The functions  $\beta$ ,  $\beta_i$  are chosen as in the previous subsections, and we also choose  $\bar{r} = 4$ ,  $\varepsilon = 0.1$ . The results are depicted in Figs. 4.11–4.13 for 870 seconds. More specifically, Fig. 4.11 shows the convergence of the distance errors  $\|x_i(t) - x_{d_i}\|$  to zero,  $\forall i \in \mathcal{N}, t \in [0, 870]$ , and Fig. 4.12 depicts the trajectories  $x_i(t)$  of the agents in the workspace,  $\forall i \in \mathcal{N}, t \in [0, 870]$ , from which it is clear that there is no collision with the workspace boundary. Finally, Fig. 4.13 shows the minimum of the distances  $\|x_i(t) - x_j(t)\| - 2r$ ,  $\forall i, j \in \mathcal{N}, i \neq j$ , and  $\|x_i(t) - c_j\| - 2r$ ,  $\forall i \in \mathcal{N}, j \in \mathcal{J}$ , defined as

$$\beta_{\min}(t) := \min \left\{ \min_{i, j \in \mathcal{N}, i \neq j} \{\|x_i(t) - x_j(t)\| - 2r\}, \min_{(i, j) \in \mathcal{N} \times \mathcal{J}} \{\|x_i(t) - c_j\| - 2r\} \right\},$$

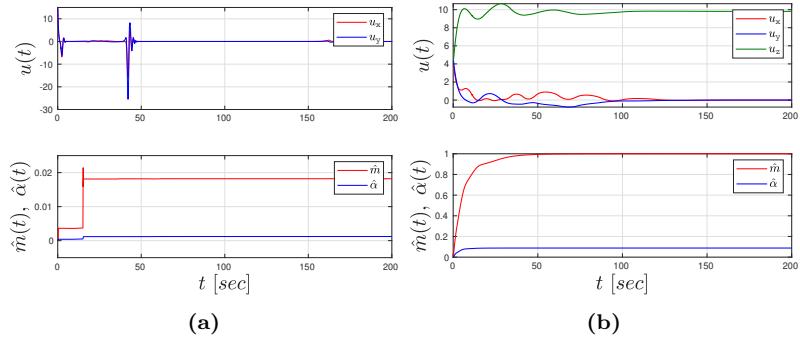
which stays strictly positive,  $\forall t \in [0, 870]$ , implying that collisions are avoided.

A video illustrating the multi-robot case can be found on

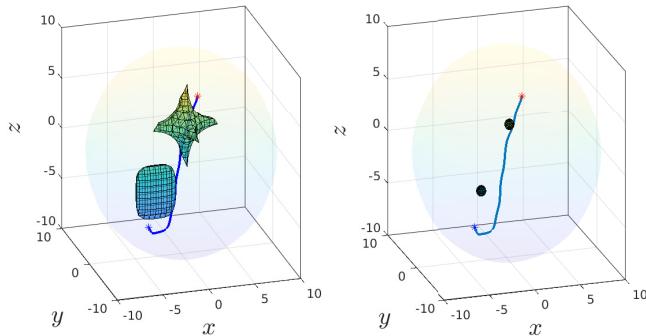
<https://vimeo.com/393443782>.



**Figure 4.7:** Left: The resulting trajectory  $x(t)$ ,  $t \in [0, 500]$  seconds, from the initial points  $-(5, 5)$  to the destination  $(3, 4)$ , in the 2D star world workspace. Right: The respective trajectory in the transformed sphere world.



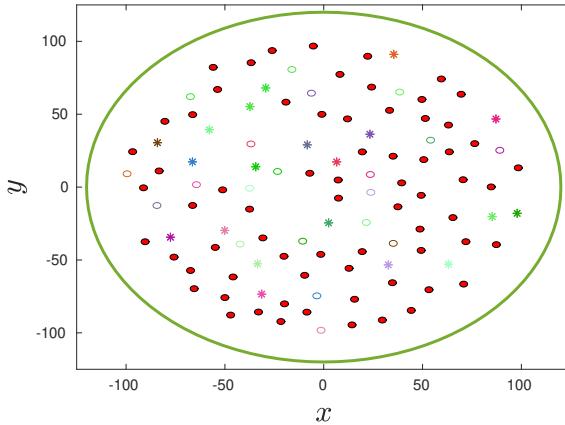
**Figure 4.8:** The input and adaptations signals  $u(t)$ ,  $\hat{\alpha}(t)$ ,  $\hat{m}(t)$ , for the 2D (a) and 3D (b) star world workspaces, for  $[0, 500]$  and  $[0, 200]$  seconds, respectively.



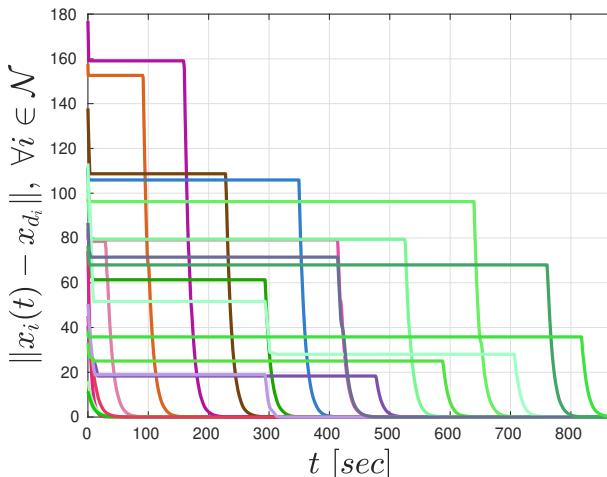
**Figure 4.9:** Left: The resulting trajectory  $x(t)$ ,  $t \in [0, 200]$  seconds, from the initial points  $-(4, 4, 2)$  to the destination  $(1, 2, 2)$ , in the 3D star world workspace. Right: The respective trajectory in the transformed sphere world.

### 4.3 Adaptive Leader-Follower Coordination with Transient Constraints

We study next the problem of leader-follower coordination of a multi-agent system in an obstacle-free workspace. In particular, we consider that a leader agent aims to navigate to a predefined position, subject to collision and connectivity constraints, as well as uncertain 2nd-order dynamics. We use innovatively a useful property of the graph's incidence matrix (see Appendix E) to obtain the desired results.



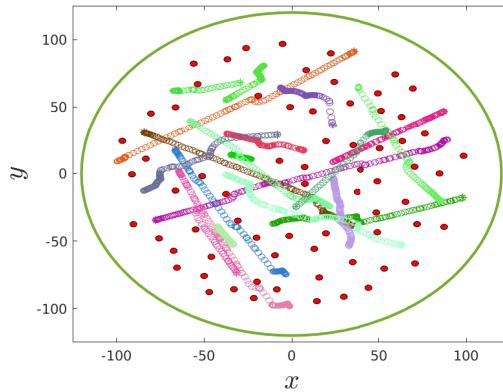
**Figure 4.10:** The initial configurations of the multi-agent scenario. The obstacles are depicted as filled red disks whereas the agents as circles. The destinations are shown with asterisk.



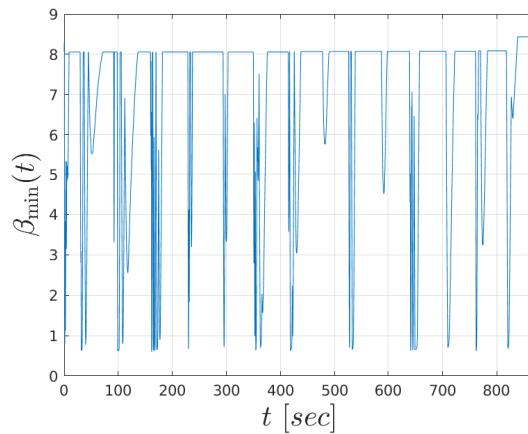
**Figure 4.11:** The resulting signals  $\|x_i(t) - x_{d_i}\|$ ,  $\forall i \in \mathcal{N}$ , shown to converge to zero for the multi-agent scenario.

### 4.3.1 Problem Formulation

Consider  $N > 1$  autonomous robotic agents, with  $\mathcal{N} := \{1, \dots, N\}$ , operating in  $\mathbb{R}^n$  and described by the spheres  $\mathcal{A}_i(x_i) := \mathcal{B}_i(x_i, r_i) = \{y \in \mathbb{R}^n : \|x_i - y\| \leq r_i\}$ , with  $x_i \in \mathbb{R}^n$  being agent  $i$ 's center, and  $r_i \in \mathbb{R}_{>0}$  its bounding radius. In contrast to the previous section, we consider now the



**Figure 4.12:** The resulting trajectories of the agents  $x_i(t)$  in the 2D workspace,  $\forall i \in \mathcal{N}, t \in [0, 870]$  seconds, for the multi-agent scenario.



**Figure 4.13:** The signal  $\beta_{\min}(t)$ , which stays strictly positive, for all  $t \in [0, 870]$ , implying that inter-agent collisions and agent-obstacle collisions are avoided.

more general Lagrangian dynamics for the agents (see Chapter 2):

$$\dot{x}_i = v_i \quad (4.32a)$$

$$M_i(x_i)\dot{v}_i + C_i(x_i, v_i)\dot{x}_i + g_i(x_i) + f_i(x_i, v_i) + d_i(t) = u_i, \quad (4.32b)$$

where  $M_i := M_i(x_i) : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times n}$  are positive definite inertia matrices, with the standard property (see (2.16))

$$0 < \underline{m}I_n \leq M_i(x) \leq \bar{m}I_n,$$

$\forall x \in \mathbb{R}^n, i \in \mathcal{N}$ , for positive constants  $\underline{m}, \bar{m}$ ,  $C_i := C_i(x_i, v_i) : \mathbb{R}^{2n} \rightarrow \mathbb{R}^{n \times n}$  are the Coriolis terms,  $g_i := g_i(x_i) : \mathbb{R}^n \rightarrow \mathbb{R}^n$  are the gravity vectors,  $f_i := f_i(x_i, v_i) : \mathbb{R}^{2n} \rightarrow \mathbb{R}^n$  are unknown vector fields that represent friction-like terms (as in (4.26)),  $d_i := d_i(t) : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^n$  are unknown external disturbances and modeling uncertainties, and  $u_i \in \mathbb{R}^n$  are the agents' control inputs,  $\forall i \in \mathcal{N}$ . The terms  $M_i$ ,  $C_i$  and  $g_i$  are continuous in their arguments, the terms  $f_i$  are Lebesgue measurable and locally bounded, and  $d_i$  are uniformly bounded. Note that here we do not require  $f_i$  and  $d_i$  to be continuous everywhere, since we will employ non-smooth analysis for the stability of the closed-loop system. Moreover, as in Chapter 2, we consider that the dynamic terms  $M_i$ ,  $C_i$ , and  $g_i$  include unknown constant dynamic parameters of the agents (e.g., masses, moments of inertia), denoted by the vectors  $\theta_i \in \mathbb{R}^\ell$ ,  $\ell \in \mathbb{N}$ ,  $\forall i \in \mathcal{N}$ . The Lagrangian system (4.32) satisfies the following well-known properties (as in Chapter 2):

**Property 4.1.** The terms  $\dot{M}_i(x) - 2C_i(x, z)$  are skew-symmetric, i.e.,  $(\dot{M}_i(x) - 2C_i(x, z))^\top = 2C_i(x, z) - \dot{M}_i(x)$  and  $y^\top(\dot{M}_i(x) - 2C_i(x, z))y = 0$ ,  $\forall x, y, z \in \mathbb{R}^n$ ,  $i \in \mathcal{N}$ .

**Property 4.2.** The dynamic terms of (4.32) can be linearly parameterized with respect to the agents' dynamic parameters. That is, for any vectors  $x, y, z, w \in \mathbb{R}^n$ , it holds that  $M_i(x)y + C_i(x, z)w + g_i(x) = Y_i(x, z, w, y)\theta_i$ ,  $\forall x, y, z, w \in \mathbb{R}^n$ , where  $Y_i : \mathbb{R}^{4n} \rightarrow \mathbb{R}^{n \times \ell}$  are known regressor matrices, and  $\theta_i \in \mathbb{R}^\ell$ ,  $\ell \in \mathbb{N}$ , are vectors of constant but unknown dynamic parameters of the agents,  $\forall i \in \mathcal{N}$ .

Moreover, we impose the following assumptions on the system (4.32), which encapsulate standard properties of friction-terms and external disturbances, similar to Assumption 4.1:

**Assumption 4.5.** It holds that  $\|f_i(x_i, v_i)\|_1 \leq \alpha_i \|v_i\|_1$ ,  $\|d_i(t)\|_1 \leq d_{b_i}$ ,  $\forall x_i, v_i \in \mathbb{R}^{2n}$ ,  $t \in \mathbb{R}_{\geq 0}$ , where  $\alpha_i$ ,  $d_{b_i}$  are *unknown* positive constants,  $i \in \mathcal{N}$ .

We aim to compensate  $f_i$  and  $d_i$  by using discontinuous adaptive control. Without loss of generality, we assume that agent  $i = 1$  corresponds to the team leader, whereas  $i > 1$  are the followers, which belong to the set  $\mathcal{N}_F := \{2, \dots, N\}$ . The task of the leader is to navigate to a desired pose  $x_d \in \mathbb{R}^n$ , and the entire team is responsible for guaranteeing collision avoidance as well as connectivity maintenance properties.

In addition, as in Sections 3.2 and 4.2.5, we consider that each agent has a limited sensing radius  $\varsigma_i \in \mathbb{R}_{>0}$ , with  $\varsigma_i > \max_{j \in \mathcal{N}}\{r_i + r_j\}$ , which implies that the agents can sense each other without colliding. Based on this, we model the topology of the multi-agent network through the undirected graph  $\mathcal{G}(x) := (\mathcal{N}, \mathcal{E}(x))$ , with  $\mathcal{E}(x) := \{(i, j) \in \mathcal{N}^2 : \|x_i - x_j\| \leq \min\{\varsigma_i, \varsigma_j\}\}$ , where  $x := [x_1^\top, \dots, x_N^\top]^\top \in \mathbb{R}^{nN}$ . We further denote  $K(x) := |\mathcal{E}(x)|$ . Given the  $k$ -th edge in the edge set  $\mathcal{E}(x)$ , we use the notation  $(k_1, k_2) \in \mathcal{N}^2$  that gives the agent indices that form edge  $k \in \mathcal{K}(x)$ , where  $k_1$  is the tail and  $k_2$  is the head of edge  $k$ , and  $\mathcal{K}(x) := \{1, \dots, K(x)\}$  is an arbitrary numbering of the edges  $\mathcal{E}(x)$ .

As discussed before, the leader agent  $i = 1$  aims at navigating to  $x_d$ . We also need to guarantee that inter-agent collisions are avoided for all times, and that some initial edges, denoted by  $\mathcal{E}_0 \subset \mathcal{E}(x(0))$ , are preserved. The motivation for that is mainly potential cooperative tasks that the agents have to accomplish, whose details are provided only to a leader agent. Then, the leader has to guide the entire team to the points of interest, which is guaranteed via graph connectivity (all agents are part of an edge). There exist, nevertheless, more sophisticated and less conservative ways to maintain graph connectivity than just maintaining part of the initial edges [? ? ?]. Such schemes are not included in the current framework. The results in this section are more general, in the sense that neither the graph connectivity of  $\mathcal{G}(x(0))$ ,  $\mathcal{G}_0 := (\mathcal{N}, \mathcal{E}_0)$ , nor the connectivity to the leader are *technical* requirements of the analysis, as shown below. Formally, the problem treated in this section is the following:

**Problem 4.2.** Consider  $N$  spherical autonomous robotic agents with dynamics (4.32). Given Properties 4.1–4.2 and Assumption 4.5, develop a decentralized control strategy that guarantees 1) achievement of the leader's task, 2) inter-agent collision avoidance, and 3) connectivity maintenance between a subset of the initially connected agents, i.e.,

1.  $\lim_{t \rightarrow \infty} (x_1(t) - x_d) = 0$ ,
2.  $\mathcal{A}_i(x_i(t)) \cap \mathcal{A}_j(x_j(t)) = \emptyset, \forall t \in \mathbb{R}_{\geq 0}, i, j \in \mathcal{N}, i \neq j$ ,
3.  $\|x_{k_1}(t) - x_{k_2}(t)\| \leq \min\{\varsigma_{m_1}, \varsigma_{m_2}\}, \forall t \in \mathbb{R}_{\geq 0}, k \in \mathcal{K}_0 \subset \mathcal{K}(x(0))$ ,

where  $\mathcal{K}_0 := \{1, \dots, K_0\}$  is an edge numbering for the edge set  $\mathcal{E}_0$ , with  $K_0 := |\mathcal{E}_0|$ .

### 4.3.2 Problem Solution

In this section we propose a decentralized control protocol for the solution of Problem 4.2.

Besides the edge set  $\mathcal{E}_0$ , with edge numbering  $\mathcal{K}_0$  and  $K_0$  edges, which needs to remain connected, consider also the complete graph  $\bar{\mathcal{G}} := (\mathcal{N}, \bar{\mathcal{E}})$ , with  $\bar{\mathcal{E}} := \{(i, j), \forall i, j \in \mathcal{N}, i < j\}$ ,  $\bar{K} := |\bar{\mathcal{E}}| = \frac{N(N-1)}{2}$ , and the edge numbering  $\bar{\mathcal{K}} := \{1, \dots, K_0, K_0 + 1, \dots, \bar{K}\}$ , where  $\{K_0 + 1, \dots, \bar{K}\}$  corresponds to the edges in  $\bar{\mathcal{E}} \setminus \mathcal{E}_0$ . Moreover, denote by  $D_0$  and  $\bar{D}$  the incidence matrices of  $\mathcal{G}_0$  and  $\bar{\mathcal{G}}$ , respectively (see Section E.1 of Appendix E).

We construct now the local collision and connectivity functions for all edges  $\bar{\mathcal{K}}$  and  $\mathcal{K}_0$ , respectively. Given positive constants  $\bar{\beta}_{\mathfrak{c}}$  and  $\bar{\beta}_{\mathfrak{n}}$ , let  $\beta_{\mathfrak{c},k} : \mathbb{R}_{\geq 0} \rightarrow [0, \bar{\beta}_{\mathfrak{c}}]$  and  $\beta_{\mathfrak{n},l} : \mathbb{R}_{\geq 0} \rightarrow [0, \bar{\beta}_{\mathfrak{n}}]$ , with

$$\begin{aligned}\beta_{\mathfrak{c},k}(x) &:= \begin{cases} \vartheta_{\mathfrak{c},k}(x) & 0 \leq x < \bar{d}_{\mathfrak{c},k}, \\ \bar{\beta}_{\mathfrak{c}} & \bar{d}_{\mathfrak{c},k} \leq x \end{cases}, \\ \beta_{\mathfrak{n},l}(x) &:= \begin{cases} \vartheta_{\mathfrak{n},l}(x) & 0 \leq x < \underline{d}_{\mathfrak{n},l}^2, \\ \bar{\beta}_{\mathfrak{n}} & \underline{d}_{\mathfrak{n},l}^2 \leq x \end{cases},\end{aligned}$$

$\forall k \in \bar{\mathcal{K}}, l \in \mathcal{K}_0$ , where  $\vartheta_{\mathfrak{c},k} : \mathbb{R}_{\geq 0} \rightarrow [0, \bar{\beta}_{\mathfrak{c}}]$ ,  $\vartheta_{\mathfrak{n},l} : \mathbb{R}_{\geq 0} \rightarrow [0, \bar{\beta}_{\mathfrak{n}}]$  are polynomials that guarantee that  $\beta_{\mathfrak{c},k}$  and  $\beta_{\mathfrak{n},l}$ , respectively, are twice continuously differentiable,  $\forall k \in \bar{\mathcal{K}}, l \in \mathcal{K}_0$ . The aforementioned functions are smooth switches, similar to the one used in Section 4.2. Then, we choose

$$\begin{aligned}\beta_{\mathfrak{c},k} &:= \beta_{\mathfrak{c},k}(\iota_k), \quad \iota_k := \iota_k(x_{k_1}, x_{k_2}) := \|x_{k_1} - x_{k_2}\|^2 - (r_{k_1} + r_{k_2})^2 \\ \beta_{\mathfrak{n},l} &:= \beta_{\mathfrak{n},l}(\nu_l), \quad \nu_l := \nu_l(x_{l_1}, x_{l_2}) := \underline{d}_{\mathfrak{n},l}^2 - \|x_{l_1} - x_{l_2}\|^2\end{aligned}$$

with  $\underline{d}_{\mathfrak{n},k} := \min\{\varsigma_{k_1}, \varsigma_{k_2}\}$  and we also set  $\bar{d}_{\mathfrak{c},k} := \underline{d}_{\mathfrak{n},k}^2 - (r_{k_1} + r_{k_2})^2$ ,  $\forall k \in \bar{\mathcal{K}}, l \in \mathcal{K}_0$ . The terms  $\bar{\beta}_{\mathfrak{c}}$ ,  $\bar{\beta}_{\mathfrak{n}}$  can be any positive constants. Note that  $\beta_{\mathfrak{c},k}$  and  $\beta_{\mathfrak{n},l}$  take into account the limited sensing capabilities of the agents and their derivatives vanish at collisions and connectivity breaks, respectively, of the respective edges. All the parameters for the construction of  $\beta_{\mathfrak{c},k}$ ,  $\beta_{\mathfrak{n},l}$  can be transmitted off-line to the agents.

Regarding the uncertain terms of (4.32), note that  $\theta_i \in \mathbb{R}^\ell$ ,  $\alpha_i \in \mathbb{R}$ , and  $d_{b_i} \in \mathbb{R}$  from Properties 4.1, 4.2 and Assumption 4.5 are unknown to the agents. Hence, we define the estimations of these terms  $\hat{\theta}_i \in \mathbb{R}^\ell$ ,  $\hat{\alpha}_i \in \mathbb{R}$ ,  $\hat{d}_{b_i} \in \mathbb{R}$ ,  $\forall i \in \mathcal{N}$ , with the respective errors  $\tilde{\theta}_i := \hat{\theta}_i - \theta_i$ ,  $\tilde{\alpha}_i := \hat{\alpha}_i - \alpha_i$ ,  $\tilde{d}_{b_i} := \hat{d}_{b_i} - d_{b_i}$ ,  $\forall i \in \mathcal{N}$ . By using adaptive control techniques, we prove in the following that these estimations compensate appropriately for the unknown terms, without necessarily converging to them. In addition, we define the leader error signal  $s_e := x_1 - x_d$  and  $\alpha_{i,k}^{\mathfrak{c}}$  and  $\alpha_{i,l}^{\mathfrak{n}}$  as:

$$\alpha_{i,k}^{\mathfrak{c}} := \begin{cases} -1, & i = k_1 \\ 1, & i = k_2 \\ 0, & \text{otherw.} \end{cases} \quad \alpha_{i,l}^{\mathfrak{n}} := \begin{cases} -1, & i = l_1 \\ 1, & i = l_2 \\ 0, & \text{otherw.} \end{cases}$$

$\forall k \in \bar{\mathcal{K}}, l \in \mathcal{K}_0, i \in \mathcal{N}$ , which provide boolean values depending on whether agent  $i$  is part (head or tail) of edge  $m$  and  $l$  (as in (3.20)). Finally, we define,  $\forall k \in \bar{\mathcal{K}}, l \in \mathcal{K}_0$ , the terms

$$\beta'_{\mathfrak{c},k} := \frac{\partial}{\partial \iota_k} \left( \frac{1}{\beta_{\mathfrak{c},k}(\iota_k)} \right), \quad \beta'_{\mathfrak{n},l} := \frac{\partial}{\partial \nu_l} \left( \frac{1}{\beta_{\mathfrak{n},l}(\nu_l)} \right),$$

which diverge to infinity in a collision and a connectivity break of the agents  $k_1, k_2$  and  $l_1, l_2$ , respectively. We propose now the following decentralized adaptive control protocol. Choose the agents' desired velocity as

$$v_{d_1} = -\gamma_e s_e + \sum_{k \in \bar{\mathcal{K}}} \alpha_{1,k}^{\mathfrak{c}} \beta'_{\mathfrak{c},k} \frac{\partial \iota_k}{\partial x_{k_1}} + \sum_{l \in \mathcal{K}_0} \alpha_{1,l}^{\mathfrak{n}} \beta'_{\mathfrak{n},l} \frac{\partial \nu_l}{\partial x_{l_1}} \quad (4.33a)$$

$$v_{d_i} = k_i \left( \sum_{k \in \bar{\mathcal{K}}} \alpha_{i,k}^{\mathfrak{c}} \beta'_{\mathfrak{c},k} \frac{\partial \iota_k}{\partial x_{k_1}} + \sum_{l \in \mathcal{K}_0} \alpha_{i,l}^{\mathfrak{n}} \beta'_{\mathfrak{n},l} \frac{\partial \nu_l}{\partial x_{l_1}} \right), \forall i \in \mathcal{N}_{\mathcal{F}} \quad (4.33b)$$

that concerns the collision avoidance and connectivity maintenance properties, with the extra term  $\gamma_e s_e$  for the leader to guarantee the navigation to  $x_d$ . The terms  $\gamma_e, k_i$  are positive constants,  $\forall i \in \mathcal{N}_{\mathcal{F}}$ . Since  $v_{d_i}$  is not the actual velocity of the agents, we define the errors  $e_{v_i} := v_i - v_{d_i}$ ,  $\forall i \in \mathcal{N}$ , and design the decentralized control laws  $u_i : \mathcal{X}_i \rightarrow \mathbb{R}^6$

$$\begin{aligned} u_i := u_i(\chi_i) = & \sum_{k \in \bar{\mathcal{M}}} \alpha_{i,k}^{\mathfrak{c}} \beta'_{\mathfrak{c},k} \frac{\partial \iota_k}{\partial x_{k_1}} + \sum_{l \in \mathcal{K}_0} \alpha_{i,l}^{\mathfrak{n}} \beta'_{\mathfrak{n},l} \frac{\partial \nu_l}{\partial x_{l_1}} - k_{v_i} e_{v_i} - \tilde{s}_{e_i} + Y_{r_i} \hat{\theta}_i \\ & - \text{sgn}(e_{v_i}) \|\dot{x}_i\|_1 \hat{\alpha}_i - \text{sgn}(e_{v_i}) \hat{d}_{b_i}, \end{aligned} \quad (4.34)$$

$\forall i \in \mathcal{N}$ , where  $\chi_i := [x^\top, v^\top, \hat{\theta}_i^\top, \hat{\alpha}_i, \hat{d}_{b_i}]^\top$ ,  $v = [v_1^\top, \dots, v_N^\top]^\top$ ,

$$\mathcal{X}_i := \{\chi_i \in \mathbb{R}^{2Nn+\ell+2} : \iota_k(x_{k_1}, x_{k_2}) > 0, \nu_l(x_{l_1}, x_{l_2}) > 0, \forall k \in \bar{\mathcal{K}}, l \in \mathcal{K}_0\},$$

$\tilde{s}_{e_1} = \gamma_e s_e$ ,  $\tilde{s}_{e_i} = 0, \forall i \in \mathcal{N}_{\mathcal{F}}$ ,  $Y_{r_i} := Y_i(x_i, v_i, v_{d_i}, \dot{v}_{d_i})$ , and  $k_{v_i}$  are positive gains. Moreover, we design the adaptation signals

$$\left. \begin{aligned} \dot{\hat{d}}_{b_i} &= \gamma_{i,d} \|e_{v_i}\|_1, \\ \dot{\hat{\alpha}}_i &= \gamma_{i,f} \|e_{v_i}\|_1 \|v_i\|_1, \\ \dot{\hat{\theta}}_i &= -\gamma_{i,\theta} Y_{r_i}^\top e_{v_i} \end{aligned} \right\} i \in \mathcal{N}, \quad (4.35)$$

with arbitrary bounded initial conditions, and positive constants  $\gamma_{i,d}, \gamma_{i,f}, \gamma_{i,\theta}, \forall i \in \mathcal{N}$ . Note from (4.34) that, unlike the usual case in the related literature, the leader contributes to the collision avoidance and connectivity maintenance properties, apart from just guaranteeing achievement of its task. Regarding the rest of the terms,  $Y_i(\cdot)\hat{\theta}$ ,  $\text{sgn}(e_{v_i})\|\dot{x}_i\|_1 \hat{\alpha}_i$ , and  $\text{sgn}(e_{v_i})\hat{d}_{b_i}$  compensate for the unknown terms  $\theta_i, f_{b_i}$ , and  $d_{b_i}$ , respectively, and  $e_{v_i}$  is a dissipative velocity term that ensures closed-loop stability. The main results of this section are summarized in the following theorem.

**Theorem 4.4.** Consider a multi-agent team  $\mathcal{N}$ , described by the dynamics (4.32) subject to Properties 4.1, 4.2 and Assumption 4.5. Then, application of the control and adaptation laws (4.34), (4.35) guarantees: 1) navigation of the leader agent to  $x_d$ , 2) connectivity maintenance of the subset  $\mathcal{E}_0$  of the initial edges, 3) inter-agent collision avoidance, and 4) boundedness of all closed loop signals, from all collision-free initial configurations, i.e.,  $\mathcal{A}_i(x_i(0)) \cap \mathcal{A}_j(x_j(0)) = \emptyset, \forall i, j \in \mathcal{N}$ , with  $i \neq j$ , providing thus a solution to Problem 4.2. Moreover, it holds that  $\lim_{t \rightarrow \infty} v_i(t) = 0, \forall i \in \mathcal{N}$ .

*Proof.* By employing (4.32), (4.34), (4.35), we can write the closed-loop system as

$$\dot{x}_i = v_i \quad (4.36a)$$

$$\dot{v}_i = -M_i(x_i)^{-1} \left( C_i(x_i, v_i) v_i + g_i(x_i) + \mathsf{K}[f_i](x_i, v_i) + d_i(t) - \mathsf{K}[u_i] \right) \quad (4.36b)$$

$$\dot{\hat{d}}_b = \gamma_{i,d} \|e_{v_i}\|_1 \quad (4.36c)$$

$$\dot{\hat{\alpha}}_i = \gamma_{i,f} \|e_{v_i}\|_1 \|v_i\|_1 \quad (4.36d)$$

$$\dot{\hat{\theta}}_i = -\gamma_{i,\theta} Y_i(x_i, v_i, v_{d_i}, \dot{v}_{d_i})^\top e_{v_i} \quad (4.36e)$$

$\forall i \in \mathcal{N}$ , where  $\mathsf{K}[f_i]$  and  $\mathsf{K}[u_i]$  are the Filippov regularizations of  $f_i$  and  $u_i$ , respectively,  $\forall i \in \mathcal{N}$ . In particular,  $\mathsf{K}[u_i]$  is formed by substituting  $\text{sgn}(e_{v_i})$  with  $\text{SGN}(e_{v_i})$  in (4.34). Let now  $\hat{d}_b := [\hat{d}_{b_1}, \dots, \hat{d}_{b_N}]^\top$ ,  $\hat{\alpha} := [\hat{\alpha}_1, \dots, \hat{\alpha}_N]^\top$ ,  $\hat{\theta} := [\hat{\theta}_1^\top, \dots, \hat{\theta}_N^\top]^\top$ ,  $\chi := [x^\top, v^\top, \hat{d}_b^\top, \hat{\alpha}^\top, \hat{\theta}^\top]^\top$  and consider the set

$$\mathcal{X} := \{\chi \in \mathbb{R}^{2Nn+2N+\ell N} : \chi_i \in \mathcal{X}_i, \forall i \in \mathcal{N}\}.$$

Since, initially the agents do not collide and  $\mathcal{E}_0$  is a subset of the initially connected agents  $\mathcal{E}(x(0))$ , it holds that  $\chi(0) \in \mathcal{X}$ . The right hand side of (4.36) is measurable in  $t$  over  $\mathbb{R}_{\geq 0}$  and Lebesgue measurable and locally bounded in  $\chi$  on  $\mathcal{X}$ . Therefore, by invoking Prop. A.1 of Appendix A, there exists at least a Filippov solution  $\mu_{LF} : [0, t_{\max}] \rightarrow \mathcal{X}$  for some  $t_{\max} > 0$ . Consider now the function

$$V_1 := \frac{\gamma_e}{2} \|s_e\|^2 + \sum_{k \in \bar{\mathcal{K}}} \frac{1}{\beta_{\mathfrak{c},k}} + \sum_{l \in \mathcal{K}_0} \frac{1}{\beta_{\mathfrak{n},l}} \quad (4.37)$$

which is well defined when  $\mu_{LF} \in \mathcal{X}$ . By considering the time derivative of  $V_1$ , and taking into account that  $\frac{\partial \iota_k}{\partial x_{k_1}} = -\frac{\partial \iota_k}{\partial x_{k_2}}, \forall k \in \bar{\mathcal{K}}$ ,  $\frac{\partial \nu_l}{\partial x_{l_1}} = -\frac{\partial \nu_l}{\partial x_{l_2}}, \forall l \in \mathcal{K}_0$ , we obtain

$$\dot{V}_1 = \gamma_e s_e^\top v_1 - \beta^\top (\tilde{D} \otimes I_n)^\top v, \quad (4.38)$$

where  $\beta := [\beta_{\mathfrak{c}}^\top, \beta_{\mathfrak{n}}^\top]^\top \in \mathbb{R}^{\bar{K}+K_0}$ ,  $\beta_{\mathfrak{c}} := [\beta'_{\mathfrak{c},1} \frac{\partial \nu_1}{\partial x_{1_1}}, \dots, \beta'_{\mathfrak{c},\bar{K}} \frac{\partial \nu_{\bar{K}}}{\partial x_{\bar{K}_1}}]^\top \in \mathbb{R}^{\bar{K}}$ ,  $\beta_{\mathfrak{n}} := [\beta'_{\mathfrak{n},1} \frac{\partial \nu_1}{\partial x_{1_1}}, \dots, \beta'_{\mathfrak{n},K_0} \frac{\partial \nu_{K_0}}{\partial x_{(K_0)_1}}]^\top \in \mathbb{R}^{K_0}$ , and  $\tilde{D} := [\bar{D}, D_0] \in \mathbb{R}^{N \times (\bar{K}+K_0)}$ , where  $\bar{D}$  and  $D_0$  are the incidence matrices corresponding to  $\bar{\mathcal{E}}$  and  $\mathcal{E}_0$ , respectively. Let now  $\tilde{d}_i^\top \in \mathbb{R}^{\bar{K}+K_0}$ ,  $i \in \mathcal{N}$ , be the rows of  $\tilde{D}$ , i.e.,  $\tilde{D} = [\tilde{d}_1, \dots, \tilde{d}_N]^\top$ . Then, (4.38) can be written as

$$\dot{V}_1 := \gamma_e s_e^\top v_1 - \sum_{i \in \mathcal{N}} \beta^\top (\tilde{d}_i \otimes I_n) v_i = (\gamma_e s_e^\top - \beta^\top (\tilde{d}_1 \otimes I_n)) v_1 - \sum_{i \in \mathcal{N}_{\mathcal{F}}} \beta^\top (\tilde{d}_i \otimes I_n) v_i$$

and (4.33) and (4.34) as

$$v_{d_1} = -\gamma_e s_e + (\tilde{d}_1 \otimes I_n)^\top \beta \quad (4.39a)$$

$$v_{d_i} = k_i (\tilde{d}_i \otimes I_n)^\top \beta, \quad \forall i \in \mathcal{N}_{\mathcal{F}} \quad (4.39b)$$

$$\begin{aligned} u_i = & (\tilde{d}_i \otimes I_n)^\top \beta - \tilde{s}_{e_i} + Y_i(x_i, v_i, v_{d_i}, \dot{v}_{d_i}) \hat{\theta}_i - k_{v_i} e_{v_i} - \text{sgn}(e_{v_i}) \|v_i\|_1 \hat{\alpha}_i \\ & - \text{sgn}(e_{v_i}) \hat{d}_{b_i}, \quad \forall i \in \mathcal{N}. \end{aligned} \quad (4.39c)$$

Achievement of the desired velocities, i.e.,  $v_i = v_{d_i}$ ,  $\forall i \in \mathcal{N}$ , would imply that

$$\dot{V}_1 = -\|\gamma_e s_e - (\tilde{d}_1 \otimes I_n)^\top \beta\|^2 - \sum_{i \in \mathcal{N}_{\mathcal{F}}} k_i \|(\tilde{d}_i \otimes I_n)^\top \beta\|^2.$$

The actual velocities of the agents, however, are not necessarily equal to the desired ones  $v_{d_i}$ , and therefore we use a backstepping-like technique to proceed. Consider the vector  $\mathbf{z} \in \mathcal{Z}$ , with

$$\mathbf{z} := \left[ s_e^\top, \left( \frac{1}{\beta_{\mathfrak{c},1}} \right)^{\frac{1}{2}}, \dots, \left( \frac{1}{\beta_{\mathfrak{c},\bar{K}}} \right)^{\frac{1}{2}}, \left( \frac{1}{\beta_{\mathfrak{n},1}} \right)^{\frac{1}{2}}, \dots, \left( \frac{1}{\beta_{\mathfrak{n},K_0}} \right)^{\frac{1}{2}}, e_v^\top, \tilde{d}_b^\top, \tilde{\alpha}^\top, \tilde{\theta}^\top \right]^\top,$$

where  $e_v := [e_{v_1}^\top, \dots, e_{v_N}^\top]^\top \in \mathbb{R}^{nN}$ ,  $\tilde{d}_b := [\tilde{d}_{b_1}, \dots, \tilde{d}_{b_N}]^\top \in \mathbb{R}^N$ ,  $\tilde{\alpha} := [\tilde{\alpha}_1, \dots, \tilde{\alpha}_N]^\top \in \mathbb{R}^N$ ,  $\tilde{\theta} := [\tilde{\theta}_1^\top, \dots, \tilde{\theta}_N^\top]^\top \in \mathbb{R}^{\ell N}$ , and  $\mathcal{Z} := \mathbb{R}^{n+(2+n+\ell)N+\bar{K}+K_0}$ . Similar to (4.36), we guarantee the existence of a Filippov solution  $\mathbf{z} : [0, t_{\max}) \rightarrow \mathcal{Z}$  for the respective closed-loop system obtained by differentiating  $\mathbf{z}$ . We aim to prove that  $\mathbf{z}(t)$  remains in a compact subset of  $\mathcal{Z}$ , which implies that  $\chi$  remains in a compact subset of  $\mathcal{X}$ . Define the barrier-like function  $V_{LF} := V_{LF}(\mathbf{z}, t) : \mathcal{Z} \times [0, t_{\max}) \rightarrow \mathbb{R}_{\geq 0}$ , with

$$V_{LF}(\mathbf{z}, t) := V_1(\mathbf{z}) + \sum_{i \in \mathcal{N}} \left\{ \frac{1}{2} e_{v_i}^\top M_i(x_i(t)) e_{v_i} + \frac{1}{2\gamma_{i,d}} \tilde{d}_{b_i}^2 + \frac{1}{2\gamma_{i,f}} \tilde{\alpha}_i^2 + \frac{1}{2\gamma_{i,\theta}} \|\tilde{\theta}_i\|^2 \right\},$$

for which, by using the fact  $m \leq M_i(x) \leq \bar{m}$ ,  $\forall x \in \mathbb{R}^n$ ,  $i \in \mathcal{N}$ , it holds that  $W_1(\mathbf{z}) \leq V_{LF}(\mathbf{z}, t) \leq W_2(\mathbf{z})$ , where  $W_1, W_2 : \mathcal{Z} \rightarrow \mathbb{R}_{\geq 0}$  are positive definite functions. Since initially the agents do not collide and  $\mathcal{E}_0$  is a

subset of the initially connected agents  $\mathcal{E}(x(0))$ ,  $V_1$ , as defined in (4.37), is well-defined, and hence  $V_{LF}(\mathbf{z}(0), 0)$ ,  $\frac{1}{\beta_{\epsilon,k}(\nu_m(0))}$ ,  $\frac{1}{\beta_{n,l}(\nu_l(0))}$  are bounded,  $\forall k \in \bar{\mathcal{K}}, l \in \mathcal{K}_0$ , i.e.,  $V_{LF}(\mathbf{z}(0), 0) \leq \bar{V}$  for a finite constant  $\bar{V}$ . By taking the derivative of  $V_{LF}$ , and in view of Lemma A.2 of Appendix A, one obtains  $\dot{V}_{LF}(\mathbf{z}(t), t) \stackrel{a.e.}{\in} \tilde{V}_{LF}(\mathbf{z}(t), t)$ , where  $\tilde{V}_{LF}(\mathbf{z}(t), t)$  is the intersection of the inner products of the all generalized gradients of  $V_{LF}$  with the right-hand size of (4.36). Since  $V_{LF}(\mathbf{z}, t)$  is continuously differentiable, the generalized gradient reduces to the standard gradient and one obtains

$$\begin{aligned}\dot{\tilde{V}}_{LF} &\subset \dot{V}_1 + \sum_{i \in \mathcal{N}} \left\{ \frac{1}{2} e_{v_i}^\top \dot{M}_i e_{v_i} + e_{v_i}^\top (u_i - C_i \dot{x}_i - g_i - f_i - d_i) - e_{v_i} M_i \dot{v}_{d_i} \right. \\ &\quad \left. + \frac{1}{\gamma_{i,f}} \tilde{\alpha}_i \dot{\alpha}_i + \frac{1}{\gamma_{i,d}} \tilde{d}_{b_i} \dot{d}_{b_i} + \frac{1}{\gamma_{i,\theta}} \tilde{\theta}_i^\top \dot{\theta}_i \right\}.\end{aligned}$$

By substituting  $v_i = e_{v_i} + v_{d_i}$  in  $C_i v_i$  and (4.38), and using Properties 4.1, 4.2, we obtain

$$\begin{aligned}\dot{\tilde{V}}_{LF} &\subset -\|\gamma_e s_e - (\tilde{d}_1 \otimes I_n)^\top \beta\|^2 - \sum_{i \in \mathcal{N}_F} k_i \|(\tilde{d}_i \otimes I_n)^\top \beta\|^2 + \gamma_e s_e^\top e_{v_1} + \\ &\quad \sum_{i \in \mathcal{N}} \left\{ e_{v_i}^\top (u_i - Y_{r_i} \theta_i - f_i - d_i - (\tilde{d}_i \otimes I_n)^\top \beta) + \frac{1}{\gamma_{i,f}} \tilde{\alpha}_i \dot{\alpha}_i + \frac{1}{\gamma_{i,d}} \tilde{d}_{b_i} \dot{d}_{b_i} \right. \\ &\quad \left. + \frac{1}{\gamma_{i,\theta}} \tilde{\theta}_i^\top \dot{\theta}_i \right\}.\end{aligned}$$

Next, by substituting the control laws (4.39), the right-hand side becomes

$$\begin{aligned}\dot{\tilde{V}}_{LF} &\subset -\|\gamma_e s_e - (\tilde{d}_1 \otimes I_n)^\top \beta\|^2 - \sum_{i \in \mathcal{N}_F} k_i \|(\tilde{d}_i \otimes I_n)^\top \beta\|^2 + \sum_{i \in \mathcal{N}} \left\{ e_{v_i}^\top (Y_{r_i} \tilde{\theta}_i \right. \\ &\quad \left. - k_{v_i} e_{v_i} - \text{SGN}(e_{v_i})(\|v_i\|_1 \hat{\alpha}_i + \hat{d}_{b_i}) - f_i - d_i) + \frac{1}{\gamma_{i,f}} \tilde{\alpha}_i \dot{\alpha}_i + \frac{1}{\gamma_{i,d}} \tilde{d}_{b_i} \dot{d}_{b_i} \right. \\ &\quad \left. + \frac{1}{\gamma_{i,\theta}} \tilde{\theta}_i^\top \dot{\theta}_i \right\}.\end{aligned}$$

By employing the property  $x^\top \text{sgn}(x) = \|x\|_1$ ,  $\forall x \in \mathbb{R}^n$  (which also implies that  $x^\top \text{SGN}(x) = \|x\|_1$ , since  $x^\top \text{SGN}(x) = \{0\}$  when  $x = 0$ ), as well as (4.39) and Assumption 4.5, we obtain

$$\begin{aligned}\max_{z \in \dot{\tilde{V}}_{LF}} \{z\} &\leq -W_\zeta(\mathbf{z}) + \sum_{i \in \mathcal{N}} \left\{ e_{v_i}^\top Y_{r_i} \tilde{\theta}_i + \alpha_i \|e_{v_i}\|_1 \|v_i\|_1 + d_{b_i} \|e_{v_i}\|_1 - k_{v_i} \|e_{v_i}\|^2 \right. \\ &\quad \left. - \|e_{v_i}\|_1 (\|v_i\|_1 \hat{\alpha}_i + \hat{d}_{b_i}) + \frac{1}{\gamma_{i,f}} \tilde{\alpha}_i \dot{\alpha}_i + \frac{1}{\gamma_{i,d}} \tilde{d}_{b_i} \dot{d}_{b_i} + \frac{1}{\gamma_{i,\theta}} \tilde{\theta}_i^\top \dot{\theta}_i \right\},\end{aligned}$$

where  $W_\zeta : \mathcal{Z} \rightarrow \mathbb{R}_{\geq 0}$ , with

$$W_\zeta(\mathbf{z}) := \|\gamma_e s_e - (\tilde{d}_1 \otimes I_n)^\top \beta\|^2 + \sum_{i \in \mathcal{N}_F} k_i \|(\tilde{d}_i \otimes I_n)^\top \beta\|^2.$$

Finally, by substituting  $\alpha_i = \hat{\alpha}_i - \tilde{\alpha}_i$ ,  $d_{b_i} = \hat{d}_{b_i} - \tilde{d}_{b_i}$ ,  $\forall i \in \mathcal{N}$ , as well as the adaptation laws (4.35), we obtain

$$\max_{z \in \dot{\tilde{V}}_{LF}} \{z\} \leq -W_\zeta(\mathbf{z}) - \sum_{i \in \mathcal{N}} k_{v_i} \|e_{v_i}\|^2 =: -W(\mathbf{z}).$$

Therefore, we conclude that  $z \leq -W(\mathbf{z})$ ,  $\forall z \in \dot{\tilde{V}}_{LF}(\mathbf{z}(t), t)$ ,  $t \in [0, t_{\max}]$ ,  $\mathbf{z} \in \mathcal{Z}$ , where  $W : \mathcal{Z} \rightarrow \mathbb{R}_{\geq 0}$  is a positive semi-definite function defined on  $\mathcal{Z}$ . Hence, the conditions of Theorem A.6 of Appendix A hold, according to which we conclude that all Filippov solutions starting in  $\mathbf{z}(0) \in \bar{\mathcal{Z}} := \{\mathbf{z} \in \mathcal{B}(0, r_\zeta) : W_2(\mathbf{z}) < \min_{\|\mathbf{z}\|=r_\zeta} W_1(\mathbf{z})\}$  are extended to  $t_{\max} = \infty$ , satisfy  $\mathbf{z}(t) \in \bar{\mathcal{Z}}$  for all  $t \in \mathbb{R}_{\geq 0}$  and any positive  $r_\zeta$ , and  $\lim_{t \rightarrow \infty} W(\mathbf{z}(t)) = 0$ . Thus, the terms  $\beta_{c,k}(\iota_k(t))$ ,  $\beta_{n,l}(\nu_l(t))$  are bounded,  $\forall t \in \mathbb{R}_{\geq 0}$ ,  $k \in \bar{K}$ ,  $l \in \bar{K}_0$ , which implies that connectivity breaks of the set  $\mathcal{E}_0$  and inter-agent collisions are avoided,  $\forall t \in \mathbb{R}_{\geq 0}$ .

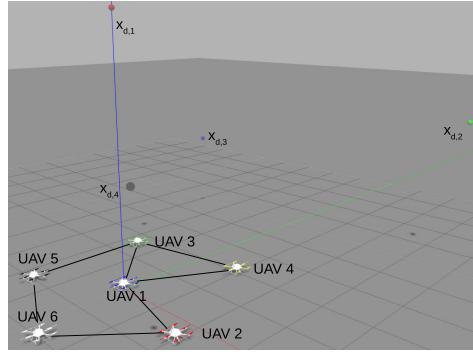
In addition, it holds that  $\lim_{t \rightarrow \infty} e_{v_i}(t) = 0$ ,  $\forall i \in \mathcal{N}$ ,  $\lim_{t \rightarrow \infty} (\tilde{d}_i \otimes I_n)^\top \beta(t) = 0$ ,  $\forall i \in \mathcal{N}_F$ , as well as  $\lim_{t \rightarrow \infty} \|\gamma_e s_e - (\tilde{d}_1 \otimes I_n)^\top \beta\| = 0$ . We employ now Property E.1 of Appendix A for incidence matrices, which dictates that  $\sum_{i \in \mathcal{N}} d_i = 0$ . Hence, it holds that

$$\lim_{t \rightarrow \infty} (\tilde{d}_1 \otimes I_n)^\top \beta = - \lim_{t \rightarrow \infty} \sum_{i \in \mathcal{N}_F} (\tilde{d}_i \otimes I_n)^\top \beta = 0,$$

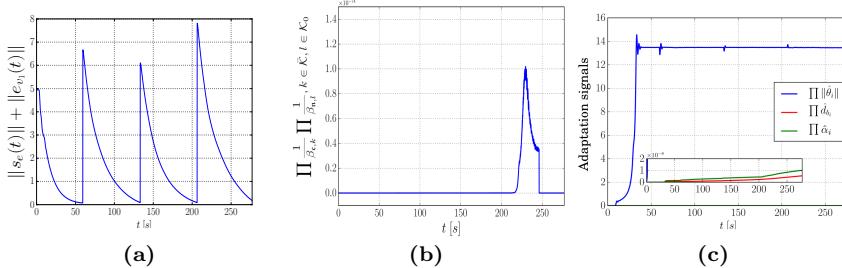
which implies that  $\lim_{t \rightarrow \infty} \|\gamma_e s_e - (\tilde{d}_1 \otimes I_n)^\top \beta\| = 0 \Rightarrow \lim_{t \rightarrow \infty} s_e = 0$ , meaning that the leader agent will converge to its destination. Moreover, one concludes that  $\lim_{t \rightarrow \infty} v_i(t) = 0$ ,  $\forall i \in \mathcal{N}$  due to (4.39). Note that  $r_\zeta$  can be any positive constant and hence the result is global with respect to  $\mathbf{z}$ , i.e., all initial configurations that are collision-free and satisfy  $\mathcal{E}_0 \subset \mathcal{E}(x(0))$ . Moreover, the fact that  $t_{\max} = \infty$  implies that there is no Zeno behavior due to the discontinuous nature of the controller.  $\square$

**Remark 4.4.** Inspection of the closed loop dynamics (4.36b) reveals that, if  $\lim_{t \rightarrow \infty} \dot{v}_i(t) = 0$ , then the follower agents will converge to an invariant set where  $d_i(t) = Y_{r_i} \tilde{\theta}_i$ .

**Remark 4.5.** Note that initial connectivity of the graphs  $\mathcal{G}(x(0))$ ,  $\mathcal{G}_0$  and connectivity to the leader are not technical requirements, as is usually the case in the related literature (e.g., [? ?]). In such cases, the leader will still



**Figure 4.14:** The initial positions of the 6 UAVs, along with the desired leader goals  $x_{d,k}$ ,  $k \in \{1, \dots, 4\}$ , and the edge set  $\mathcal{E}_0$ .



**Figure 4.15:** (a): The leader signal  $\|s_e(t)\| + \|e_{v_1}(t)\|$ , which converges to zero for every navigation objective; (b) the product  $\prod_{k \in \bar{\mathcal{K}}} \frac{1}{\beta_{c,k}(\nu_k(t))}$   $\prod_{l \in \mathcal{K}_0} \frac{1}{\beta_{n,l}(\nu_l(t))}$ , which remains bounded, proving thus the collision and connectivity properties (the zero values stem from the computer's lower numerical limits); (c) the adaptation signals  $\prod_{i \in \{1, \dots, 6\}} \|\hat{\theta}_i(t)\|$ ,  $\prod_{i \in \{1, \dots, 6\}} \hat{d}_{b_i}(t)$ ,  $\prod_{i \in \{1, \dots, 6\}} \hat{\alpha}_i(t)$ , which remain bounded,  $\forall t \in [0, 277]$  s.

converge to  $x_d$ , inter-agent collisions will not occur, and the edges of  $\mathcal{E}_0$ , will be preserved. Regarding the unknown terms  $f_i$ ,  $d_i$ ,  $\theta_i$ , note from Theorem 4.4 and its proof that these are successfully compensated, without the need of convergence of the respective errors to zero. Finally, observe that the framework can be also applied to the multi-robot navigation problem, via alternating between leaders and followers and appropriate prioritization.

### 4.3.3 Simulation Results

We conducted simulations with  $N = 6$  UAVs in  $\mathbb{R}^3$  using the realistic robotic simulator Gazebo [? ]. We considered bounding radii  $r_i = 0.35$ m,

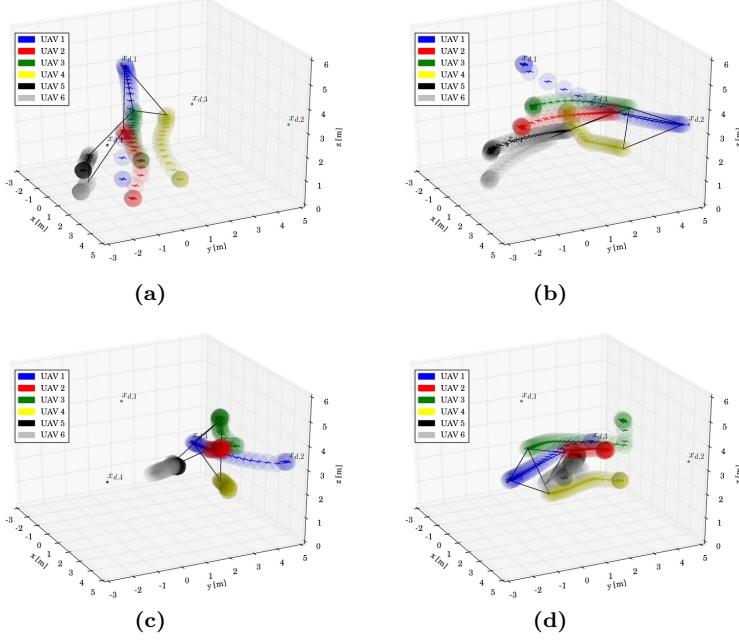
sensing ranges  $\varsigma_i = 3\text{m}$ ,  $\forall i \in \mathcal{N}$ , and initial positions  $x_1(0) = [0, 0, 0.1]^\top$ ,  $x_2(0) = [2, -0.5, 0.1]^\top$ ,  $x_3(0) = [-1.5, 1.5, 0.1]^\top$ ,  $x_4(0) = [1, 2, 0.1]^\top$ ,  $x_5(0) = [-1.5, -1, 0.1]^\top$ , and  $x_6(0) = [0.5, -1.5, 0.1]^\top \text{ m}$ , (see Fig. 4.14). We also considered that the leader has 4 navigation objectives, that is, to sequentially navigate to the points  $x_{d,1} = [0, 0, 5]^\top$ ,  $x_{d,2} = [4, 5, 3]^\top$ ,  $x_{d,3} = [-2, 4, 2]^\top$ ,  $x_{d,4} = [3, -2, 3]^\top \text{ m}$  (pictured as small spheres in Fig. 4.14). Since this work provides asymptotic results with respect to the error  $s_e$ , the leader switches navigation goal each time it gets closer than 0.075m to the current goal, i.e.,  $\|s_e\| \leq 0.075\text{m}$ . We also considered

$$\mathcal{E}_0 = \{(1, 2), (1, 3), (1, 4), (3, 4), (3, 5), (5, 6), (2, 6)\},$$

as shown in Fig. 4.14 via straight black lines. The unknown parameters  $\theta_i$  concerned the UAVs' mass and the gravity constant. The control gains and parameters were set as  $\gamma_e = 0.7$ ,  $k_i = 5$ ,  $\forall i \in \{2, \dots, 6\}$ , and  $\gamma_{i,\theta} = 0.1$ ,  $\gamma_{i,d} = 0.01$ ,  $\gamma_{i,f} = 0.1$ ,  $k_{v_i} = 2$ ,  $\forall i \in \{1, \dots, 6\}$ . The simulation results are shown in Figs. 4.15–4.17 for  $t \in [0, 277] \text{ s}$ . More specifically, Fig. 4.15 shows (a) the evolution of the signal  $\|s_e(t)\| + \|e_{v_1}(t)\|$ , which converges to zero for each navigation objective, (b) the evolution of the product  $\prod_{k \in \bar{\mathcal{K}}} \frac{1}{\beta_{c,k}(\nu_k(t))} \prod_{l \in \mathcal{K}_0} \frac{1}{\beta_{n,l}(\nu_l(t))}$ , which remains bounded, verifying thus the collision avoidance and connectivity maintenance properties, and (c) the evolution of the products of the adaptation signals  $\prod_{i \in \{1, \dots, 6\}} \|\hat{\theta}_i(t)\|$ ,  $\prod_{i \in \{1, \dots, 6\}} \hat{d}_{b_i}(t)$ ,  $\prod_{i \in \{1, \dots, 6\}} \hat{\alpha}_i(t)$ , which remain bounded, verifying thus the boundedness of the individual signals. Moreover, Fig. 4.16 depicts the evolution of the multi-agent system along the 4 navigation objectives, with the connectivity of  $\mathcal{E}_0$  (straight black lines), and Fig. 4.17 shows the control inputs of the UAVs. The simulations were carried out in a ROS-Python interface of an i7-8750H laptop computer with 12 cores at 2.2GHz and 16GB of RAM and an illustrating video can be found in <https://youtu.be/bzzXC-v2hEM>.

## 4.4 Closed-Form Collision Avoidance of Ellipsoidal Multi-Agent Systems

The previous sections, as well as Section 3.2 of the previous chapter, considered spherical agents, which is a common assumption also in the related literature. In this section, we turn our attention to robotic agents whose volume is approximated as an ellipsoid in  $\mathbb{R}^3$ , since such an approximation is more realistic for robotic agents (see e.g., Fig. 4.18). We develop a class of *closed-form* barrier functions that approximate the distance between two such ellipsoids and design an adaptive control scheme for the collision avoid-



**Figure 4.16:** The motion of the multi-agent system as the leader navigates to  $x_{d,1}$  (a), ...,  $x_{d,4}$  (d). The connectivity of  $\mathcal{E}_0$  is also pictured via straight lines.

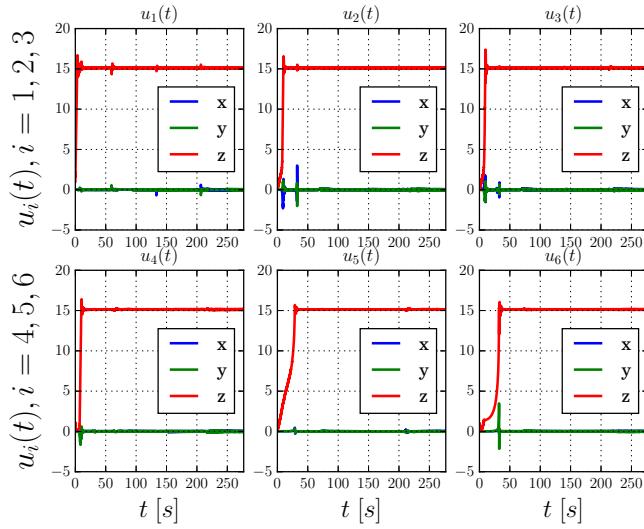
ance of the multi-agent system, subject to some primary task and 2nd-order uncertain dynamics, like in the previous section.

#### 4.4.1 Problem Formulation

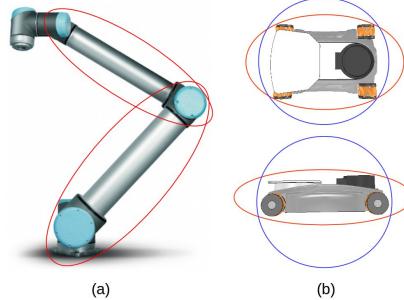
Consider  $N > 1$  ellipsoidal autonomous agents, with  $\mathcal{N} := \{1, \dots, N\}$ , operating in  $\mathbb{SE}(3)$ , and described now by the ellipsoids

$$\mathcal{A}_i(x_{s_i}) := \{y \in \mathbb{R}^4 : y^\top A_i(x_{s_i})y \leq 0\};$$

$x_{s_i} := [p_i^\top, \zeta_i^\top]^\top \in \mathbb{M} := \mathbb{R}^3 \times \mathbb{S}^3$  is the  $i$ th agent's center of mass pose, where  $p_i \in \mathbb{R}^3$  is its inertial position and  $\zeta_i := [\varphi_i, \epsilon_i^\top]^\top \in \mathbb{S}^3$  its unit quaternion-based orientation, with  $\varphi_i \in \mathbb{R}$ ,  $\epsilon_i \in \mathbb{R}^3$  its scalar and vector parts, respectively, subject to  $\|\zeta_i\| = 1$ ;  $A_i(x_{s_i}) := T_i^{-\top}(x_{s_i}) \hat{A}_i T_i^{-1}(x_{s_i})$ , with  $\hat{A}_i := \text{diag}\{l_{x,i}^{-2}, l_{y,i}^{-2}, l_{z,i}^{-2}, -1\}$ , corresponding to the principal axis lengths  $l_{x,i}, l_{y,i}, l_{z,i} \in \mathbb{R}_{>0}$  of agent  $i$ 's ellipsoid, and  $T_i \in \mathbb{SE}(3)$  is the transformation matrix describing the translation and orientation of agent  $i$ 's center of mass,  $\forall i \in \mathcal{N}$ . The agents' motion follows the standard Lagrangian dynamics



**Figure 4.17:** The resulting control inputs  $u_i(t)$ ,  $i \in \{1, \dots, 6\}$ ,  $t \in [0, 277]$  s.



**Figure 4.18:** Ellipsoid approximation of (a) the rigid links of a robotic manipulator, (b) a mobile robot (top and front view).

(similar to (4.32)):

$$\dot{x}_{s_i} = \bar{E}_\zeta(\zeta_i)v_i \quad (4.40a)$$

$$M_i(x_{s_i})\dot{v}_i + C_i(x_{s_i}, v_i)v_i + g_i(x_{s_i}) + f_i(x_{s_i}, v_i) + d_i(t) = u_i, \quad (4.40b)$$

where  $v_i := [\dot{p}_i^\top, \omega_i^\top]^\top$  is agent  $i$ 's velocity, with  $\omega_i \in \mathbb{R}^3$  being its angular velocity,  $\bar{E}_\zeta : \mathbb{S}^3 \rightarrow \mathbb{R}^{7 \times 6}$  is the matrix mapping the quaternion rates to velocities, defined as  $\bar{E}_\zeta := \text{diag}\{I_3, E(\zeta)\}$  and  $E(\zeta)$  as defined in Section 2.2.1. The rest of the terms are the same as in (4.32), with unknown dynamic parameters in  $M_i$ ,  $C_i$ ,  $g_i$ , and unknown  $f_i$ ,  $d_i$ ,  $\forall i \in \mathcal{N}$ . Here we also consider

that  $u_i$  is decomposed as  $u_i = u_{f,i} + u_{s,i}$ , where  $u_{f,i}$  is a bounded term that is responsible for some (potentially cooperative) task, and  $u_{s,i}$  is a control term to be designed in order to achieve multi-agent decentralized collision avoidance,  $\forall i \in \mathcal{N}$ . More specifically, we consider that  $\phi_s(x_s) \in \mathbb{R}_{\geq 0}$  is a term that corresponds to the cooperative task dictated by  $u_{f,i}$ , with

$$u_{f,i} = \bar{E}_\zeta(\zeta_i)^\top \frac{\partial \phi_s(x_s)}{\partial x_{s_i}},$$

$\forall i \in \mathcal{N}$ ,  $c_1(x_s) \leq \phi_s(x_s) \leq c_2(x_s)$ , for continuous positive definite functions  $c_1, c_2$ , and nonempty sets  $\{x_s \in \mathcal{X}_s : x_s = \phi_s^{-1}(y)\}$ ,  $\forall y \in \mathbb{R}_{\geq 0}$ , where  $x_s := [x_{s_1}^\top, \dots, x_{s_N}^\top]^\top$ , and

$$\mathcal{X}_s := \{x_s \in \mathbb{M}^N : \mathcal{A}_i(x_{s_i}) \cap \mathcal{A}_j(x_{s_j}) = \emptyset, \forall i, j \in \mathcal{N}, i \neq j\};$$

$\phi_s$  can be also a function of  $\tilde{x}_s := [p_1^\top - p_2^\top, \dots, p_N^\top - p_{N-1}^\top, \zeta_1^\top, \dots, \zeta_N^\top]^\top$  that concerns potential formation control objectives. Then  $\mathcal{X}_s$  becomes

$$\{\tilde{x}_s \in \mathbb{R}^{\frac{3N(N-1)}{2}} \times \mathbb{S}^{3N} : \mathcal{A}_i(x_{s_i}) \cap \mathcal{A}_j(x_{s_j}) = \emptyset, \forall i, j \in \mathcal{N}, i \neq j\}.$$

The conditions for  $\phi$  are satisfied by standard quadratic functions, e.g.,

$$\phi_s(x_s) = \sum_{i \in \mathcal{N}} \{\|p_i - p_{d_i}\|^2 + e_{\zeta_i}^2\}$$

(for multi-agent navigation) or

$$\phi_s(\tilde{x}) = \sum_{(i,j) \in \mathsf{F}} \{\|p_i - p_j - p_{d_{i,j}}\|^2 + e_{\zeta_{i,j}}^2\}$$

(for formation) for sufficiently distant  $p_{d_i}, p_{d_{i,j}}$ , where  $\mathsf{F}$  is a potential formation set and  $e_{\zeta_i}, e_{\zeta_{i,j}}$  represent appropriate quaternion errors (see Section 2.2.3). Note that  $\phi_s$  and  $u_{f,i}$  are *not* responsible for collision avoidance or compensating model uncertainties.

The terms  $M_i$  and  $C_i$  satisfy Property 4.1,  $\forall i \in \mathcal{N}$ , as well as a slightly modified version of Property 4.2 that accounts only for  $g$ :

**Property 4.3.** The gravity terms of (4.40) can be written as  $g_i(z) = Y_{g_i}(z)\theta_{g_i}$ ,  $\forall z \in \mathbb{M}, i \in \mathcal{N}$ , where  $Y_{g_i} : \mathbb{M} \rightarrow \mathbb{R}^{6 \times \ell}$  are known continuous matrices, and  $\theta_{g_i} \in \mathbb{R}^\ell$ ,  $\ell \in \mathbb{N}$ , are constant but unknown dynamic parameters of the agents,  $\forall i \in \mathcal{N}$ .

Moreover, the unknown disturbances  $d_i$  satisfy Assumption 4.5 for unknown  $d_{b_i}$ ,  $\forall i \in \mathcal{N}$ , whereas we impose a stronger assumption on  $f_i$  for simplicity:

**Property 4.4.** [?] The friction terms are dissipative, i.e.,  $v_i^\top f_i(x_{s_i}, v_i) > 0$ ,  $\forall x_{s_i} \in \mathbb{M}, v_i \neq 0, i \in \mathcal{N}$ .

As before, we consider that each robot has a limited sensing radius  $\varsigma_i \in \mathbb{R}_{>0}$ , with the restriction now  $\varsigma_i > \max\{l_{x,i}, l_{y,i}, l_{z,i}\} + \max_{j \in \mathcal{N}} \{\max\{l_{x,j}, l_{y,j}, l_{z,j}\}\} + \varepsilon$  for an arbitrarily small positive constant  $\varepsilon$ , which implies that the agents can sense each other without colliding. Based on this, the undirected time-varying graph that models the topology of the multi-robot network becomes now  $\mathcal{G}(p) := (\mathcal{N}, \mathcal{E}(p))$ , with  $\mathcal{E}(p) := \{(i, j) \in \mathcal{N}^2 : \|p_i - p_j\| \leq \min\{\varsigma_i, \varsigma_j\}\}$ ,  $p := [p_1^\top, \dots, p_N^\top]^\top$ , and we further define the agent time-varying neighborhood  $\mathcal{N}_i(p) := \{j \in \mathcal{N} : \|p_i - p_j\| < \varsigma_i\}$ ,  $\forall i \in \mathcal{N}$ . Moreover, we consider again the complete graph  $\bar{\mathcal{G}} := (\mathcal{N}, \bar{\mathcal{E}})$ , with  $\bar{\mathcal{E}} := \{(i, j), \forall i, j \in \mathcal{N}, i < j\}$ ,  $\bar{K} := |\bar{\mathcal{E}}| = \frac{N(N-1)}{2}$  and an edge numbering set  $\bar{\mathcal{K}} := \{1, \dots, \bar{K}\}$ . Finally, we use the same notation for  $(k_1, k_2)$  that give the robot indices that form edge  $k$ .

As discussed before, the agents need to avoid collisions with each other, while executing their task, dictated by  $u_{f,i}$ . To that end, we aim to design *closed-form* barrier functions and decentralized feedback control laws  $u_{s,i}$  that guarantee collision avoidance among the ellipsoidal agents, while compensating appropriately for the model uncertainties and the external disturbances. Formally, the treated problem is the following:

**Problem 4.3.** Given  $N$  3D ellipsoidal autonomous agents with the uncertain Lagrangian dynamics (4.40) executing tasks dictated by  $u_{f,i}$ , design

1. closed-form barrier functions that encode collision avoidance of the agents,
2. decentralized control laws in  $u_{s,i}$  that guarantee inter-agent collision avoidance, i.e.,  $\mathcal{A}_i(x_{s_i}(t)) \cap \mathcal{A}_j(x_{s_j}(t)) = \emptyset$ ,  $\forall i, j \in \mathcal{N}, i \neq j$ , as well as boundedness of all closed loop signals.

#### 4.4.2 Problem Solution

This section describes the proposed solution to Problem 4.3. In order to deal with the ellipsoidal collision avoidance, we employ results from computer graphics that are related to detection of ellipsoid collision and we build appropriate barrier functions whose boundedness implies the collision-free trajectories. Moreover, we use adaptive and discontinuous control laws to appropriately compensate for the uncertainties and external disturbances of (4.40).

We employ first the results described in Proposition G.8 of Appendix G to build an appropriate ellipsoidal barrier function. Note, however, that these results concern planar ellipsoids and cannot be straightforwardly extended

to the 3D case, which is the case of the considered multi-agent system. For that reason, we consider the respective planar projections. For an ellipsoid  $\mathcal{A}_i, i \in \mathcal{N}$ , we denote as  $\mathcal{A}_i^{xy}, \mathcal{A}_i^{xz}, \mathcal{A}_i^{yz}$  its projections on the planes  $x-y$ ,  $x-z$  and  $y-z$ , respectively, with corresponding matrix terms  $A_i^{xy}, A_i^{xz}, A_i^{yz}$ , i.e.,

$$\mathcal{A}_i^s(x_{s_i}) := \{y \in \mathbb{R}^3 : y^\top A_i^s(x_{s_i})y \leq 0\}, \forall s \in \{xy, xz, yz\}.$$

Note that in order for  $\mathcal{A}_i, \mathcal{A}_j$  to collide (touch externally), all their projections on the three planes must also collide, i.e.,

$$\begin{aligned} \mathcal{A}_i(x_{s_i}) \cap \mathcal{A}_j(x_{s_j}) \neq \emptyset \wedge \text{Int}(\mathcal{A}_i(x_{s_i})) \cap \text{Int}(\mathcal{A}_j(x_{s_j})) = \emptyset &\Leftrightarrow \\ \mathcal{A}_i^s(x_{s_i}) \cap \mathcal{A}_j^s(x_{s_j}) \neq \emptyset \wedge \text{Int}(\mathcal{A}_i^s(x_{s_i})) \cap \text{Int}(\mathcal{A}_j^s(x_{s_j})) = \emptyset, \forall s &\in \{xy, xz, yz\}. \end{aligned}$$

Therefore,  $\mathcal{A}_i$  and  $\mathcal{A}_j$  do not collide if and only if  $\mathcal{A}_i^s(x_{s_i}) \cap \mathcal{A}_j^s(x_{s_j}) = \emptyset$  for some  $s \in \{xy, xz, yz\}$ . In view of Proposition G.8 of Appendix G, that means that the characteristic equations

$$f_{i,j}^s(\lambda) := \det(\lambda A_i^s(x_{s_i}) - A_j^s(x_{s_j})) = 0$$

must always have one positive real root and two negative distinct roots for at least one  $s \in \{xy, xz, yz\}$ . Hence, by denoting the discriminant of  $f_{i,j}^s(\lambda) = 0$  as  $\Delta_{i,j}^s(x_{s_i}, x_{s_j})$ , Proposition G.7 of Appendix G suggests that  $\Delta_{i,j}^s(x_{s_i}, x_{s_j})$  must remain always positive for at least one  $s \in \{xy, xz, yz\}$ , since a collision would imply  $\Delta_{i,j}^s(x_{s_i}, x_{s_j}) = 0, \forall s \in \{xy, xz, yz\}$ . Therefore, by defining the smooth function [?]

$$\sigma(z) := \begin{cases} \exp(-\frac{1}{z}), & z > 0 \\ 0, & z \leq 0 \end{cases} \quad (4.41)$$

we conclude that  $\mathcal{A}_i$  and  $\mathcal{A}_j$  do not collide if and only if

$$\sigma(\Delta_{i,j}^{xy}(x_{s_i}, x_{s_j})) + \sigma(\Delta_{i,j}^{xz}(x_{s_i}, x_{s_j})) + \sigma(\Delta_{i,j}^{yz}(x_{s_i}, x_{s_j})) > 0,$$

since a collision would result in  $\Delta_{i,j}^s(x_{s_i}, x_{s_j}) = 0 \Leftrightarrow \sigma(\Delta_{i,j}^s(x_{s_i}, x_{s_j})) = 0, \forall s \in \{xy, xz, yz\}$ . We aim now at defining a decentralized continuously differentiable function for each edge  $k \in \bar{\mathcal{K}}$  that incorporates the collision avoidance property of agents  $k_1, k_2$ . We need first the following result regarding the discriminant of  $f_{i,j}^s(\lambda) = 0$ :

**Proposition 4.2.** Let  $\Delta_1, \Delta_2$  be the discriminants of  $f_1(\lambda) := \det(\lambda A - B) = 0, f_2(\lambda) := \det(\lambda B - A) = 0$ , respectively, where  $A, B \in \mathbb{R}^{3 \times 3}$ . Then  $\Delta_1 = \Delta_2$ .

*Proof.* Let

$$\det(\lambda A - B) = 0 \Leftrightarrow f_1(\lambda) := c_3\lambda^3 + c_2\lambda^2 + c_1\lambda + c_0 = 0,$$

with  $c_\ell \in \mathbb{R}$ ,  $\forall \ell \in \{0, \dots, 3\}$ . It can be verified that

$$\det(\lambda B - A) = 0 \Leftrightarrow f_2(\lambda) = -c_0\lambda^3 - c_1\lambda^2 - c_2\lambda - c_3 = 0.$$

Let  $\lambda_1, \lambda_2, \lambda_3$  be the solutions of  $f_1(\lambda) = 0$ , i.e.  $f_1(\lambda_1) = f_1(\lambda_2) = f_1(\lambda_3) = 0$ , and  $\lambda_1\lambda_2\lambda_3 = -\frac{c_0}{c_3}$ . By substituting  $\frac{1}{\lambda_\ell}$  in  $f_2(\lambda)$ ,  $\ell \in \{1, 2, 3\}$ , we obtain

$$-c_0\lambda_\ell^{-3} - c_1\lambda_\ell^{-2} - c_2\lambda_\ell^{-1} - c_3 = -(c_3\lambda_\ell^3 + c_2\lambda_\ell^2 + c_1\lambda_\ell + c_0) = -f_1(\lambda_\ell) = 0.$$

Hence,  $\frac{1}{\lambda_1}, \frac{1}{\lambda_2}, \frac{1}{\lambda_3}$  are the solutions of  $f_2(\lambda) = 0$ . The discriminants of  $f_1(\lambda) = 0$  and  $f_2(\lambda) = 0$  are

$$\Delta_1 = c_3^4(\lambda_1 - \lambda_2)^2(\lambda_1 - \lambda_3)^2(\lambda_2 - \lambda_3)^2$$

and

$$\begin{aligned}\Delta_2 &= (-c_0)^4 (\lambda_1^{-1} - \lambda_2^{-1})^2 (\lambda_1^{-1} - \lambda_3^{-1})^2 (\lambda_2^{-1} - \lambda_3^{-1})^2 \\ &= c_0^4(\lambda_1\lambda_2\lambda_3)^{-4}(\lambda_2 - \lambda_1)^2(\lambda_3 - \lambda_1)^2(\lambda_3 - \lambda_2)^2,\end{aligned}$$

respectively, which, by substituting  $c_0 = -c_3\lambda_1\lambda_2\lambda_3$ , becomes  $\Delta_2 = \Delta_1$ .  $\square$

Therefore, we conclude that the discriminants  $\Delta_{i,j}^s()$  and  $\Delta_{j,i}^s()$  of  $\det(\lambda A_i^s(x_{s_i}) - A_j^s(x_{s_j})) = 0$  and  $\det(\lambda A_j^s(x_{s_j}) - A_i^s(x_{s_i})) = 0$ , respectively, are the same, for all  $s \in \{xy, xz, yz\}$ . Hence, we can define uniquely for each edge  $k \in \bar{\mathcal{K}}$  the continuously differentiable function  $\Delta_k : \mathbb{K}^2 \rightarrow \mathbb{R}_{\geq 0}$ , with

$$\begin{aligned}\Delta_k(x_{s_{k_1}}, x_{s_{k_2}}) &:= \sigma(\Delta_{k_1, k_2}^{xy}(x_{s_{k_1}}, x_{s_{k_2}})) + \sigma(\Delta_{k_1, k_2}^{xz}(x_{s_{k_1}}, x_{s_{k_2}})) \\ &\quad + \sigma(\Delta_{k_1, k_2}^{yz}(x_{s_{k_1}}, x_{s_{k_2}})),\end{aligned}\tag{4.42}$$

which needs to remain positive for all times in order to achieve the collision avoidance property, i.e.,  $\Delta_k(x_{s_{k_1}}(t), x_{s_{k_2}}(t)) > 0$ ,  $\forall t \in \mathbb{R}_{\geq 0}, k \in \bar{\mathcal{K}}$ . Note that, in view of Proposition 4.2, the agents  $k_1$  and  $k_2$  can calculate (4.42) based on  $\Delta_{k_1, k_2}^s(x_{s_{k_1}}, x_{s_{k_2}})$  and  $\Delta_{k_2, k_1}^s(x_{s_{k_2}}, x_{s_{k_1}})$ , respectively,  $\forall s \in \{xy, xz, yz\}, k \in \bar{\mathcal{K}}$ .

We still need to incorporate the fact that agents have a limited sensing radius, and that agent  $i$  does not have access to the functions  $\Delta_{i,j}^s(x_{s_i}, x_{s_j})$ , when  $j \notin \mathcal{N}_i(p)$ . To that end, we define first the greatest lower bound of  $\Delta_k$  when both agents  $k_1, k_2$  are in each other's sensing radius, i.e.,

$$\tilde{\Delta}_k := \inf_{\substack{(x_{s_{k_1}}, x_{s_{k_2}}) \in \mathbb{M}^2 \\ \|p_{k_1} - p_{k_2}\| \leq \min\{\varsigma_{k_1}, \varsigma_{k_2}\}}} \{\Delta_k(x_{s_{k_1}}, x_{s_{k_2}})\}, \forall k \in \bar{\mathcal{K}}.\tag{4.43}$$

Since  $\varsigma_i > \max\{l_{x,i}, l_{y,i}, l_{z,i}\} + \max_{j \in \mathcal{N}} \{\max\{l_{x,j}, l_{y,j}, l_{z,j}\}\} + \varepsilon$ ,  $\forall i \in \mathcal{N}$ , it follows that there exists a positive constant  $\varepsilon_\Delta$  such that  $\tilde{\Delta}_k \geq \varepsilon_\Delta > 0$ ,  $\forall k \in \bar{\mathcal{K}}$ .

$\bar{\mathcal{K}}$ . Next, we define the smooth switching functions  $\beta_k : \mathbb{R}_{\geq 0} \rightarrow [0, \bar{\beta}_m]$ , with [? ]

$$\beta_k(z) = \bar{\beta}_k \frac{\sigma(z)}{\sigma(z) + \sigma(\bar{\Delta}_k - z)}, \quad (4.44)$$

where  $\bar{\Delta}_k$  is a positive constant satisfying  $\bar{\Delta}_k < \tilde{\Delta}_k$ ,  $\forall k \in \bar{\mathcal{K}}$ . Then, by choosing  $\beta_k := \beta_k(\gamma_\sigma \Delta_k(x_{s_{k_1}}, x_{s_{k_2}}))$ , where  $\gamma_\sigma$  is a positive scaling constant, we incorporate the limited sensing radius of the agents in the collision avoidance scheme, since  $\frac{\partial \beta_k(z)}{\partial z}$  vanishes when  $k_1 \notin \mathcal{N}_{k_2}(p)$  or  $k_2 \notin \mathcal{N}_{k_1}(p)$ , i.e., when at least one of the agents that form edge  $k$  lies outside the sensing range of the other agent. Note that  $\beta_k$  are similar to the switches defined in Section 4.3.2. The terms  $\bar{\beta}_k$  can be any positive constants,  $\forall k \in \bar{\mathcal{K}}$ . All the necessary information for the construction of the functions  $\beta_k$ ,  $\Delta_k$ , i.e., the constants  $\bar{\Delta}_k$ ,  $\bar{\beta}_k$  and the lengths  $l_{x,i}$ ,  $l_{y,i}$ ,  $l_{z,i}$ ,  $i \in \mathcal{N}$ , can be transmitted off-line to the agents.

We can now define a suitable barrier function for each edge  $k \in \bar{\mathcal{K}}$  as any continuously differentiable function  $b_k : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$  with the property  $\lim_{z \rightarrow 0} b_k(z) = \infty$ , e.g.,  $b_k(z) = \frac{1}{z}$ ,  $k \in \bar{\mathcal{K}}$ . The barrier function for edge  $k$  is then  $b_k := b_k(\beta_k)$ ,  $\forall k \in \bar{\mathcal{K}}$ .

We propose now a decentralized feedback control law for the solution of Problem 4.3. Firstly, as in Section 4.3.2, we define the estimations of the unknown terms  $\theta_{g_i} \in \mathbb{R}^\ell$  and  $d_{b_i} \in \mathbb{R}$  as  $\hat{\theta}_{g_i} \in \mathbb{R}^\ell$  and  $\hat{d}_{b_i} \in \mathbb{R}$ , with the respective errors  $\tilde{\theta}_{g_i} := \hat{\theta}_{g_i} - \theta_{g_i}$  and  $\tilde{d}_{b_i} := \hat{d}_{b_i} - d_{b_i}$ ,  $\forall i \in \mathcal{N}$ . By using adaptive and discontinuous control techniques, we prove in the following that these estimations compensate appropriately for the unknown terms, without necessarily converging to them. In particular, we design the feedback control laws for  $u_{s,i} : \mathcal{X}_{s_i} \rightarrow \mathbb{R}^6$  as

$$u_{s,i} := u_{s,i}(\chi_{s_i}) = \sum_{k \in \bar{\mathcal{K}}} \alpha_{i,k} \kappa_k \bar{E}_\zeta(\zeta_i)^\top \frac{\partial \Delta_k}{\partial x_{s_i}} + Y_i(x_{s_i}) \hat{\theta}_{g_i} - k_{v_i} v_i - \hat{d}_{b_i} \text{sgn}(v_i), \quad (4.45)$$

where  $\chi_{s_i} := [x_s^\top, v_i^\top, \hat{\theta}_{g_i}^\top, \hat{d}_{b_i}]^\top$ ,  $\mathcal{X}_{s_i} := \mathcal{X}_s \times \mathbb{R}^{7+\ell}$ , with  $\mathcal{X}_s$  as defined in Section 4.4.1. Moreover,  $\alpha_{i,k} = -1$  if agent  $i$  is part of edge  $k$ , and  $\alpha_{i,k} = 0$  otherwise,  $\forall i \in \mathcal{N}$ ,  $k \in \bar{\mathcal{K}}$ ,  $\kappa_k := \frac{\partial b_k(\beta_k)}{\partial \beta_k} \frac{\partial \beta_k(\Delta_k)}{\partial \Delta_k}$ ,  $\forall k \in \bar{\mathcal{K}}$ , and  $k_{v_i}$  are positive constant gains. Finally, we design the associated adaptation laws

$$\left. \begin{array}{lcl} \dot{\hat{\theta}}_{g_i} &:=& -\gamma_{i,\theta} Y_i(x_{s_i})^\top v_i \\ \dot{\hat{d}}_{b_i} &:=& \gamma_{i,d} \|v_i\|_1 \end{array} \right\} \forall i \in \mathcal{N}, \quad (4.46)$$

with arbitrary bounded initial conditions, where  $\gamma_{\theta,i}$  and  $\gamma_{d,i}$  are positive gains,  $\forall i \in \mathcal{N}$ . The correctness of (4.45)-(4.46) is shown in the following theorem:

**Theorem 4.5.** Consider a multi-agent system comprised of 3D ellipsoidal agents and subject to the dynamics (4.40) at a collision-free initial configuration, i.e.,  $\mathcal{A}_i(x_{s_i}(0)) \cap \mathcal{A}_j(x_{s_j}(0)) = \emptyset$ ,  $\forall i, j \in \mathcal{N}$  with  $i \neq j$ . Then, application of the control and adaptation laws (4.45), (4.46) guarantees that the agents avoid collisions for all times, i.e.,  $\mathcal{A}_i(x_{s_i}(t)) \cap \mathcal{A}_j(x_{s_j}(t)) = \emptyset$ ,  $\forall i, j \in \mathcal{N}$  with  $i \neq j$ ,  $t \in \mathbb{R}_{\geq 0}$ , with all closed loop signals being bounded. Moreover,  $\lim_{t \rightarrow \infty} v_i(t) = 0, \forall i \in \mathcal{N}$ .

*Proof.* Consider the vector  $\chi_s := [x_s^\top, v^\top, \tilde{\theta}_g^\top, \tilde{d}_b^\top]^\top \in \tilde{\mathcal{X}}_s := \mathcal{X}_s \times \mathbb{R}^{7N+\ell N}$ ,  $v := [v_1^\top, \dots, v_N^\top]^\top \in \mathbb{R}^{6N}$ ,  $\tilde{d}_b := [\tilde{d}_{b_1}, \dots, \tilde{d}_{b_N}]^\top \in \mathbb{R}^N$ ,  $\tilde{\theta}_g := [\tilde{\theta}_{g_1}^\top, \dots, \tilde{\theta}_{g_N}^\top]^\top \in \mathbb{R}^{\ell N}$ . Since the initial configuration is collision-free, it holds that  $\chi_s(0) \in \tilde{\mathcal{X}}_s$ . By combining (4.40), (4.45), and (4.46), it can be verified that the conditions of Prop. 3 of Prop. A.1 in Appendix A are satisfied and hence we conclude that at least one Filippov solution exists and any such solution satisfies  $\chi_s : [0, t_{\max}) \rightarrow \tilde{\mathcal{X}}_s$  for a positive  $t_{\max}$ . Define

$$\mathbf{z}_s := [\phi_s, b_1, \dots, b_{\bar{M}}, v^\top, \tilde{\theta}_g^\top, \tilde{d}_b^\top]^\top \in \mathcal{Z}_s := \mathbb{R}^{\bar{K}+7N+\ell N+1},$$

where  $\phi_s$  is the cooperative term defined in Section 4.4.1. Note that  $\mathbf{z}_s(0) \in \mathcal{Z}_s$  and, for any finite  $r_s$ ,  $\mathbf{z}_s \in \bar{\mathcal{B}}(0, r_s) \subset \mathcal{Z}_s \Leftrightarrow \chi_s \in \tilde{\mathcal{X}}_s$ , which we prove in the following. Define the function

$$V_s := V_s(\mathbf{z}_s) := \phi_s + \sum_{k \in \bar{\mathcal{K}}} b_k + \sum_{i \in \mathcal{N}} \left\{ \frac{1}{2} v_i^\top M_i(x_{s_i}) v_i + \frac{1}{2\gamma_{i,d}} \tilde{d}_{b_i}^2 + \frac{1}{2\gamma_{i,\theta}} \|\tilde{\theta}_{g_i}\|^2 \right\},$$

for which it holds that  $W_{s_1}(\mathbf{z}_s) \leq V_s(\mathbf{z}_s) \leq W_{s_2}(\mathbf{z}_s)$  for positive definite functions  $W_{s_1}, W_{s_2}$  on  $\mathcal{Z}_s$ . Since  $\mathbf{z}_s(0) \in \mathcal{Z}_s$ , we conclude that  $V_s(\mathbf{z}_s(0))$  is well defined, and hence there exists a finite constant  $\bar{V}_s$  such  $V_s(\mathbf{z}_s(0)) \leq \bar{V}_s$  and  $b_k(0) \leq \bar{V}_s$ ,  $\forall k \in \bar{\mathcal{K}}$ . By differentiating  $V_s$  along the solutions of the closed loop system and in view of Lemma A.2 we obtain  $\dot{V}_s \in \dot{\tilde{V}}_s := \cap_{\xi \in \partial V_s(\mathbf{z}_s)} \xi^\top \mathbf{K}[\dot{\mathbf{z}}_s]$ . Since  $V$  is continuously differentiable, the generalized gradient reduces to the standard gradient and therefore, after using Properties 4.1, 4.3, and grouping terms, we obtain

$$\begin{aligned} \max_{z \in \dot{\tilde{V}}_s} \{z\} &\leq \sum_{i \in \mathcal{N}} \left\{ \sum_{k \in \bar{\mathcal{K}}} \left[ \alpha_{i,k} \kappa_k \frac{\partial \Delta_k}{\partial x_{s_i}}^\top \bar{E}_\zeta(\zeta_i) \right] v_i + \|v_i\|_1 \|d_i(t)\|_1 + v_i^\top \left( u_i \right. \right. \\ &\quad \left. \left. - Y_i(x_{s_i}) \theta_{g_i} + \bar{E}_\zeta(\zeta_i)^\top \frac{\partial \phi(x_s)}{\partial x_{s_i}} \right) - v_i^\top f_i(v_i) + \frac{1}{\gamma_{i,d}} \tilde{d}_{b_i} \dot{d}_{b_i} + \frac{1}{\gamma_{i,\theta}} \tilde{\theta}_{g_i}^\top \dot{\theta}_{g_i} \right\}, \end{aligned}$$

By also using Property 4.4 and Assumption 4.5, substituting  $u_i = u_{f,i} + u_{s,i}$  with  $u_{f,i} = \bar{E}_\zeta(\zeta_i)^\top \frac{\partial \phi(x_s)}{\partial x_{s_i}}$  and (4.45), the adaptation laws (4.46), and using

$\tilde{d}_{b_i} = \hat{d}_{b_i} - d_{b_i}$ ,  $\tilde{\theta}_{g_i} = \hat{\theta}_{g_i} - \theta_{g_i}$  and the property  $z^\top \text{sign}(z) = \|z\|_1$ ,  $\forall z \in \mathbb{R}^n$ , we obtain

$$\max_{z \in \dot{\tilde{V}}_s} \{z\} \leq - \sum_{i \in \mathcal{N}} k_{v_i} \|v_i\|^2 =: W_s(\mathbf{z}_s).$$

Therefore,  $z \leq -W_s(\mathbf{z}_s(t))$ ,  $\forall z \in \dot{\tilde{V}}_s(\mathbf{z}_s(t))$ ,  $t \in [0, t_{\max})$ , where  $W_s : \mathcal{Z}_s \rightarrow \mathbb{R}_{\geq 0}$  is a positive semi-definite function defined on  $\mathcal{Z}_s$ . Hence, by applying Theorem A.6 of Appendix A, we conclude that  $t_{\max} = \infty$ ,  $\mathbf{z}_s(t)$  is bounded in the compact set  $\{\mathbf{z}_s \in \bar{\mathcal{B}}(0, r_s) : W_{s_2}(\mathbf{z}_s) \leq c\}$ ,  $\forall t \in \mathbb{R}_{\geq 0}$  for any  $r_s$  and  $c$  satisfying  $\bar{\mathcal{B}}(0, r_s) \subset \mathcal{Z}_s$ ,  $c < \min_{\|\mathbf{x}\|=r_s} W_{s_1}(\mathbf{z}_s)$ , and  $\lim_{t \rightarrow \infty} W_s(\mathbf{z}_s(t)) = 0 \Rightarrow \lim_{t \rightarrow \infty} v(t) = 0$ . Note that, since the sets  $\{x_s \in \mathcal{X}_s : x_s = \phi_s^{-1}(y)\}$  are nonempty,  $r_s$  can be chosen arbitrarily large, corresponding to all collision-free initial configurations. Therefore, inter-agent collisions are avoided, and the adaptation signals  $\hat{\theta}_{g_i}$ ,  $\hat{d}_{b_i}$ , remain bounded,  $\forall i \in \mathcal{N}$ ,  $t \in \mathbb{R}_{\geq 0}$ . The continuity of the terms  $Y_i(\cdot)$  implies also their boundedness and hence the boundedness of the control signals (4.45), (4.46),  $t \in \mathbb{R}_{\geq 0}$ .  $\square$

**Remark 4.6.** It can be verified that  $\det(\lambda A_{k_1}^s(x_{s_{k_1}}) - A_{k_2}^s(x_{s_{k_2}}))$ , and hence  $b_k$ , are functions of  $p_{k_1} - p_{k_2}$ ,  $\zeta_{k_1}$ ,  $\zeta_{k_2}$ . Therefore, if  $\phi_s$  is a function of  $\tilde{x}$ , the aforementioned analysis still holds by setting  $\mathcal{X}_s = \{\tilde{x} \in \mathbb{R}^{\frac{3N(N-1)}{2}} \times \mathbb{S}^3 : \mathcal{A}_i(x_{s_i}) \cap \mathcal{A}_j(x_{s_j}) = \emptyset, \forall i, j \in \mathcal{N}, i \neq j\}$ . Moreover, note that achievement of the objectives expressed by  $\phi_s$  is not pursued here and may not be necessarily guaranteed due to the potentially counteracting terms of  $u_i$ .

**Remark 4.7.** Since  $\Delta_{i,j}^s = \Delta_{j,i}^s$  (due to Proposition 4.2),  $\forall i, j \in \mathcal{N}$ ,  $i \neq j$ , the control scheme can be extended to directed communication graphs, by setting for the  $i$ th agent  $b_{i,j} = b_{i,j}(\beta_{i,j}(\Delta_{i,j}(x_{s_i}, x_{s_j})))$ ,  $\forall j \in \mathcal{N} \setminus \{i\}$ , with  $\Delta_{i,j}(x_{s_i}, x_{s_j})$  as in (4.42) and  $\beta_{i,j}$  as in (4.44),  $\tilde{\Delta}_{i,j}$  as in (4.43), and appropriately modifying the control law. Similarly, collision avoidance with static environment obstacles could be incorporated in the overall scheme.

#### 4.4.3 Simulation Results

We consider a simulation example with  $N = 8$  rigid bodies in  $\mathbb{SE}(3)$ , described by ellipsoids with axes lengths  $l_{x,i} = 0.5\text{m}$ ,  $l_{y,i} = 0.3\text{m}$ ,  $l_{z,i} = 0.2\text{m}$ ,  $\forall i \in \mathcal{N}$ . The initial poses are (in m)

$$\begin{aligned} p_1 &= [3, 3, 0]^\top, & p_2 &= -[3, 3, 0]^\top \\ p_3 &= [3, -3, 0]^\top, & p_4 &= [-3, 3, 0]^\top \\ p_5 &= [3, 3, 3]^\top, & p_6 &= -[3, 3, 3]^\top \\ p_7 &= [3, -3, 3]^\top, & p_8 &= [-3, 3, -3]^\top \end{aligned}$$

$$\begin{aligned}\zeta_1 &= \zeta_8 = [0.769, 0.1696, 0.6153, 0.0358]^\top \\ \zeta_2 &= \zeta_6 = [0.8488, -0.3913, -0.0598, -0.3505]^\top \\ \zeta_3 &= \zeta_5 = [0.7638, -0.5283, -0.3275, -0.1738]^\top \\ \zeta_4 &= \zeta_7 = [0.7257, 0.3081, 0.3714, 0.4904]^\top\end{aligned}$$

We consider that  $\phi_s(x_s)$  describes an independent multi-agent navigation objective, with desired configurations as

$$\begin{aligned}p_{1_d} &= p_2, p_{2_d} = p_1, p_{3_d} = p_4, p_{4_d} = p_3, \\ p_{5_d} &= p_6, p_{6_d} = p_5, p_{7_d} = p_8, p_{8_d} = p_7,\end{aligned}$$

and  $\zeta_{i_d} = [1, 0, 0, 0]^\top$ ,  $\forall i \in \mathcal{N}$ . We set the errors  $e_{p_i} := p_i - p_{i_d}$  and  $e_{\zeta_i} := [e_{\varphi_i}, e_{\epsilon_i}^\top]^\top := \zeta_{i_d} \cdot \zeta_i^+$ , and  $e_{\varphi_i}, e_{\epsilon_i}$  are the scalar and vector parts, respectively, of the quaternion error (see Section 2.2). The desired quaternion configuration is achieved when  $e_{\zeta_i} = [\pm 1, 0, 0, 0]^\top$  and hence the function  $\phi_s(x_s)$  is chosen as

$$\phi_s = \sum_{i \in \mathcal{N}} \left( \frac{1}{2} \|p_i - p_{i_d}\|^2 + 1 - e_{\varphi_i}^2 \right),$$

with

$$\dot{\phi}_s = \sum_{i \in \mathcal{N}} ((p_i - p_{i_d})^\top \dot{p}_i - e_{\varphi_i} e_{\epsilon_i}^\top \omega_i).$$

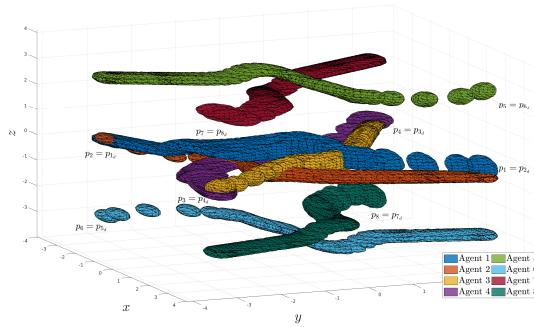
The control inputs  $u_{f,i}$  are therefore chosen as

$$u_{f,i} = [p_{i_d}^\top - p_i^\top, e_{\varphi_i} e_{\epsilon_i}^\top]^\top,$$

$\forall i \in \mathcal{N}$ . The agent masses and moments of inertia are chosen randomly in the interval  $(0, 0.2]$ . We also set  $f_i(x_i, v_i) = m_{f_i} \sin(w_{f_i} t + \phi_{f_i}) v_i$ ,  $d_i(t) = (1/m_{f_i}) \sin(w_{f_i} t + \phi_{f_i})$ ,  $\forall i \in \mathcal{N}$ , with the terms  $m_{f_i}$ ,  $w_{f_i}$ , and  $\phi_{f_i}$  chosen randomly in the interval  $(0, 5]$ ,  $\forall i \in \mathcal{N}$ . We choose  $b_k = \frac{1}{\beta_k}$ , with  $\beta_k = 1$ ,  $\bar{\Delta}_k = 10^4$ ,  $\gamma_\sigma = 10^{-40}$ ,  $\forall k \in \bar{\mathcal{K}}$ , and  $\hat{\theta}_{g_i}(0) = 0.1$ ,  $\hat{d}_{b_i}(0) = 0.2$ ,  $k_{v_i} = 1$ ,  $\forall i \in \mathcal{N}$ . The expressions for  $\Delta_k(x_{s_{k_1}}, x_{s_{k_2}})$  were derived by using the symbolic toolbox of MATLAB. Fig. 4.19 shows a 3D plot of the agent trajectories, and Fig. 4.20 shows the minimum of the barrier functions  $\min_{k \in \bar{\mathcal{K}}} \{\beta_k(t)\}$  (left), which is always positive, and the signals  $\gamma_i(t) := \|p_i - p_{i_d}\|^2 + 1 - e_{\varphi_i}^2$  and  $v_i(t)$  (right),  $\forall i \in \mathcal{N}$ ,  $t \in [0, 15]$ . Finally, Fig. 4.21 depicts the control inputs of the agents. A short video that demonstrates the aforementioned simulation example can be found in <https://youtu.be/IAni7zIMM7k>.

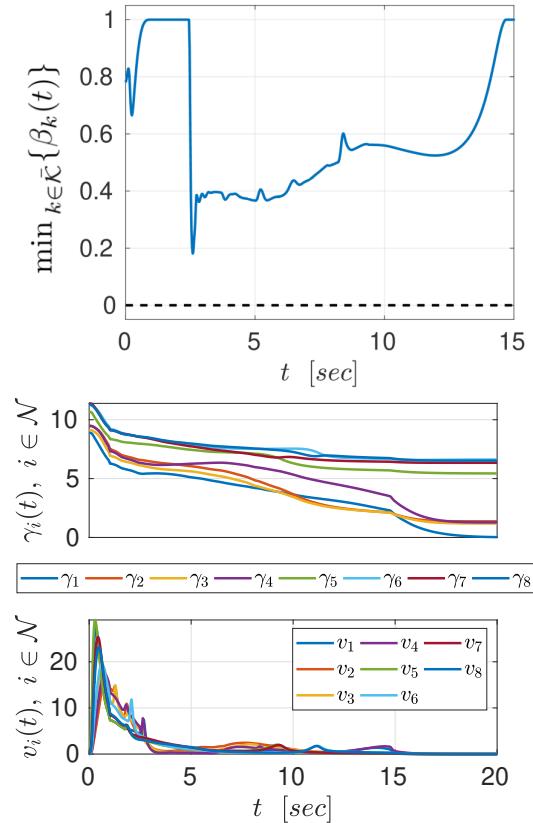
## 4.5 Conclusion

This chapter presented several continuous control algorithms for multi-agent coordination of systems with uncertain dynamics. Firstly, we develop

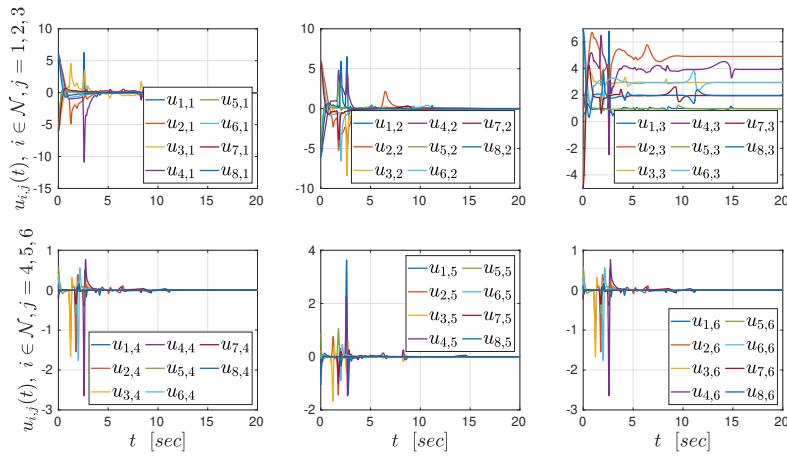


**Figure 4.19:** The evolution of agent trajectories  $\forall t \in [0, 20]$  sec.

an adaptive control scheme for the almost global navigation of a single robotic agent in a workspace with obstacles, which is then extended to a decentralized multi-robot scheme. Secondly, we develop a decentralized adaptive multi-agent algorithm for the leader-follower coordination: A leader agent converges to a predefined goal point while the entire team avoids collision with each other and maintains connectivity. Finally, we develop a closed-form barrier function that encodes the distance between 3D ellipsoids, and design a decentralized collision avoidance control scheme for a team of ellipsoidal robotic agents, while compensating for the dynamic uncertainties.



**Figure 4.20:** Top: The evolution of the minimum of the functions  $\min_{k \in \bar{\mathcal{K}}} \{\beta_k(t)\}$ . Bottom: The evolution of the signals  $\gamma_i(t)$  and  $v_i(t)$ ,  $\forall i \in \mathcal{N}$ ,  $\forall t \in [0, 20]$  sec.



**Figure 4.21:** The control inputs of the agents  $u_i(t)$ ,  $\forall t \in [0, 20]$  sec,  $i \in \mathcal{N}$ .

## Chapter 5

# Abstractions of Multi-Agent and Multi-Agent-Object Systems

After designing continuous solutions to multi-agent problems, such as cooperative manipulation, formation, and navigation, we are ready to transit to the problem of multi-agent planning under temporal logic tasks. The content of the previous chapters can act as a means to obtain well-defined discrete representations (abstractions) of the continuous systems in hand. Therefore, this chapter addresses the motion and task planning of multi-agent *and* multi-agent-object systems (systems comprised of multiple robotic agents and objects) subject to temporal logic constraints, focusing both on the abstraction technique as well as the control synthesis for the accomplishment of the tasks.

More specifically, this chapter can be divided in two main parts. The first part tackles the motion planning of multi-robot teams under *local* linear temporal tasks, i.e., when each robotic agent has its own task. The second part addresses the case where *unactuated* objects of the workspace must satisfy a certain temporal logic task, with the robotic agents being responsible for their accomplishment.

## 5.1 Introduction

Temporal-logic-based motion planning has gained significant attention in recent years, as it provides a fully automated correct-by-design controller synthesis approach for autonomous robots. Temporal logics such as linear temporal logic (LTL) and metric interval temporal logic (MITL) provide formal high-level languages that can describe complex planning objectives. As already discussed in the previous chapters, standard control problems are restricted to point-to-point navigation, multi-agent formation control, or consensus. Ultimately, however, 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. The task specification is given as a temporal logic formula with respect to the discretized abstraction of the robot motion modeled as a finite transition system [? ? ? ? ]. Then a high-level discrete plan is found by off-the-shelf model-checking algorithms, given the finite transition system and the task specification [? ]. Temporal logics have been extensively used in the related literature for both single- and multi-agent systems, e.g., [? ? ? ? ? ? ? ? ? ? ? ? ? ? ? ? ? ].

A special and important class of autonomous robotic systems is the class of unmanned aerial vehicles (UAV), which can provide efficient multi-agent solutions in several problems, e.g., coverage or inspection. Control of aerial vehicles in a multi-agent setting has been well studied in the related literature. The standard problem of formation control for a team of aerial vehicles is addressed in [? ? ? ? ? ? ? ], whereas [? ? ? ? ? ] consider leader-follower formation approaches, where the latter also treats the problem of collision avoidance with static obstacles in the environment; [? ], [? ? ] and [? ] employ dynamic programming, Model Predictive Control and reachable set algorithms, respectively, for inter-agent collision avoidance, which is tackled also in [? ]. In [? ] the cooperative evader pursuit problem is treated. Aerial vehicles and temporal logic-based planning is considered in [? ], which addresses the vehicle routing problem using MTL specifications and in [? ], which approaches the LTL motion planning using MILP optimization techniques, both in a centralized manner. Markov Decision Processes are used for the LTL planning in [? ]. 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 [? ]. 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 [? ? ? ? ? ] for general systems and [? ? ] 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 [? ? ? ]), 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.

Furthermore, 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. More specifically, 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 robotic agents. *Time constraints* can be incorporated in the motion planning temporal logic-based problem via specific logics, such as Metric and Metric Interval Temporal Logic (MTL, MITL) [? ? ?], as well as Time Window Temporal Logic (TWTL), or Signal Temporal Logic (STL). Such languages have been for multi-agent motion planning in several works (e.g., [? ? ? ?]).

This chapter addresses the motion planning problem of multi-agent systems as well multi-agent-object systems subject to complex tasks, expressed as temporal logic specifications. Firstly, we develop decentralized control protocols for the navigation of a multi-robot team among predefined regions or interest in the workspace, while taking into collision and/or connectivity properties. We consider separately the cases of (i) aerial vehicles, and (ii) mobile robotic manipulators. This allows us to abstract the continuous multi-agent dynamics as discrete transition systems (abstractions), which then can be used to obtain a path that satisfies the given *local* LTL specifications, by employing formal method-based methodologies.

Secondly, we provide, similar to the first case, appropriate discrete abstractions for multi-robot-object systems, encoding the behavior of the robots as well as the *unactuated objects* in the workspace. The proposed abstraction design involves both multi-robot *safe* navigation as well as cooperative object transportation. The abstracted systems are then used to derive paths that satisfy the robotic agents' and the objects' (possibly timed) temporal goals.

Although the proposed control schemes from the previous chapters can be used, we provide new control ideas as alternatives for the derivation of the discrete abstractions.

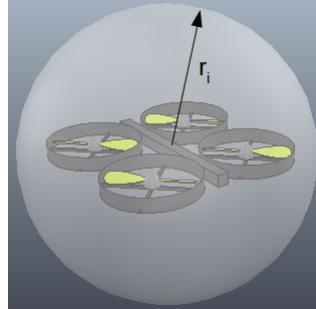


Figure 5.1: Bounding sphere of an aerial vehicle.

## 5.2 Decentralized Motion Planning with Collision Avoidance for a Team of UAVs under High Level Goals

We first describe a decentralized hybrid control algorithm for the motion planning of aerial vehicles subject to Linear Temporal Logic (LTL) specifications.

### 5.2.1 Problem Formulation

Consider  $N$  aerial agents operating in a static workspace that is bounded by a large sphere in 3-D space  $\mathcal{W} := \mathcal{B}(p_0, r_0)$ , where  $p_0 \in \mathbb{R}^3$  and  $r_0 \in \mathbb{R}_{>0}$  are the center and radius of  $\mathcal{W}$ . Within  $\mathcal{W}$  there exist  $K$  smaller spheres around points of interest, which are described by  $\pi_k := \bar{\mathcal{B}}(p_{\pi_k}, r_{\pi_k}) \subset \mathcal{W}$ , where  $p_{\pi_k} \in \mathbb{R}^3, r_{\pi_k} \in \mathbb{R}_{>0}$  are the central point and radius, respectively, of  $\pi_k$ . We denote the set of all  $\pi_k$  as  $\Pi = \{\pi_1, \dots, \pi_K\}$ . Moreover, we introduce a set of atomic propositions  $\Psi_i$  for each agent  $i \in \{1, \dots, N\}$  that indicates certain properties of interest of agent  $i$  in  $\Pi$  and are expressed as boolean variables. The properties satisfied at each region  $\pi_k$  are provided by the labeling function  $\mathcal{L}_i : \Pi \rightarrow 2^{\Psi_i}$ , which assigns to each region  $\pi_k, k \in \mathcal{K}_R := \{1, \dots, K\}$  the subset of the atomic propositions  $\Psi_i$  that are true in that region.

Each agent  $i \in \mathcal{N} := \{1, \dots, N\}$  occupies a bounding sphere  $\bar{\mathcal{B}}(p_i, r_i)$ , where  $p_i \in \mathbb{R}^3$  is the center and  $r_i \in \mathbb{R}_{>0}$  the radius of the sphere (Fig. 5.1). We also consider that  $r_i < r_{\pi_k}, \forall i \in \mathcal{N}, k \in \mathcal{K}_R$ , 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:

$$\dot{p}_i = u_i, i \in \mathcal{N}. \quad (5.1)$$

Moreover, similar to the previous chapter, we consider that agent  $i$  has a limited sensing range of  $\varsigma_i > \max_{i,j \in \mathcal{N}} (r_i + r_j)$ . Therefore, by defining the neighboring set  $\mathcal{N}_i := \{j \in \mathcal{N}, \text{ s.t. } \|p_i - p_j\| \leq \varsigma_i\}$ , agent  $i$  knows at each configuration the position of all  $p_j, \forall j \in \mathcal{N}_i$  as well as its own position  $p_i$ . The workspace is assumed to be perfectly known, i.e.,  $p_{\pi_k}, r_{\pi_k}$  are known to all agents, for all  $k \in \mathcal{K}_{\mathcal{R}}$ .

With the above ingredients, we provide the following definitions:

**Definition 5.1.** An agent  $i \in \mathcal{N}$  is in a region  $\pi_k, k \in \mathcal{K}_{\mathcal{R}}$  at a configuration  $p_i$ , denoted as  $\mathcal{A}_i(p_i) \in \pi_k$ , if and only if  $\bar{\mathcal{B}}(p_i, r_i) \subseteq \bar{\mathcal{B}}(p_{\pi_k}, r_{\pi_k})$ .

**Definition 5.2.** Assume that  $\mathcal{A}_i(p_i(t_0)) \in \pi_k, i \in \mathcal{N}, k \in \mathcal{K}_{\mathcal{R}}$  for some  $t_0 \geq 0$ . Then there exists a transition for agent  $i$  from region  $\pi_k$  to region  $\pi_{k'}, k' \in \mathcal{K}_{\mathcal{R}}$ , denoted as  $\pi_k \rightarrow_i \pi_{k'}$ , if and only if there exists a finite  $t_f \geq 0$  such that

1.  $\mathcal{A}_i(p_i(t_f)) \in \pi_{k'}$ ,
2.  $\bar{\mathcal{B}}(p_i(t), r_i) \subset \mathcal{W}$ ,
3.  $\bar{\mathcal{B}}(p_i(t), r_i) \cap \bar{\mathcal{B}}(p_{\pi_m}, r_{\pi_m}) = \emptyset$ ,
4.  $\bar{\mathcal{B}}(p_i(t), r_i) \cap \bar{\mathcal{B}}(p_{i'}(t), r_{i'}) = \emptyset, \forall m \in \mathcal{K}_{\mathcal{R}}$  with  $m \neq k, k', \forall i' \in \mathcal{N}$  with  $i' \neq i$  and  $t \in [0, t_f]$ .

Loosely speaking, an agent  $i$  can transit between two regions of interest  $\pi_k$  and  $\pi_{k'}$ , if there exists a bounded control trajectory  $u_i$  in (5.1) that takes agent  $i$  from  $\pi_k$  to  $\pi_{k'}$  while avoiding entering all other regions, colliding with the other agents, or exiting the workspace boundary.

Our goal is to control the multi-agent system subject to (5.1) so that each agent's behavior obeys a given specification over its atomic propositions  $\Psi_i$ .

**Definition 5.3.** Given a trajectory  $p_i(t)$  of agent  $i$ , its corresponding *behavior* is given by the infinite sequence  $\mathfrak{b}_i(\check{\psi}_i) := (p_{i_1}, \check{\psi}_{i_1})(p_{i_2}, \check{\psi}_{i_2}) \dots$ , with  $\check{\psi}_{i_m} \in 2^{\Psi_i}$  and  $\mathcal{A}_i(p_{i_m}) \in \pi_{k_m}, \check{\psi}_{i_m} \in \mathcal{L}_i(\pi_{k_m}), k_m \in \mathcal{K}_{\mathcal{R}}, \forall m \in \mathbb{N}$ .

The satisfaction of a LTL formula is provided by the following definition (see Appendix F for more details on LTL formulas).

**Definition 5.4.** The behavior  $\mathfrak{b}_i(\check{\psi}_i)$  satisfies an LTL formula  $\Phi$  if and only if  $\check{\psi}_i \models \Phi$ .

The control objectives are given for each agent separately as LTL formulas  $\Phi_i$  over  $\Psi_i, i \in \mathcal{N}$ . An LTL formula is satisfied if there exists a behavior  $\mathfrak{b}_i(\check{\psi}_i)$  of agent  $i$  that satisfies  $\Phi_i$ . Formally, the problem treated in this section is the following:

**Problem 5.1.** Given a set of aerial vehicles  $N$  subject to the dynamics (5.1) and  $N$  LTL formulas  $\Phi_i$ , over the respective atomic propositions  $\Psi_i, i \in \mathcal{N}$ , achieve behaviors  $\mathbf{b}_i$  that (i) yield satisfaction of  $\Phi_i, \forall i \in \mathcal{N}$  and (ii) guarantee inter-agent collision avoidance.

### 5.2.2 Problem Solution

We provide here the proposed solution to Problem 5.1, which consists of two main layers, that is, the design of a continuous control scheme, and the derivation of a high level path that satisfies  $\Phi_i$ .

#### Continuous Control Design

The first ingredient of our solution is the development of a decentralized feedback control law that establishes a transition relation  $\pi_k \rightarrow_i \pi_{k'}, \forall k, k' \in \mathcal{K}_{\mathcal{R}}$  according to Def. 5.2. The proposed approach is based on the concept of *Decentralized Navigation Functions*, introduced in [?], for which an overview can be found in Appendix C. 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  $\pi_k \rightarrow_i \pi_{k'}$ , as defined in Def. 5.2.

Initially, we define the set of “undesired” regions as  $\Pi_{k,k'} := \{\pi_m \in \Pi, m \in \mathcal{K}_{\mathcal{R}} \setminus \{k, k'\}\}$  and the corresponding free space  $\mathcal{F}_{i_{k,k'}} := \{p \in \mathcal{W}^N : \bar{\mathcal{B}}(p_i, r_i) \cap \bar{\mathcal{B}}(p_j, r_j) = \emptyset, \forall j \in \mathcal{N} \setminus \{j\}, \bar{\mathcal{B}}(p_i, r_i) \cap \pi = \emptyset, \forall \pi \in \Pi_{k,k'}\}$ , with  $p := [p_1^\top, \dots, p_N^\top]^\top$ . As the goal configuration we consider the centroid  $p_{\pi_{k'}}$  of  $\pi_{k'}$  and we construct the function  $\gamma_{i_{k'}} : \mathbb{R}^3 \rightarrow \mathbb{R}_{\geq 0}$  with  $\gamma_{i_{k'}}(p_i) := \|p_i - p_{\pi_{k'}}\|^2$ . For the collision avoidance between the agents, we employ the function  $G_i : \mathcal{F}_{i_{k,k'}} \rightarrow \mathbb{R}$  as defined in [?], which encodes the distances among the agents.

Moreover, we 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  $\alpha_{i_{k,k'}} : \mathbb{R}^3 \rightarrow \mathbb{R}$  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  $\alpha_{i,0} : \mathbb{R}^3 \rightarrow \mathbb{R}$  is a measure of the distance of agent  $i$  from the workspace boundary  $\alpha_{i,0}(p_i) := (r_0 - r_i)^2 - \|p_i - p_0\|^2$  and the function  $\alpha_{i,m} : \mathbb{R}^3 \rightarrow \mathbb{R}$  is a measure of the distance of agent  $i$  from the undesired regions  $\alpha_{i,m}(p_i) := \|p_i - p_m\|^2 - (r_i + r_m)^2$ .

With the above ingredients, we construct the following navigation function  $\varphi_{i_{k,k'}} : \mathcal{F}_{i_{k,k'}} \rightarrow [0, 1]$ :

$$\varphi_{i_{k,k'}}(p) := \frac{\gamma_{i_{k'}}(p_i) + f_{G_i}(G_i)}{(\gamma_{i_{k'}}^{\lambda_i}(p_i) + G_i(p)\alpha_{i_{k,k'}}(p_i))^{1/\lambda_i}}$$

for agent  $i$ , with  $\lambda_i > 0$  and the following vector field:

$$c_{i_{k,k'}}(p) := \begin{cases} -k_{g_i} \frac{\partial \varphi_{i_{k,k'}}(p)}{\partial p_i}, & \text{if } \pi_k \neq \pi_{k'} \\ 0 & \text{if } \pi_k \equiv \pi_{k'} \end{cases} \quad (5.2)$$

for all  $t \geq t_0$ , with  $k_{g_i} > 0$  and  $f_{G_i}(G_i)$ , defined in [?], is a term that handles inter-agent collisions when an agent has reached its destination.

The navigation field (5.2) guarantees that agent  $i$  will not enter the undesired regions or collide with the other agents and  $\lim_{t \rightarrow \infty} p_i(t) = p_{\pi_{k'}}$ . The latter property of asymptotic convergence along with the assumption that  $r_i < r_{\pi_k}, \forall i \in \mathcal{N}, k \in \mathcal{K}_{\mathcal{R}}$ , implies that there exists a finite time instant  $t_{i,k'}^f \geq t_0$  such that  $p_i(t_{i,k'}^f) \in \bar{\mathcal{B}}(p_{\pi_{k'}}, r_{\pi_{k'}})$  and more specifically that  $\mathcal{A}_i(p_i(t_{i,k'}^f)) \in \pi_{k'}$ , which is the desired behavior. The time instant  $t_{i,k'}^f$  can be chosen from the set  $\{t \geq t_0, \mathcal{A}_i(p_i(t)) \in \pi_{k'}\}$ .

Note, however, that once agent  $i$  leaves region  $\pi_k$ , there is no guarantee that it will not enter that region again (note that  $\mathcal{F}_{i_{k,k'}}$  includes  $\pi_k$ ), which might be undesirable. Therefore, we define the set  $\Pi_{\emptyset,k'} := \{\pi_m \in \Pi, m \in \mathcal{K}_{\mathcal{R}} \setminus \{k'\}\}$  and the corresponding free space  $\mathcal{F}_{i_{\emptyset,k'}} := \{p \in \mathcal{W}^{\mathcal{N}} : \bar{\mathcal{B}}(p_i, r_i) \cap \bar{\mathcal{B}}(p_j, r_j) = \emptyset, \forall j \in \mathcal{N} \setminus \{i\}, \bar{\mathcal{B}}(p_i, r_i) \cap \pi = \emptyset, \forall \pi \in \Pi_{\emptyset,k'}\}$ , and we construct the function  $\varphi_{i_{\emptyset,k'}} : \mathcal{F}_{i_{\emptyset,k'}} \rightarrow [0, 1]$ :

$$\varphi_{i_{\emptyset,k'}}(p) := \frac{\gamma_{i_{k'}}(p_i) + f_{G_i}(G_i)}{(\gamma_{i_{k'}}^{\lambda_i}(p_i) + G_i(p) \alpha_{i_{\emptyset,k'}}(p_i))^{1/\lambda_i}}$$

where  $\alpha_{i_{\emptyset,k'}}(p_i) := \alpha_{i,0}(p_i) \prod_{m \in \Pi_{\emptyset,k'}} \alpha_{i,m}(p_i)$ , with corresponding vector field:

$$c_{i_{\emptyset,k'}}(p) := -k_{g_i} \frac{\partial \varphi_{i_{\emptyset,k'}}(p)}{\partial p_i}, \quad (5.3)$$

which guarantees that region  $\pi_k$  will be also avoided. Therefore, we develop a switching control protocol that employs (5.2) until agent  $i$  is out of region  $\pi_k$  and then switches to (5.3) until  $t = t_{i,k'}^f$ . Consider the following switching function:

$$s_{\text{sat}}(x) := \frac{1}{2}(\text{sat}(2x - 1) + 1)$$

and the time instant  $t'_{i,k}$  that represents the moment that agent  $i$  is out of region  $\pi_k$ , i.e.,  $t'_{i,k} := \min\{t \geq t_0, \bar{\mathcal{B}}(p_i(t), r_i) \cap \bar{\mathcal{B}}(p_{\pi_k}, r_{\pi_k}) = \emptyset\}$ . Then, we propose the following switching control protocol  $u_i : \mathcal{F}_{i_{k,k'}} \cup \mathcal{F}_{i_{\emptyset,k'}} \rightarrow \mathbb{R}^3$ :

$$u_i := u_i(p) = \begin{cases} c_{i_{k,k'}}(p), & t \in [t_0, t'_{i,k}) \\ (1 - s_{\text{sat}}(\iota_{i,k}))c_{i_{k,k'}}(p) + s_{\text{sat}}(\iota_{i,k})c_{i_{\emptyset,k'}}(p), & t \in [t'_{i,k}, t_{i,k'}^f) \end{cases} \quad (5.4)$$

where  $\iota_{i,k} := \frac{t - t'_{i,k}}{\nu_i}$ , and  $\nu_i$  is a design parameter indicating the time period of the switching process, with  $t'_{i,k'} - t'_{i,k} > \nu_i > 0$ . Invoking the continuity of  $p_i(t)$ , we obtain  $\bar{\mathcal{B}}(p_i(t'_{i,k'}), r_i) \subset \bar{\mathcal{B}}(p_{\pi_{k'}}, r_{\pi_{k'}})$  and hence the control protocol (5.4) guarantees, for sufficiently small  $\nu_i$ , that agent  $i$  will navigate from  $\pi_k$  to  $\pi_{k'}$  in finite time without entering any other regions or colliding with other agents and therefore establishes a transition  $\pi_k \rightarrow_i \pi_{k'}$ .

### 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 5.2.2, 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 [? ]:

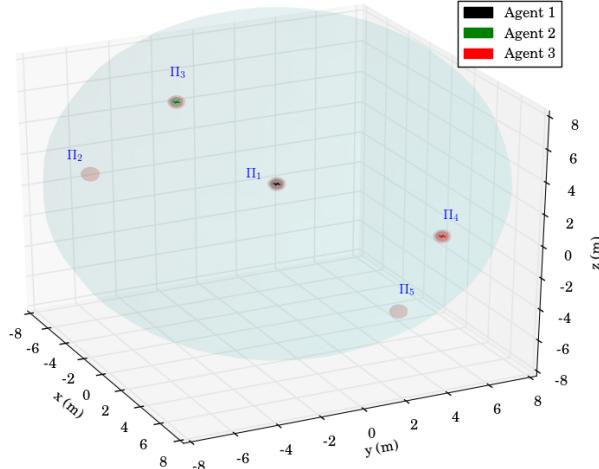
**Definition 5.5.** The motion of each agent  $i \in \mathcal{N}$  in  $\mathcal{W}$  is modeled by the following Transition System (TS):

$$\mathcal{T}_i = (\Pi_i, \Pi_i^{\text{init}}, \rightarrow_i, \Psi_i, \mathcal{L}_i),$$

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. 5.1,  $\Pi_i^{\text{init}} \subseteq \Pi_i$  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 5.2.2 and  $\Psi_i, \mathcal{L}_i$  are the atomic propositions and labeling function respectively, as defined in Section 5.2.1.

After the definition of  $\mathcal{T}_i$ , we translate each given LTL formula  $\Phi_i, i \in \mathcal{N}$  into a Büchi automaton  $\mathcal{C}_i$  and we form the product  $\tilde{\mathcal{T}}_i = \mathcal{T}_i \times \mathcal{C}_i$ . The accepting runs of  $\tilde{\mathcal{T}}_i$  satisfy  $\Phi_i$  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., [? ].

Following the aforementioned methodology, we obtain a high-level plan for each agent as sequences of regions and atomic propositions  $r_{\mathcal{T},i} := \pi_{i_1} \pi_{i_2} \dots$  and  $\psi_i := \psi_{i_1} \psi_{i_2} \dots$  with  $i_m \in \mathcal{K}_{\mathcal{R}}, \psi_{i_m} \in 2^{\Psi_i}, \psi_{i_m} \in \mathcal{L}_i(\pi_{i_m}), \forall m \in \mathbb{N}$  and  $\check{\psi}_i \models \Phi_i, \forall i \in \mathcal{N}$ .



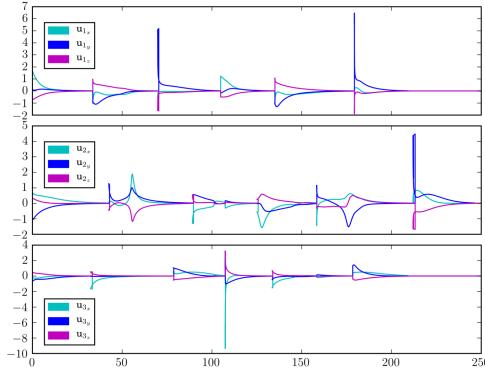
**Figure 5.2:** Initial workspace of the simulation studies. The grey spheres represent the regions of interest while the black, green and red crosses represent agents 1,2 and 3, respectively, along with their bounding spheres.

The execution of  $r_{\mathcal{T},i}, \check{\psi}_i$  produces a trajectory  $p_i(t)$  that corresponds to the behavior  $\mathfrak{b}_i(\check{\psi}_i) = (p_{i_1}(t), \check{\psi}_{i_1})(p_{i_2}(t), \check{\psi}_{i_2}) \dots$ , with  $\mathcal{A}_i(p_{i_m}) \in \pi_{i_m}$  and  $\check{\psi}_{i_m} \in \mathcal{L}_i(\pi_{i_m})$ ,  $\forall m \in \mathbb{N}$ . Therefore, since  $\check{\psi}_i \models \Phi_i$ , the behavior  $\mathfrak{b}_i$  yields satisfaction of the formula  $\Phi_i$ . 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 5.2.2.

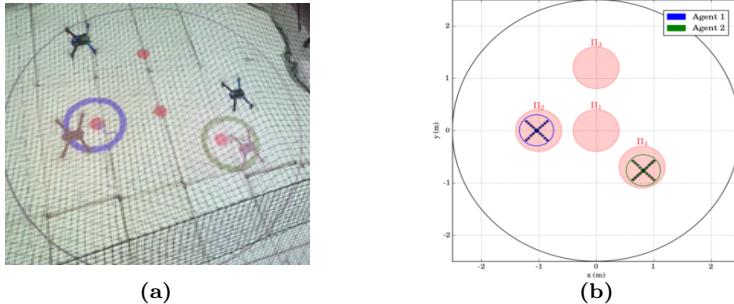
**Remark 5.1.** The proposed control algorithm is decentralized in the sense that each agent derives and executes its own plan without communicating with the rest of the team. The only information that each agent has is the position of its neighboring agents that lie in its limited sensing radius. It is worth mentioning, nevertheless, that the workspace boundary and regions of interest have to satisfy certain assumptions, such as having a sufficient distance from each other or being sufficiently sparse.

### 5.2.3 Simulation and Experimental Results

To demonstrate the efficiency of the proposed algorithm, we consider  $N = 3$  aerial vehicles with  $r_i = 0.3\text{m}$ ,  $\varsigma_i = 0.65\text{m}$ ,  $\forall i = \{1, 2, 3\}$ , operating in a workspace  $\mathcal{W}$  with  $r_0 = 10\text{m}$  and  $p_0 = [0, 0, 0]^\top \text{m}$ . Moreover, we consider  $K = 5$  spherical regions of interest with  $r_{\pi_k} = 0.4\text{m}$ ,  $\forall k = \{1, \dots, 5\}$  and  $p_{\pi_1} = [0, 0, 2]^\top \text{m}$ ,  $p_{\pi_2} = [1, -9, 5]^\top \text{m}$ ,  $p_{\pi_3} = [-8, -1, 4]^\top \text{m}$ ,  $p_{\pi_4} = [2, 7, -2]^\top \text{m}$  and  $p_{\pi_5} = [7.5, 2, -3]^\top \text{m}$ . The initial configurations of the



**Figure 5.3:** The resulting 3-dimensional control signals of the 3 agents for the simulation studies. Top: agent 1, middle: agent 2, bottom: agent 3.



**Figure 5.4:** 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  $\pi_1, \dots, \pi_4$ .

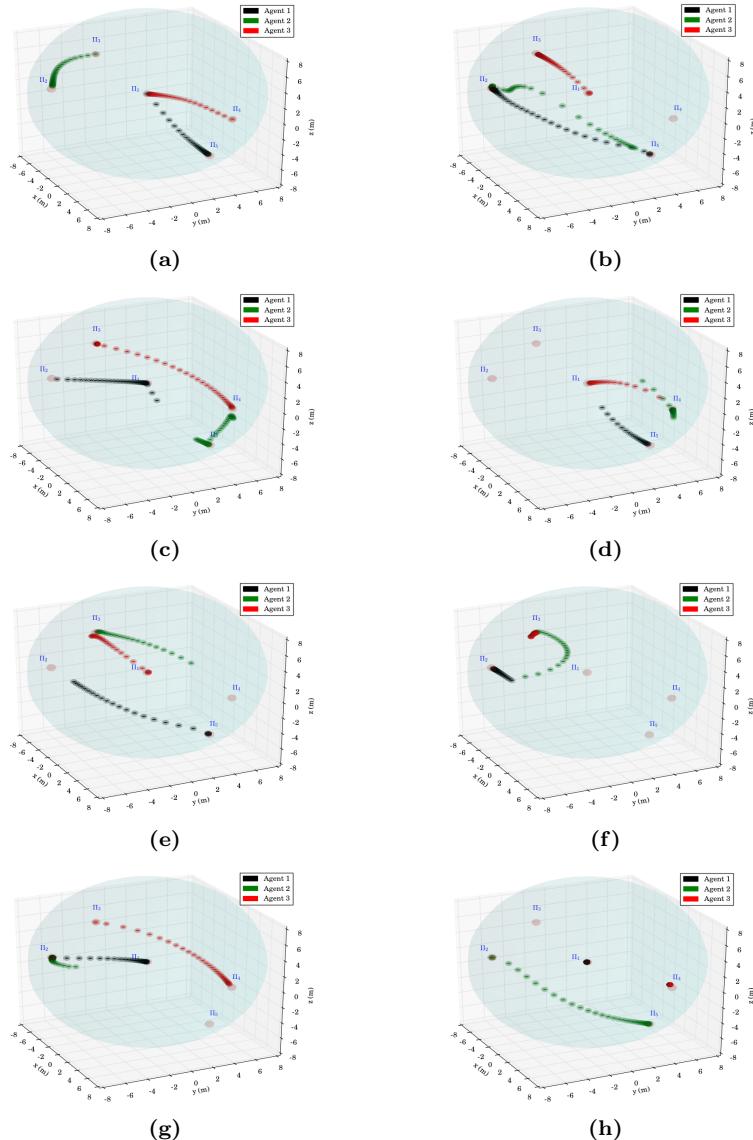
agents are taken as  $p_1(0) = p_{\pi_1}, p_2(0) = p_{\pi_3}, p_3(0) = p_{\pi_4}$  and therefore,  $\mathcal{A}_1(p_1(0)) \in \pi_1, \mathcal{A}_2(p_2(0)) \in \pi_3$  and  $\mathcal{A}_3(p_3(0)) \in \pi_4$ . An illustration of the described workspace is depicted in Fig. 5.2

We consider that agent 2 is assigned with inspection tasks and has the atomic propositions  $\Psi_2 = \{\text{"ins}_a\text{", "ins}_b\text{", "ins}_c\text{", "ins}_d\text{", "obs"}\}$  with  $\mathcal{L}_2(\pi_1) = \{\text{"obs"}\}$ ,  $\mathcal{L}_2(\pi_2) = \{\text{"ins}_a\}\text{, } \mathcal{L}_2(\pi_3) = \{\text{"ins}_b\}\text{, } \mathcal{L}_2(\pi_4) = \{\text{"ins}_c\}$  and  $\mathcal{L}_2(\pi_5) = \{\text{"ins}_d\}$ , where we have considered that region  $\pi_1$  is an undesired (“obstacle”) region for this agent. More specifically, the task for agent 2 is the continuous inspection of the workspace while avoiding

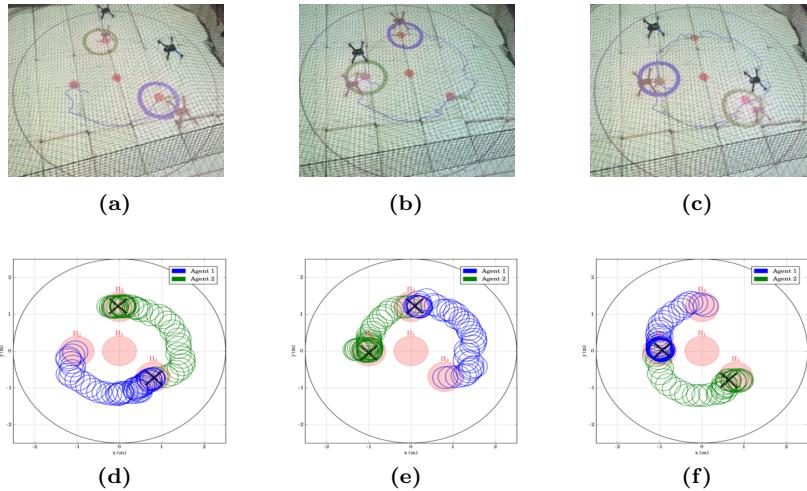
region  $\pi_1$ . The corresponding LTL specification is  $\Phi_2 = (\square\neg\text{``obs''}) \wedge \square(\diamond\text{``ins}_a'' \wedge \diamond\text{``ins}_b'' \wedge \diamond\text{``ins}_c'' \wedge \diamond\text{``ins}_d'')$ . Agents 1 and 3 are interested in moving around resources scattered in the workspace and have propositions  $\Psi_1 = \Psi_3 = \{\text{``res}_a'', \text{``res}_b'', \text{``res}_c'', \text{``res}_d'', \text{``res}_e''\}$  with  $\mathcal{L}_1(\pi_1) = \mathcal{L}_3(\pi_1) = \{\text{res}_a\}$ ,  $\mathcal{L}_1(\pi_2) = \mathcal{L}_3(\pi_2) = \{\text{res}_b\}$ ,  $\mathcal{L}_1(\pi_3) = \mathcal{L}_3(\pi_3) = \{\text{res}_c\}$ ,  $\mathcal{L}_1(\pi_4) = \mathcal{L}_3(\pi_4) = \{\text{res}_d\}$  and  $\mathcal{L}_1(\pi_5) = \mathcal{L}_3(\pi_5) = \{\text{res}_e\}$ . We assume that “ $\text{res}_a$ ” is shared between the two agents whereas “ $\text{res}_b$ ” and “ $\text{res}_e$ ” have to be accessed only by agent 1 and “ $\text{res}_c$ ” and “ $\text{res}_d$ ” only by agent 3. The corresponding specifications are  $\Phi_1 = \square\neg(\text{``res}_c'' \vee \text{``res}_d'') \wedge \square\diamond(\text{``res}_a'' \bigcirc \text{``res}_e'' \bigcirc \text{``res}_b'')$  and  $\Phi_3 = \square\neg(\text{``res}_b'' \vee \text{``res}_e'') \wedge \square\diamond(\text{``res}_a'' \bigcirc \text{``res}_c'' \bigcirc \text{``res}_d'')$ , where we have also included a specific order for the access of the resources. Next, we employ the off-the-shelf tool LTL2BA [?] to create the Büchi automata  $\mathcal{C}_i, i = \{1, 2, 3\}$  and by following the procedure described in Section 5.2.2, 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  $\Phi_1, \Phi_2, \Phi_3$ . Regarding the continuous control protocol, we chose  $k_{g_i} = 15, \lambda_i = 5, \forall i \in \{1, 2, 3\}$  in (5.2), (5.3) and the switching duration in (5.4) was calculated online as  $\nu_i = 0.1t'_{i,k}$ , where we assume that the large distance between the regions  $\pi_k$  (see Fig. 5.2) implies that  $t'_{i,k'} > 1.1t'_{i,k}$  and thus,  $\nu_i < t'_{i,k'} - t'_{i,k}$ . The simulation results are depicted in Fig. 5.3 and 5.5. In particular, Fig. 5.5 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. 5.3 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.

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  $\varsigma_i = 0.65\text{m}$ , upper control input bound  $|u_m| \leq 1\text{m/s}$ ,  $m \in \{x, y, z\}$ , and bounding spheres with radius  $r_i = 0.3\text{m}, \forall i \in \{1, 2\}$ . We considered two 2-dimensional scenarios in a workspace  $\mathcal{W}$  with  $p_0 = [0, 0]^\top$  and  $r_0 = 2.5\text{m}$ .

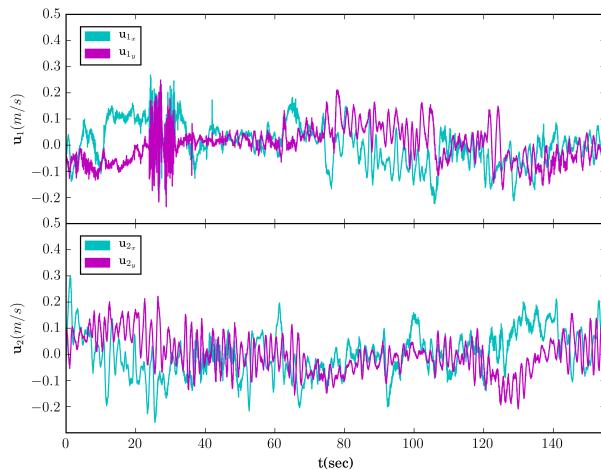
The first scenario included 4 regions of interest  $\Pi = \{\pi_1, \dots, \pi_4\}$  in  $\mathcal{W}$ , with  $r_{\pi_k} = 0.4, \forall k \in \{1, \dots, 4\}$  and  $p_{\pi_1} = [0, 0]^\top \text{m}, p_{\pi_2} = [-1, 0]^\top \text{m}, p_{\pi_3} = [0, 1.25]^\top \text{m}$  and  $p_{\pi_4} = [0.8, -0.7]^\top \text{m}$ . 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. 5.4). We also defined the atomic propositions  $\Psi_1 = \Psi_2 = \{\text{``obs''}, \text{``a''}, \text{``b''}, \text{``c''}\}$  with  $L_1(\pi_1) = L_2(\pi_1) = \{\text{``obs''}\}, L_1(\pi_2) = L_2(\pi_2) = \{\text{``a''}\}, L_1(\pi_3) = L_2(\pi_3) = \{\text{``b''}\}, L_1(\pi_4) = L_2(\pi_4) = \{\text{``c''}\}$ . 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:  $\Phi_1 = \Phi_2 = \square\neg\text{``obs''} \wedge \square\diamond(\text{``a''} \bigcirc \text{``c''} \bigcirc \text{``b''})$ . By following the procedure described in



**Figure 5.5:** 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.



**Figure 5.6:** 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):  $\pi_2 \rightarrow_1 \pi_4, \pi_4 \rightarrow_2 \pi_3$ , (b), (e):  $\pi_4 \rightarrow_1 \pi_3, \pi_3 \rightarrow_2 \pi_2$ , (c), (f):  $\pi_3 \rightarrow_1 \pi_2, \pi_2 \rightarrow_2 \pi_4$ .



**Figure 5.7:** The resulting 2-dimensional control signals of the 2 agents for the first experimental scenario. Top: agent 1, bottom: agent 2.

Section 5.2.2 we obtained the paths  $p_1 = (\pi_2\pi_4\pi_3)^\omega, p_2 = (\pi_4\pi_2\pi_3)^\omega$ . Fig. 5.6 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. 5.7 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  $p_{\pi_1} = [-1, -1.7]^\top \text{m}$ ,  $p_{\pi_2} = [-1.3, 1.3]^\top \text{m}$  and  $p_{\pi_3} = [1.2, 0]^\top \text{m}$ . The initial positions of the agents were taken such that  $\mathcal{A}_1(p_1(0)) \in \pi_1$  and  $\mathcal{A}_2(p_2(0)) \in \pi_2$  (see Fig. 5.8). We also defined the atomic propositions  $\Psi_1 = \Psi_2 = \{\text{"res}_a\text{"}, \text{"res}_b\text{"}, \text{"base"\}}$ , corresponding to a base and several resources in the workspace, with  $L_1(\pi_1) = L_2(\pi_1) = \{\text{"res}_a\text{"}\}$ ,  $L_1(\pi_2) = L_2(\pi_2) = \{\text{"base"\}\}$ ,  $L_1(\pi_3) = L_2(\pi_3) = \{\text{"res}_b\text{"}\}$ . We considered that the agents had to transfer the resources to the “base” in  $\pi_2$ ; both agents were responsible for “res<sub>a</sub>” but only agent 1 should access “res<sub>b</sub>”. The specifications were translated to the formulas  $\Phi_1 = \square(\Diamond(\text{"res}_a\text{"} \bigcirc \text{"base"\}) \wedge \Diamond(\text{"res}_b\text{"} \bigcirc \text{"base"\}))$ ,  $\Phi_2 = \square \neg \text{"res}_b\text{"} \wedge \square \Diamond(\text{"res}_a\text{"} \bigcirc \text{"base"\})$  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 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. 5.10, and the corresponding control inputs are shown in Fig. 5.9. The figures 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 (5.2), (5.3) and the switching duration in (5.4) as  $\nu_i = 0.1t'_{i,k}, \forall i \in \{1, 2\}$ .

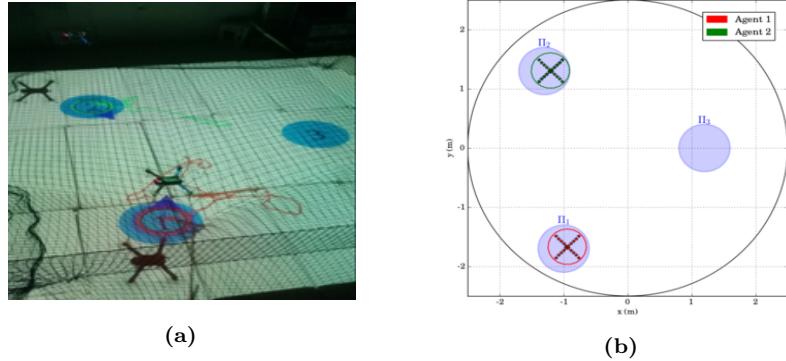
The simulations and experiments were conducted in Python environment using an Intel Core i7 2.4 GHz personal computer with 4 GB of RAM, and are clearly demonstrated in the video found in <https://youtu.be/dO77ZYEFHIE>.

## 5.3 Robust Decentralized Abstractions for Multiple Mobile Manipulators

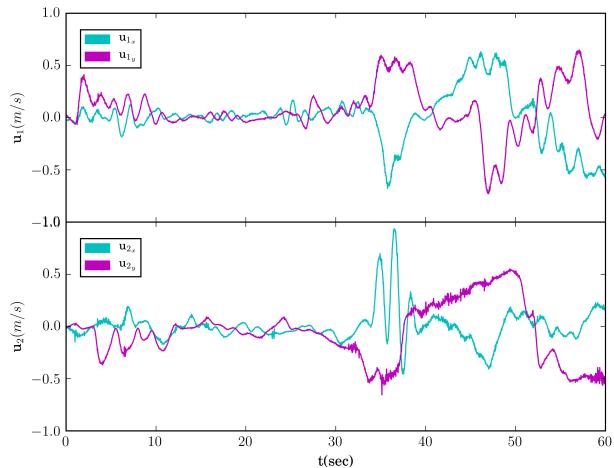
We now turn our attention to a class of more complex systems, that is, mobile manipulators, which, unlike the previous section, have more complex and uncertain dynamics. In fact, we provide a more explicit dynamics formulation than just a sphere/ellipsoid, which was done so far. We describe a decentralized control algorithm that allows the derivation of a discrete abstraction of the multi-agent dynamics.

### 5.3.1 Problem Formulation

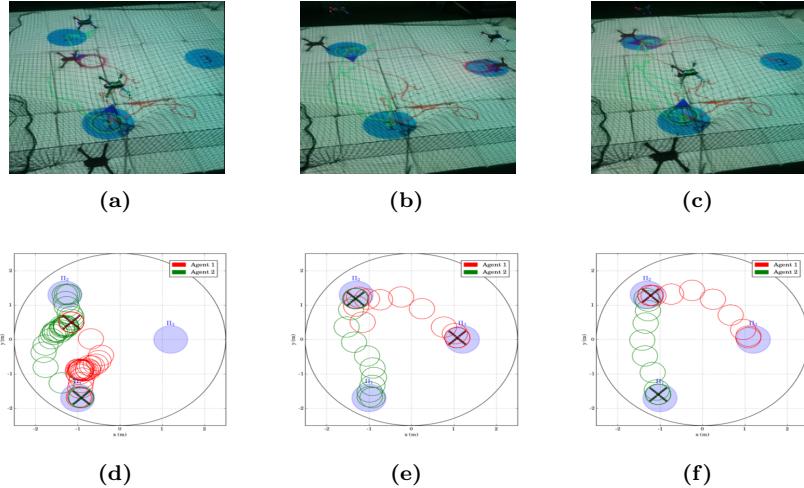
Like before, consider  $N \in \mathbb{N}$  fully actuated agents, with index set  $\mathcal{N}$ , composed by a robotic arm mounted on an omnidirectional mobile base,



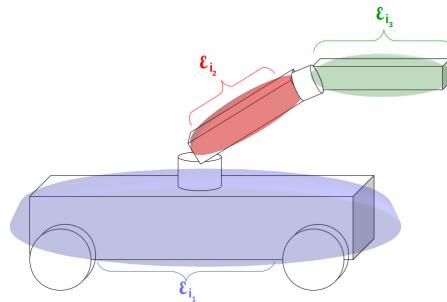
**Figure 5.8:** Initial workspace for the second experimental scenario. (a): The UAVs with the projection of their bounding spheres, (with red and green), and the regions of interest (blue disks). (b): Top view of the described workspace. The UAVs are represented by the red and green circled X's and the regions of interest by the blue disks  $\pi_1, \dots, \pi_3$ .



**Figure 5.9:** The resulting 2-dimensional control signals of the 2 agents for the second experimental scenario. Top: agent 1, bottom: agent 2.



**Figure 5.10:** 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):  $\pi_1 \rightarrow_1 \pi_2, \pi_2 \rightarrow_2 \pi_1$ , (b), (e):  $\pi_2 \rightarrow_1 \pi_3, \pi_1 \rightarrow_2 \pi_2$ , (c), (f):  $\pi_3 \rightarrow_1 \pi_2, \pi_2 \rightarrow_2 \pi_1$ .



**Figure 5.11:** An agent that consists of  $\ell_i = 3$  rigid links.

operating in a static workspace  $\mathcal{W}$  that is bounded by a large sphere in 3D space, i.e.  $\mathcal{W} = \mathcal{B}(p_0, r_0) = \{p \in \mathbb{R}^3 \text{ s.t. } \|p - p_0\| < r_0\}$ , where  $p_0 \in \mathbb{R}^3$  is the center of  $\mathcal{W}$ , and  $r_0 \in \mathbb{R}_{>0}$  is its radius. Without loss of generality, we consider that  $p_0 = 0$ , corresponding to an inertial frame of reference. As in the previous section, we consider that within  $\mathcal{W}$  there exist  $K$  disjoint spheres around points of interest, which are described by  $\pi_k = \bar{\mathcal{B}}(p_{\pi_k}, r_{\pi_k})$ ,  $k \in \mathcal{K}_R$ , where  $p_k \in \mathbb{R}^3$  and  $r_{\pi_k} \in \mathbb{R}_{>0}$  are the center and radius of the  $k$ th 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

$$T_{\pi_k} = \begin{bmatrix} I_3 & p_{\pi_k} \\ 0_{1 \times 3} & -r_{\pi_k}^2 \end{bmatrix}, \forall k \in \mathcal{K}_R.$$

The dynamic model of each agent is given by the second-order Lagrangian dynamics (see (2.1) of Chapter 2):

$$B_i(q_i)\ddot{q}_i + C_{q_i}(q_i, \dot{q}_i)\dot{q}_i + g_{q_i}(q_i) + f_i(q_i, \dot{q}_i) = \tau_i, \quad (5.5)$$

$\forall i \in \mathcal{N}$ , 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), with  $q := [q_1^\top, \dots, q_N^\top]^\top$ , and the rest of the terms as in (2.1) with a slight change of notation;  $f_i(\cdot)$  here represents friction-like terms. Without loss of generality, we assume that  $n_i = n \in \mathbb{N}, \forall i \in \mathcal{N}$ . In addition, we denote by  $p_{B_i} := p_{B_i}(q_i) : \mathbb{R}^n \rightarrow \mathbb{R}^3$  the inertial position of the mobile base of agent  $i$ . Moreover, the matrix  $\dot{B}_i - 2C_{q_i}$  is skew-symmetric [?], and we further make the following assumption, similar to Assumption 4.1:

**Assumption 5.1.** There exist positive constants  $\alpha_i$  such that  $\|f_i(q_i, \dot{q}_i)\| \leq \alpha_i \|q_i\| \|\dot{q}_i\|$ ,  $\forall (q_i, \dot{q}_i) \in \mathbb{R}^n \times \mathbb{R}^n$ ,  $i \in \mathcal{N}$ .

For the inter-agent collisions, we will use the ideas of Section 4.4 and ellipsoid collision. We consider hence that each agent is composed by  $\ell_i$  rigid links (see Fig. 5.11) with  $\mathcal{Q}_i = \{1, \dots, \ell_i\}$  the corresponding index set. Each link of agent  $i$  is approximated by the ellipsoid set [?]  $\mathcal{E}_{i_m}(q_i) = \{z \in \mathbb{R}^4 \text{ s.t. } z^\top E_{i_m}(q_i)z \leq 0\}$ ;  $z = [p^\top, 1]^\top$  is the homogeneous coordinates of  $p \in \mathbb{R}^3$ , and  $E_{i_m} : \mathbb{R}^n \rightarrow \mathbb{R}^{4 \times 4}$  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  $\hat{E}_{i_m} = \text{diag}\{l_{i_m,x}^{-2}, l_{i_m,y}^{-2}, l_{i_m,z}^{-2}, -1\}$  corresponds to the positive lengths  $l_{i_m,x}, l_{i_m,y}, l_{i_m,z}$  of the principal axes of the ellipsoid, and  $T_{i_m} : \mathbb{R}^n \rightarrow \mathbb{R}^{4 \times 4}$  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  $\mathcal{E}_{i_m}$ :

$$T_{i_m}(q_i) = \begin{bmatrix} R_{i_m}(q_i) & p_{i_m}(q_i) \\ 0_{1 \times 3}^\top & 1 \end{bmatrix},$$

with  $R_{im} : \mathbb{R}^n \rightarrow \mathbb{SO}^3$  being the rotation matrix of the link,  $\forall m \in \mathcal{Q}_i, i \in \mathcal{N}$ . For an ellipsoid  $\mathcal{E}_{im}$ ,  $i \in \mathcal{N}, m \in \mathcal{Q}_i$ , we denote by  $\mathcal{E}_{im}^{xy}, \mathcal{E}_{im}^{xz}, \mathcal{E}_{im}^{yz}$  its projections on the planes  $x-y$ ,  $x-z$  and  $y-z$ , respectively, with corresponding matrix terms  $E_{im}^{xy}, E_{im}^{xz}, E_{im}^{yz}$ .

By following the procedure of Section 4.4, we conclude that a sufficient condition for  $\mathcal{E}_{im}$  and  $\mathcal{E}_{jl}$  not to collide is  $\sigma(\Delta_{im,jl}^{xy}) + \sigma(\Delta_{im,jl}^{xz}) + \sigma(\Delta_{im,jl}^{yz}) > 0$ , with  $\sigma()$  as defined in 4.41 and  $\Delta_{im,jl}^s$  is the discriminant of the equation  $\det(\lambda E_{im}^s(q_i) - E_{jl}^s(q_j)) = 0$ ,  $\forall i, j \in \mathcal{N}, i \neq j, m \in \mathcal{Q}_i, l \in \mathcal{Q}_j$ , where the subscript  $s \in \{xy, yz, xz\}$  stands for the planar ellipsoid matrices.

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.  $\bar{d}_{B_i} := \sup_{p_i \in \bigcup_{m \in \mathcal{Q}_i} \mathcal{E}_{im}(q_i)} \{\|p_{B_i}(q_i) - p_i\|\}$ . We also denote  $\bar{d}_B = [\bar{d}_{B_1}, \dots, \bar{d}_{B_N}]^\top$ . Moreover, we consider that each agent has a sensor located at the center of its mobile base  $p_{B_i}$  with a sensing radius  $\varsigma_i \geq 2 \max_{i \in \mathcal{N}} \{\bar{d}_{B_i}\} + \varepsilon_d$ , where  $\varepsilon_d$  is an arbitrarily small positive constant. Hence, each agent has the sensing sphere  $\mathcal{D}_i(q_i) := \{p \in \mathbb{R}^3 \text{ s.t. } \|p - p_{B_i}(q_i)\| \leq \varsigma_i\}$  and its neighborhood set at each time instant is defined as  $\mathcal{N}_i(q_i) := \{j \in \mathcal{N} \setminus \{i\} \text{ s.t. } \|p_{B_i}(q_i) - p_{B_j}(q_j)\| \leq \varsigma_i\}$ .

As in Section 5.2.1, 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  $\tilde{\mathcal{N}}_i \subseteq \mathcal{N}_i(q_i(0)), i \in \mathcal{N}$ , of the initial neighbors of the agents must stay connected through their motion in the workspace, similarly to Section 4.3. 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  $\tau_i, i \in \mathcal{N}$ , for the appropriate transitions of the agents among the predefined regions of interest in the workspace.

Next, we provide the equivalent definitions to Def. 5.1 and 5.2

**Definition 5.6.** An agent  $i \in \mathcal{N}$  is in region  $k \in \mathcal{K}_R$  at a configuration  $q_i \in \mathbb{R}^n$ , denoted as  $\mathcal{A}_i(q_i) \in \pi_k$ , if and only if  $\mathcal{E}_{im}(q_i) \subset \pi_k, \forall m \in \mathcal{Q}_i$ .

**Definition 5.7.** Agents  $i, j \in \mathcal{N}$ , 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  $\mathcal{E}_{im}(q_i) \cap \mathcal{E}_{jl}(q_j) = \emptyset, \forall m \in \mathcal{Q}_i, l \in \mathcal{Q}_j$ .

Given the aforementioned discussion, we make the following assumptions regarding the agents and the validity of the workspace

**Assumption 5.2.** The regions of interest are

- (i) large enough such that all the agents can fit, i.e., given a specific  $k \in \mathcal{K}_R$ , there exist  $q_i, i \in \mathcal{N}$ , such that  $\mathcal{A}_i(q_i) \in \pi_k, \forall i \in \mathcal{N}$ , with  $\mathcal{A}_i(q_i) \not\equiv \mathcal{A}_j(q_j), \forall i, j \in \mathcal{N}$ , with  $i \neq j$ .
- (ii) sufficiently far from each other and the obstacle workspace, i.e.,

$$\begin{aligned} \|p_{\pi_k} - p_{\pi_{k'}}\| &\geq \max_{i \in \mathcal{N}}\{2\bar{d}_{B_i}\} + r_{\pi_k} + r_{\pi_{k'}} + \epsilon_\pi, \\ r_0 - \|p_k\| &\geq \max_{i \in \mathcal{N}}\{2\bar{d}_{B_i}\}, \end{aligned}$$

$\forall k, k' \in \mathcal{K}_R, k \neq k'$ , where  $\epsilon_\pi$  is an arbitrarily small positive constant.

Next, in order to proceed, we need the following definition.

**Definition 5.8.** Assume that  $\mathcal{A}_i(q_i(t_0)) \in \pi_k, i \in \mathcal{N}$ , for some  $t_0 \in \mathbb{R}_{\geq 0}, k \in \mathcal{K}_R$ , with  $\mathcal{A}_i(q_i(t_0)) \not\equiv \mathcal{A}_j(q_j(t_0)), \forall j \in \mathcal{N} \setminus \{i\}$ . There exists a transition for agent  $i$  between  $\pi_k$  and  $\pi_{k'}, k' \in \mathcal{K}_R$ , 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

- $\mathcal{A}_i(q_i(t_f)) \in \pi_{k'}$
- $\mathcal{A}_i(q_i(t)) \not\equiv \mathcal{A}_j(q_j(t)), \forall j \in \mathcal{N} \setminus \{i\}$ ,
- $\mathcal{E}_{i_m}(q_i(t)) \cap \mathcal{E}_{i_\ell}(q_i(t)), \forall m, \ell \in \mathcal{Q}_i, m \neq \ell$ ,
- $\mathcal{E}_{i_m}(q_i(t)) \cap \pi_z = \emptyset, \forall m \in \mathcal{Q}_i, z \in \mathcal{K}_R \setminus \{k, k'\}$ ,
- $\mathcal{E}_{i_m}(q_i(t)) \subset \mathcal{W}, \forall m \in \mathcal{Q}_i$ ,

$\forall t \in [t_0, t_f]$ .

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 [? ]

$$\mathcal{T}_i = (\Pi, \Pi_{i,0}, \xrightarrow{i}, \Psi_i, \mathcal{L}_i, \mathcal{H}_i), \quad (5.6)$$

where  $\Pi = \{\pi_1, \dots, \pi_K\}$  is the set of regions of interest that the agents can be at, according to Def. 5.6.  $\Pi_{i,0} \subseteq \Pi$  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. 5.8,  $\Psi_i$  is a set of given atomic propositions, represented as boolean variables, that hold in the regions of interest,  $\mathcal{L}_i : \Pi \rightarrow 2^{\Psi_i}$  is a labeling function, and  $\mathcal{H}_i : \Pi \rightarrow \Pi^{|\tilde{\mathcal{N}}_i|}$  is a function that maps the region that agent  $i$  occupies to

the regions the initial neighbors  $\tilde{\mathcal{N}}_i$  of agent  $i$  are at. Therefore, the treated problem is the design of bounded controllers  $\tau_i$  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  $\tilde{\mathcal{N}}_i, \forall i \in \mathcal{N}$ . Another desired property important in applications involving robotic manipulators, is the nonsingularity of the Jacobian matrix  $J_i : \mathbb{R}^n \rightarrow \mathbb{R}^{6 \times n}$ , that transforms the generalized coordinate rates of agent  $i \in \mathcal{N}$  to generalized velocities [?] (also defined in Chapter 2). That is, the agents should always remain in the sets  $S_i = \{q_i \in \mathbb{R}^n \text{ s.t. } \det(J_i(q_i)J_i(q_i)^\top) > 0\}, \forall i \in \mathcal{N}$ .

Formally, we define the problem treated in this section as follows:

**Problem 5.2.** Consider  $N$  mobile manipulators with dynamics (5.5) and  $K$  regions of interest  $\pi_k, k \in \mathcal{K}_R$ , with  $\dot{q}_i(t_0) < \infty, \mathcal{A}_i(q_i(t_0)) \in \pi_{k_i}, k_i \in \mathcal{K}_R, q_i(t_0) \in S_i, \forall i \in \mathcal{N}$  and  $\mathcal{A}_i(q_i(t_0)) \neq \mathcal{A}_j(q_j(t_0)), \mathcal{E}_{i_m}(q_i(t_0)) \cap \mathcal{E}_{i_\ell}(q_i(t_0)) = \emptyset, \forall i, j \in \mathcal{N}, i \neq j, m, \ell \in Q_i, m \neq \ell$ .

Given nonempty subsets of the initial edge sets  $\tilde{\mathcal{N}}_i \subseteq \mathcal{N}_i(q_i(0)) \subseteq \mathcal{N}, \forall i \in \mathcal{N}$ , as well as the indices  $k'_i \in \mathcal{K}_R, i \in \mathcal{N}$ , such that  $\|p_{k'_i} - p_{k'_j}\| + r_{\pi_{k'_i}} + r_{\pi_{k'_j}} \leq \varsigma_i, \forall j \in \tilde{\mathcal{N}}_i, i \in \mathcal{N}$ , design decentralized controllers  $\tau_i$  such that, for all  $i \in \mathcal{N}$ :

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.  $j_i^* \in \mathcal{N}_i(q_i(t)), \forall j_i^* \in \tilde{\mathcal{N}}_i, t \in [t_0, t_{f_i}],$
4.  $q_i(t) \in S_i, \forall t \in [t_0, t_{f_i}].$

The aforementioned specifications concern 1) the agent transitions according to Def. 5.8, 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  $\tilde{\mathcal{N}}_i$  are nonempty implies that the sensing radius of each agent  $i$  covers the regions  $\pi_{k_j}$  of the agents in the neighboring set  $\tilde{\mathcal{N}}_i$ . Similarly, the condition  $\|p_{k'_i} - p_{k'_j}\| + r_{\pi_{k'_i}} + r_{\pi_{k'_j}} \leq \varsigma_i, \forall j \in \tilde{\mathcal{N}}_i$ , 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  $\varsigma_i$  should be large enough to allow transitions of the multi-agent system to the entire workspace.

### 5.3.2 Problem Solution

To solve Problem 5.2, we use the concept of potential fields, as done in Section 5.2. Nevertheless, we do not provide an explicit closed-form expression of the potential function, but provide appropriate conditions.

Let  $\varphi_i$  be a *decentralized potential function*, with the following properties:

- (i) The function  $\varphi_i(q)$  is not defined, i.e.,  $\varphi_i(q) = \infty$ ,  $\forall i \in \mathcal{N}$ , when a collision or a connectivity break occurs,
- (ii) The critical points of  $\varphi_i$  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*  $\gamma_i : \mathbb{R}^n \rightarrow \mathbb{R}_{\geq 0}$ , which should vanish when  $\mathcal{A}_i(q_i) \in \pi_{k'_i}$ , and an *obstacle function*,  $\beta_i : \mathbb{R}^{Nn} \rightarrow \mathbb{R}_{\geq 0}$  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  $\varphi_i$  is not taken into account.

The control objective of agent  $i$ , i.e., reaching the region of interest  $\pi_{k'_i}$ , is encoded in the function  $\gamma_i := \gamma_i(q_i) : \mathbb{R}^n \rightarrow \mathbb{R}_{\geq 0}$ , defined as

$$\gamma_i(q_i) := \|q_i - q_{k'_i}\|^2,$$

where  $q_{k'_i}$  is a configuration such that  $r_{\pi_{k'_i}} - \|p_{B_i}(q_{k'_i}) - p_{k'_i}\| \leq \bar{d}_{B_i} - \epsilon_q$ , for an arbitrarily small positive constant  $\epsilon_q$ , which implies  $\mathcal{A}_i(q_{k'_i}) \in \pi_{k'_i}$ ,  $\forall i \in \mathcal{N}$ . In case that multiple agents have the same target, i.e., there exists at least one  $j \in \mathcal{N} \setminus \{i\}$  such that  $\pi_{k'_j} = \pi_{k'_i}$ , then we assume that  $\mathcal{A}_i(q_{k'_i}) \not\equiv \mathcal{A}_j(q_{k'_j})$ .

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  $J_i(q_i)$ ,  $\forall i \in \mathcal{N}$  are encoded by a function  $\beta_i$ , defined next.

As mentioned before, a sufficient condition for ellipsoids  $\mathcal{E}_{i_m}$  and  $\mathcal{E}_{j_l}$  not to collide, is  $\Delta_{i_m,j_l}(q_i, q_j) = \sigma(\Delta_{i_m,j_l}^{xy}(q_i, q_j) + \sigma(\Delta_{i_m,j_l}^{xz}(q_i, q_j)) + \sigma(\Delta_{i_m,j_l}^{yz}(q_i, q_j)) > 0$ ,  $\forall m \in \mathcal{Q}_i, l \in \mathcal{Q}_j, i, j \in \mathcal{N}$ , and  $\sigma$  as defined in (4.41).

Additionally, we define the greatest lower bound of the  $\Delta_{i_m,j_l}$  when the point  $p_{j_l}$  is on the boundary of the sensing radius  $\partial D_i(q_i)$  of agent  $i$ , as  $\tilde{\Delta}_{i_m,j_l} = \inf_{\substack{(q_i, q_j) \in \mathbb{R}^{2n} \\ \|p_{B_i}(q_i) - p_{j_l}(q_j)\| = \varsigma_i}} \{\Delta_{i_m,j_l}(q_i, q_j)\}$ ,  $\forall m \in \mathcal{Q}_i, l \in \mathcal{Q}_j, i, j \in \mathcal{N}$ .

Since  $\varsigma_i > 2 \max_{i \in \mathcal{N}} \{\bar{d}_{B_i}\} + \epsilon_d$ , it follows that there exists a positive constant  $\epsilon_\Delta$  such that  $\tilde{\Delta}_{i_m,j_l} \geq \epsilon_\Delta > 0$ ,  $\forall m \in \mathcal{Q}_i, l \in \mathcal{Q}_j, i, j \in \mathcal{N}, i \neq j$ .

We further define the function  $\zeta_{ij} := \zeta_{ij}(q_i, q_j) : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ , with  $\zeta_{ij}(q_i, q_j) := \varsigma_i^2 - \|p_{B_i}(q_i) - p_{B_j}(q_j)\|^2$ , and the distance functions  $\beta_{\mathbf{c}, i_m, j_l} := \beta_{\mathbf{c}, i_m, j_l}(\Delta_{i_m, j_l}) : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ ,  $\beta_{\mathbf{n}, ij} := \beta_{\mathbf{n}, ij}(\zeta_{ij}) : \mathbb{R} \rightarrow \mathbb{R}_{\geq 0}$ ,  $\beta_{iw} := \beta_{iw}(\|p_{B_i}\|^2) : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}$  as

$$\begin{aligned}\beta_{\mathbf{c}, i_m, j_l}(\Delta_{i_m, j_l}) &:= \begin{cases} \vartheta_{\mathbf{c}, i}(\Delta_{i_m, j_l}), & 0 \leq \Delta_{i_m, j_l} < \bar{\Delta}_{i_m, j_l}, \\ \bar{\Delta}_{i_m, j_l}, & \bar{\Delta}_{i_m, j_l} \leq \Delta_{i_m, j_l}, \end{cases} \\ \beta_{\mathbf{n}, ij}(\zeta_{ij}) &:= \begin{cases} 0, & \eta_{ij,c} < 0, \\ \vartheta_{\mathbf{n}, i}(\zeta_{ij}), & 0 \leq \eta_{ij,c} < \varsigma_i^2, \\ d_{\text{con}_i}^2, & \varsigma_i^2 \leq \zeta_{ij}, \end{cases} \\ \beta_{iw}(\|p_{B_i}\|^2) &= (r_0 - \bar{d}_{B_i})^2 - \|p_{B_i}\|^2,\end{aligned}$$

where  $\bar{\Delta}_{i_m, j_l}$  is a constant satisfying  $0 < \bar{\Delta}_{i_m, j_l} \leq \bar{\Delta}_{i_m, j_l}, \forall m \in \mathcal{Q}_i, l \in \mathcal{Q}_j, i, j \in \mathcal{N}, i \neq j$ , and  $\vartheta_{\mathbf{c}, i}, \vartheta_{\mathbf{n}, i}$  are *strictly increasing* polynomials appropriately selected to guarantee that the functions  $\beta_{\mathbf{c}, i_m, j_l}$ , and  $\beta_{\mathbf{n}, ij}$ , respectively, are twice continuously differentiable everywhere, with  $\vartheta_{\mathbf{c}, i}(0) = \vartheta_{\mathbf{n}, i}(0) = 0, \forall i \in \mathcal{N}$ . Note that the functions defined above use only local information in the sensing range  $\varsigma_i$  of agent  $i$ . Similarly,  $\beta_{iw}$  encodes the collision of agent  $i$  with the workspace boundary.

Finally, we choose the function  $\beta_i := \beta_i(q) : \mathbb{R}^{Nn} \rightarrow \mathbb{R}_{\geq 0}$  as

$$\begin{aligned}\beta_i(q) &= (\det(J_i(q_i)J_i(q_i)^\top))^2 \beta_{iw}(\|p_{B_i}\|^2) \prod_{j \in \tilde{\mathcal{N}}_i} \beta_{\mathbf{n}, ij}(\zeta_{ij}) \\ &\quad \prod_{(m, j, l) \in \tilde{T}} \beta_{\mathbf{c}, i_m, j_l}(\Delta_{i_m, j_l}) \prod_{(m, k) \in \tilde{L}} \Delta_{i_m, \pi_k}(q_i),\end{aligned}$$

$\forall i \in \mathcal{N}$ , where  $\tilde{T} := \mathcal{Q}_i \times \mathcal{N} \times \mathcal{Q}_j, \tilde{L} := \mathcal{Q}_i \times (\mathcal{K}_{\mathcal{R}} \setminus \{k_i, k'_i\})$ . Note that we have included the term  $(\det(J_i J_i^\top))^2$  to also account for singularities of  $J_i, \forall i \in \mathcal{N}$  and the term  $\prod_{(m, j, l) \in \tilde{T}} \beta_{\mathbf{c}, i_m, j_l}(\Delta_{i_m, j_l})$  takes into account also the collisions between the ellipsoidal rigid bodies of agent  $i$ .

With the introduced notation, the properties of the functions  $\varphi_i$  are:

- (i)  $\beta_i(q) \rightarrow 0 \Leftrightarrow (\varphi_i(q) \rightarrow \infty), \forall i \in \mathcal{N}$ ,
- (ii)  $\nabla_{q_i} \varphi_i(q)|_{q_i=q_i^*} = 0, \forall q_i^* \in \mathbb{R}^n$  s.t.  $\gamma_i(q_i^*) = 0$  and the regions of attraction of the points  $\{q \in \mathbb{R}^{Nn} : \nabla_{q_i} \varphi_i(q)|_{q_i=\tilde{q}_i} = 0, \gamma_i(\tilde{q}_i) \neq 0\}, i \in \mathcal{N}$ , 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 of this section:

**Theorem 5.1.** Under the Assumptions 5.1-5.2, the decentralized control laws  $\tau_i := \tau_i(q, \dot{q}_i, \hat{\alpha}_i) : \mathbb{D}_i \times \mathbb{R}^{n+1} \rightarrow \mathbb{R}^n$ , with

$$\tau_i(q, \dot{q}_i, \hat{\alpha}_i) = g_i(q_i) - \nabla_{q_i} \varphi_i(q) - \sum_{j \in \mathcal{N}_i(q_i)} \nabla_{q_i} \varphi_j(q) - \hat{\alpha}_i \|q_i\| \dot{q}_i - k_{v_i} \dot{q}_i, \quad (5.7)$$

with  $k_{v_i}$  positive gain constants,  $\forall i \in \mathcal{N}$ , along with the adaptation laws

$$\dot{\hat{\alpha}}_i = k_{\alpha_i} \|\dot{q}_i\|^2 \|q_i\|, \quad (5.8)$$

with  $\hat{\alpha}_i(t_0) < \infty$ ,  $k_{\alpha_i} \in \mathbb{R}_{\geq 0}$  positive gain constants,  $\forall i \in \mathcal{N}$ , guarantee the transitions  $(\pi_{k_i}, t_0) \xrightarrow{i} (\pi_{k'_i}, t_{f_i})$  for finite  $t_{f_i}, i \in \mathcal{N}$  for almost all initial conditions, while ensuring  $\beta_i > 0, \forall i \in \mathcal{N}$ , as well as the boundedness of all closed loop signals, providing, therefore, a solution to Problem 5.2.

*Proof.* The closed loop system of (5.5) is written as:

$$M_i(q_i)\ddot{q}_i + N_i(q_i, \dot{q}_i)\dot{q}_i + f_i(q_i, \dot{q}_i) = -\nabla_{q_i} \varphi_i(q_i) - k_{v_i} \dot{q}_i - \hat{\alpha}_i \|q_i\| \dot{q}_i - \sum_{j \in \mathcal{N}_i(q_i)} \nabla_{q_i} \varphi_j(q), \quad (5.9)$$

$\forall i \in \mathcal{N}$ . Due to Assumption 5.2, 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  $V := V(\varphi, \dot{q}, \tilde{\alpha}, q) : \mathbb{R}^N \times \mathbb{R}^{Nn} \times \mathbb{R}^N \times \mathbb{D}_1 \times \cdots \times \mathbb{D}_N \rightarrow \mathbb{R}_{\geq 0}$ , with

$$V := \sum_{i \in \mathcal{N}} \varphi_i(q) + \frac{1}{2} \dot{q}_i^\top M_i(q_i) \dot{q}_i + \frac{1}{2k_{\alpha_i}} \tilde{\alpha}_i^2$$

where  $\varphi$  and  $\tilde{\alpha}$  are the stack vectors containing all  $\varphi_i$  and  $\tilde{\alpha}_i$ , respectively,  $i \in \mathcal{N}$ , and  $\tilde{\alpha}_i := \hat{\alpha}_i - \alpha_i, \forall i \in \mathcal{N}$ . Note that, since there are no collision or singularities at  $t_0$ , the functions  $\beta_i(q), i \in \mathcal{N}$ , are strictly positive at  $t_0$  which implies the boundedness of  $V$  at  $t_0$ . Therefore, since  $\dot{q}_i(t_0) < \infty$  and  $\hat{\alpha}_i(t_0) < \infty, \forall i \in \mathcal{N}$ , there exists a positive and finite constant  $M < \infty$  such that  $V_0 := V(t_0) \leq M$ .

By differentiating  $V$ , substituting the dynamics (5.5), employing the skew symmetry of  $M_i - 2N_i$  as well as  $\sum_{i \in \mathcal{N}} (\nabla_{q_i} \varphi_i(q)^\top \dot{q}_i + \sum_{j \in \mathcal{N}_i(q_i)} \nabla_{q_j} \varphi_i(q)^\top \dot{q}_j) = \sum_{i \in \mathcal{N}} (\nabla_{q_i} \varphi_i(q)^\top + \sum_{j \in \mathcal{N}_i(q_i)} \nabla_{q_j} \varphi_i(q)^\top) \dot{q}_i$ , we obtain

$$\begin{aligned} \dot{V} = \sum_{i \in \mathcal{N}} & \left\{ \dot{q}_i^\top \left( \nabla_{q_i} \varphi_i(q) + \sum_{j \in \mathcal{N}_i(q_i)} \nabla_{q_j} \varphi_i(q) + \tau_i - g_i(q_i) \right) - \dot{q}_i^\top f_i(q_i, \dot{q}_i) \right. \\ & \left. + \frac{1}{k_{\alpha_i}} \tilde{\alpha}_i \dot{\hat{\alpha}}_i \right\}, \end{aligned}$$

which, by substituting the control and adaptation laws (5.7) and (5.8), becomes:

$$\begin{aligned}\dot{V} &= \sum_{i \in \mathcal{N}} \left\{ -k_{v_i} \|\dot{q}_i\|^2 - \hat{\alpha}_i \|\dot{q}_i\|^2 \|q_i\| - \dot{q}_i^\top f_i(q_i, \dot{q}_i) + \tilde{\alpha}_i \|\dot{q}_i\|^2 \|q_i\| \right\} \\ &\leq \sum_{i \in \mathcal{N}} \left\{ -k_{v_i} \|\dot{q}_i\|^2 - (\hat{\alpha}_i - \alpha_i - \tilde{\alpha}_i) \|\dot{q}_i\|^2 \|q_i\| \right\}\end{aligned}$$

where we have used the property  $\|f_i(q_i, \dot{q}_i)\| \leq \alpha_i \|q_i\| \|\dot{q}_i\|$ . Since  $\tilde{\alpha}_i = \hat{\alpha}_i - \alpha_i$ , we obtain  $\dot{V} \leq -\sum_{i \in \mathcal{N}} k_{v_i} \|\dot{q}_i\|^2$ , 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  $\tilde{\alpha}_i, \varphi_i, \dot{q}_i$  and hence of  $\dot{\alpha}_i, \forall i \in \mathcal{N}, t \geq t_0$ . Therefore, we conclude that  $\beta_i(q(t)) > 0, \forall t \geq t_0, i \in \mathcal{N}$ .

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

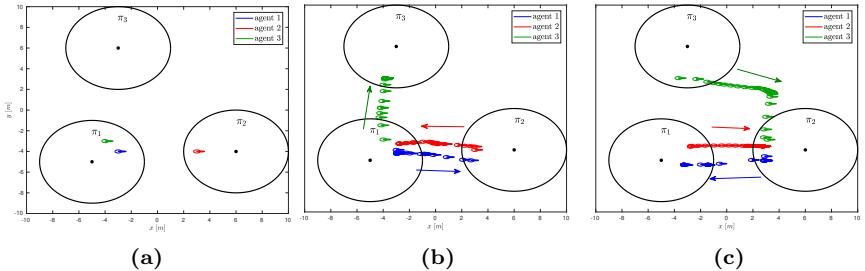
$$S_C := \{(q, \dot{q}) \in \mathbb{D}_1 \times \cdots \times \mathbb{D}_N \times \mathbb{R}^{Nn} \text{ s.t. } \dot{q} = 0_{Nn \times 1}\}.$$

One can easily conclude that  $\ddot{q}_i = 0, \forall i \in \mathcal{N}$ , in  $S_C$  and thus we conclude for the closed loop system (5.9) that  $\nabla_{q_i} \varphi_i(q) = 0, \forall i \in \mathcal{N}$ , since  $\|f_i(q_i, 0)\| \leq 0, \forall q_i \in \mathbb{R}^n$ , in view of Assumption 5.1. Therefore, by invoking the properties of  $\varphi_i(q)$ , each agent  $i \in \mathcal{N}$  will converge to a critical point of  $\varphi_i$ , i.e., all the configurations where  $\nabla_{q_i} \varphi_i(q) = 0, \forall i \in \mathcal{N}$ . However, due to properties of  $\varphi_i(q)$ , the initial conditions that lead to configurations  $\tilde{q}_i$  such that  $\nabla_{q_i} \varphi_i(q)|_{q_i=\tilde{q}_i} = 0$  and  $\gamma_i(\tilde{q}_i) \neq 0$  are sets of measure zero in the configuration space [?]. Hence, the agents will converge to the configurations where  $\gamma_i(q_i) = 0$  from almost all initial conditions, i.e.,  $\lim_{t \rightarrow \infty} \gamma_i(q_i(t)) = 0$ . Therefore, since  $r_{\pi_{k'_i}} - \|p_{B_i}(q_{k'_i}) - p_{k'_i}\| \leq \bar{d}_{B_i} - \epsilon_q$ , it can be concluded that there exists a finite time instance  $t_{f_i}$  such that  $\mathcal{A}_i(q_i(t_{f_i})) \in \pi_{k'_i}, \forall i \in \mathcal{N}$  and hence, each agent  $i$  will be at its goal region  $\pi_{k'_i}$  at time  $t_{f_i}, \forall i \in \mathcal{N}$ . In addition, the boundedness of  $q_i, \dot{q}_i$  implies the boundedness of the adaptation laws  $\dot{\alpha}_i, \forall i \in \mathcal{N}$ . Hence, the control laws (5.7) are also bounded.

□

## Hybrid Control Framework

Due to the proposed continuous control protocol, the transitions  $(\pi_{k_i}, t_0) \xrightarrow{i} (\pi_{k'_i}, t_{f_i})$  of Problem 5.2 are well-defined, according to Def. 5.8. Moreover, since all the agents  $i \in \mathcal{N}$  remain connected with the subset of their initial neighbors  $\tilde{\mathcal{N}}_i$  and there exist finite constants  $t_{f_i}$ , such that  $\mathcal{A}_i(q_i(t_{f_i})) \in$



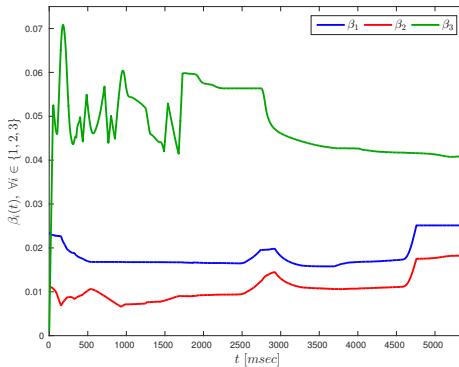
**Figure 5.12:** (a): The initial position of the agents in the workspace of the simulation example. (b): The first transition of the agents in the workspace. Agent 1 transits from  $\pi_1$  to  $\pi_2$ , agent 2 from  $\pi_2$  to  $\pi_1$ , and agent 3 from  $\pi_1$  to  $\pi_3$ . (c): The second transition of the agents in the workspace. Agent 1 transits from  $\pi_2$  to  $\pi_1$ , agent 2 from  $\pi_1$  to  $\pi_2$ , and agent 3 from  $\pi_3$  to  $\pi_2$ .

$\pi_{k'_i}, \forall i \in \mathcal{N}$ , all the agents are aware of their neighbors state, when a transition is performed. Hence, the transition system (5.6) is well defined,  $\forall i \in \mathcal{N}$ . Consider, therefore, that  $\mathcal{A}_i(q_i(0)) \in \pi_{k_{i,0}}, k_{i,0} \in \mathcal{K}_{\mathcal{R}}, \forall i \in \mathcal{N}$ , as well as a given desired path for each agent, that does not violate the connectivity condition of Problem 5.2. Then, the iterative application of the control protocol (5.7) 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 5.2.** 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 future research.

### 5.3.3 Simulation Results

To demonstrate the validity of the proposed methodology, we consider the simplified example of three agents in a workspace with  $r_0 = 12\text{m}$  and three regions of interest, with  $r_{\pi_k} = 4\text{m}$ ,  $\forall k \in \{1, 2, 3\}$ . Each agent consists of a mobile base and a rigid link connected with a rotational joint, with  $\bar{d}_{B_i} = 1\text{m}$ ,  $\forall i \in \{1, 2, 3\}$ . We also choose  $p_1 = [-5, -5]^\top \text{m}$ ,  $p_2 = [6, -4]^\top \text{m}$ ,  $p_3 = [-3, 6]^\top \text{m}$ . The initial base positions are taken as  $p_{B_1} = [-3, -4]^\top \text{m}$ ,  $p_{B_2} = [3, -4]^\top \text{m}$ ,  $p_{B_3} = [-4, -5]^\top \text{m}$ , 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. 5.12(a)). The control

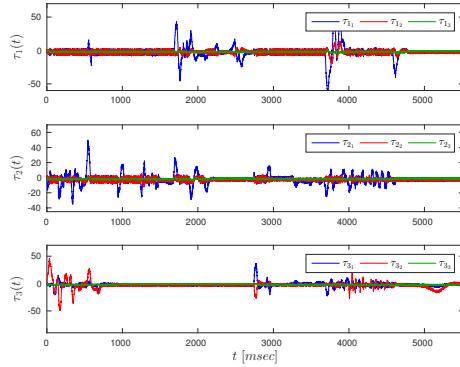


**Figure 5.13:** The obstacle functions  $\beta_i, i \in \{1, 2, 3\}$ , which remain strictly positive.

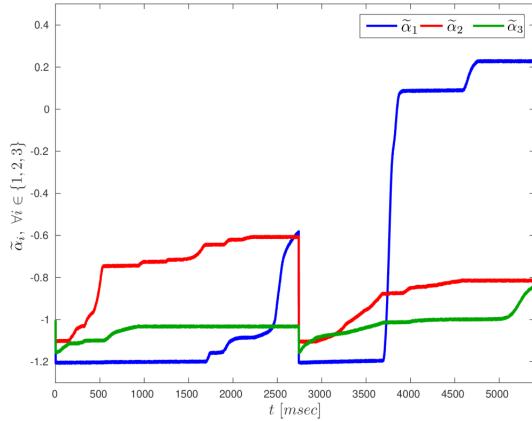
inputs for the agents are the 2D force acting on the mobile base, and the joint torque of the link. We also consider a sensing radius of  $d_{\text{con}_i} = 8\text{m}$  and the subsets of initial neighbors as  $\tilde{\mathcal{N}}_1 = \{2\}$ ,  $\tilde{\mathcal{N}}_2 = \{1, 3\}$ , and  $\tilde{\mathcal{N}}_3 = \{2\}$ , 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 are chosen as  $k_i = 5$ ,  $k_{v_i} = 10$ , and  $k_{\alpha_i} = 0.01, \forall i \in \{1, 2, 3\}$ . We employ the potential field from [? ]. The simulation results are depicted in Fig. 5.12–5.15. In particular, Fig. 5.12(b) and 5.12(c) illustrate the two consecutive transitions of the agents. Fig. 5.13 depicts the obstacle functions  $\beta_i$  which are strictly positive,  $\forall i \in \{1, 2, 3\}$ . Finally, the control inputs are given in Fig. 5.14 and the parameter errors  $\tilde{\alpha}$  are shown in Fig. 5.15, which indicates their boundedness. As proven in the theoretical analysis, the transitions are successfully performed while satisfying all the desired specifications.

## 5.4 Timed Abstractions for Distributed Cooperative Manipulation

We now switch our attention to multi-agent-object systems. Such systems include, except for a number of robotic agents, a certain number of *unactuated* objects (items in the environment). We consider that these objects, along with the agents, have themselves some local tasks to complete, expressed



**Figure 5.14:** The resulting control inputs  $\tau_i, \forall i \in \{1, 2, 3\}$  for the two transitions.



**Figure 5.15:** The parameter deviations  $\tilde{\alpha}_i, \forall i \in \{1, 2, 3\}$ , which are shown to be bounded.

as temporal specifications with respect to their locations. This section considers the problem of motion planning for one unactuated object, which is grasped by a number of robotic agents, under *timed* temporal specifications, and in particular, Metric Interval Temporal Logic (MITL) specifications. In particular, we design appropriate well-defined *timed* abstractions for a cooperatively manipulated object that allows us to express and solve the object motion planning problem under MITL formulas.

### 5.4.1 Problem Formulation

Consider a bounded workspace  $\mathcal{W} \subset \mathbb{R}^3$  containing  $N$  robotic agents rigidly grasping an object, similar to what is shown in Fig. 2.1. 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 setup considered here is the same as in Section 2.2, which we briefly recap. 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.

The dynamics of the agents are (see eq. (2.3))

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

whereas the object's

$$\begin{aligned} \dot{x}_o &= J_o(\eta_o)v_o \\ M_o(\eta_o)\ddot{v}_o + C_o(\eta_o, \omega_o)v_o + g_o + d_o(x_o, \dot{x}_o, t) &= h_o, \end{aligned}$$

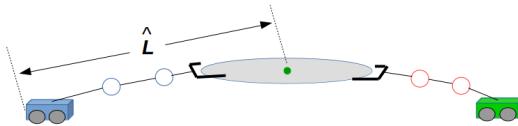
and the coupled dynamics by

$$\widetilde{M}(x)\ddot{v}_o + \widetilde{C}(x)v_o + \widetilde{g}(x) + \widetilde{d}(x, t) = G(q)^\top \bar{u}, \quad (5.10)$$

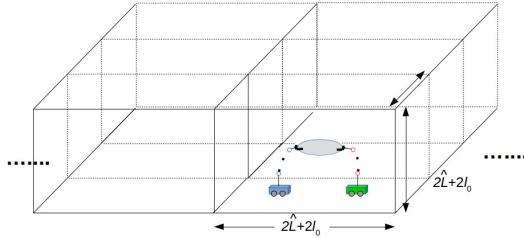
with the coupling terms as in (2.15) and  $x = [q^\top, \dot{q}^\top, \eta_o^\top, \omega_o^\top]$ .

### Workspace Partition

As already mentioned, 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.



**Figure 5.16:** An example of the system shown in Fig. 2.1 in the configuration that produces  $\hat{L}$ .



**Figure 5.17:** The workspace partition according to the bounding box of the coupled system.

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  $p_s \in \mathcal{W}$  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  $\hat{L} \geq \sup_{p_s \in \mathcal{S}_q} \|p_s - p_O(q)\|$ , 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. 5.16 shows  $\hat{L}$  for the system of Fig. 2.1. It is straightforward to conclude that

$$\mathcal{S}_q \subset \mathcal{B}(p_O(q), \hat{L}), \forall q \in \mathbb{R}^n. \quad (5.11)$$

Next, we partition the workspace  $\mathcal{W}$  into  $R$  equally sized rectangular regions  $\Pi = \{\pi_1, \dots, \pi_R\}$ , whose geometric centers are denoted by  $p_{\pi_j}^c \in \mathcal{W}, j \in \{1, \dots, R\}$ . 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  $\pi_j$  can be formally defined as follows:

$$\pi_j := \{p \in \mathcal{W} \text{ s.t. } (p)_k \in [(p_{\pi_j}^c)_k - \hat{L} - l_0, (p_{\pi_j}^c)_k + \hat{L} + l_0], \forall k \in \{x, y, z\}\},$$

with  $\|p_{\pi_{j+1}}^c - p_{\pi_j}^c\| = (2\hat{L} + 2l_0), \forall j \in \{1, \dots, R-1\}$ , and  $(p_{\pi_j}^c)_z := \hat{L} + l_0, \forall j \in \{1, \dots, R\}$ ;  $(\cdot)_k, k \in \{x, y, z\}$ , denotes the  $k$ -th coordinate. An illustration of the aforementioned partition is depicted in Fig. 5.17.

Note that each  $\pi_j$  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  $\pi_j$  as the set of its adjacent regions, i.e.,  $\mathcal{D}(\pi_j) := \{\pi_{j'} \in \Pi \text{ s.t. } \|p_{\pi_j}^c - p_{\pi_{j'}}^c\| = (2\hat{L} + 2l_0)\}$ , 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 5.9.** The coupled object-agents system is in region  $\pi_j$  at a configuration  $q$ , denoted as  $\mathcal{A}(q) \in \pi_j$ , if and only if the following hold:

1.  $\mathcal{S}_q \subset \pi_j$
2.  $\|p_o(q) - p_{\pi_j}^c\| < l_0$ .

**Definition 5.10.** 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  $\pi_j$  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

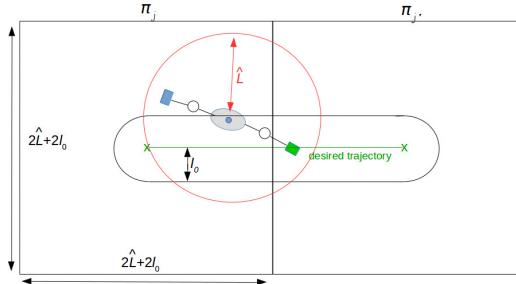
1.  $\mathcal{A}(q(t_0 + \delta t_{j,j'})) \in \pi_{j'}$ ,
2.  $\mathcal{S}_{q(t)} \subset \pi_i \cup \pi_j, \forall t \in [t_0, t_0 + \delta t_{j,j'}]$ .

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 5.10.

## Specification

Given the workspace partition, we can introduce a set of atomic propositions  $\Psi$  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  $\mathcal{L} : \Pi \rightarrow 2^\Psi$  assigns to each region  $\pi_j$  the subset of the atomic propositions  $\Psi$  that are true in  $\pi_j$ . We next provide the timed behavior, similar to Section 5.2.

**Definition 5.11.** Given a time trajectory  $q(t), t \geq 0$ , a *timed sequence* of  $q$  is the infinite sequence  $\mathbf{s}_t := (q(t_1), t_1)(q(t_2), t_2) \dots$ , 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  $\mathbf{s}_t$  is the infinite sequence  $\mathbf{b}_t := (\psi_1, t_1)(\psi_2, t_2) \dots$ , with  $\psi_m \in 2^\Psi, \psi_m \in \mathcal{L}(\pi_{j_m})$  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  $\mathcal{A}(q(t_m)) \in \pi_{j_m}$ .



**Figure 5.18:** Top view of a transition between two adjacent regions  $\pi_j$  and  $\pi_{j'}$ . Since  $p_O \in \mathcal{B}(p_{j,j'}(t), l_0)$ , we conclude that  $\mathcal{S}_q \subset \mathcal{B}(p_O, \hat{L}) \subset \mathcal{B}(p_{j,j'}(t), l_0 + \hat{L}) \subset \pi_j \cup \pi_{j'}$ .

The satisfaction of a MITL formula is provided by the following definition (see Appendix F for more details on MITL formulas).

**Definition 5.12.** The timed sequence  $s_t$  satisfies a MITL formula  $\Phi$  if and only if  $b_t \models \Phi$ .

We are now ready to state the problem treated in this section.

**Problem 5.3.** Given  $N$  agents rigidly grasping an object in  $\mathcal{W}$  subject to the coupled dynamics (5.10), the workspace partition  $\Pi$  such that  $\mathcal{A}(q(0)) \in \pi_{j_0}, j_0 \in \{1, \dots, R\}$ , a MITL formula  $\Phi$  over  $\Psi$  and the labeling function  $\mathcal{L}$ , derive a control strategy that achieves a timed sequence  $s_t$  which yields the satisfaction of  $\Phi$ .

## 5.4.2 Main Results

### Control Design

The first ingredient of the proposed solution is the design of a decentralized control protocol  $u$  such that a transition relation between two adjacent regions according to Definition 5.10 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  $\mathcal{S}_{q(t)} \subset \pi_j \cup \pi_{j'}, \forall t \in [t_0, t_0 + \delta t_{j,j'}]$ , for a predefined arbitrary constant  $\delta t_{j,j'} \in \mathbb{R}_{\geq 0}$  corresponding to the transition  $\pi_j \xrightarrow{\tau} \pi_{j'}$ .

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  $p_{\pi_j}^c$  and  $p_{\pi_{j'}}^c$ , i.e. define  $p_{j,j'} : [t_0, \infty) \rightarrow \mathbb{R}^3$ , such that  $p_{j,j'}(t_0) =$

$$p_{\pi_j}^c, p_{j,j'}(t) = p_{\pi_{j'}}^c, \forall t \geq t_0 + \delta t_{j,j'} \text{ and}$$

$$\mathcal{B}(p_{j,j'}(t), \hat{L} + l_0) \subset \pi_j \cup \pi_{j'}, \quad \forall t \geq t_0. \quad (5.12)$$

An example of  $p_{j,j'}$  is

$$p_{j,j'}(t) = \begin{cases} \frac{p_{\pi_{j'}}^c - p_{\pi_j}^c}{\delta t_{j,j'}} t + \frac{p_{\pi_j}^c (\delta t_{j,j'} - 1) - p_{\pi_{j'}}^c}{\delta t_{j,j'}} t_0, & t \in [t_0, t_0 + \delta t_{j,j'}] \\ p_{\pi_{j'}}^c, & t \in [t_0 + \delta t_{j,j'}, \infty) \end{cases} \quad (5.13)$$

The intuition behind the solution of Problem 5.3 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.,  $\|p_O(t) - p_{j,j'}(t)\| < l_0, \forall t \geq t_0$ , then  $\|p_O(t_0 + \delta t_{j,j'}) - p_{\pi_j}^c\| < l_0$  and, by invoking (5.11) and (5.12), we obtain  $\mathcal{S}_{q(t)} \subset \mathcal{B}(p_O(t), \hat{L}) \subset \mathcal{B}(p_{j,j'}(t), \hat{L} + l_0) \subset \pi_j \cup \pi_{j'}, \forall t \geq t_0$  (and therefore  $t \in [t_0, t_0 + \delta t_{j,j'}]$ ), and thus the requirements of Definition 5.10 for the transition relation are met. Fig. 5.18 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  $\eta_{j,j'} := [\phi_{j,j'}, \theta_{j,j'}, \psi_{j,j'}]^\top : [t_0, \infty) \rightarrow \mathbb{T}$  with bounded time derivative, that has to be tracked by the object's center of mass. We choose  $\theta_{j,j'}(t) \in [-\bar{\theta}, \bar{\theta}] \subset (-\frac{\pi}{2}, \frac{\pi}{2}), \forall t \in \mathbb{R}_{\geq 0}$ , with  $\bar{\theta} \in (0, \frac{\pi}{2})$ , so as to ensure the singularity avoidance of  $J_O(\eta_O)$ . We form, therefore, the desired pose trajectory  $x_{j,j'} : [t_0, \infty) \rightarrow \mathbb{M}$ , with  $x_{j,j'}(t) := [p_{j,j'}(t)^\top, \eta_{j,j'}(t)^\top]^\top$ . In case of multiple consecutive transitions  $\dots \pi_h \xrightarrow{\mathcal{T}} \pi_j \xrightarrow{\mathcal{T}} \pi_{j'} \xrightarrow{\mathcal{T}} \pi_{h'} \dots$  over the intervals  $\dots, \delta t_{h,j}, \delta t_{j,j'}, \delta t_{j',h'}, \dots$ , the desired orientation trajectories  $\dots, \eta_{h,j}(t), \eta_{j,j'}(t), \eta_{j',h'}(t), \dots$  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 5.3 is equivalent to a problem of trajectory tracking within certain bounds.

A suitable methodology for the control design in hand is that of prescribed performance control, which was used for the cooperative manipulation problem in Section 2.2.4. We describe it briefly here and associate it with the abstraction problem.

We consider first the associated position and orientation error as in (2.29):

$$e_s := [e_{s_x}, e_{s_y}, e_{s_z}, e_{s_\phi}, e_{s_\theta}, e_{s_\psi}]^\top := x_O - x_{j,j'}(t). \quad (5.14)$$

Following that section as well as Appendix B, the mathematical expressions of prescribed performance are given by the inequalities:

$$-\rho_{s_k}(t) < e_{s_k}(t) < \rho_{s_k}(t), \quad \forall k \in \mathcal{K}, \quad (5.15)$$

$\forall t \in [t_0, \infty)$ , where  $\mathcal{K} = \{x, y, z, \phi, \theta, \psi\}$ ,  $\rho_{s_k} := \rho_{s_k}(t) : [t_0, \infty) \rightarrow \mathbb{R}_{>0}$  with

$$\rho_{s_k}(t) = (\rho_{s_k,0} - \rho_{s_k,\infty}) \exp(-l_{s_k} t) + \rho_{s_k,\infty}, \quad \forall k \in \mathcal{K}, \quad (5.16)$$

as in (2.31).

The proposed prescribed performance control protocol does not incorporate any information on the agents' or the object's dynamics or the external disturbances and guarantees (5.15) for all  $t \in [t_0, \infty)$  and hence  $[t_0, t_0 + \delta t_{j,j'}]$ , which, by appropriately selecting  $\rho_{s_k}(t), k \in \mathcal{K}$  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.

As in Section 2.2.4, consider the following steps: **Step I-a.** Select the corresponding functions  $\rho_{s_k}$  as in (5.16) with

- (i)  $\rho_{s_\theta,0} = \rho_{s_\theta}(t_0) = \theta^*, \rho_{s_k,0} = \rho_{s_k}(t_0) = l_0, \forall k \in \{x, y, z\}$   $\rho_{s_k,0} = \rho_{s_k}(t_0) > \|e_{s_k}(t_0)\|, \forall k \in \{\phi, \psi\}$ ,
- (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  $\theta^*$  is a positive constant satisfying  $\theta^* + \bar{\theta} < \frac{\pi}{2}$ .

**Step I-b.** Introduce the normalized errors

$$\xi_s := [\xi_{s_x}, \dots, \xi_{s_\psi}]^\top := \rho_s^{-1} e_s,$$

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

$$\begin{aligned} \varepsilon_s &:= \varepsilon_s(\xi_s) := [\varepsilon_{s_x}, \dots, \varepsilon_{s_\psi}]^\top := \left[ \ln \left( \frac{1+\xi_{s_x}}{1-\xi_{s_x}} \right), \dots, \ln \left( \frac{1+\xi_{s_\psi}}{1-\xi_{s_\psi}} \right) \right]^\top \\ r_s &:= r_s(\xi_s) := \text{diag}\{r_{s_k}(\xi_{s_k})\}_{k \in \mathcal{K}} := \text{diag} \left\{ \left[ \frac{\partial \varepsilon_{s_k}}{\partial \xi_{s_k}} \right]_{k \in \mathcal{K}} \right\} \\ &= \text{diag} \left\{ \left[ \frac{2}{1 - \xi_{s_k}^2} \right]_{k \in \mathcal{K}} \right\} \end{aligned}$$

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

$$v_r := v_r(\xi_s, t) := -g_s J_O \left( \eta_d(t) + \rho_{s_\eta}(t) \xi_{s_\eta} \right)^{-1} \rho_s(t)^{-1} r_s(\xi_s) \varepsilon_s,$$

where  $\rho_{s_\eta} := \rho_{s_\eta}(t) := \text{diag}\{\rho_{s_\phi}, \rho_{s_\theta}, \rho_{s_\psi}\}$ ,  $\xi_{s_\eta} := [\xi_{s_\phi}, \xi_{s_\eta}, \xi_{s_\phi}]^\top$ , and we have further used the relation  $\xi_s = \rho_s^{-1}(x_o - x_d)$  from (2.29) and (2.32).

**Step II-a.** Define the velocity error vector

$$e_v := [e_{v_x}, \dots, e_{v_\psi}]^\top := v_o - v_r,$$

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

**Step II-b.** Define the normalized velocity error

$$\xi_v := [\xi_{v_x}, \dots, \xi_{v_\psi}]^\top := \rho_v^{-1} e_v,$$

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

$$\begin{aligned} \varepsilon_v := \varepsilon_v(\xi_v) &:= [\varepsilon_{v_x}, \dots, \varepsilon_{v_\psi}]^\top := \left[ \ln \left( \frac{1+\xi_{v_x}}{1-\xi_{v_x}} \right), \dots, \ln \left( \frac{1+\xi_{v_\psi}}{1-\xi_{v_\psi}} \right) \right]^\top \\ r_v(\xi_v) &:= \text{diag}\{r_{v_k}(\xi_{v_k})\}_{k \in \mathcal{K}} := \text{diag} \left\{ \left[ \frac{\partial \varepsilon_{v_k}}{\partial \xi_{v_k}} \right]_{k \in \mathcal{K}} \right\} \\ &= \text{diag} \left\{ \left[ \frac{2}{1 - \xi_{v_k}^2} \right]_{k \in \mathcal{K}} \right\}, \end{aligned} \quad (5.17)$$

and design the decentralized feedback control protocol for each agent  $i \in \mathcal{N}$  as  $u_i : \mathsf{S}_i \times (-1, 1)^6 \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ , with

$$u_i := u_i(q_i, \xi_v, t) := -g_v J_{M_i}(q_i) \rho_v^{-1} r_v(\xi_v) \varepsilon_v(\xi_v), \quad (5.18)$$

where  $g_v$  is a positive constant gain and  $J_{M_i}$  as defined in (2.24).

The control law (5.18) can be written in vector form:

$$u = U_j^{j'} := -g_v G_M^+(q) \rho_v^{-1} r_v(\xi_v) \varepsilon_v(\xi_v), \quad (5.19)$$

where  $G_M^+(q)$  as in (2.40), and the notation  $U_j^{j'}$  stands for the transition from  $\pi_j$  to  $\pi_{j'}$ .

The next theorem summarizes the results of this section.

**Theorem 5.2.** Consider  $N$  agents rigidly grasping an object with unknown coupled dynamics (5.10) and  $\mathcal{A}(q(t_0)) \in \pi_j$ ,  $j \in \{1, \dots, R\}$  as well as  $|\theta(t_0) - \theta_{j,j'}(t_0)| < \theta^*$ . Then, the distributed control protocol (5.14)-(5.17) 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  $\pi_j$  and  $\pi_{j'}$  for the coupled object-agents system, according to Definition 5.10.

*Proof.* By following the proof of Theorem 2.2 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  $|e_{s_k}(t)| < \rho_{s_k}(t)$ ,  $\forall k \in \mathcal{K}$  and thus  $|e_{s_k}(t)| < l_0$ ,  $\forall k \in \{x, y, z\}$ ,  $t \in [t_0, \infty)$ , since  $\rho_{s_k,0} = l_0$ ,  $\forall k \in \{x, y, z\}$ . Therefore,  $p_O(q(t)) \in \mathcal{B}(p_{j,j'}(t), l_0)$ ,  $\forall t \geq t_0$  and, consequently,  $p_O(q(t_0 + \delta t_{j,j'})) \in \mathcal{B}(p_{\pi_{j'}}^c, l_0)$ , since  $p_{j,j'}(t_0 + \delta t_{j,j'}) = p_{\pi_{j'}}^c$ .

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 (5.11) and (5.12), we conclude that  $\mathcal{S}_{q(t)} \subset \pi_j \cup \pi_{j'}, \forall t \in [t_0, t_0 + \delta t_{j,j'}] \subset [t_0, \infty)$ , and therefore a transition relation with time duration  $\delta t_{j,j'}$  is successfully established. Finally, according to the proof of Theorem 2.2, it holds  $|\theta_O(t)| < \frac{\pi}{2}, \forall t \geq t_0$  and hence representation singularities are provably avoided.  $\square$

### High-Level Timed Plan Generation

The second part of the proposed solution is the derivation of a high-level plan that satisfies the given MITL formula  $\Phi$  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) [?]

$$\mathcal{T} = \{\Pi, \Pi_0, \xrightarrow{\mathcal{T}}, \Psi, \mathcal{L}, \gamma_{\mathcal{T}}\},$$

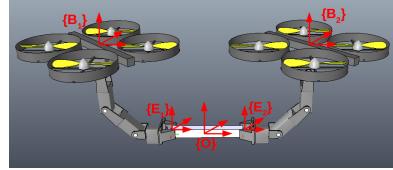
where

- $\Pi$  is the set of states defined in Section 5.4.1,
- $\Pi_0 \subset \Pi$  is a set of initial states,
- $\xrightarrow{\mathcal{T}} \subseteq \Pi \times \Pi$  is a transition relation according to Definition 5.10,
- $\Psi$  and  $\mathcal{L}$  are the atomic propositions and the labeling function, respectively, as defined in Section 5.4.1, and
- $\gamma_{\mathcal{T}} : (\xrightarrow{\mathcal{T}}) \rightarrow \mathbb{R}_{\geq 0}$  is a map that assigns to each transition its time duration, i.e.,  $\gamma_{\mathcal{T}}(\pi_j \xrightarrow{\mathcal{T}} \pi_{j'}) = \delta t_{j,j'}$ .

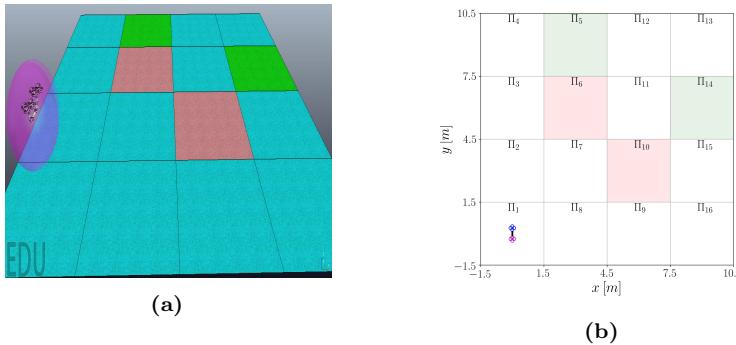
Therefore, by designing the switching protocol  $U_{r_j}^{r_{j+1}}(t)$  from (5.19):

$$U_{r_j}^{r_{j+1}}(t) = -g_v G_M^+(q(t)) \rho_v(t)^{-1} r_v(\xi_v(t)) \varepsilon_v(\xi_v(t)), \forall t \in [t_j, t_j + \delta t_{r_j, r_{j+1}}),$$

$j \in \mathbb{N}$ , 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  $r_{\mathcal{WTS}} := (\pi_{r_1}, t_1)(\pi_{r_2}, t_2) \dots$ , 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*  $w_{\mathcal{WTS}}(r) := (\mathcal{L}(\pi_{r_1}), t_1)(\mathcal{L}(\pi_{r_2}), t_2) \dots$  over  $\Psi$  where  $\mathcal{L}(\pi_{r_j}), j \in \mathbb{N}$ , is the subset of the atomic propositions  $\Psi$  that are true when  $\mathcal{A}(q(t_j)) \in \pi_{r_j}$ .



**Figure 5.19:** The aerial robots employed in the simulation rigidly grasping an object.



**Figure 5.20:** 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.

The given MITL formula  $\Phi$  is translated into a *Timed Büchi Automaton*  $\mathcal{A}_\phi^t$  [? ] and the product  $\mathcal{A}_p = \mathcal{T} \otimes \mathcal{A}_\phi^t$  is built [? ]. The projection of the accepting runs of  $\mathcal{A}_p$  onto  $\mathcal{T}$  provides a *timed run*  $r_{w\tau}$  of  $\mathcal{T}$  that satisfies  $\phi$ ;  $r_{w\tau}$  has the form  $r_{w\tau} = (\pi_{r_1}, t_1)(\pi_{r_2}, t_2)\dots$ , i.e., an infinite sequence<sup>1</sup> of regions  $\pi_{r_j}$  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 can be found in [? ? ? ].

The execution of  $r_{w\tau} = (\pi_{r_1}, t_1)(\pi_{r_2}, t_2)\dots$  produces a trajectory  $q(t), t \in \mathbb{R}_{\geq 0}$ , with timed sequence  $s_t = (q(t_1), t_1)(q(t_2), t_2)\dots$ , with  $\mathcal{A}(q(t_j)) \in \pi_{r_j}, \forall j \in \mathbb{N}$ . Following Definition 5.11,  $s_t$  has the timed behavior  $b_t = (\psi_1, t_1)(\psi_1, t_2)\dots$  with  $\psi_j \in \mathcal{L}(\pi_{r_j})$ , for  $\mathcal{A}(q(t_j)) \in \pi_{r_j}, \forall j \in \mathbb{N}$ . The latter implies that  $s_t \models \Phi$  and therefore that  $b_t$  satisfies  $\Phi$ . The aforementioned discussion is summarized as follows:

**Theorem 5.3.** *The execution of  $r_{w\tau} = (\pi_{r_1}, t_1)(\pi_{r_2}, t_2)\dots$  of  $\mathcal{T}$  that satis-*

<sup>1</sup>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 [? ].

fies  $\Phi$  guarantees a timed behavior  $s_t$  of the coupled object-agents system that yields the satisfaction of  $\Phi$  and provides, therefore, a solution to Problem 5.3.

### 5.4.3 Simulation Results

The validity of the proposed framework is verified through a simulation study in the Virtual Robot Experimentation Platform (V-REP) [? ]. We consider a rectangular rigid body of dimensions  $0.025 \times 0.2 \times 0.025$  m<sup>3</sup> 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 revolute degrees of freedom as depicted in Fig. 5.19. We consider that the quadrotor is fully actuated, as mentioned in Section 5.4.1, and there exists an embedded algorithm that translates the generalized force of the quadrotor base to the actual motor inputs.

The initial conditions of the system are taken as  $p_o(0) = [0, 0, 1.5]^\top$  m,  $\eta_o(0) = [0, 0, 0]^\top$  rad. The workspace is partitioned into  $R = 16$  regions, with  $\hat{L} = 0.75$  m and  $l_0 = 0.5$  m. Fig. 5.20 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  $\Psi = \{\text{"green}_1\text{", "green}_2\text{", "red", "obs"}\}$ , representing goal ("green<sub>1</sub>", "green<sub>2</sub>") and obstacle ('obs') regions with  $\mathcal{L}(\pi_5) = \{\text{"green}_1\text{"}\}$ ,  $\mathcal{L}(\pi_{14}) = \{\text{"green}_2\text{"}\}$ ,  $\mathcal{L}(\pi_6) = \mathcal{L}(\pi_{10}) = \{\text{"obs"}\}$  and  $\mathcal{L}(\pi_j) = \emptyset$ , for the remaining regions.

We consider the MITL formula

$$\Phi = (\square_{[0, \infty)} \neg \text{"obs"}) \wedge \diamond_{[0, 60]} (\text{"green}_1 \wedge \text{"green}_2),$$

which describes the following behavior: the coupled system

1. must always avoid the obstacle regions,
2. must visit the greens region in the first 60 seconds.

By following the procedure described in Section 5.4.2, we obtain the accepting timed run

$$r_{w\tau} = (\pi_{r_1}, t_1)(\pi_{r_2}, t_2) \cdots = (\pi_1, 0)(\pi_2, 6)(\pi_3, 12)(\pi_4, 18)(\pi_5, 24)(\pi_{12}, 30) \\ (\pi_{13}, 36)(\pi_{14}, 42)(\pi_{11}, 48)(\pi_{12}, 54)(\pi_5, 60).$$

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 (5.13) and  $\eta_{r_j, r_{j'}}(t) = [0, 0, \frac{\pi}{4} \sin(\frac{\pi}{3}(t - t_{r_j}))]^\top$ , 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  $\rho_{s_k, 0} = \rho_{s_k}(t_{r_j}) = l_0 = 0.5$  [m],  $l_{s_k} = 0.5$ ,  $\rho_{s_k, \infty} = \lim_{t \rightarrow \infty} \rho_{s_k}(t) = 0.1$  [m],  $\forall k \in \{x, y, z\}$ ,  $\rho_{s_k, 0} =$

$\rho_{s_k}(t_{r_j}) = \frac{\pi}{2}$  [rad],  $l_{s_k} = 0.5$ ,  $\rho_{s,k,\infty} = \lim_{t \rightarrow \infty} \rho_{s_k}(t) = \frac{\pi}{12}$  r,  $\forall k \in \{\phi, \theta, \psi\}$ ,  $\rho_{v_k,0} = \rho_{v_k}(t_{r_j}) = 2|e_{v_k}(t_{r_j})| + 0.5$ ,  $l_{v_k} = 0.5$  and  $\rho_{v_k,\infty} = \lim_{t \rightarrow \infty} \rho_{v_k}(t) = 0.1$ ,  $k \in \mathcal{K}, j \in \{1, \dots, 10\}$ . The control gains are chosen as  $g_s = 1$ ,  $g_v = 10$ , and the agents are set to contribute equally to the object motion.

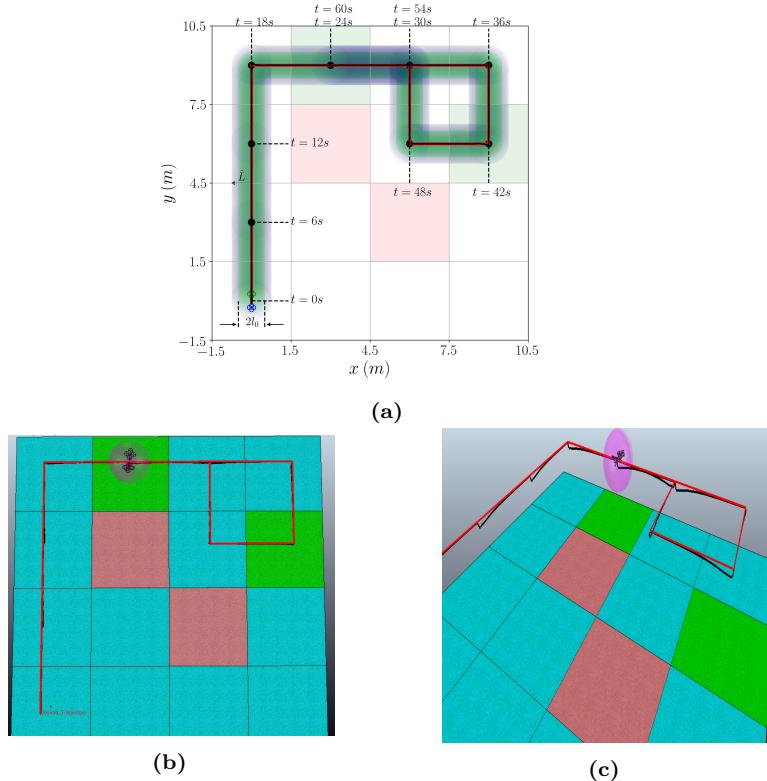
The simulation results are depicted in Figs. 5.21–5.24. More specifically, Fig. 5.21 depicts the timed transitions of the coupled object-agents system, from which it can be deduced that  $p_o(t) \in \mathcal{B}(p_{r_j, r_{j'}}, l_0)$  and therefore  $\mathcal{S}_q(t) \subset \pi_{r_j} \cup \pi_{r_{j'}}$ ,  $\forall j \in \{1, \dots, 10\}$ . Moreover, Fig. 5.22 and 5.23 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  $\pi_{r_j} \rightarrow \pi_{r_{j'}}$ ,  $j \in \{1, \dots, 10\}$ . Finally, the resulted control inputs  $\tau_1, \tau_2$  for the two agents are shown in Fig. 5.24. A video showing the aforementioned simulation paradigm can be found on <https://youtu.be/AiAt9NqL1jo>.

## 5.5 Planning and Control for Multi-Robot-Object Systems under Temporal Logic Formulas

The final section of this chapter considers the general case of a multi-robot-object system, with  $N > 1$  agents and  $M > 1$  objects. Unlike the previous section, the objects are now not assumed to be grasped in the starting configuration. Moreover, temporal logic specifications are imposed both to the robotic agents and the objects, whose behavior depends on the agent actions.

### 5.5.1 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.,  $\mathcal{W} := \mathcal{B}(0, r_0)$ , where  $r_0 \in \mathbb{R}_{>0}$  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  $\pi_k := \bar{\mathcal{B}}(p_{\pi_k}, r_{\pi_k})$ , where  $p_{\pi_k} \in \mathbb{R}^3$  is the center and  $r_{\pi_k} \in \mathbb{R}_{>0}$  the radius of  $\pi_k$ . We denote the set of all  $\pi_k$  as  $\Pi := \{\pi_1, \dots, \pi_K\}$  and  $\mathcal{K}_{\mathcal{R}} := \{1, \dots, K\}$ . Moreover, we introduce disjoint sets of atomic propositions  $\Psi_i, \Psi_j^o$ , expressed as boolean variables, that represent services provided to agent  $i \in \mathcal{N}$  and object  $j \in \mathcal{M} := \{1, \dots, M\}$  in  $\Pi$ . The services provided at each region  $\pi_k$  are given by the labeling functions  $\mathcal{L}_i : \Pi \rightarrow 2^{\Psi_i}, \mathcal{L}_j^o : \Pi \rightarrow 2^{\Psi_j^o}$ , which assign to each region  $\pi_k, k \in \mathcal{K}_{\mathcal{R}}$ , the subset of services  $\Psi_i$  and  $\Psi_j^o$ , respectively, that can be provided in that region to agent  $i \in \mathcal{N}$  and object  $j \in \mathcal{M}$ , respectively. In

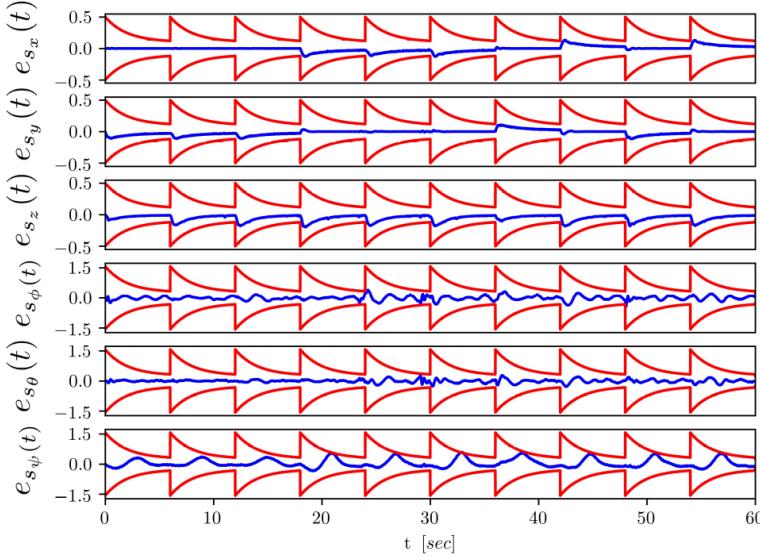


**Figure 5.21:** (a): The overall desired object trajectory (with red), the actual object trajectory (with black), the domain specified by  $\mathcal{B}(p_{r_j, r_{j'}}(t), l_0)$ ,  $\forall j \in \{1, \dots, 10\}$  (with green), and the domain specified by  $\mathcal{B}(p_O(t), \hat{L})$  (with blue), for  $t \in [0, 60]$  s. (b), (c): Illustration of the system at the final region at  $t = 60s$  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.

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  $init : \mathcal{N} \rightarrow \mathcal{K}_{\mathcal{R}}$ ,  $init_O : \mathcal{M} \rightarrow \mathcal{K}_{\mathcal{R}}$  specify the initial region indices.

The notation and modeling is identical to the one of the previous section and of Chapter 2, which we briefly recap. For this section, we denote by  $[z_i]_{i \in \mathcal{A}}$  the stack column vector of the vectors/scalars  $z_i$ ,  $i \in \mathcal{A}$ , where  $\mathcal{A}$  is an index set.

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



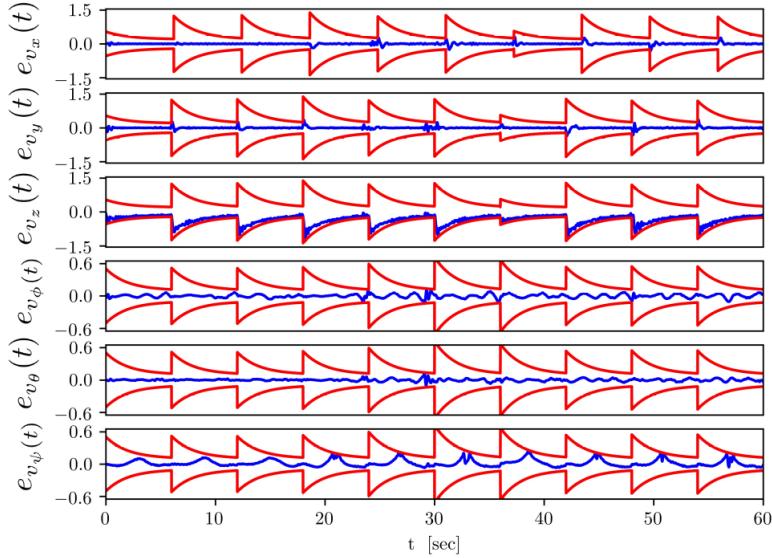
**Figure 5.22:** The pose errors  $e_s(t)$  (with blue) along with the performance functions  $\rho_s(t)$  (with red).

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  $i$ th end-effector, denoted by  $p_{E_i} = p_{E_i}(q_i) : \mathbb{R}^{n_i} \rightarrow \mathbb{R}^3$  and  $\eta_{E_i} := \eta_{E_i}(q_i) : \mathbb{R}^{n_i} \rightarrow \mathbb{T}$ , respectively, expressed in an inertial reference frame, can be derived by the forward kinematics. The generalized velocity of each agent's end-effector  $v_i = [\dot{p}_{E_i}^\top, \omega_{E_i}^\top]^\top \in \mathbb{R}^6$  is given by  $v_i = J_i(q_i)\dot{q}_i$ , where  $J_i = J_i(q_i) : \mathbb{R}^{n_i} \rightarrow \mathbb{R}^{6 \times n_i}$  is the geometric Jacobian matrix,  $\forall i \in \mathcal{N}$ . The matrix inverse of  $J_i$  is well defined in the set away from *kinematic singularities*,  $S_i := \{q_i \in \mathbb{R}^{n_i} : \det(J_i(q_i)J_i(q_i)^\top) > 0\}$ ,  $\forall i \in \mathcal{N}$ . The joint- and task-space dynamics of each agent are

$$B_i(q_i)\ddot{q}_i + C_{q_i}(q_i, \dot{q}_i)\dot{q}_i + g_{q_i}(q_i) = \tau_i - J_i(q_i)^\top h_i \quad (5.20a)$$

$$M_i(q_i)\ddot{v}_i + C_i(q_i, \dot{q}_i)v_i + g_i(q_i) = u_i - h_i, \quad (5.20b)$$

with the standard dynamic terms (see previous chapters). Note again that the terms of (5.20b) are only defined in  $S_i$ , away from singular configurations. Avoidance of such configurations is not explicitly taken account here. 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.



**Figure 5.23:** The velocity errors  $e_v(t)$  (with blue) along with the performance functions  $\rho_v(t)$  (with red).

We consider that each agent  $i$ , for a given  $q_i$ , covers a spherical region  $\mathcal{A}_i := \bar{\mathcal{B}}_i(c_i(q_i), r_i) \subset \mathbb{R}^3$  of constant radius  $r_i \in \mathbb{R}_{>0}$  that bounds its volume for that given  $q_i$ , where  $c_i := c_i(q_i) : \mathbb{R}^{n_i} \rightarrow \mathbb{R}^3$  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  $\epsilon_i > 0$ ,  $i \in \mathcal{N}$ , via an analogous relation.

Regarding the objects, we slightly change the notation with respect to the previous chapters and denote by  $x_j^o := [(p_j^o)^\top, (\eta_j^o)^\top]^\top \in \mathbb{M}$ ,  $v_j^o := [(\dot{p}_j^o)^\top, (\omega_j^o)^\top]^\top \in \mathbb{R}^{12}$ ,  $\forall j \in \mathcal{M}$ , the pose and generalized velocity of the  $j$ th object's center of mass. The object dynamic equations are given by the standard Newton-Euler form:

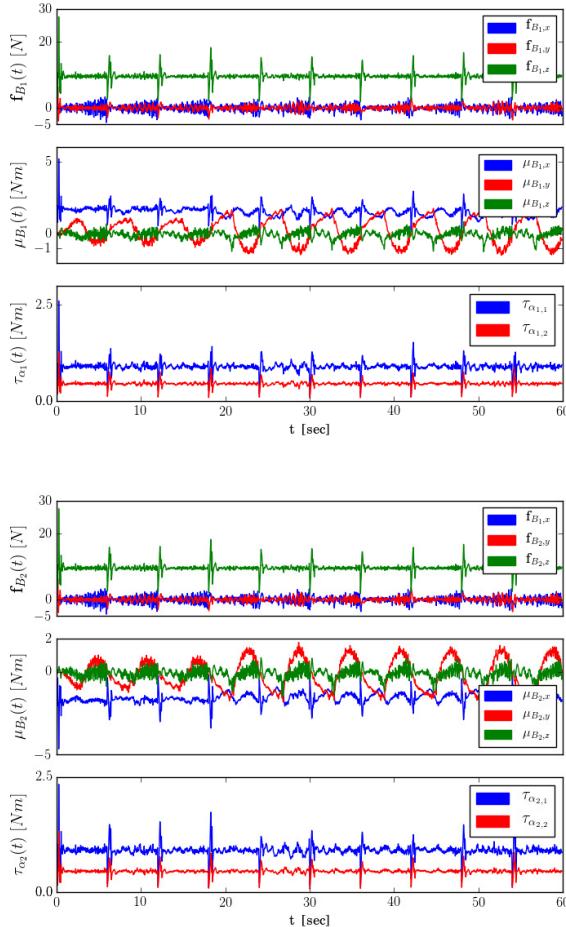
$$\dot{x}_j^o = J_j^o(x_j^o)v_j^o, \quad (5.21a)$$

$$M_o(\eta_j^o)\ddot{v}_j^o + C_o(\eta_j^o, \omega_j^o)v_j^o + g_o = h_j^o. \quad (5.21b)$$

Similarly to the agents, each object's volume is represented by the spherical set  $\mathcal{O}_j := \mathcal{O}_j(p_j^o) := \bar{\mathcal{B}}_j(p_j^o, r_j^o) \subset \mathbb{R}^3$  of a constant radius  $r_j^o \in \mathbb{R}_{>0}$ ,  $\forall j \in \mathcal{M}$ <sup>2</sup>.

---

<sup>2</sup>Different center might be considered to obtain less conservative spherical volume.



**Figure 5.24:** The resulting control inputs  $\tau_i = [f_{B_i}^\top, \mu_{B_i}^\top, \tau_{\alpha_{i,1}}, \tau_{\alpha_{i,2}}]$  for  $i = 1$  and  $i = 2$ ;  $f_{B_i}, \mu_{B_i}, \tau_{\alpha_i}$  are the quadrotor base forces and torques and the manipulator torque commands, respectively.

Similarly to (2.14), the coupled dynamics between an object  $j \in \mathcal{M}$  and a subset  $\mathcal{V} \subseteq \mathcal{N}$  of agents that grasp it rigidly is given by

$$\widetilde{M}_{\mathcal{V},j} \dot{v}_j^O + \widetilde{C}_{\mathcal{V},j} v_j^O + \widetilde{g}_{\mathcal{V},j} = G_{\mathcal{V},j} u_{\mathcal{V}}, \quad (5.22)$$

where

$$\begin{aligned}\widetilde{M}_{\mathcal{V},j} &:= \widetilde{M}_{\mathcal{V},j}(x_{\mathcal{V},j}) := M_O + G_{\mathcal{V},j} M_{\mathcal{V}} G_{\mathcal{V},j}^\top \\ \widetilde{C}_{\mathcal{V},j} &:= \widetilde{C}_{\mathcal{V},j}(x_{\mathcal{V},j}) := C_O + G_{\mathcal{V},j} M_{\mathcal{V}} \dot{G}_{\mathcal{V},j}^\top + G_{\mathcal{V},j} C_{\mathcal{V}} G_{\mathcal{V},j}^\top \\ \widetilde{g}_{\mathcal{V},j} &:= \widetilde{g}_{\mathcal{V},j}(x_{\mathcal{V},j}) := g_O + G_{\mathcal{V},j} g_{\mathcal{V}} \\ G_{\mathcal{V},j} &:= [(J_{1,j}^O)^\top, \dots, (J_{|\mathcal{V}|,j}^O)^\top],\end{aligned}$$

$J_{i,j}^O \in \mathbb{R}^{6 \times 6}$ ,  $i \in \mathcal{V}$ , is the object-to-agent Jacobian matrix (see (2.10)), and  $x_{\mathcal{V},j}$  is the overall state  $x_{\mathcal{V},j} := [q_{\mathcal{V}}^\top, \dot{q}_{\mathcal{V}}^\top, (x_j^O)^\top, (v_j^O)^\top]^\top \in \mathsf{S}_{\mathcal{V}} \times \mathbb{R}^{n_{\mathcal{V}}+6} \times \mathbb{M}$ , where  $\mathsf{S}_{\mathcal{V}} := \prod_{i \in \mathcal{V}} \mathsf{S}_i$ , and  $n_{\mathcal{V}} := |\mathcal{V}|$ . The subscript  $\mathcal{V}$  here corresponds to the agents of the set  $\mathcal{V}$ . We also use the following Lemma from Chapter 2 that is necessary for the following analysis.

**Lemma 5.1.** *The matrices  $B_i(q_i)$  and  $\widetilde{M}_{\mathcal{V},j}$  are symmetric and positive definite and the matrices  $\dot{B}_i - 2C_{q_i}$  and  $\widetilde{M}_{\mathcal{V},j} - 2\widetilde{C}_{\mathcal{V},j}$  are skew symmetric,  $\forall i \in \mathcal{N}, j \in \mathcal{M}, \mathcal{V} \subseteq \mathcal{N}$ .*

Regarding the volume of the coupled agents-object system, we denote by  $\mathcal{AO}_{\mathcal{V},j} := \mathcal{AO}_{\mathcal{V},j}(p_j^O) := \bar{\mathcal{B}}(p_j^O, r_{\mathcal{V},j}) \subset \mathbb{R}^3$  the sphere centered at  $p_j^O$  with constant radius  $r_{\mathcal{V},j} \in \mathbb{R}_{>0}$ , which is large enough to cover the volume of the coupled system in all configurations  $q_{\mathcal{V}}$ <sup>3</sup>. This conservative formulation emanates from the sphere-world restriction of the multi-agent navigation function framework [? ? ]. In order to take into account other spaces, ideas from [?] could be employed or extensions of the respective works of [? ], [?] to the multi-agent case could be developed.

Moreover, in order to take into account the introduced agents' power capabilities  $\mathbf{c}_i$ ,  $i \in \mathcal{N}$ , we consider a function  $\Lambda \in \{\text{True}, \text{False}\}$  that outputs whether the agents that grasp an object are able to transport the object, based on their power capabilities. For instance,  $\Lambda(m_j^O, \mathbf{c}_{\mathcal{V}}) = \text{True}$ , where  $m_j^O \in \mathbb{R}_{>0}$  is the mass of object  $j$  and  $\mathbf{c}_{\mathcal{V}} := [\mathbf{c}_i]_{i \in \mathcal{V}}$ , implies that the agents  $\mathcal{V}$  have sufficient power capabilities to cooperatively transport object  $j$ .

Next, we define the boolean functions  $\mathcal{AG}_{i,j} : \mathbb{R}^{n_i} \times \mathbb{M} \rightarrow \{\text{True}, \text{False}\}$ ,  $i \in \mathcal{N}, j \in \mathcal{M}$ , to denote whether agent  $i \in \mathcal{N}$  rigidly grasps an object  $j \in \mathcal{M}$  at a given configuration  $q_i, x_j^O$ ; We also define  $\mathcal{AG}_{i,0} : \mathbb{R}^{n_i} \times \mathbb{M}^M \rightarrow \{\text{True}, \text{False}\}$ , to denote that agent  $i$  does not grasp any objects, i.e.,  $\mathcal{AG}_{i,j}(q_i, x_j^O) =$

<sup>3</sup> $r_{\mathcal{V},j}$  can be chosen as the largest distance of the object's center of mass to a point in the agents' volume over all possible  $q_{\mathcal{V}}$  (see previous section).

$\text{False}, \forall j \in \mathcal{M} \Leftrightarrow \mathcal{AG}_{i,0}(q_i, x^o) = \text{True}, \forall i \in \mathcal{N}$ , where  $x^o := [x_j^o]_{j \in \mathcal{M}} \in \mathbb{M}^M$ . Note also that  $\mathcal{AG}_{i,\ell}(q_i, x_\ell^o) = \text{True}, \ell \in \mathcal{M} \Leftrightarrow \mathcal{AG}_{i,j}(q_i, x_j^o) = \text{False}, \forall j \in \mathcal{M} \setminus \{\ell\}$ , i.e., agent  $i$  can grasp at most one object at a time.

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  $\pi_k$  and a number  $\tilde{N} \in \mathbb{N}$  of sphere radii (of agents and/or objects) the procedure can be seen as a function  $\mathcal{P}_s := [\mathcal{P}_{s,0}, \mathcal{P}_{s,1}^\top]^\top$ , where  $\mathcal{P}_{s,0} : \mathbb{R}_{\geq 0}^{\tilde{N}+1} \rightarrow \{\text{True}, \text{False}\}$  outputs whether the spheres fit in the region  $\pi_k$  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,  $P_{s,0}(r_{\pi_2}, r_1, r_3, r_1^o, r_5^o)$  determines whether the agents 1,3 and the objects 1,5 fit in region  $\pi_2$ , without colliding with each other;  $(q_1, q_3, x_1^o, x_5^o) = P_{s,1}(r_{\pi_2}, r_1, r_3, r_1^o, r_5^o)$  provides a set of configurations such that  $\mathcal{A}_1(q_1), \mathcal{A}_3(q_3), \mathcal{O}_1(x_1^o), \mathcal{O}_5(x_5^o) \subset \pi_2$  and the pairwise intersections of the respective sets are empty. The problem of finding an algorithm  $\mathcal{P}_s$  is a special case of the sphere packing problem [? ]. 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, as in the previous sections.

**Definition 5.13. (Transition)** Consider that  $\mathcal{A}_i(q_i(t_0)) \subset \pi_k$ , for some  $i \in \mathcal{N}, k \in \mathcal{K}_R, t_0 \in \mathbb{R}_{\geq 0}$ , and

$$\mathcal{A}_i(q_i(t_0)) \cap \left( \mathcal{A}_l(q_l(t_0)) \cup \mathcal{O}_j(p_j^o(t_0)) \cup \mathcal{AO}_{V,\ell}(p_\ell^o(t_0)) \right) = \emptyset, \quad (5.23)$$

for all  $l \in \mathcal{N} \setminus \{i\}$ ,  $j \in \mathcal{M}$ , and any  $V \subseteq \mathcal{N} \setminus \{i\}$ ,  $\ell \in \mathcal{M}$  such that  $\mathcal{AG}_{h,\ell}(q_h(t_0), x_\ell^o(t_0)) = \text{True}, \forall h \in V$ . Then, there exists a transition for agent  $i$  from region  $\pi_k$  to  $\pi_{k'}$ ,  $k' \in \mathcal{K}_R$ , denoted as  $\pi_k \rightarrow_i \pi_{k'}$ , if there exists a finite  $t_f \geq t_0$  such that  $\mathcal{A}_i(q_i(t_f)) \subset \pi_{k'}, \mathcal{A}_i(q_i(t)) \subset \mathcal{W}, \mathcal{A}_i(q_i(t)) \cap \pi_m = \emptyset, \forall t \in [t_0, t_f], m \in \mathcal{K}_R \setminus \{k, k'\}$  and (5.23) holds for all  $t \in [t_0, t_f]$ .

**Definition 5.14. (Grasping)** Consider that  $\mathcal{A}_i(q_i(t_0)) \subset \pi_k, \mathcal{O}_j(x_j^o(t_0)) \subset \pi_k, k \in \mathcal{K}_R$  for some  $i \in \mathcal{N}, j \in \mathcal{M}, t_0 \in \mathbb{R}_{\geq 0}$ , and (5.23) holds. Then, agent  $i$  grasps object  $j$ , denoted as  $i \xrightarrow{g} j$ , if there exists a finite  $t_f \geq t_0$  such that  $\mathcal{AG}_{i,j}(q_i(t_f), p_j^o(t_f)) = \text{True}, \mathcal{A}_i(q_i(t)) \subset \pi_k, \mathcal{O}_j(p_j^o(t)) \subset \pi_k, k \in \mathcal{K}_R, \forall t \in [t_0, t_f]$ , and (5.23) holds for all objects except for  $j$  and all  $t \in [t_0, t_f]$ .

**Definition 5.15. (Releasing)** Consider that  $\mathcal{A}_i(q_i(t_0)) \subset \pi_k, \mathcal{O}_j(p_j^o(t_0)) \subset \pi_k, k \in \mathcal{K}_R$  for some  $i \in \mathcal{N}, j \in \mathcal{M}, t_0 \in \mathbb{R}_{\geq 0}$ , with  $\mathcal{AG}_{i,j}(q_i(t_0), x_j^o(t_0)) = \text{True}$ , and (5.23) holding for all objects except for  $j$ . Then, agent  $i$  releases

object  $j$ , denoted as  $i \xrightarrow{r} j$ , if there exists a finite  $t_f \geq t_0$  such that  $\mathcal{AG}_{i,0}(q_i(t_f), x^o(t_f)) = \text{True}$ ,  $\mathcal{A}_i(q_i(t)) \subset \pi_k$ ,  $\mathcal{O}_j(p_j^o(t)) \subset \pi_k$ ,  $k \in \mathcal{K}_R$ ,  $t \in [t_0, t_f]$ , and (5.23) holding for all objects except for  $j$  and all  $t \in [t_0, t_f]$ .

**Definition 5.16. (Transportation)** Consider a nonempty subset of agents  $\mathcal{V} \subseteq \mathcal{N}$  and an object  $j \in \mathcal{M}$  such that  $\mathcal{AG}_{i,j}(q_i(t_0), x_j^o(t_0)) = \text{True}$ ,  $\forall i \in \mathcal{V}$  and  $\mathcal{AO}_{\mathcal{V},j}(p_j^o(t_0)) \subset \pi_k$  for some  $k \in \mathcal{K}_R$ ,  $t_0 \geq 0$ , with

$$\mathcal{AO}_{\mathcal{V},j}(p_j^o(t_0)) \cap \left( \mathcal{A}_l(q_l(t_0)) \cup \mathcal{O}_\ell(p_\ell^o(t_0)) \cap \mathcal{AO}_{\mathcal{V}',j'}(p_{j'}^o(t_0)) \right) = \emptyset, \quad (5.24)$$

for all  $l \in \mathcal{N} \setminus \mathcal{V}$ ,  $\ell \in \mathcal{M} \setminus \{j\}$ , and any  $\mathcal{V}' \subseteq \mathcal{N} \setminus \mathcal{V}$ ,  $j' \in \mathcal{M} \setminus \{j\}$  such that  $\mathcal{AG}_{h,j'}(q_h(t_0), x_h^o(t_0)) = \text{True}$ ,  $\forall h \in \mathcal{V}'$ . Then, the team of agents  $\mathcal{V}$  transports the object  $j$  from region  $\pi_k$  to region  $\pi_{k'}$ ,  $k' \in \mathcal{K}_R$ , denoted as  $\pi_k \xrightarrow{T} \pi_{k'}$ , if there exists a finite  $t_f \geq t_0$  such that  $\mathcal{AO}_{\mathcal{V},j}(p_j^o(t_f)) \subset \pi_{k'}$ ,  $\mathcal{AO}_{\mathcal{V},j}(p_j^o(t)) \subset \mathcal{W}$ ,  $\mathcal{AG}_{i,j}(q_i(t), x_j^o(t)) = \text{True}$ ,  $\forall i \in \mathcal{V}$ ,  $\mathcal{AO}_{\mathcal{V},j}(p_j^o(t)) \cap \pi_m = \emptyset$ ,  $\forall m \in \mathcal{K}_R \setminus \{k, k'\}$ ,  $t \in [t_0, t_f]$ , and (5.24) holding for all  $t \in [t_0, t_f]$ .

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  $\tau_i$ ,  $u_i$ , 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 are assumed to be avoided.

Our goal is to control the multi-agent system such that the agents and the objects obey a given specification over their atomic propositions  $\Psi_i, \Psi_j^o, \forall i \in \mathcal{N}, j \in \mathcal{M}$ . Given the trajectories  $q_i(t), x_j^o(t), t \in \mathbb{R}_{\geq 0}$ , of agent  $i$  and object  $j$ , respectively, their corresponding *behaviors* are given by the infinite sequences

$$\begin{aligned} \mathfrak{b}_i &:= (q_i(t), \check{\psi}_i) := (q_i(t_{i,1}), \check{\psi}_{i,1})(q_i(t_{i,2}), \check{\psi}_{i,2}) \dots, \\ \mathfrak{b}_j^o &:= (x_j^o(t), \check{\psi}_j^o) := (x_j^o(t_{j,1}), \check{\psi}_{j,1}^o)(x_j^o(t_{j,2}), \check{\psi}_{j,2}^o) \dots, \end{aligned}$$

with  $t_{i,\ell+1} > t_{i,\ell} \geq 0, t_{j,\ell+1}^o > t_{j,\ell}^o \geq 0, \forall \ell \in \mathbb{N}$ , representing specific time stamps. The sequences  $\check{\psi}_i, \check{\psi}_j^o$  are the services provided to the agent and the object, respectively, over their trajectories, i.e.,  $\check{\psi}_{i,\ell} \in 2^{\Psi_i}, \check{\psi}_{j,l}^o \in 2^{\Psi_j^o}$  with  $\mathcal{A}_i(q_i(t_{i,\ell})) \subset \pi_{k_{i,\ell}}$ ,  $\check{\psi}_{i,\ell} \in \mathcal{L}_i(\pi_{k_{i,\ell}})$  and  $\mathcal{O}_j(x_j^o(t_{j,l}^o)) \subset \pi_{k_{j,l}^o}$ ,  $\check{\psi}_{j,l}^o \in \mathcal{L}_j^o(\pi_{k_{j,l}^o})$ ,  $k_{i,\ell}, k_{j,l}^o \in \mathcal{K}_R, \forall l \in \mathbb{N}, i \in \mathcal{N}, j \in \mathcal{M}$ , where  $\mathcal{L}_i$  and  $\mathcal{L}_j^o$  are the previously defined labeling functions. The following Lemma then follows:

**Lemma 5.2.** *The behaviors  $\mathfrak{b}_i, \mathfrak{b}_j^o$  satisfy formulas  $\Phi_i, \Phi_{\mathcal{O}_j}$  if  $\check{\psi}_i \models \Phi_i$  and  $\check{\psi}_j^o \models \Phi_{\mathcal{O}_j}^o$ , respectively.*

The control objectives are given as LTL formulas  $\Phi_i, \Phi_j^o$  over  $\Psi_i, \Psi_j^o$ , respectively,  $\forall i \in \mathcal{N}, j \in \mathcal{M}$ . The LTL formulas  $\Phi_i, \Phi_j^o$  are satisfied if there exist behaviors  $b_i, b_j^o$  of agent  $i$  and object  $j$  that satisfy  $\Phi_i, \Phi_j^o$ . We are now ready to give a formal problem statement consider in this section:

**Problem 5.4.** Consider  $N$  robotic agents and  $M$  objects in  $\mathcal{W}$  subject to the dynamics (5.20) and (5.21), respectively, not colliding at  $t = 0$ , and

$$\dot{q}_i(0) = 0, v_j^o = 0, \mathcal{A}_i(q_i(0)) \subset \pi_{\text{init}(i)}, \mathcal{O}_j(x_j^o(0)) \subset \pi_{\text{init}_o(j)}, \forall i \in \mathcal{N}, j \in \mathcal{M},$$

Given the disjoint sets  $\Psi_i, \Psi_j^o$ ,  $N$  LTL formulas  $\Phi_i$  over  $\Psi_i$  and  $M$  LTL formulas  $\Phi_j^o$  over  $\Psi_j^o$ , develop a control strategy that achieves behaviors  $b_i, b_j^o$  which yield the satisfaction of  $\Phi_i, \Phi_j^o, \forall i \in \mathcal{N}, j \in \mathcal{M}$ .

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  $\mathcal{P}_{s,0}(r_{\pi_k}, [r_i]_{i \in \{i \in \mathcal{N}: \text{init}(i)=k\}}, [r_j^o]_{j \in \{j \in \mathcal{M}: \text{init}_o(j)=k\}}) = \text{True}, \forall k \in \mathcal{K}_R$ .

## 5.5.2 Problem Solution

### Continuous Control Design

The first ingredient of our solution is the development of feedback control laws that establish agent transitions and object transportsations as defined in Def. 5.13 and 5.16, respectively. Although the control protocols of Sections 2.2.7, 2.2.8, 4.2.5 can be applied, we focus on an alternative design that follows the concept of multi-robot navigation functions (see Appendix C). Moreover, we do not focus on the grasping/releasing actions of Def. 5.14, 5.15 and we refer to some existing methodologies that can derive the corresponding control laws (e.g., [?], [?]).

Assume that the conditions of Problem 5.4 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. 5.13 and 5.16, respectively. Let  $\mathcal{Z}, \mathcal{V}, \mathcal{G}, \mathcal{R} \subseteq \mathcal{N}$  denote disjoint sets of agents corresponding to transition, transportation, grasping and releasing actions, respectively, with  $|\mathcal{Z}| + |\mathcal{V}| + |\mathcal{G}| + |\mathcal{R}| \leq |\mathcal{N}|$  and  $\mathcal{A}_z(q_z(t_0)) \subset \pi_{k_z}, \mathcal{A}_v(q_v(t_0)) \subset \pi_{k_v}, \mathcal{A}_g(q_g(t_0)) \subset \pi_{k_g}, \mathcal{A}_\rho(q_\rho(t_0)) \subset \pi_{k_\rho}$ , where  $k_z, k_v, k_g, k_\rho \in \mathcal{K}_R, \forall z \in \mathcal{Z}, v \in \mathcal{V}, g \in \mathcal{G}, \rho \in \mathcal{R}$ . Note that there might be idle agents in some regions, not performing any actions, i.e., the set  $\mathcal{N} \setminus (\mathcal{Z} \cup \mathcal{V} \cup \mathcal{G} \cup \mathcal{R})$  might not be empty.

More specifically, regarding the transportation actions, we consider that the set  $\mathcal{V}$  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.  $\mathcal{V} = \mathcal{V}_1 \cup \mathcal{V}_2 \cup \dots \cup \mathcal{V}_{\bar{T}}$ , and  $\mathcal{A}_\nu(q_\nu(t_0)) \subset \pi_{k_{\mathcal{V}_m}}, \forall \nu \in \mathcal{V}_m, m \in \{1, \dots, \bar{T}\}$ , where  $k_{\mathcal{V}_m} \in \mathcal{K}_{\mathcal{R}}, \forall m \in \{1, \dots, \bar{T}\}$ . Let also  $\mathcal{S} := \{s_{\mathcal{V}_1}, s_{\mathcal{V}_2}, \dots, s_{\mathcal{V}_{\bar{T}}}\}$ ,  $\mathcal{X} := \{[x_g]_{g \in \mathcal{G}}\}$ ,  $\mathcal{Y} := \{[y_\rho]_{\rho \in \mathcal{R}}\} \subseteq \mathcal{M}$  be disjoint sets of objects to be transported, grasped, and released, respectively. More specifically, each team  $\mathcal{V}_m$  in the set  $\mathcal{V}$  will transport cooperatively object  $s_{\mathcal{V}_m}, m \in \{1, \dots, \bar{T}\}$ , 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  $t_0$ :

- $\mathcal{AG}_{\rho, y_\rho}(q_\rho(t_0), x_{y_\rho}^O(t_0)) = \text{True}, \forall \rho \in \mathcal{R}$ ,
- $\mathcal{AG}_{z, 0}(q_z(t_0), x^O(t_0)) = \text{True}, \forall z \in \mathcal{Z}$ ,
- $\mathcal{AG}_{g, 0}(q_g(t_0), x^O(t_0)) = \text{True}, \forall g \in \mathcal{G}$ ,
- $\mathcal{AG}_{\nu, s_{\mathcal{V}_m}}(q_\nu(t_0), x_{s_{\mathcal{V}_m}}^O(t_0)) = \text{True}, \forall \nu \in \mathcal{V}_m, m \in \{1, \dots, \bar{T}\}$ ,
- $\mathcal{AO}_{\mathcal{V}_m, s_{\mathcal{V}_m}}(p_{s_{\mathcal{V}_m}}^O(t_0)) \subset \pi_{k_{\mathcal{V}_m}}, \forall m \in \{1, \dots, \bar{T}\}$ ,
- $\mathcal{O}_{x_g}(p_{x_g}^O(t_0)) \subset \pi_{k_g}, \forall g \in \mathcal{G}$ ,
- $\mathcal{O}_{y_\rho}(p_{y_\rho}^O(t_0)) \subset \pi_{k_\rho}, \forall \rho \in \mathcal{R}$ ,

which mean, intuitively, that the objects  $s_{\mathcal{V}_m}, x_g, y_\rho$  to be transported, grasped, released, are in the regions  $\pi_{k_{\mathcal{V}_m}}, \pi_{k_g}, \pi_{k_\rho}$ , 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. 5.13 [5.16] hold.

In the following, we design  $\tau_z$  and  $u_\nu$  such that  $\pi_{k_z} \rightarrow_z \pi_{k'_z}$  and  $\pi_{k_{\mathcal{V}_m}} \xrightarrow{\mathcal{T}}_{\mathcal{V}_m, s_{\mathcal{V}_m}} \pi_{k'_{\mathcal{V}_m}}$ , with  $k'_z, k'_{\mathcal{V}_m} \in \mathcal{K}_{\mathcal{R}}, \forall z \in \mathcal{Z}, m \in \{1, \dots, \bar{T}\}$ , assuming that (i) there exist appropriate  $u_g$  and  $u_\rho$  that guarantee  $g \xrightarrow{g} x_g$  and  $\rho \xrightarrow{r} y_\rho$  in  $\pi_{k_g}, \pi_{k_\rho}$ , 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.,

$$\begin{aligned} \mathcal{P}_{s, 0} \left( r_{\pi_k}, [r_z]_{z \in \mathcal{Q}_{\mathcal{Z}, k}}, [r_g]_{g \in \mathcal{Q}_{\mathcal{G}, k}}, [r_\rho]_{\rho \in \mathcal{Q}_{\mathcal{R}, k}}, [r_{\mathcal{V}_m, s_{\mathcal{V}_m}}]_{m \in \mathcal{Q}_{\mathcal{V}, k}}, \right. \\ \left. [r_{x_g}^O]_{g \in \mathcal{Q}_{\mathcal{G}, k}}, [r_{y_\rho}^O]_{\rho \in \mathcal{Q}_{\mathcal{R}, k}} \right) = \text{True} \end{aligned} \quad (5.25)$$

$\forall k \in \mathcal{K}_{\mathcal{R}}$ , where we define the sets:  $\mathcal{Q}_{\mathcal{Z}, k} := \{z \in \mathcal{Z} : k'_z = k\}$ ,  $\mathcal{Q}_{\mathcal{G}, k} := \{g \in \mathcal{G} : k_g = k\}$ ,  $\mathcal{Q}_{\mathcal{R}, k} := \{\rho \in \mathcal{R} : k_\rho = k\}$ ,  $\mathcal{Q}_{\mathcal{V}, k} := \{m \in \{1, \dots, \bar{T}\} : k'_{\mathcal{V}_m} = k\}$ , that correspond to the indices of the agents and objects that are in region  $k \in \mathcal{K}_{\mathcal{R}}$ .

**Example 5.1.** 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,  $\mathcal{K}_{\mathcal{R}} = \{1, \dots, 4\}$ . Let  $t_0 = 0$  and, according to Problem 5.4, 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  $\pi_1, \pi_2$  and  $\pi_3$ , respectively. We assume that agents 1, 5 grasp object 1, and agents 3, 4 grasp object 3, i.e.,  $\mathcal{AG}_{1,1}(q_1(0), x_1^o(0)) = \mathcal{AG}_{5,1}(q_5(0), x_1^o(0)) = \mathcal{AG}_{3,3}(q_3(0), x_3^o(0)) = \mathcal{AG}_{4,3}(q_4(0), x_4^o(0)) = \mathcal{AG}_{2,0}(q_2(0), x^o(0)) = \mathcal{AG}_{6,0}(q_6(0), x^o(0)) = \text{True}$ . Agents 1 and 5 aim to cooperatively transport object 1 to  $\pi_4$ , agent 2 aims to grasp object 2, agents 3 and 4 aim to cooperatively transport object 3 to  $\pi_1$  and agent 6 aims to perform a transition to region  $\pi_2$ . Therefore,  $\mathcal{Z} = \{6\}, \bar{T} = 2, \mathcal{V}_1 = \{1, 5\}, \mathcal{V}_2 = \{3, 4\}, \mathcal{V} = \mathcal{V}_1 \cup \mathcal{V}_2 = \{1, 5, 4, 3\}, \mathcal{G} = \{2\}, \mathcal{R} = \emptyset, s_{\mathcal{V}_1} = 1, s_{\mathcal{V}_2} = 2, \mathcal{S} = \{s_{\mathcal{V}_1}, s_{\mathcal{V}_2}\} = \{1, 2\}, \mathcal{X} = \{x_2\} = \{2\}, \mathcal{Y} = \emptyset$ . Moreover, the region indices  $k_z, k_v, k_g, k_r, k_{\mathcal{V}_m}, k'_z, k'_{\mathcal{V}_m}, z \in \mathcal{Z} = \{6\}, v \in \mathcal{V} = \{1, 5, 4, 3\}, g \in \mathcal{G} = \{2\}, r \in \mathcal{R} = \emptyset, m \in \{1, 2\}$ , take the form  $k_6 = 4, k_1 = k_5 = 1, k_2 = 2, k_3 = k_4 = 3, k_{\mathcal{V}_1} = 1, k_{\mathcal{V}_3} = 3, k'_6 = 2, k'_{\mathcal{V}_1} = 4, k'_{\mathcal{V}_2} = 1$ . Finally, the actions that need to be performed by the agents are  $\pi_1 \xrightarrow{T} \pi_4, 2 \xrightarrow{g} 2, \pi_3 \xrightarrow{T} \pi_1$  and  $\pi_4 \rightarrow \pi_2$ .

Next, for each region  $\pi_k$ , we compute from  $\mathcal{P}_s$  a set of configurations for the agents and objects in this region. More specifically,

$$([q_z^*]_{z \in \mathcal{Q}_{\mathcal{Z},k}}, [q_g^*]_{g \in \mathcal{Q}_{\mathcal{G},k}}, [q_\rho^*]_{\rho \in \mathcal{Q}_{\mathcal{R},k}}, [x_{s_{\mathcal{V}_m}}^{O*}]_{m \in \mathcal{Q}_{\mathcal{V},k}}, [x_{x_g}^{O*}]_{g \in \mathcal{Q}_{\mathcal{G},k}}, [x_{y_\rho}^{O*}]_{\rho \in \mathcal{Q}_{\mathcal{R},k}}) = \\ \mathcal{P}_{s,1} \left( r_{\pi_k}, [r_z]_{z \in \mathcal{Q}_{\mathcal{Z},k}}, [r_g]_{g \in \mathcal{Q}_{\mathcal{G},k}}, [r_\rho]_{\rho \in \mathcal{Q}_{\mathcal{R},k}}, [r_{\mathcal{V}_m, s_{\mathcal{V}_m}}]_{m \in \mathcal{Q}_{\mathcal{V},k}}, [r_{x_g}^o]_{g \in \mathcal{Q}_{\mathcal{G},k}}, [r_{y_\rho}^o]_{\rho \in \mathcal{Q}_{\mathcal{R},k}} \right),$$

where we have used the notation of (5.25). Hence, we now have the goal configurations for the agents  $\mathcal{Z}$  performing the transitions as well as agents  $\mathcal{V}$  performing the cooperative transports.

Following Section C.1, we define the error functions  $\gamma_z : \mathbb{R}^{n_z} \rightarrow \mathbb{R}_{\geq 0}$  with  $\gamma_z(q_z) := \|q_z - q_z^*\|^2, \forall z \in \mathcal{Z}$ , and  $\gamma_{\mathcal{V}_m} : \mathbb{M} \rightarrow \mathbb{R}_{\geq 0}$  as  $\gamma_{\mathcal{V}_m}(x_{s_{\mathcal{V}_m}}^o) := \|p_{s_{\mathcal{V}_m}}^o - p_{s_{\mathcal{V}_m}}^{O*}\|^2$ , where  $p_{s_{\mathcal{V}_m}}^{O*}$  is the position part of  $x_{s_{\mathcal{V}_m}}^{O*}$ .

Regarding the grasping agents  $g \in \mathcal{G}$ , these are assumed to operate in the sphere with the fixed center  $c_g(q_g)$  and radius  $r_g$ . Regarding the releasing agent  $\rho \in \mathcal{R}$  and the respective objects  $y_\rho, \rho$ , these are assumed to operate in the sphere with the fixed center  $c_\rho(q_\rho)$  and radius  $r_\rho$ .

Based on the above, we define the following collision functions:

$$\begin{aligned}
 \beta_{i,l}(q_i, q_l) &:= \|c_i(q_i) - c_l(q_l)\|^2 - (r_i + r_l)^2, \forall i, l \in \mathcal{N} \setminus \mathcal{V}, i \neq l, \\
 \beta_{i,O_j}(q_i) &:= \|c_i(q_i) - p_j^O\|^2 - (r_i + r_j^O)^2, \forall i \in \mathcal{N} \setminus \mathcal{V}, j \in \mathcal{M} \setminus (\mathcal{S} \cup \mathcal{Y}) \\
 \beta_{i,\nu_m}(q_i, x_{s_{\nu_m}}^O) &:= \|c_i(q_i) - p_{s_{\nu_m}}^O\|^2 - (r_i + r_{\nu_m, s_{\nu_m}})^2, \\
 &\quad \forall i \in \mathcal{N} \setminus \mathcal{V}, m \in \{1, \dots, \bar{T}\}, \\
 \beta_{\nu_m, \nu_\ell}(x_{s_{\nu_m}}^O, x_{s_{\nu_\ell}}^O) &:= \|p_{s_{\nu_m}}^O - p_{s_{\nu_\ell}}^O\|^2 - (r_{\nu_m, s_{\nu_m}} + r_{\nu_\ell, s_{\nu_\ell}})^2, \\
 &\quad \forall m, \ell \in \{1, \dots, \bar{T}\}, m \neq \ell, \\
 \beta_{\nu_m, O_j}(x_{s_{\nu_m}}^O) &:= \|p_{s_{\nu_m}}^O - p_j^O\|^2 - (r_{\nu_m, s_{\nu_m}} + r_j^O)^2, \\
 &\quad \forall m \in \{1, \dots, \bar{T}\}, j \in \mathcal{M} \setminus (\mathcal{S} \cup \mathcal{Y}), \\
 \beta_{i, \pi_k}(q_i) &:= \|c_i(q_i) - p_{\pi_k}\|^2 - (r_i + r_{\pi_k})^2, \forall i \in \mathcal{Z}, k \in \mathcal{K}_{\mathcal{R}} \setminus \{k_z, k'_z\}, \\
 \beta_{\nu_m, \pi_k}(x_{s_{\nu_m}}^O) &:= \|p_{s_{\nu_m}}^O - p_{\pi_k}\|^2 - (r_{\nu_m, s_{\nu_m}} + r_{\pi_k})^2, \\
 &\quad \forall m \in \{1, \dots, \bar{T}\}, k \in \mathcal{K}_{\mathcal{R}} \setminus \{k_{\nu_m}, k'_{\nu_m}\}, \\
 \beta_{i, w}(q_i) &:= (r_0 - r_i)^2 - \|c_i(q_i)\|^2, \forall i \in \mathcal{N} \setminus \mathcal{V} \\
 \beta_{\nu_m, w}(x_{s_{\nu_m}}^O) &:= (r_0 - r_{\nu_m, s_{\nu_m}})^2 - \|p_{s_{\nu_m}}^O\|^2, \forall m \in \{1, \dots, \bar{T}\},
 \end{aligned}$$

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 C.1, we can form the total obstacle function  $G : \mathbb{R}^{n_z} \times \mathbb{M}^{|\mathcal{S}|} \rightarrow \mathbb{R}_{\geq 0}$ ,  $n_z := \sum_{z \in \mathcal{Z}} n_z$ , and thus, define the navigation function [? ?]  $\varphi : \mathcal{F} \rightarrow [0, 1]$  as

$$\varphi(q_z, x_s^O) := \frac{\gamma(q_z, x_s^O)}{\left(\gamma(q_z, x_s^O)^\kappa + G(q_z, x_s^O)\right)^{\frac{1}{\kappa}}},$$

where  $q_z := [q_z]_{z \in \mathcal{Z}}$ ,  $x_s^O := [x_{s_{\nu_m}}^O]_{m \in \{1, \dots, \bar{T}\}} \in \mathbb{M}^{|\mathcal{S}|}$ ,  $\gamma(q_z, x_s^O) := \sum_{z \in \mathcal{Z}} \gamma_z(q_z) + \sum_{m \in \{1, \dots, \bar{T}\}} \gamma_{\nu_m}(x_{s_{\nu_m}}^O)$ ,  $\mathcal{F}$  is a subset of  $\mathbb{R}^{n_z} \times \mathbb{M}^M$  where the collision functions are positive, and  $\kappa > 0$  is a positive gain used to derive the proof correctness of  $\varphi$  [? ?]. Note that, a sufficient condition for avoidance of the undesired regions and avoidance of collisions and singularities is  $\varphi(q_z, x_s^O) < 1$ .

Next, we design the feedback control protocols  $\tau_z : \mathcal{F} \times \mathbb{R}^{n_z} \rightarrow \mathbb{R}^6$ ,  $u_\ell :$

$\mathcal{F} \times S_\ell \times \mathbb{R}^6$ ,  $\forall z \in \mathcal{Z}, \ell \in \mathcal{V}_m, m \in \{1, \dots, \bar{T}\}$  as follows:

$$\tau_z = \tau_z(q_z, x_s^O, \dot{q}_z) := g_{q_z} - \nabla_{q_z} \varphi(q_z, x_s^O) - K_z \dot{q}_z, \quad (5.26a)$$

$$u_\ell = u_\ell(q_z, x_s^O, q_\ell, v_{s_{\mathcal{V}_m}}^O) := \left( J_{\ell, s_{\mathcal{V}_m}}^O \right)^{-\top} \left\{ \left( g_O - \left( J_{s_{\mathcal{V}_m}}^O \right)^\top \nabla_{x_{s_{\mathcal{V}_m}}^O} \varphi(q_z, x_s^O) - v_{s_{\mathcal{V}_m}}^O \right) \right\} + g_\ell, \quad (5.26b)$$

where  $K_z = \text{diag}\{k_z\} \in \mathbb{R}^{n_z \times n_z}$ , with  $k_z > 0, \forall z \in \mathcal{Z}$ , is a constant positive definite gain matrix. To characterize the solutions of the closed-loop system, we consider the function

$$V := \varphi(q_z, x_s^O) + \frac{1}{2} \sum_{z \in \mathcal{Z}} \dot{q}_z^\top B_z(q_z) \dot{q}_z + \frac{1}{2} \sum_{m \in \{1, \dots, \bar{T}\}} \left( v_{s_{\mathcal{V}_m}}^O \right)^\top \tilde{M}_{v_m, s_{\mathcal{V}_m}} v_{s_{\mathcal{V}_m}}^O.$$

Since no collisions occur and the robots and objects have zero velocity at  $t_0$ , we conclude that  $V_0 := V(t_0) = \varphi(q_z(t_0), x_s^O(t_0)) =: \varphi_0 < 1$ . By differentiating  $V$  and substituting (5.20), (5.22), we obtain

$$\begin{aligned} \dot{V} &= \sum_{z \in \mathcal{Z}} \left\{ \nabla_{q_z} \varphi(q_z, x_s^O)^\top \dot{q}_z + \dot{q}_z^\top (\tau_z - C_{q_z} \dot{q}_z - g_{q_z}) + \frac{1}{2} \dot{q}_z^\top \dot{M}_z \dot{q}_z \right\} \\ &\quad + \sum_{m \in \{1, \dots, \bar{T}\}} \left\{ \nabla_{x_{s_{\mathcal{V}_m}}^O} \varphi(q_z, x_s^O)^\top \dot{x}_{s_{\mathcal{V}_m}}^O + \left( v_{s_{\mathcal{V}_m}}^O \right)^\top \left( \sum_{\ell \in \mathcal{V}_m} [J_{\ell, s_{\mathcal{V}_m}}^O]^\top u_\ell - g_O \right. \right. \\ &\quad \left. \left. - \sum_{\ell \in \mathcal{V}_m} [J_{\ell, s_{\mathcal{V}_m}}^O]^\top g_\ell - \tilde{C}_{v_m, s_{\mathcal{V}_m}} \right) + \frac{1}{2} \left( v_{s_{\mathcal{V}_m}}^O \right)^\top \dot{\tilde{M}}_{v_m, s_{\mathcal{V}_m}} v_{s_{\mathcal{V}_m}}^O \right\}, \end{aligned}$$

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. By employing Lemma 5.1 as well as (5.21a),  $\dot{V}$  becomes:

$$\begin{aligned} \dot{V} &= \sum_{z \in \mathcal{Z}} \dot{q}_z^\top \left( \nabla_{q_z} \varphi(q_z, x_s^O) + \tau_z - g_{q_z} \right) + \\ &\quad \sum_{m \in \{1, \dots, \bar{T}\}} \left( v_{s_{\mathcal{V}_m}}^O \right)^\top \left( \sum_{\ell \in \mathcal{V}_m} [J_{\ell, s_{\mathcal{V}_m}}^O]^\top (u_\ell - g_\ell) - g_O + [J_{s_{\mathcal{V}_m}}^O]^\top \nabla_{x_{s_{\mathcal{V}_m}}^O} \varphi(q_z, x_s^O) \right), \end{aligned}$$

and after substituting (5.26):

$$\dot{V} = - \sum_{z \in \mathcal{Z}} \dot{q}_z K_z \dot{q}_z - \sum_{m \in \{1, \dots, \bar{T}\}} \|v_{s_{\mathcal{V}_m}}^O\|^2,$$

which is strictly negative unless  $\dot{q}_z = 0, v_{s_{\mathcal{V}_m}}^O = 0, \forall z \in \mathcal{Z}, m \in \{1, \dots, \bar{T}\}$ . Since  $J_{\ell, s_{\mathcal{V}_m}}^O$  is always non-singular, and  $J_\ell(q_\ell(t))$  has full-rank by assumption for the maximal solution,  $\forall \ell \in \mathcal{V}_m, m \in \{1, \dots, \bar{T}\}$ , the latter implies also

that  $\dot{q}_\ell = 0, \forall \ell \in \mathcal{V}_m, m \in \tilde{\mathcal{V}}$ . Hence,  $V(t) \leq V_0 < 1, \forall t \in [t_0, t_{\max}]$ , which suggests that  $\varphi(q_z(t), x_s^o(t)) \leq \varphi_0 < 1 \forall t \geq t_0$ . Moreover, according to La Salle's Invariance Principle [? ], the system will converge to the largest invariant set contained in the set where  $\dot{q}_z = 0, v_{s_{\mathcal{V}_m}}^o = 0, \forall z \in \mathcal{Z}, m \in \{1, \dots, \bar{T}\}$ . We can also conclude that  $\lim_{t \rightarrow \infty} \ddot{q}_z(t) = 0, \dot{v}_{s_{\mathcal{V}_m}}^o = 0$ , which, by employing (5.26), (5.20), (5.22), and the assumption of non-singular  $J_{s_{\mathcal{V}_m}}^o, \forall t \in \mathbb{R}_{\geq 0}$ , implies that  $\nabla_{q_z} \varphi(q_z, x_s^o) = 0, \nabla_{x_{s_{\mathcal{V}_m}}^o} \varphi(q_z, x_s^o) = 0, \forall z \in \mathcal{Z}, m \in \{1, \dots, \bar{T}\}$ . Since  $\varphi$  is a navigation function [? ], by setting  $\kappa$  large enough, this condition is true only at the destination configurations (i.e., where  $\gamma(q_z, x_s^o) = 0$ ) and a set of isolated saddle points, whose region of attraction is a set of measure zero [? ? ]. Thus, the system converges to the destination configuration from almost everywhere, i.e.,  $\|q_z(t) - q_z^*\| \rightarrow 0$  and  $\|p_{s_{\mathcal{V}_m}}^o(t) - p_{s_{\mathcal{V}_m}}^{o^*}\| \rightarrow 0$ . 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  $\mathcal{AO}_{\mathcal{V}_m, s_{\mathcal{V}_m}}(p_{s_{\mathcal{V}_m}}^o(t_{f_m})) \subset \pi_{k'_{\mathcal{V}_m}}$ , with inter-agent collision avoidance,  $\forall z \in \mathcal{Z}, m \in \{1, \dots, \bar{T}\}$ . Since the actions  $g \xrightarrow{g} x_g, \rho \xrightarrow{r} y_\rho$  are also performed, we denote as  $t_{f_g}, t_{f_\rho}$  the times that these actions have been completed,  $g \in \mathcal{G}, \rho \in \mathcal{R}$ . Hence, by setting  $t_f := \max\{\max_{z \in \mathcal{Z}} t_{f_z}, \max_{m \in \{1, \dots, \bar{T}\}} t_{f_m}, \max_{g \in \mathcal{G}} t_{f_g}, \max_{\rho \in \mathcal{R}} t_{f_\rho}\}$ , all the actions of all agents will be completed at  $t_f$ .

It should be noted that [? ] does not take into account static obstacles. Since, however, the results are an extension of [? ], intuition suggests that the results are valid for sufficiently distant obstacles (in our case, the regions of interest).

### High-Level Plan Generation

The second part of the solution is the derivation of a high-level plan that satisfies the given LTL formulas  $\Phi_i$  and  $\Phi_j^o$  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 transports  $\pi_k \rightarrow_i \pi_{k'}$  and  $\pi_k \xrightarrow{T} \pi_{k'}$ , 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 5.17.** 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;  $\bar{\Pi} := \Pi_1 \times \dots \times \Pi_N$  and  $\bar{\Pi}^o := \Pi_1^o \times \dots \times \Pi_M^o$  are the set of states-regions that the agents and

the objects can be at, with  $\Pi_i = \Pi_j^o = \Pi, \forall i \in \mathcal{N}, j \in \mathcal{M}$ ;  $\mathcal{AG} := \mathcal{AG}_1 \times \dots \times \mathcal{AG}_N$  is the set of boolean grasping variables introduced in Section 5.5.1, with  $\mathcal{AG}_i := \{\mathcal{AG}_{i,0}\} \cup \{\mathcal{AG}_{i,j}\}_{j \in \mathcal{M}}$ ,  $\forall i \in \mathcal{N}$ . By defining  $\bar{\pi} := (\pi_{k_1}, \dots, \pi_{k_N}), \bar{\pi}_o := (\pi_{k_1^o}, \dots, \pi_{k_M^o}), \bar{w} = (w_1, \dots, w_N)$ , with  $\pi_{k_i}, \pi_{k_j^o} \in \Pi$  (i.e.,  $k_i, k_j^o \in \mathcal{K}_R, \forall i \in \mathcal{N}, j \in \mathcal{M}$ ) 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  $\Pi_s$ , i.e.,  $(\bar{\pi}, \bar{\pi}_o, \bar{w}) \in \Pi_s$  if

- a)  $\mathcal{P}_{s,0}(r_{\pi_k}, [r_i]_{i \in \{i \in \mathcal{N}: k_i=k\}}, [r_j^o]_{j \in \{j \in \mathcal{M}: k_j^o=k\}}) = \text{True}$ , i.e., the respective agents and objects fit in the region,  $\forall k \in \mathcal{K}_R$ ,
  - b)  $k_i = k_j^o$  for all  $i \in \mathcal{N}, j \in \mathcal{M}$  such that  $w_i = \mathcal{AG}_{i,j} = \text{True}$ , 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 5.4.
3.  $\rightarrow_s \subset \Pi_s \times \Pi_s$  is a transition relation defined as follows: given the states  $\pi_s, \tilde{\pi}_s \in \Pi_s$ , with

$$\begin{aligned} \pi_s &:= (\bar{\pi}, \bar{\pi}_o, \bar{w}) := (\pi_{k_1}, \dots, \pi_{k_N}, \pi_{k_1^o}, \dots, \pi_{k_M^o}, w_1, \dots, w_N), \\ \tilde{\pi}_s &:= (\tilde{\bar{\pi}}, \tilde{\bar{\pi}}_o, \tilde{\bar{w}}) := (\pi_{\tilde{k}_1}, \dots, \pi_{\tilde{k}_N}, \pi_{\tilde{k}_1^o}, \dots, \pi_{\tilde{k}_M^o}, \tilde{w}_1, \dots, \tilde{w}_N), \end{aligned} \quad (5.27)$$

a transition  $\pi_s \rightarrow_s \tilde{\pi}_s$  occurs if all the following hold:

- a)  $\nexists i \in \mathcal{N}, j \in \mathcal{M}$  such that  $w_i = \mathcal{AG}_{i,j} = \text{True}, \tilde{w}_i = \mathcal{AG}_{i,0} = \text{True}$ , (or  $w_i = \mathcal{AG}_{i,0} = \text{True}, \tilde{w}_i = \mathcal{AG}_{i,j} = \text{True}$ ) and  $k_i \neq \tilde{k}_i$ , i.e., there are no simultaneous grasp/release and navigation actions,
- b)  $\nexists i \in \mathcal{N}, j \in \mathcal{M}$  such that  $w_i = \mathcal{AG}_{i,j} = \text{True}, \tilde{w}_i = \mathcal{AG}_{i,0} = \text{True}$ , (or  $w_i = \mathcal{AG}_{i,0} = \text{True}, \tilde{w}_i = \mathcal{AG}_{i,j} = \text{True}$ ) and  $k_i = k_j^o \neq \tilde{k}_i = \tilde{k}_j^o$ , 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  $w_i = \mathcal{AG}_{i,j} = \text{True}$  and  $\tilde{w}_i = \mathcal{AG}_{i,j'} = \text{True}$  ( $w_i = \mathcal{AG}_{i,j'} = \text{True}$  and  $\tilde{w}_i = \mathcal{AG}_{i,j'} = \text{True}$ ), i.e., there are no simultaneous grasp and release actions,
- d)  $\nexists j \in \mathcal{M}$  such that  $k_j^o \neq \tilde{k}_j^o$  and  $w_i \neq \mathcal{AG}_{i,j}, \forall i \in \mathcal{N}$  ( or  $\tilde{w}_i \neq \mathcal{AG}_{i,j}, \forall i \in \mathcal{N}$ ), i.e., there is no transportation of a non-grasped object,
- e)  $\nexists j \in \mathcal{M}, \mathcal{V} \subseteq \mathcal{N}$  such that  $k_j^o \neq \tilde{k}_j^o$  and  $\Lambda(m_j^o, \mathbf{c}_V) = \text{False}$ , where  $w_i = \tilde{w}_i = \mathcal{AG}_{i,j} = \text{True} \Leftrightarrow i \in \mathcal{V}$ , 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  $\bar{\Psi}^O = \bigcup_{j \in \mathcal{M}} \Psi_j^O$ , are the atomic propositions of the agents and objects, respectively, as defined in Section 5.5.1.
5.  $\mathcal{L} : \Pi_s \rightarrow 2^\Psi$  is a labeling function defined as follows: Given a state  $\pi_s$  as in (5.27) and  $\check{\psi}_s := \left( \bigcup_{i \in \mathcal{N}} \check{\psi}_i \right) \cup \left( \bigcup_{j \in \mathcal{M}} \check{\psi}_j^O \right)$  with  $\check{\psi}_i \in 2^{\Psi_i}$ ,  $\check{\psi}_j^O \in 2^{\Psi_j^O}$ , then  $\check{\psi}_s \in \mathcal{L}(\pi_s)$  if  $\check{\psi}_i \in \mathcal{L}_i(\pi_{k_i})$  and  $\check{\psi}_j^O \in \mathcal{L}_j^O(\pi_{k_j^O})$ ,  $\forall i \in \mathcal{N}, j \in \mathcal{M}$ .
6.  $\Lambda$  and  $P_s$  as defined in Section 5.5.1.
7.  $\chi : (\rightarrow_s) \rightarrow \mathbb{R}_{\geq 0}$  is a function that assigns a cost to each transition  $\pi_s \rightarrow_s \tilde{\pi}_s$ . This cost might be related to the distance of the agents' regions in  $\pi_s$  to the ones in  $\tilde{\pi}_s$ , combined with the cost efficiency of the agents involved in transport tasks (according to  $c_i, i \in \mathcal{N}$ ).

Next, we form the global LTL formula  $\Phi := (\wedge_{i \in \mathcal{N}} \Phi_i) \wedge (\wedge_{j \in \mathcal{M}} \Phi_j^O)$  over the set  $\Psi$ . Then, we translate  $\Phi$  to a Büchi Automaton  $\mathcal{BA}$  and we build the product  $\mathcal{TS} := \mathcal{TS} \times \mathcal{BA}$ . Using basic graph-search theory, we can find the accepting runs of  $\mathcal{TS}$  that satisfy  $\Phi$  and minimize the total cost  $\chi$ . 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., [? ].

Following the aforementioned methodology, we obtain a high-level plan as sequences of states and atomic propositions  $\pi_{pl} := \pi_{s,1} \pi_{s,2} \dots$  and  $\check{\psi}_{pl} := \check{\psi}_{s,1} \check{\psi}_{s,2} \dots \models \Phi$ , which minimizes the cost  $\chi$ , with

$$\begin{aligned} \pi_{s,\ell} &:= (\bar{\pi}_\ell, \bar{\pi}_{O,\ell}, \bar{w}_\ell) \in \Pi_s, \forall \ell \in \mathbb{N}, \\ \check{\psi}_{s,\ell} &:= \left( \bigcup_{i \in \mathcal{N}} \check{\psi}_{i,\ell} \right) \cup \left( \bigcup_{j \in \mathcal{M}} \check{\psi}_{j,\ell}^O \right) \in 2^\Psi, \mathcal{L}(\pi_{s,\ell}), \forall \ell \in \mathbb{N}, \end{aligned}$$

where

- $\bar{\pi}_\ell := \pi_{k_{1,\ell}}, \dots, \pi_{k_{N,\ell}}$ , with  $k_{i,\ell} \in \mathcal{K}_R$ ,  $\forall i \in \mathcal{N}$ ,
- $\bar{\pi}_{O,\ell} := \pi_{k_{1,\ell}^O}, \dots, \pi_{k_{N,\ell}^O}$ , with  $k_{j,\ell}^O \in \mathcal{K}_R$ ,  $\forall j \in \mathcal{M}$ ,
- $\bar{w}_\ell := w_{1,\ell}, \dots, w_{N,\ell}$ , with  $w_{i,\ell} \in \mathcal{AG}_i$ ,  $\forall i \in \mathcal{N}$ ,
- $\check{\psi}_{i,\ell} \in 2^{\Psi_i}, \mathcal{L}_i(\pi_{k_{i,\ell}}), \forall i \in \mathcal{N}$ ,
- $\check{\psi}_{j,\ell}^O \in 2^{\Psi_j^O}, \mathcal{L}_j^O(\pi_{k_{j,\ell}^O}), \forall j \in \mathcal{M}$ .

The path  $\pi_{\text{pl}}$  is then projected to the individual sequences of the regions  $\pi_{k_{j,1}^O} \pi_{k_{j,2}^O} \dots$  for each object  $j \in \mathcal{M}$ , as well as to the individual sequences of the regions  $\pi_{k_{i,1}} \pi_{k_{i,2}} \dots$  and the boolean grasping variables  $w_{i,1} w_{i,2} \dots$  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. 5.17, we obtain that  $\check{\psi}_{i,\ell} \in \mathcal{L}_i(\pi_{k_{i,\ell}})$ ,  $\check{\psi}_{j,\ell}^O \in \mathcal{L}_j^O(\pi_{k_{j,\ell}^O})$ ,  $\forall i \in \mathcal{N}, j \in \mathcal{M}, \ell \in \mathbb{N}$ . Therefore, since  $\Phi = (\wedge_{i \in \mathcal{N}} \Phi_i) \wedge (\wedge_{j \in \mathcal{M}} \Phi_{O,j})$  is satisfied by  $\check{\psi}_{\text{pl}}$ , we conclude that  $\check{\psi}_{i,1} \check{\psi}_{i,2} \dots \models \Phi_i$  and  $\check{\psi}_{j,1}^O \check{\psi}_{j,2}^O \dots \models \Phi_j^O, \forall i \in \mathcal{N}, j \in \mathcal{M}$ .

The sequences  $\pi_{k_{i,1}} \pi_{k_{i,2}} \dots$ ,  $\check{\psi}_{i,1} \psi_{i,2} \dots$  and  $\pi_{k_{j,1}^O} \pi_{k_{j,2}^O} \dots$ ,  $\check{\psi}_{j,1}^O \check{\psi}_{j,2}^O \dots$  over  $\Pi, 2^{\Psi_i}$  and  $\Pi, 2^{\Psi_j^O}$ , respectively, produce the trajectories  $q_i(t)$  and  $x_j^O(t), \forall i \in \mathcal{N}, j \in \mathcal{M}$ . The corresponding behaviors are

$$\begin{aligned}\mathfrak{b}_i &= (q_i(t), \check{\psi}_i) = (q_i(t_{i,1}), \check{\psi}_{i,1})(q_i(t_{i,2}), \check{\psi}_{i,2}) \dots \\ \mathfrak{b}_j^O &= (x_j^O(t), \check{\psi}_j^O) = (x_j^O(t_{j,1}^O), \check{\psi}_{j,1}^O)(x_j^O(t_{j,2}^O), \check{\psi}_{j,2}^O) \dots,\end{aligned}$$

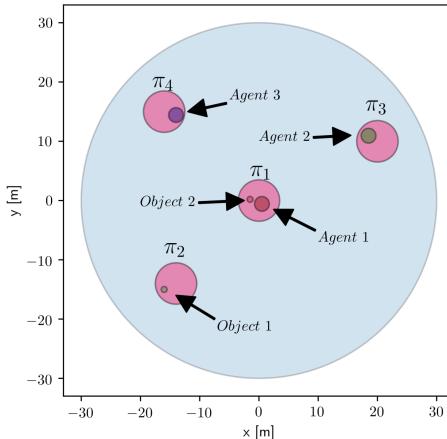
respectively, according to Section 5.5.1, with  $\mathcal{A}_i(q_i(t_{i,\ell})) \subset \pi_{k_{i,\ell}}, \check{\psi}_{i,\ell} \in \mathcal{L}_i(\pi_{k_{i,\ell}})$  and  $\mathcal{O}_j(x_j^O(t_{j,m}^O)) \in \pi_{k_{j,\ell}^O}, \check{\psi}_{j,\ell}^O \in \mathcal{L}_j^O(\pi_{k_{j,\ell}^O})$ . Thus, it is guaranteed that  $\check{\psi}_i \models \Phi_i, \check{\psi}_j^O \models \Phi_j^O$  and consequently, the behaviors  $\mathfrak{b}_i$  and  $\mathfrak{b}_j^O$  satisfy the formulas  $\Phi_i$  and  $\Phi_j^O$ , respectively,  $\forall i \in \mathcal{N}, j \in \mathcal{M}$ . The aforementioned reasoning is summarized in the next theorem:

**Theorem 5.4.** *The execution of the path  $(\pi_{\text{pl}}, \psi_{\text{pl}})$  of  $\mathcal{TS}$  guarantees behaviors  $\mathfrak{b}_i, \mathfrak{b}_j^O$  that yield the satisfaction of  $\Phi_i$  and  $\Phi_j^O$ , respectively,  $\forall i \in \mathcal{N}, j \in \mathcal{M}$ , providing, therefore, a solution to Problem 5.4.*

**Remark 5.3.** 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.

### 5.5.3 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 of radius  $r_{\pi_k} = 3.5\text{m}$ ,  $\forall k \in \mathcal{K}_{\mathcal{R}}$ , centered at  $p_{\pi_1} = [0, 0, 0]^T, p_{\pi_2} = [-14, -14, 0]^T \text{m}, p_{\pi_3} = [20, -10, 0]^T \text{m}, p_{\pi_4} = [-16, 15, 0]^T$ , respectively (see Fig. 5.25). Moreover, we consider two cuboid objects of bounding radius  $r_j^O = 0.5\text{m}$ , and mass  $m_j^O = 0.5\text{kg}, \forall j \in \{1, 2\}$ , initiated at  $x_1^O(0) = [-16, 15, 0.5, 0, 0, 0]^T \text{ (m,rad)}, x_2^O(0) = [-1.5, 0.2, 0.5, 0, 0, 0]^T \text{ (m,rad)}$ ,



**Figure 5.25:** 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.

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  $0.5 \times 0.5 \times 0.2 \text{ m}^3$  and mass  $0.5\text{kg}$ , and the two arm links have length  $1\text{m}$  and mass  $0.5\text{kg}$  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,  $\mathcal{L}_i(\pi_k) = \{\text{"}i\text{-}\pi_k\text{"}\}$ ,  $\mathcal{L}_j^o(\pi_k) = \{\text{"}O_j\text{-}\pi_k\text{"}\}$ ,  $\forall k \in \mathcal{K}_R$ ,  $i \in \mathcal{N}$ ,  $j \in \mathcal{M}$ . 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  $t = 0$ . We also assume that  $\mathcal{AG}_{i,0}(q_i(0), x^o(0)) =$

**Table 5.1:** The agent actions for the discrete path of the first simulation example

$\pi_{s,\ell}$	Actions	$\pi_{s,\ell}$	Actions
$\pi_{s,1}$	(-)	$\pi_{s,14}$	$(\pi_1 \xrightarrow{T} \{\pi_1, \pi_2\}, 2 \pi_2)$
$\pi_{s,2}$	$(-, \pi_3 \rightarrow_2 \pi_1)$	$\pi_{s,15}$	$(1 \xrightarrow{r} 2, 2 \xrightarrow{r} 2)$
$\pi_{s,3}$	$(1 \xrightarrow{g} 2, 2 \xrightarrow{g} 2)$	$\pi_{s,16}$	$(\pi_2 \rightarrow_2 \pi_4, \pi_2 \rightarrow_2 \pi_4)$
$\pi_{s,4}$	$(\pi_1 \xrightarrow{T} \{\pi_1, \pi_2\}, 2 \pi_4)$	$\pi_{s,17}$	$(1 \xrightarrow{g} 1, 2 \xrightarrow{g} 1)$
$\pi_{s,5}$	$(\pi_4 \xrightarrow{T} \{\pi_1, \pi_2\}, 2 \pi_1)$	$\pi_{s,18}$	$(\pi_4 \xrightarrow{T} \{\pi_1, \pi_2\}, 1 \pi_1)$
$\pi_{s,6}$	$(1 \xrightarrow{r} 2, 2 \xrightarrow{r} 2)$	$\pi_{s,19}$	$(\pi_1 \xrightarrow{T} \{\pi_1, \pi_2\}, 1 \pi_4)$
$\pi_{s,7}$	$(\pi_1 \rightarrow_1 \pi_2, \pi_1 \rightarrow_2 \pi_2)$	$\pi_{s,20}^*$	$(-, 2 \xrightarrow{r} 1)$
$\pi_{s,8}$	$(1 \xrightarrow{g} 1, 2 \xrightarrow{g} 1)$	$\pi_{s,21}^*$	$(-, \pi_4 \rightarrow_2 \pi_3)$
$\pi_{s,9}$	$(\pi_2 \xrightarrow{T} \{\pi_1, \pi_2\}, 1 \pi_4)$	$\pi_{s,22}^*$	$(-, \pi_3 \rightarrow_2 \pi_4)$
$\pi_{s,10}$	$(1 \xrightarrow{r} 1, 2 \xrightarrow{r} 1)$	$\pi_{s,23}^*$	$(-, 2 \xrightarrow{g} 1)$
$\pi_{s,11}$	$(-, \pi_4 \rightarrow_2 \pi_3)$	$\pi_{s,24}^*$	$(\pi_4 \xrightarrow{T} \{\pi_1, \pi_2\}, 1 \pi_1)$
$\pi_{s,12}$	$(\pi_4 \rightarrow_2 \pi_1, \pi_3 \rightarrow_2 \pi_1)$	$\pi_{s,25}^*$	$(\pi_1 \xrightarrow{T} \{\pi_1, \pi_2\}, 1 \pi_4)$
$\pi_{s,13}$	$(1 \xrightarrow{g} 2, 2 \xrightarrow{g} 2)$		

$\top, \forall i \in \{1, 2\}$ . We represent the agents' power capabilities with the scalars  $c_1 = 2, c_2 = 4$  and construct the functions  $\Lambda(m_1^o, c_V) = \top$  if and only if  $\sum_{\ell \in V} c_\ell \geq 5$ , with  $\mathcal{AG}_{\ell,1} = \top \Leftrightarrow \ell \in V$ , and  $\Lambda(m_2^o, c_V) = \top$  if and only if  $\sum_{\ell \in V} c_\ell \geq 6$ , with  $\mathcal{AG}_{\ell,2} = \top \Leftrightarrow \ell \in V$ , 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  $\chi$ , we simply choose the sum of the distances of the transition and transportation regions, i.e., given  $\pi_s, \tilde{\pi}_s$  as in (5.27) such that  $\pi_s \rightarrow_s \tilde{\pi}_s$ , we have that

$$\chi = \sum_{i \in \{1, 2\}} \{\|p_{\pi_{k_i}} - p_{\tilde{\pi}_{k_i}}\|^2\} + \sum_{j \in \{1, 2\}} \|p_{\pi_{k_j^o}} - p_{\tilde{\pi}_{k_j^o}}\|^2\}.$$

The LTL formula is taken as

$$(\square \neg "1-\pi_3") \wedge (\square \lozenge "2-\pi_3") \wedge (\square \lozenge "O_1-\pi_1") \wedge \\ \square ("O_1-\pi_1" \rightarrow \bigcirc "O_1-\pi_4") \wedge (\lozenge "O_2-\pi_4"),$$

which represents the following behavior. Agent 1 must never go to region  $\pi_3$ , which must be visited by agent 2 infinitely many times,

object 1 must be taken infinitely often to region  $\pi_1$ , always followed by a visit in region  $\pi_4$ , and object 2 must be eventually taken to region  $\pi_4$ .

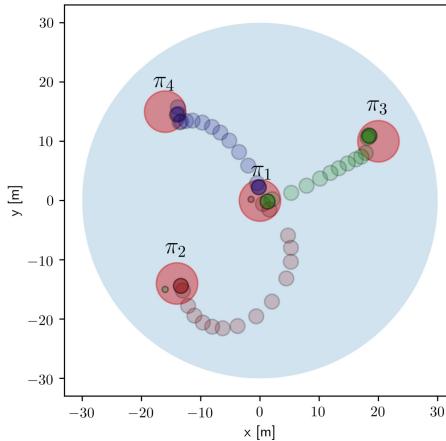
The resulting transition system  $\mathcal{TS}$  consists of 560 reachable states and 7680 transitions and it was created in 3.19 sec. The Büchi automaton  $\mathcal{BA}$  contains 7 states and 29 transitions and the product  $\widetilde{\mathcal{TS}}$  contains 3920 states and 50976 transitions. Table 5.1 shows the actions of the agents for the derived path, which is the sequence of states  $\pi_{s,1}\pi_{s,2}\dots\dots(\pi_{s,20}^*,\dots,\pi_{s,25}^*)^\omega$ , where the states with (\*) constitute the suffix that is run infinitely many times. Loosely speaking, the derived path describes the following behavior: Agent 2 goes first to  $\pi_1$  to grasp and transfer object 2 to  $\pi_4$  and back to  $\pi_1$  with agent 1. The two agents then navigate to  $\pi_2$  to take object 1 to  $\pi_4$ . In the following, after agent 2 goes to  $\pi_3$ , they both go to  $\pi_1$  to transfer object 2 to  $\pi_2$ . Then, they navigate to  $\pi_4$  to transfer object 1 to  $\pi_1$  and back. Finally, the actions that are run infinitely many times consist of agent 2 going to from  $\pi_4$  to  $\pi_3$  and back, and transferring object 1 to  $\pi_1$  and  $\pi_4$  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 the first case,  $q_3(0) = [-14, 15, \frac{\pi}{4}, \frac{\pi}{4}]^\top ([m, rad])$  implying  $\mathcal{A}_3(q_3(0)) \in \pi_4$ , with  $\mathcal{AG}_{3,0}(q_i(0), x^o(0)) = \top$ ,  $c_3 = 3$ , and no collisions occurring at  $t = 0$ . The functions  $\Lambda$  and  $\chi$  are the same as in the first case. The formula in this scenario is

$$(\square\neg“1-\pi_3”) \wedge (\square\Diamond“2-\pi_3”) \wedge (\square\Diamond“O_1-\pi_1”) \wedge \\ \square(“O_1-\pi_1” \rightarrow \Diamond“O_1-\pi_4”) \wedge (\square\Diamond“O_2-\pi_3”),$$

which represents the following behavior. Agent 1 must never visit region  $\pi_3$ , which must be visited infinitely many times by agent 2, object 1 must be taken infinitely many times to region  $\pi_1$ , eventually followed by a visit in region  $\pi_4$ , and object 2 must be taken infinitely many times to region  $\pi_2$ .

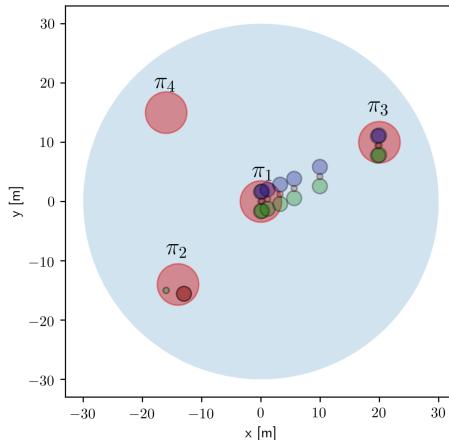
The resulting transition system  $\mathcal{TS}$  consists of 3112 reachable states and 154960 transitions and it was created in 100.74 sec. The Büchi automaton  $\mathcal{BA}$  contains 9 states and 49 transitions and the product  $\widetilde{\mathcal{TS}}$  contains 28008 states and 1890625 transitions. Table 5.2 shows the agent actions for the derived path as the sequence of states



**Figure 5.26:** The transition  $\pi_{s,1} \rightarrow_s \pi_{s,2}$  (a), that corresponds to the navigation of the agents  $\pi_1 \rightarrow_1 \pi_2$ ,  $\pi_3 \rightarrow_2 \pi_1$ ,  $\pi_4 \rightarrow_3 \pi_1$ .

$\pi_{s,1}\pi_{s,2}\dots(\pi_{s,10}^*,\pi_{s,11}^*)^\omega$ . In this case, the three agents navigate first to regions  $\pi_2$ ,  $\pi_1$ , and  $\pi_1$ , respectively, and agents 2 and 3 take object 2 to  $\pi_3$ . Next, agent 3 goes to  $\pi_2$  to transfer object 1 to  $\pi_1$  and then  $\pi_4$  with agent 1. The latter transports occur infinitely often. The time taken for the construction of the product  $\widehat{\mathcal{TS}}$  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. 5.26 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  $K_z = \text{diag}\{0.01, 0.01, 0.01\}$ ,  $\forall z \in \{1, 2, 3\}$ . Moreover, Fig. 5.27 depicts the transportation of object 2 by agents 2 and 3, i.e.,  $\pi_1 \xrightarrow{T} \{\pi_2, \pi_3\} \pi_3$ , that corresponds to  $\pi_{s,3} \rightarrow_s \pi_{s,4}$ .



**Figure 5.27:** The transition  $\pi_{s,3} \rightarrow_s \pi_{s,4}$  (b), that corresponds to the transportation  $\pi_1 \xrightarrow{T} \{\pi_2, \pi_3\} \pi_3$ .

## 5.6 Conclusion

This chapter presented hybrid control strategies for multi-agent systems and multi-agent-object systems under complex specifications expressed as temporal logic formulas. We considered firstly multi-agent teams of aerial vehicles and mobile manipulators with uncertain dynamics, by providing local agent abstractions as well local paths that satisfy the respective agents' LTL formulas. Secondly, we incorporated specifications of unactuated objects of the environment. Based on the previous chapters, we designed appropriate timed abstractions for a single object grasped by two agents in a partitioned workspace, and designed a timed path for it to follow, satisfying its timed specifications, expressed as MITL formulas. Next, we devised a hybrid control scheme for a system comprised of multiple robotic agents and objects that have local LTL formulas over a set of regions of interest in the workspace. We provided a multi-agent-object abstraction as well as a path that satisfies the local specifications.

**Table 5.2:** The agent actions for the discrete path of the second simulation example

$\pi_{s,\ell}$	Actions
$\pi_{s,1}$	(-)
$\pi_{s,2}$	$(\pi_1 \rightarrow_1 \pi_2, \pi_3 \rightarrow_2 \pi_1, \pi_4 \rightarrow_3 \pi_1)$
$\pi_{s,3}$	$(-, 2 \xrightarrow{g} 1, 3 \xrightarrow{g} 2)$
$\pi_{s,4}$	$(-, \pi_1 \xrightarrow[T]{\{2,3\},2} \pi_3,)$
$\pi_{s,5}$	$(-, -, 3 \xrightarrow{r} 2)$
$\pi_{s,6}$	$(-, -, \pi_3 \rightarrow_3 \pi_2)$
$\pi_{s,7}$	$(1 \xrightarrow{g} 1, 3 \xrightarrow{g} 1)$
$\pi_{s,8}$	$(\pi_2 \xrightarrow[T]{\{1,3\},1} \pi_1, -)$
$\pi_{s,9}$	$(\pi_1 \xrightarrow[T]{\{1,3\},1} \pi_4, -)$
$\pi_{s,10}^*$	$(\pi_4 \xrightarrow[T]{\{1,3\},1} \pi_1, -)$
$\pi_{s,11}^*$	$(\pi_1 \xrightarrow[T]{\{1,3\},1} \pi_4, -)$

# Chapter 6

# Single-Agent Extensions

This final chapter considers some additional problems for single-agent systems. Firstly, we consider the optimal motion planning of a single robot in a workspace with obstacles, under time temporal constraints. Unlike Section 5.4, we do not resort to a complete partition of the workspace, making thus the proposed algorithm more efficient. Moreover, a novel reconfiguration scheme guarantees that the obtained path is asymptotically optimal.

Secondly, we integrate adaptive control methodologies with sampling-based motion planning for high-dimensional complex systems, such as robotic manipulators. In particular, a standard adaptive control scheme is developed that compensates for the uncertain Lagrangian dynamics of the system and allows tracking of a predefined trajectory within certain bounds. These bounds are then passed to a RRT-variant planner that outputs a feasible collision-free geometric path to follow.

Finally, we develop an extension of the standard Prescribed Performance Control methodology (see Appendix B) that guarantees compliance with funnel constraints as well as *asymptotic stability*. The developed scheme applies for control-affine 2nd-order systems with *completely unknown* dynamic terms.

## 6.1 Introduction

The first part of this chapter deals with robot motion planning under timed temporal constraints in an obstacle-cluttered workspace. As already discussed in the previous chapters, temporal logic-based motion planning has gained significant attention in recent years, since it provides a fully automated correct-by-design control synthesis approach for autonomous robots. An attribute that makes the problem both more interesting and challenging is the incorporation of time constraints in the temporal specification, as done in Section 5.4.

In this case, however, we do not fully partition the workspace and take into account the environment obstacles via a continuous feedback control

scheme proposed in [? ]. The latter guarantees *timed* collision-free navigation and allows thus the discretization of the robot motion as a weighted transition system among a predefined set of regions of interest, as in Sections 5.2 and 5.5. Subsequently, we employ formal verification techniques to derive a plan that satisfies the *untimed* specification and recast the assignment of the transition times as a convex optimization problem thereby achieving satisfaction of the timed specification. The transition times are recalculated after each transition, incorporating newly acquired information, and resulting in decreased control effort. In that sense, the proposed scheme is asymptotically optimal with respect to the robot control effort.

The second part of the chapter focuses on the motion planning problem of complex high-dimensional systems (e.g., robotic manipulators) with dynamic uncertainties in obstacle-cluttered environments. In particular, we integrate sampling-based motion planning and adaptive control techniques to provide a computationally efficient framework that navigates the system to a desired goal while provably avoiding obstacles and compensating for the uncertain dynamics.

For complex systems in high-dimensional spaces, closed-form feedback control fails to guarantee global solutions, and randomized planning has been introduced to overcome the respective scalability issues of standard motion planners (e.g.,  $A^*$ ); [? ? ?] introduce the notions of probabilistic roadmaps (PRM) and random trees (RRT, EST), respectively, which constitute efficient and probabilistically complete solutions to multiple- and single-query, respectively, high-dimensional motion planning problems. The intuition behind these algorithms is the addition of random sampled states of the free space to a discrete graph/tree, promoting the search of the unexplored free space.

Furthermore, although the initial works derive geometric solutions in the configuration space, trees have been extended to kinodynamic planning, where the robot dynamics  $\dot{x} = f(x, t, u)$  are taken into account [? ? ? ? ]. In these algorithms, the robot dynamics are simulated forward in time, possibly by randomly sampling inputs, in order to find a feasible path. Except for the randomized inputs, the incremental step as well as the duration of this forward simulation are often also chosen randomly. In high dimensional spaces, this randomness might require excessive tuning of the aforementioned parameters in order to find a solution in a reasonable amount of time. Along the lines of PRM, [? ] and [? ] introduce the notion of LQR-trees, which constitute trees of trajectories that probabilistically cover the state space. In that way, every controllable initial condition belongs to the region of attraction (funnel) of a trajectory and is thus driven to the goal via local linearization of the dynamics and optimal feedback control. Dynamics linearization and reachability sets were also recently used to develop an

optimal kinodynamic algorithm, namely R3T [? ].

A potential drawback of the aforementioned algorithms on kinodynamic motion planning is their strong dependence on the robot dynamics, which in general might be uncertain/unknown. The accurate identification of the dynamic models of real robots is a very tedious and often ineffective procedure. Therefore, the robot model used in standard forward simulation-based kinodynamic algorithms might deviate from the actual dynamics, outputting hence paths that might be colliding with obstacles or difficult to be realized by the actual robotic system. Similar to the LQR-trees, [?] proposes an algorithm that builds trees of funnels based on the (known) bounds of model disturbances, restricted however to polynomial robot dynamics. Planning under uncertainty has been also considered in a stochastic framework and via belief trees [? ? ? ? ]. These approaches, however, usually deal with linearized dynamics, and/or propagate the uncertainties on the planning horizon, constraining thus excessively the free space.

In this chapter, we propose a two-layer framework that integrates “intelligent” feedback control protocols with geometric motion planning for high-dimensional Lagrangian holonomic systems (e.g., robotic manipulators). Firstly, motivated by the difficulty of measuring accurately the robotic system’s dynamical parameters (like masses, and moments of inertia) as well as potential external disturbances, we design a feedback control scheme that does not use any information on these parameters/disturbances. The control scheme is a variation of standard adaptive control design, and aims at achieving tracking of a given trajectory for the robot, which is assumed to obey 2nd-order dynamics. The tracking of the trajectory is achieved within certain bounds that stem from the aforementioned uncertainties/disturbances. These bounds create an implicit funnel around the trajectory, which can be further shrunk by appropriate tuning of the control parameters, the latter being a standard procedure in adaptive control design. This funnel is then incorporated in a RRT-like algorithm, which outputs a path connecting an initial configuration to the goal. The construction of the RRT and the employed control protocol guarantee that the robot will follow the derived path without colliding with the workspace obstacles. In that way, by using appropriate feedback control, the proposed methodology “relieves” the sampling-based motion planner of the robot dynamics and their uncertainties, hence the problem of constructing a path becomes purely geometrical. The motion planner relies only on the performance of the control layer, encoded in the aforementioned bounds. Similar ideas were pursued in [?] and [?]; [?], however, just provides a general idea of interfacing the planning and control layers, without elaborating on a particular systematic control technique, while [?] considers mainly predictive controllers for linear systems, without avoiding the forward simulation of the available system model. The proposed framework exhibits the following important characteristics: 1) The robot

dynamics are not forward simulated and hence they are decoupled from the motion planner. Consequently, even though a 2nd-order system is considered, the motion planner is purely geometrical and depends on the geometry of the configuration space as well as the bounds of the robot uncertainties. 2) We do not resort to linearization of the dynamics and computation of basins of attraction around the output trajectories, since the designed feedback control protocol applies directly to the nonlinear model. Finally, the proposed algorithm is expected, in practice, to exhibit lower complexity than standard kinodynamic planning algorithms, since it is purely geometrical and does not simulate any differential equations. The proposed methodology is validated using a UR5 robotic manipulator in V-REP environment [? ].

The third part of this chapter deals with the problem of *asymptotic stability* subject to funnel constraints for a class of 2nd-order uncertain systems, which have been mostly studied through robust and adaptive control, as well as neural network/fuzzy logic control [? ? ]. There exists a variety of works achieving both asymptotic and “practical” (ultimately bounded errors) stability under the presence of model uncertainties (e.g., [? ? ? ? ? ? ? ? ? ? ? ? ]). The majority of the related works that achieve asymptotic stability assume parametric uncertainty of the underlying dynamics, and employ standard adaptive control techniques to compensate for them. Neural network approximations and fuzzy logic controllers have been also extensively used (e.g., [? ? ? ? ]), being valid, however, only in certain compact sets of the state space, and possibly yielding complex structures. Asymptotic stability subject to parametric and structural uncertainties is guaranteed in [?] under a set of initial conditions, where gain tuning and growth conditions on the unknown terms are assumed. The same property is achieved in [? ], where the controller uses partial information of the input matrix, as well as gain tuning.

A well-studied special instance of adaptive control is funnel control, where the output of the system is confined to a predefined funnel [? ? ? ]. It is a model-free control scheme of high-gain type, with numerous applications during the last years. Examples include chemical reactors [? ], robotic manipulation [? ] (and Section 2.2.4), vehicle platooning [? ? ], temporal logic planning (see Section 5.4), and multi-agent systems [? ? ? ] (and also Section 3.2). The intuition behind funnel control is the incorporation of an adaptive gain in the control scheme, which increases (in absolute value) as the system’s output reaches the funnel’s boundary. In that way, the system’s output is “pushed” to always remain inside the funnel. Funnel control has been developed for both linear (e.g., [? ]) and nonlinear systems (e.g., [? ? ? ]), involving parametric (e.g., [? ]) as well as structural (e.g., [? ]) dynamic uncertainties, for a wide class of systems. A funnel bang-bang controller for SISO systems was developed in [? ].

An important property that most related funnel-control works fail to achieve is that of asymptotic stability subject to unknown nonlinear dynamics. Traditional funnel control guarantees only confinement of the system output in a prespecified funnel, and thus the closest property to asymptotic stability that can be achieved is that of “practical stability”, where the funnel converges arbitrarily close to zero. The latter, however, might yield undesired large inputs due to the small funnel values, and can be problematic in real-time systems. On the other hand, with potential guarantees of asymptotic stability, the funnel is not needed to converge close to zero, and can be used in order to encode just transient constraints for the system. Asymptotic tracking subject to transient constraints has been considered in several works [? ? ? ?]; [? ? ? ] consider linear systems (LTI and double integrators), whereas [?] assumes known model structure, with the uncertainties being only parametric; Along with the funnel confinement objective, finite-time stability has been also considered in [?] for a Lagrangian-dynamics model. One can conclude that the aforementioned works cannot be extended in a straightforward manner to nonlinear systems where the dynamic terms have both *parametric and structural* uncertainties. In addition, a class of systems for which funnel control has not been taken into account in the related works is the non-smooth type, i.e., systems with discontinuous right-hand side. Such models are motivated by real-time systems, where several dynamic terms (e.g., friction) can be accurately modeled by discontinuous functions of the state.

The third part of this chapter considers the asymptotic tracking control problem subject to transient constraints imposed by a predefined funnel for a class of MIMO systems satisfying a loose set of assumptions. The control design combines adaptive and discontinuous control techniques and its region of attraction is independent of the system (unknown) dynamics, and relies on the initial funnel condition. If the latter is a design parameter, the results can be rendered global. It is worth noting that asymptotic stability has not been guaranteed in the related literature for such systems under the mild considered assumptions.

## 6.2 Reconfigurable Motion Planning and Control in Obstacle Cluttered Environments under Timed Temporal Tasks

We first tackle the problem of single-robot motion planning in workspace with obstacles under time temporal constraints. We develop a novel reconfigurable control scheme that achieves asymptotic optimality of the derived paths.

### 6.2.1 Problem Formulation

Consider a robotic agent operating in an open bounded subset  $\mathcal{W}$  of the 2-dimensional Euclidean space. In addition, the workspace is populated with  $m \in \mathbb{N}$  connected, closed sets  $\{O_i\}_{i \in \mathcal{J}}$ , indexed by the set  $\mathcal{J} := \{1, \dots, m\}$ , representing obstacles. Accordingly, we define the free space as

$$\mathcal{F} := \mathcal{W} \setminus \bigcup_{i \in \mathcal{J}} O_i,$$

**Remark 6.1.** To facilitate the exposition, we assume that all the data describing the workspace are known *a priori*. The analysis remains the same for the case of initially unknown workspaces where obstacles are discovered along the way.

The agent is assumed to be a point<sup>1</sup> described by the position variable  $x \in \mathbb{R}^2$  which is governed by the single integrator dynamics,

$$\dot{x} = u, \quad u \in \mathbb{R}^2. \quad (6.1)$$

Moreover, similarly to the previous sections, we consider that there exist  $K$  points of interest in the free space, denoted by  $c_{\pi_k} \in \mathcal{F}$ , for every  $k \in \mathcal{K}_R := \{1, \dots, K\}$ , with  $\Pi := \{c_{\pi_1}, \dots, c_{\pi_K}\}$ , that correspond to certain properties of interest (*e.g.*, gas station, obstacle region, repairing area, etc.) These properties of interest are expressed as boolean variables via the finite set of atomic propositions  $\Psi$ . The properties satisfied at each point are provided by the labeling function  $\mathcal{L} : \Pi \rightarrow 2^\Psi$ , which assigns to each point  $c_{\pi_k}, k \in \mathcal{K}_R$ , the subset of the atomic propositions that hold true in that point.

Since, in practice, the aforementioned properties shared by a point of interest are naturally inherited to some neighborhood of that point we define for each  $k \in \mathcal{K}_R$ , the *region of interest*  $\pi_k$  corresponding to the point of interest  $c_{\pi_k}$  as the set

$$\pi_k := \bar{\mathcal{B}}(c_{\pi_k}, r_{\pi_k}) \cap \mathcal{F}, \quad r_{\pi_k} \in \mathbb{R}_{>0}.$$

We also let  $\pi_{\mathcal{W}} := \mathcal{F} \setminus (\cup_{k \in \mathcal{K}_R} \pi_k)$  be the subset of the free space outside the regions of interest. We define thus the set  $\tilde{\Pi} := \{\pi_k\}_{k \in \mathcal{K}_R} \cup \{\pi_{\mathcal{W}}\}$  as well as the corresponding labeling function as  $\tilde{\mathcal{L}} : \tilde{\Pi} \rightarrow 2^\Psi$ , with  $\mathcal{L}(c_{\pi_k}) = \{p\} \Leftrightarrow \tilde{\mathcal{L}}(\pi_k) = \{p\}$ ,  $\forall k \in \mathcal{K}_R$ , and  $\tilde{\mathcal{L}}(\pi_{\mathcal{W}}) = \emptyset$ . The agent is assumed to be in a region  $\pi_k$ ,  $k \in \mathcal{K}_R$ , in  $\pi_{\mathcal{W}}$ , simply when  $x \in \pi_k$  and  $x \in \pi_{\mathcal{W}}$ , respectively. We assume that, for all  $k \in \mathcal{K}_R$ , the location of the points  $c_{\pi_k}$  as well as the radii  $r_{\pi_k}$  are known.

---

<sup>1</sup>Treating a robot with volume can be achieved by initially “transferring” its volume to the other workspace entities (*e.g.*, obstacles) and subsequently considering it as a point.

We make the following standard assumptions [? ] regarding the geometry of the workspace and the regions of interest.

**Assumption 6.1.** The collection of sets comprised of all obstacles and regions of interest is pairwise disjoint.

The aforementioned assumption simply states that the obstacles/regions of interest are sufficiently away from each other as well as the workspace boundary.

As already mentioned, we are interested in defining timed temporal formulas over the atomic propositions  $\Psi$ , and hence, over the regions of interest  $\Pi$  of  $\mathcal{F}$ . To that end, we need to discretize the system using a finite set of states. We will achieve that by guaranteeing timed transitions between the regions of interest in  $\Pi$  and by building a well-defined timed transition system among them. We first need the following definition regarding the transitions of the agent.

**Definition 6.1.** Assume that  $x(t_k) \in \mathcal{F}$ , for a  $t_k \in \mathbb{R}_{\geq 0}$ , i.e., the agent is either in a region  $\pi_k$ , for some  $k \in \mathcal{K}_R$ , or in  $\pi_W$ . Then, given  $\delta \in \mathbb{R}_{>0}$ , there exists a *timed transition* to  $\pi_\ell$ ,  $\ell \in \mathcal{K}_R$ , denoted as  $\pi_k \xrightarrow{\delta} \pi_\ell$  (or  $\pi_W \xrightarrow{\delta} \pi_\ell$ ), if there exists a time-varying feedback control law  $u : \mathcal{F} \times [t_k, t_\ell] \rightarrow \mathbb{R}^2$ , with  $t_\ell \geq t_k + \delta$ , such that the solution  $x$  of the closed loop system (6.1) satisfies the following:

1.  $x(t) \in \pi_l$ , for all  $t \in [t_k + \delta, t_l]$ ,
2.  $x(t) \in \mathcal{F}$ , for all  $t \in [t_k, t_\ell]$ ,
3.  $x(t) \notin \pi_m$ , for all  $m \in \mathcal{M}_s$ ,  $t \in [t_k, t_\ell]$ ,

where  $\mathcal{M}_s := \mathcal{K}_R \setminus \{k, \ell\}$  if  $x(t_k) \in \pi_k$  and  $\mathcal{M}_s := \mathcal{K}_R \setminus \{\ell\}$  if  $x(t_k) \in \pi_W$ .

Intuitively, according to (6.1), the agent has to transit between two regions  $\pi_k, \pi_\ell$  (or  $\pi_W$  and  $\pi_\ell$ ), while avoiding all other regions of interest, obstacles, as well as the workspace boundary. In what follows, we sometimes use  $\pi_k \xrightarrow{\delta} \pi_\ell$  instead of  $\pi_k \rightarrow \pi_\ell$  to emphasize the transition time  $\delta$ . We have included the space outside the regions  $\pi_W$  to account for initial conditions that might satisfy  $x(t_k) \notin \cup_{k \in \mathcal{K}_R} \pi_k$ . Next, we define the behavior of the agent, in order to formulate the problem of timed specifications.

**Definition 6.2.** Consider an agent trajectory  $x : [t_0, \infty) \rightarrow \mathcal{F}$  of (6.1), where  $t_0 \in \mathbb{R}_{\geq 0}$ . Then, a *timed behavior* of  $x$  is the infinite sequence  $b := (x(t_1), \psi_1, t_1)(x(t_2), \psi_2, t_2) \dots$ , where  $t_1 t_2 \dots$  is a time sequence according to Def. F.5 of Appendix F,  $x(t_0) \in \tilde{\Pi}$ ,  $x(t_l) \in \pi_{j_l}$ ,  $j_l \in \mathcal{K}_R$ ,  $\forall l \in \mathbb{N}$ , and  $\psi_l = \mathcal{L}(\pi_{j_l}) \subseteq 2^\Psi$ , i.e., the subset of atomic propositions that are true when  $x(t_j) \in \pi_{j_l}, \forall l \in \mathbb{N}$ .

The specifications in this section are expressed via a Metric Interval Temporal Logic (MITL) formula  $\Phi$  (see Appendix F for more details), although other timed variants could be used. The timed behavior  $\mathbf{b}$  satisfies a timed formula  $\Phi$  if and only if  $\mathbf{b}_\psi := (\psi_0, t_0)(\psi_1, t_1)\dots \models \Phi$ .

We are now ready to state the problem addressed in this section.

**Problem 6.1.** Consider a robot with dynamics governed by (6.1), operating in the workspace  $\mathcal{W}$ , with initial position  $x(0) \in \mathcal{F}$ . Given a timed formula  $\Phi$  over  $\Psi$  and a labeling function  $\tilde{\mathcal{L}}$ , develop a control strategy that results in a solution  $x : [0, \infty) \rightarrow \mathcal{F}$ , which achieves a timed behavior  $\mathbf{b}$  that yields the satisfaction of  $\Phi$ .

### 6.2.2 Problem Solution

In this section we present the proposed solution, which consists of two layers: (i) a tuning-free continuous control law that guarantees the navigation of the agent to a desired point from *all* obstacle-collision-free configurations, and (ii) a discrete time plan over the regions of interest for the robot to follow, which employs formal verification and optimization techniques and is updated on-line.

#### Motion Controller

The first part of the proposed solution is the design of a control protocol such that a transition to a region of interest is established, according to Def. 6.1. Assume, therefore, that  $x(t_k) \in \mathcal{F}$ , and more specifically,  $x(t_k) \in \pi_k$  ( $x(t_k) \in \pi_{\mathcal{W}}$ ) for some  $t_k \in \mathbb{R}_{\geq 0}$  and  $k \in \mathcal{K}_{\mathcal{R}}$ . Given  $\delta \in \mathbb{R}_{>0}$ , we wish to find a time-varying state-feedback control law  $u : \mathcal{F} \times [t_k, t_\ell] \rightarrow \mathbb{R}^2$ , with  $t_\ell \geq t_k + \delta$ , such that  $\pi_k \xrightarrow{\delta} \pi_\ell$  ( $\pi_{\mathcal{W}} \xrightarrow{\delta} \pi_\ell$ ). To that end, we first redefine the free space as

$$\mathcal{F} := \mathcal{W} \setminus \left( \bigcup_{i \in \mathcal{J}} \mathcal{O}_i \cup \bigcup_{m \in \mathcal{M}_s} \pi_m \right),$$

so that regions of interest that shall not be crossed during the transition are regarded as obstacles.

Following the previous work [?], the tuple  $(x(t_k), c_{\pi_l}, \delta)$  constitutes a well-defined instance of the *Prescribed Time Scale Navigation Problem* [?, Problem 1] in  $\mathcal{F}$ . Theorem 2 of the aforementioned work suggests that the construction of the required feedback law  $u : \mathcal{F} \times [t_k, t_\ell] \rightarrow \mathbb{R}^2$  reduces to the problem of smoothly transforming the free space  $\mathcal{F}$  to a topologically equivalent, yet geometrically simpler, space.

More specifically, we require a diffeomorphism  $T : \mathcal{F} \rightarrow \mathcal{P}$  where  $\mathcal{P}$  is a point world [?]; an open disk modulo a finite set with cardinality equal to the number of obstacles and regions of interest  $|\mathcal{J}| + |\mathcal{M}_s|$ . Under the prevailing Assumption 6.1 [? , Theorem 1] provides a computationally efficient method to determine the space  $\mathcal{P}$  and the mapping  $T$ .

This allows us to apply the conclusions of [? , Theorems 1, 2] which yield a feedback law  $u : \mathcal{F} \times [t_k, t_\ell] \rightarrow \mathbb{R}^2$  such that the closed-loop system satisfies the properties of 6.1 therefore establishing the existence of the required timed transition. More details can be found in [? ].

### High-Level Plan Generation

The second part of our solution is the derivation of a high-level timed plan over the regions of interest, which satisfies the given timed formula  $\Phi$ . This plan will be generated using standard techniques from automata-based formal verification and optimization methodologies. Thanks to the proposed control law of the previous section that allows the transitions in the set  $\tilde{\Pi}$  in predefined time intervals, we can abstract the motion of the robotic agent as a finite transition system  $\mathcal{T} := \{\tilde{\Pi}, \tilde{\Pi}_0, \rightarrow, \Psi, \tilde{\mathcal{L}}, \gamma\}$ , where  $\tilde{\Pi}$  is the set of states defined in Section 6.2.1,  $\tilde{\Pi}_0 \in \tilde{\Pi}$  is the initial state,  $\rightarrow := \tilde{\Pi} \times \tilde{\Pi}$  is a transition relation according to Def. 6.1,  $\Psi$  and  $\tilde{\mathcal{L}}$  are the atomic propositions and the labeling function, respectively, as defined in Section 6.2.1, and  $\gamma : (\rightarrow) \rightarrow \mathbb{R}_{>0}$  is a cost associated with each transition. More specifically, we consider as cost the distance the agent has to cover from a region  $\pi_k$  (or  $\pi_{\mathcal{W}}$ ) to a region  $\pi_\ell$ . However, this cost is highly dependent on the initial robot configuration and the number and position of the obstacles between the initial and the goal regions, and cannot be computed explicitly. Therefore, we initially set  $\gamma(\pi_k \rightarrow \pi_\ell) = \|c_k - c_\ell\|$ ,  $\gamma(\pi_k \rightarrow \pi_k) = 0$ , and  $\gamma(\pi_{\mathcal{W}} \rightarrow \pi_k) = \gamma(\pi_k \rightarrow \pi_{\mathcal{W}}) = \|c_k - x(0)\|$ , for all  $k, \ell \in \mathcal{K}_{\mathcal{R}}$  with  $k \neq \ell$ , and proceed with the derivation of the timed plan as a timed sequence of regions in  $\Pi$ .

Firstly, the timed formula  $\Phi$  over the atomic propositions  $\Psi$  is translated to the TBA  $\mathcal{A}_t = (Q, Q_0, \text{CL}, \Psi, E, F)$  (see Appendix F for more details) using off-the-shelf tools [? ]. Secondly, we calculate the product Büchi Automaton  $\mathcal{A}_{\mathcal{P}}$  as  $\mathcal{A}_{\mathcal{P}} := \mathcal{T} \otimes \mathcal{A}_t = (S, S_0, \rightarrow_{\mathcal{P}}, F_{\mathcal{P}}, \gamma_{\mathcal{P}})$ , where

- $S = \tilde{\Pi} \times Q$ ,
- $S_0 = \tilde{\Pi} \times Q_0$ ,
- $\rightarrow_{\mathcal{P}} \subset S \times \Phi(\text{CL}) \times 2^C \times S$  gives the set of edges;  $e := (s, g, R, s') \in \rightarrow_{\mathcal{P}}$ , with  $s := (\pi, q)$ ,  $s' := (\pi', q') \in S$  if and only if (i)  $(q, g, R, \mathcal{L}(\pi), q') \in E$  and (ii)  $q = q'$ ,  $(\pi, \pi') \in \rightarrow$ .

- $F_{\mathcal{P}} \subseteq \tilde{\Pi} \times F$  with  $s := (\pi, q) \in F_{\mathcal{P}}$  if and only if  $q \in F$  and  $(s, \Phi(\text{CL}), R, S) \in E$  for some state in  $S$ , i.e., there is always a transition from  $s$ , for all the the possible valuations of the clocks CL.
- $\gamma_{\mathcal{P}} : (\rightarrow_{\mathcal{P}}^*) \rightarrow \mathbb{R}_{>0}$ , with  $\gamma_{\mathcal{P}}((s, g, R, s')) = \gamma(\pi \rightarrow \pi')$ , where  $(\rightarrow_{\mathcal{P}}^*) := \{((\pi, q), g, R, (\pi', q')) \in \rightarrow_{\mathcal{P}} : \pi \neq \pi'\}\}.$

We use the abbreviation  $s \xrightarrow{I} s'$  for  $(s, g, R, s') \in \rightarrow_{\mathcal{P}}$ , where  $I := \{g, R\}$ . Note that the product  $\mathcal{A}_{\mathcal{P}}$  consists of a finite number of states, and therefore we can employ graph-search techniques to find the optimal timed path, with respect to the cost  $\gamma_{\mathcal{P}}$ , from the initial states  $S_0$  to the accepting states  $F_{\mathcal{P}}$ , which will satisfy the given timed formula  $\Phi$  [? ]. This path will contain a finite prefix — a finite sequence of states to be visited — and a infinite suffix — a specific sequence of states to be visited infinitely many times [? ? ]. Moreover, note that the motion controller developed in Section [section 6.2.2](#) can guarantee the safe navigation among two regions of interest in any predefined time interval.

By viewing  $\mathcal{A}_{\mathcal{P}}$  as a graph, we can find a path that starts at the initial states  $S_0$  and traverses an accepting state in  $F_{\mathcal{P}}$  infinitely many times. Such a path has the form

$$\bar{s}_{p_1} \xrightarrow{I_{1,2}} \bar{s}_{p_2} \xrightarrow{I_{2,3}} \dots \xrightarrow{I_{L-1,L}} \bar{s}_{p_L} \xrightarrow{I_{L,L+1}} \left( \bar{s}_{p_{L+1}} \xrightarrow{I_{L+1,L+2}} \dots \xrightarrow{I_{L+Z-1,L+Z}} \bar{s}_{p_{L+Z}} \right)^{\omega}$$

Here,  $\bar{s}_{p_j}$ , for  $j \in \{1, \dots, L + Z\}$ , denotes the sequence of states

$$\bar{s}_{p_j} := (\pi_{p_j}, q_{j_1}) \xrightarrow{I_{j_1,2}} \dots \xrightarrow{I_{j_{(\ell_j-1)},\ell_j}} (\pi_{p_j}, q_{j_{\ell_j}}),$$

with  $\pi_{p_j} \in \tilde{\Pi}$ ,  $q_{j_\iota} \in Q$ , for  $j \in \{1, \dots, L + Z\}$ ,  $\iota \in \{1, \dots, \ell_j\}$ , and  $\ell_j \in \{1, \dots, |S|\}$ . Moreover,  $q_{(j+1)_1} = q_{j_{\ell_j}}$ ,  $q_{(L+Z)\ell_{(L+Z)}} = q_{(L+1)_1}$  and

$$I_{j,j+1} := \{g_{j,j+1}, R_{j,j+1}\}, \quad I_{j_{\iota},\iota+1} := \{g_{j_{\iota},\iota+1}, R_{j_{\iota},\iota+1}\},$$

indicating the corresponding guards and reset maps, for  $j \in \{1, 1, \dots, L + Z - 1\}$ ,  $\iota \in \{1, \dots, \ell_j - 1\}$ . The transition set  $I_{L+Z,L+1}$  is defined similarly. Loosely speaking, the path consists of consecutive (at most  $|S|$ ) transitions of the form  $(\pi_j, q_{j_\iota}) \xrightarrow{(\cdot)} (\pi_j, q_{j_{(\iota+1)}})$  among states in  $\mathcal{A}_t$ , where  $\pi_j$  is fixed, and transitions of the form  $(\pi_{p_j}, q_{j_{\ell_j}}) \xrightarrow{(\cdot)} (\pi_{p_{(j+1)}}, q_{(j+1)_1})$  among the states of  $\mathcal{T}$ , where  $q_{(j+1)_1} = q_{j_{\ell_j}}$  is fixed.

Note that we have not yet associated any time intervals with the transitions  $(\pi_{p_j}, q_{j_{\ell_j}}) \xrightarrow{(\cdot)} (\pi_{p_{(j+1)}}, q_{(j+1)_1})$ , which correspond to physical transitions among the regions of interest. We do that now by using the transition guards  $g_{j,j+1}, g_{j_{\ell_j}, j+1}$ . More specifically, consider the transitions

$$(\pi_{p_j}, q_{j_1}) \xrightarrow{I_{j_1,2}} (\pi_{p_j}, q_{j_2}) \xrightarrow{I_{j_2,3}} \dots \xrightarrow{I_{j_{\ell_j-1}, \ell_j}} (\pi_{p_j}, q_{j_{\ell_j}}) \xrightarrow{I_{j,j+1}} (\pi_{p_{j+1}}, q_{(j+1)_1}),$$

that encode the physical transition from  $\pi_{p_j}$  to  $\pi_{p_{j+1}}$  in  $\mathcal{A}_{\mathcal{P}}$ . The intersection of the respective guards  $g_{j,j+1}, g_{j_{\ell_j}, j+1}$ ,  $\ell \in \{1, \dots, \ell_{j-1}\}$ , provides a time interval of the form  $\mathcal{I}_{j,j+1} \in \{[a, b], [a, b), (a, b], (a, b), [a, \infty), (a, \infty)\}$ , with  $a, b \in \mathbb{Q}_{>0}$ ,  $b > a$ , such that,  $t_{j,j+1} \in \mathcal{I}_{j,j+1} \Rightarrow t_{j,j+1} \models g_{j,j+1}, t_{j,j+1} \models g_{j_{\ell_j}, j+1}$ , for  $\ell \in \{1, \dots, \ell_{j-1}\}$ , where  $t_{j,j+1}$  is the time duration of the navigation  $\pi_j \xrightarrow{t_{j,j+1}} \pi_{j+1}$ . Note that  $\mathcal{I}_{j,j+1}$  might be a function of the previous transition duration  $t_{j-1,j}$ .

Since  $\mathcal{I}_{j,j+1}$  is, in general, an infinite set, and we have, thus, infinitely many choices for  $t_{j,j+1}$ , we propose a procedure for assigning the time durations  $t_{j,j+1}$ , for each  $j \in \{1, L+Z, -1\}$ , and  $t_{L+Z, L+1}$ . In particular, we formulate the transition times assignment as a convex optimization problem. To that end, let  $t_p := [t_{1,2}, \dots, t_{L+Z-1, L+Z}, t_{L+Z, L+1}]^\top \in \mathbb{R}_{>0}^{L+Z+1}$  be the concatenation of the transition times constituting the variable of the following optimization problem:

$$\underset{t_p}{\text{minimize}} \sum_{j=1}^{L+Z} \left( \frac{\gamma(\pi_{p_j} \rightarrow \pi_{p_{j+1}})}{t_{j,j+1}} \right) + \frac{\gamma(\pi_{p_{L+Z}} \rightarrow \pi_{p_{L+1}})}{t_{L+Z, L+1}}, \quad (6.2a)$$

$$\text{subject to } t_{j,j+1} \in \mathcal{I}_{j,j+1}, \text{ for all } j \in \{1, L+Z, -1\}, \quad (6.2b)$$

$$t_{L+Z, L+1} \in \mathcal{I}_{L+Z, L+1}. \quad (6.2c)$$

Note that the objective function is a convex function of  $t_p$  and the constraints can be expressed as linear inequalities on the problem variables. Thus, the above optimization problem is convex and can be efficiently solved using off-the-shelf software. The choice of this particular cost function is motivated by the following two observations: (i) the time assigned to a transition is an increasing function of the transition cost, and (ii) brief transition times are penalized. Furthermore, the imposed constraints guarantee the satisfaction of the formula provided that transitions are executed within the specified transition times. We also report empirical evidence from numerical simulations suggesting that reduction in control effort is achieved.

After solving the aforementioned optimization problem and obtaining the time durations  $t_p$ , the robot performs the first transition using the motion controller presented in [section 6.2.2](#) where  $\delta$  is taken equal to the corresponding transition time. Once the transition  $\pi_{p_j} \xrightarrow{t_{j,j+1}} \pi_{p_{j+1}}$  is completed, the

corresponding transition cost  $\gamma(\pi_j \rightarrow \pi_{j+1})$  is updated by being set equal to the length of the integral curve of the closed-loop system for the duration of the transition. The updated value of the transition cost is in some sense more accurate than the initial estimate based on the Euclidean distance since the existence of obstacles can potentially obstruct the straight line path between two regions of interest.

Having acquired this new information the associated optimization problem can be solved to acquire new values for the transition times. We note that after each transition the constraints of the optimization problem are altered. In particular, the TBA of the formula has to be shifted forward by an amount of time equal to the last performed transition which induces a change in the guards and, therefore, to the optimization problem constraints. We assume that the time needed for solving the optimization problem is short enough so that satisfaction of the formula is not jeopardized. This assumption is reasonable enough primarily owing to the problem's low computational complexity and, secondarily, the fact that the previously computed values of the transition times are a good prior for initiating the numerical solver. Nevertheless, computational overhead can be accounted for in the constraints by allocating the required time, or optimization could be performed en route to the next region of interest with the transition times adjusted in an any-time fashion.

### 6.2.3 Simulation Results

To demonstrate the proposed scheme, we consider a task and motion planning problem for a robot operating in a planar office environment. In particular, we consider three points of interest and therefore have  $\Pi = \{c_{\pi_k}\}_{k \in \mathcal{K}_R}$ , where  $\mathcal{K}_R = \{1, 2, 3\}$ . The corresponding regions of interest  $\pi_k = \mathcal{B}(c_{\pi_k}, r)$ , where  $r_{\pi_k} = 0.2$  for  $k \in \mathcal{K}_R$ , define the set  $\Pi = \{\pi_k\}_{k \in \mathcal{K}_R}$ . The set of atomic prepositions is  $\Psi = \tilde{\Pi}$  and the labeling function  $\mathcal{L} : \Pi \rightarrow 2^\Psi$  is defined as  $\pi_k \mapsto \{\pi_k\}$ ,  $k \in \mathcal{K}_R$ . The scenario setting is illustrated in [Figure 6.1](#).

We require that the robot “*always visits each region of interest at least once every 120 time units*” which is equivalent to the MITL formula  $\Phi = \bigwedge_{k \in \mathcal{K}_R} (\square \diamond [0, 120] \pi_k)$ . The robot is initially located at  $c_{\pi_1} \in \pi_1$  and, therefore, the infinitely repeating cycle of transitions  $\pi_1 \xrightarrow{t_{1,2}(1)} \pi_2 \xrightarrow{t_{2,3}(1)} \pi_3 \xrightarrow{t_{3,1}(1)} \pi_1 \xrightarrow{t_{1,2}(2)} \dots$  with appropriately assigned transition times is an accepting run. Let  $t : 0, 1, \dots \rightarrow \mathbb{Q}_{>0}^3$ ,  $\kappa \mapsto [t_{1,2}(\kappa), t_{2,3}(\kappa), t_{3,1}(\kappa)]^\top$  which is defined recursively as follows:  $t(0) := [0, 0, 0]^\top$ , then assuming  $t(\kappa)$  is defined for some  $\kappa \in \mathbb{N}_0$ ,

$$t(\kappa + 1) := (U_3^\top)^\kappa \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} U_3^\kappa \hat{t}(\kappa) + (U_3^\top)^\kappa \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} U_3^\kappa t(\kappa),$$

where  $U_3 := \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$  is the upper shift matrix and  $\hat{t}(\kappa)$  is the solution of the optimization problem (6.2) under the following constraints:

$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} U_3^\kappa \hat{t}(\kappa) \leq \begin{bmatrix} 120 \\ 120 \\ 120 \end{bmatrix} - \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} U_3^\kappa t(\kappa).$$

The motion controller results in collision-free trajectories (Figure 6.1), and by performing each transition time in the time derived from the optimization procedure results in a run that satisfies the formula  $\Phi$  (see bottom of Figure 6.2). Finally, it is worth noting that the transition times converge in just a few steps as illustrated on the top part of Figure 6.2 and the overall control effort per suffix execution is reduced (Table 6.1).

**Table 6.1: CONTROL EFFORT PER SUFFIX EXECUTION**

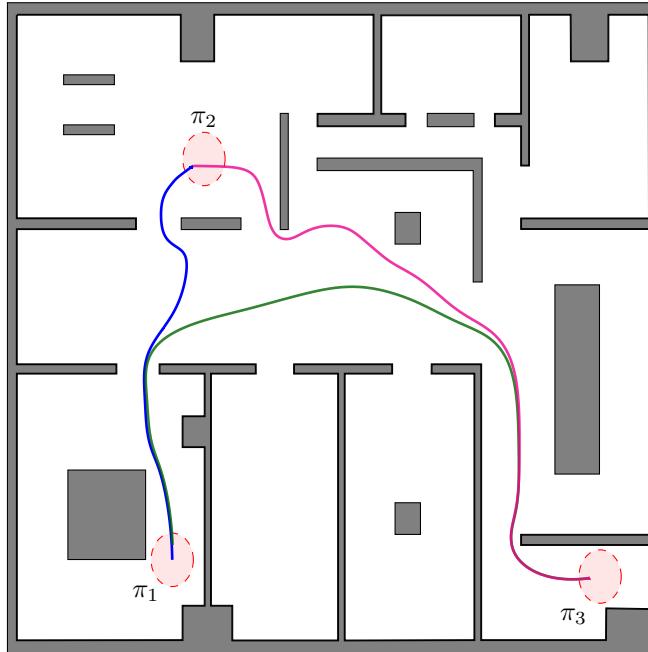
Cycle	1	2	3	4	5
$\int \ u(x(\tau), \tau)\ ^2 d\tau$	8.06	7.67	7.67	7.65	7.66

## 6.3 Sampling-based Motion Planning for Uncertain High-dimensional Systems via Adaptive Control

We turn now our attention to the motion planning problem for uncertain high-dimensional systems, such as robotic manipulators. We integrate sampling-based motion planning techniques with intelligent adaptive control methodologies to tackle the problem of uncertain dynamics and collision-free navigation.

### 6.3.1 Problem Formulation

Consider a robotic system with state  $(q, \dot{q}) \in \mathbb{T} \times \mathbb{R}^n \subset \mathbb{R}^{2n}$ ,  $n \in \mathbb{N}$ , representing its positions and velocities. Usual robotic structures (e.g., robotic manipulators) might consist of translational and rotational joints, which we define here as  $q_{tr} \in \mathbb{R}^{n_{tr}}$  and  $q_r \in [0, 2\pi)^{n_r}$ , respectively, with  $n_{tr} + n_r = n$ , and hence  $\mathbb{T} := \mathcal{W}_{tr} \times [0, 2\pi)^{n_r}$ , where  $\mathcal{W}_{tr}$  is a closed subset of  $\mathbb{R}^{n_{tr}}$ . Without loss of generality, we assume that  $q = [q_{tr}^\top, q_r^\top]^\top$ . We consider that the equations of motion of the robot obey the standard 2nd-order



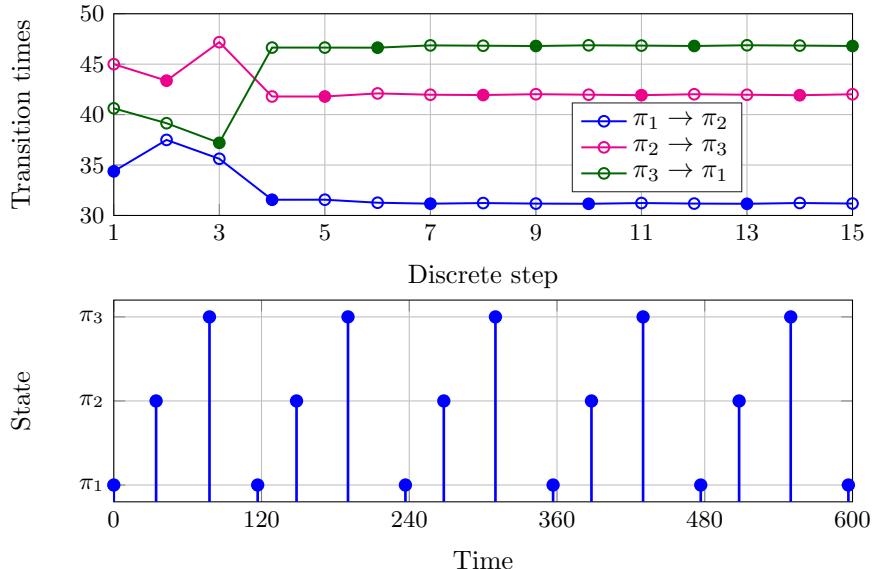
**Figure 6.1:** Workspace overview. The red discs correspond to the three regions of interest. The plotted paths are the resulting trajectories from the first out of the five executions of the suffix.

Lagrangian dynamics (2.1)

$$B(q)\ddot{q} + C_q(q, \dot{q})\dot{q} + g_q(q) + d_q(q, \dot{q}, t) = \tau, \quad (6.3)$$

with the various terms as in (2.1). We assume here that  $d(\cdot)$  is continuous and uniformly bounded by a *known* bound  $\bar{d}$  as  $\|d(t)\| \leq \bar{d}, \forall t \geq 0$ . We remind the reader that the dynamical terms  $B(q)$ ,  $C(q, \dot{q})$ ,  $g(q)$  of (6.3) depend on the dynamical parameters of the robot, i.e., its mass and moment of inertia. These parameters are assumed to be *unknown*, and hence they cannot be used in the planning and control modules. The same applies to the function  $d(\cdot)$ . Nevertheless, as will be shown later, having satisfying estimates for these terms renders the planning module for the robot less conservative in terms of collision checking.

We consider that the robot operates in a workspace  $\mathcal{W} \subset \mathbb{R}^3$  filled with obstacles occupying a closed set  $\mathcal{O} \subset \mathbb{R}^3$ . We denote the set of points that consist the volume of the robot at configuration  $q$  as  $\mathcal{A}(q) \subset \mathbb{R}^3$ . The collision-free space is defined as the open set  $\mathcal{A}_{\text{free}} := \{q \in \mathbb{T} : \mathcal{A}(q) \cap \mathcal{O} = \emptyset\}$ . Our goal is to achieve safe navigation of the robot to a predefined goal



**Figure 6.2:** (top) The transition times calculated before each transition; filled marks correspond to actually used transition times. (bottom) The resulting timed run of the transition system.

region  $Q_g \subset \mathcal{A}_{\text{free}}$  from an initial configuration  $q(0) \in \mathcal{A}_{\text{free}}$  via a path  $q_p : [0, \sigma] \rightarrow \mathcal{A}_{\text{free}}$  satisfying  $q_p(0) = q(0)$  and  $q_p(\sigma) \in Q_g$ , for some positive  $\sigma$ .

The problem we consider is the following:

**Problem 6.2.** Given  $q(0) \in \mathcal{A}_{\text{free}}$  and  $Q_g \subset \mathcal{A}_{\text{free}}$ , respectively, design a control trajectory  $u : [0, t_f] \rightarrow \mathbb{R}^n$ , for some finite  $t_f > 0$ , such that the solution  $q^*(t)$  of (6.3) satisfies  $q^*(t) \in \mathcal{A}_{\text{free}}$ ,  $\forall t \in [0, t_f]$ , and  $q^*(t_f) \in Q_g$ .

The feasibility of Problem 6.2 is established in the following assumption.

**Assumption 6.2.** There exists a (at least twice differentiable) path  $q_p : [0, \sigma] \rightarrow \mathcal{A}_{\text{free}}$  such that  $q_p(0) = q(0)$  and  $q_p(\sigma) \in Q_g$ .

### 6.3.2 Problem Solution

We present here the proposed solution for Problem 6.2. Our methodology follows a two-layer approach, consisting of a robust trajectory-tracking control design and a higher-level sampling-based motion planner. Firstly, we use an adaptive control protocol that compensates for the uncertain dynamical parameters of the robot and forces the system to evolve in a

funnel around a desired trajectory, whose size depends on the initial estimates of the dynamical parameters and the bound of the external disturbances. Secondly, we develop a geometric sampling-based motion planner that uses this funnel to find a collision free trajectory from the initial to the goal configuration. Intuitively, the robust control design helps the motion planner procedure, which does not have to take into account the complete dynamics (6.3).

## Control Design

We first recap the dynamics linear parameterization with respect to the aforementioned unknown parameters, denoted by  $\theta \in \mathbb{R}^l$ ,  $l \in \mathbb{N}$ . More specifically, similarly to Chapter 2, it holds that

$$B(a)d + C(a, b)c + g(a) = Y(a, b, c, d)\theta, \quad (6.4)$$

$\forall a \in \mathbb{T}, b, c, d \in \mathbb{R}^{3n}$ , where  $Y(\cdot)$  is a matrix independent of  $\theta$ . Let  $q_d := [q_{d,tr}^\top, q_{d,r}^\top]^\top : [0, t_f] \rightarrow \mathbb{T}$  be a reference trajectory, with  $q_{d,tr} \in \mathbb{R}^{n_{tr}}$  and  $q_{d,r} \in [0, 2\pi)^{n_r}$  being its translational and rotational parts, respectively. Such a trajectory will be the output of the sampling-based motion planning algorithm that will be developed in the next section. We wish to design the control input  $\tau$  of (6.3) such that  $q(t)$  converges close to  $q_d(t)$ , despite the uncertainty in  $\theta$ . We show how such a design can be used in the motion planning of a robotic system with dynamical uncertainties by developing a suitable variant of a standard adaptive control scheme.

We start by defining the appropriate error metric between  $q = [q_{tr}^\top, q_r^\top]^\top$  and  $q_d = [q_{d,tr}^\top, q_{d,r}^\top]^\top$ , which represents their distance. Regarding the translational part, we define the standard Euclidean error  $e_{tr} := q_{tr} - q_{d,tr}$ . For the rotation part, however, the same error  $e_r := q_r - q_{d,r}$  does not represent the minimum distance metric, since  $q_r$  evolves on the  $n_r$ -dimensional sphere, and its use might cause conservative or infeasible results in the planning layer. Hence, unlike standard adaptive control schemes for robotic manipulators, which drive the Euclidean difference  $e_r(t)$  to zero (e.g., [? ?]), we use the chordal metric  $d_C(x, y) := 1 - \cos(x, y) \in [0, 2]$ ,  $\forall x, y \in [0, 2\pi)$ , or  $\bar{d}_C(x, y) := \sum_{j \in \{1, \dots, \ell\}} d_C(x_j, y_j)$  for vectors  $x = [x_1, \dots, x_\ell], y = [y_1, \dots, y_\ell] \in [0, 2\pi)^\ell$ . Nevertheless, note that rotational joints subject to upper and/or lower mechanical limits evolve in  $\mathbb{R}$  rather than the unit circle and should hence be included in  $q_{tr}$  instead of  $q_r$ .

We are now ready to define a suitable distance metric for  $\mathbb{T}$  as follows: for  $x := [x_{tr}^\top, x_r^\top]^\top$ ,  $y := [y_{tr}^\top, y_r^\top]^\top \in \mathbb{T}$  we define  $d_{\mathbb{T}}$  as

$$d_{\mathbb{T}}(x, y) := \|x_{tr} - y_{tr}\|^2 + \bar{d}_C(x_r, y_r).$$

Note, however, that the chordal metric induces a limitation with respect to tracking on the unit sphere. Consider  $d_C(q_{r_j}, q_{d,r_j}) = 1 - \cos(e_{r_j})$ , where

we further define  $e_{r_j} := q_{r_j} - q_{d,r_j}$  as the  $j$ th element of  $e_r$ ,  $j \in \{1, \dots, n_r\}$ . Differentiation yields

$$\dot{d}_C(q_{r_j}, q_{d,r_j}) = \sin(e_{r_j})\dot{e}_{r_j}, \quad \forall j \in \{1, \dots, n_r\},$$

which is zero when  $e_{r_j} = 0$  or  $e_{r_j} = \pi$ . The second case is an undesired equilibrium, which implies that the point  $e_{r_j} = 0$  cannot be stabilized from *all* initial conditions using a continuous controller. This is an inherent property of dynamics on the unit sphere due to topological obstructions ([?]). In the following, we devise a control scheme that, except for driving  $q(t)$  to  $q_d(t)$ , guarantees that  $e_{r_j}(t) \neq \pi$ ,  $\forall t \in (0, t_f]$ , provided that  $e_{r_j}(0) \neq 0$ ,  $\forall j \in \{1, \dots, n_r\}$ . To do that, we define the mapping

$$\mathsf{H}(x, y) := \left[ \tan\left(\frac{x_1 - y_1}{2}\right), \dots, \tan\left(\frac{x_{n_r} - y_{n_r}}{2}\right) \right]^\top \in \left(-\frac{\pi}{2}, \frac{\pi}{2}\right)^{n_r} \quad (6.5)$$

for vectors  $x = [x_1, \dots, x_{n_r}]^\top$ ,  $y = [y_1, \dots, y_{n_r}]^\top \in [0, 2\pi)^{n_r}$ , as well as the signal  $\eta_r := \mathsf{H}(q_r, q_{d,r})$ . Note that  $\eta_r$  is not defined when  $e_{r_j} = \pi$  for some  $j \in \{1, \dots, n_r\}$ , which we exploit in the control design.

We define first the reference signals for  $\dot{q}_{tr}, \dot{q}_r$  as  $\alpha_q := [\alpha_{tr}^\top, \alpha_r^\top]^\top$ , with

$$\alpha_{tr} := -K_{tr}e_{tr} + \dot{q}_{d,tr}, \quad (6.6a)$$

$$\alpha_r := \begin{bmatrix} \alpha_{r_1} \\ \vdots \\ \alpha_{r_{n_r}} \end{bmatrix} := \begin{bmatrix} \dot{q}_{d,r_1} - k_{r_1} \cos\left(\frac{e_{r_1}}{2}\right) \sin\left(\frac{e_{r_1}}{2}\right) \\ \vdots \\ \dot{q}_{d,r_{n_r}} - k_{r_{n_r}} \cos\left(\frac{e_{r_{n_r}}}{2}\right) \sin\left(\frac{e_{r_{n_r}}}{2}\right) \end{bmatrix}, \quad (6.6b)$$

where  $K_{tr} \in \mathbb{R}^{n_{tr} \times n_{tr}}$  is a symmetric positive definite gain matrix, and  $k_{r_j} > 0$  are positive gain constants,  $\forall j \in \{1, \dots, n_r\}$ . Define also the associated velocity error

$$e_{v_q} := \dot{q} - \alpha_q,$$

and the estimate  $\hat{\theta} \in \mathbb{R}^l$  of  $\theta$ , as well as the error  $e_\theta := \hat{\theta} - \theta \in \mathbb{R}^l$ .

Let now  $\mathcal{R}_j$  be defined as

$$\mathcal{R}_j := \begin{cases} [0, \pi) & \text{if } e_{r_j}(0) \in [0, \pi), \\ (\pi, 2\pi] & \text{if } e_{r_j}(0) \in (\pi, 2\pi], \end{cases}$$

and design the control law as  $\tau : \mathcal{W}_{tr} \times \mathcal{R}_1 \times \dots \mathcal{R}_{n_r} \times \mathbb{R}^6 \times \mathbb{R}^l \rightarrow \mathbb{R}^n$ , with

$$\tau = \tau(e_x, e_{v_q}, \hat{\theta}) := Y_\alpha \hat{\theta} - K_v e_{v_q} - e_x, \quad (6.7)$$

where  $Y_\alpha := Y(q, \dot{q}, \alpha_q, \dot{\alpha}_q)$ , with  $Y(\cdot)$  as given in (6.4), the signal  $e_x$  is defined as

$$e_x := \left[ e_{tr}, \frac{\tan\left(\frac{e_{r_1}}{2}\right)}{\cos\left(\frac{e_{r_1}}{2}\right)^2}, \dots, \frac{\tan\left(\frac{e_{r_{n_r}}}{2}\right)}{\cos\left(\frac{e_{r_{n_r}}}{2}\right)^2} \right]^\top$$

and  $K_v \in \mathbb{R}^{n \times n}$  is a symmetric and positive definite gain matrix. Moreover, design the evolution of  $\hat{\theta}$  as

$$\dot{\hat{\theta}} = -\Gamma \left( \frac{1}{\underline{k}_v} Y_\alpha^\top e_{v_q} + \sigma_\theta \hat{\theta} \right), \quad (6.8)$$

with any initial condition  $\hat{\theta}(0) \in \mathbb{R}^\ell$ ,  $\underline{k}_v := \lambda_{\min}(K_v)$ , and  $\Gamma \in \mathbb{R}^{\ell \times \ell}$  being a symmetric positive definite gain matrix, and  $\sigma_\theta$  a positive constant. Note that the control law (6.7) is well-defined when  $e_x \in \mathcal{R}_j$ , since  $e_{r_j} \neq \pi$ ,  $\forall j \in \{1, \dots, n_r\}$ .

The correctness of the aforementioned control scheme is proven in the subsequent theorem.

**Theorem 6.1.** Consider the dynamics (6.3), a reference trajectory  $q_d : [0, t_f] \rightarrow \mathbb{T}$ , as well as the constant

$$V_0 := \frac{1}{2} \|e_{tr}(0)\|^2 + \|\eta_r(0)\|^2 + \frac{1}{2\underline{k}_v} e_{v_q}(0)^\top B(q(0)) e_{v_q}(0) + \frac{1}{2} e_\theta(0)^\top \Gamma^{-1} e_\theta(0),$$

Then, if  $e_{r_j}(0) \neq \pi$ ,  $\forall j \in \{1, \dots, n_r\}$ , the control protocol (6.7)-(6.8) guarantees that

$$\|e_{tr}(t)\| \leq \bar{e}_{tr} := \max \left\{ 2V_0, \sqrt{\frac{d_x}{\underline{k}_{tr}}} \right\}, \quad \|\eta_r(t)\| \leq \bar{\eta}_r := \max \left\{ V_0, \sqrt{\frac{d_x}{\underline{k}_r}} \right\}, \quad (6.9a)$$

$\forall t \in [0, t_f]$ , where  $d_x$  is a positive constant satisfying  $d_x \geq \frac{d^2}{2\underline{k}_v^2} + \frac{\sigma_\theta}{2} \|\theta\|^2$ , and  $\underline{k}_{tr} := \lambda_{\min}(K_{tr})$ ,  $\underline{k}_r := \min\{k_{r_1}, \dots, k_{r_{n_r}}\}$ . Moreover, it holds that  $e_{r_j}(t) \neq \pi$ ,  $\forall j \in \{1, \dots, n_r\}$ , and all closed-loop signals remain bounded, for all  $t \in [0, t_f]$ .

*Proof.* Let  $x_R := [e_{tr}^\top, \eta_r^\top, e_{v_q}^\top, e_\theta]^\top$  and consider the candidate Lyapunov function

$$V(x_R) := \frac{1}{2} \|e_{tr}\|^2 + \|\eta_r\|^2 + \frac{1}{2\underline{k}_v} e_{v_q}^\top B(q) e_{v_q} + \frac{1}{2} e_\theta^\top \Gamma^{-1} e_\theta,$$

Since  $e_{r_j}(0) \neq \pi$ ,  $V(x_R(0))$  is bounded by a constant  $V(x_R(0)) \leq \bar{V}$ .

Differentiation of  $V$  yields

$$\begin{aligned} \dot{V} &= e_{tr}^\top (\dot{q}_{tr} - \dot{q}_{d,tr}) + \sum_{j \in \{1, \dots, n_r\}} \frac{\tan\left(\frac{e_{r_j}}{2}\right)}{\cos\left(\frac{e_{r_j}}{2}\right)^2} (\dot{q}_{r_j} - \dot{q}_{d,r_j}) + \frac{1}{\underline{k}_v} e_{v_q}^\top (\tau - C_q \dot{q} \\ &\quad - g_q - d) + \frac{1}{2\underline{k}_v} e_{v_q}^\top \dot{B} e_{v_q} - \frac{1}{\underline{k}_v} e_{v_q}^\top B \dot{\alpha}_q + e_\theta^\top \Gamma^{-1} \dot{\hat{\theta}}, \end{aligned}$$

which, by substituting  $\dot{q} = e_{v_q} + \alpha_q$ , becomes

$$\begin{aligned}\dot{V} = & -e_{tr}^\top K_{tr} e_{tr} - \eta_r^\top K_r \eta_r + \frac{1}{2\underline{k}_v} e_{v_q}^\top \dot{B} e_{v_q} - \frac{1}{\underline{k}_v} e_{v_q}^\top C_q e_{v_q} + \\ & \frac{1}{\underline{k}_v} e_{v_q}^\top (\tau + e_x - B\dot{\alpha}_q - C_q \alpha_q - g_q - d) + e_\theta^\top \Gamma^{-1} \dot{\hat{\theta}},\end{aligned}$$

where  $K_r := \text{diag}\{k_{r_1}, \dots, k_{r_{n_r}}\}$ . By using the skew symmetric property of  $\dot{B} - 2C$  and the dynamics' linear parameterization (6.4), we obtain

$$\dot{V} = -e_{tr}^\top K_{tr} e_{tr} - \eta_r^\top K_r \eta_r + \frac{1}{\underline{k}_v} e_{v_q}^\top (u + e_x - Y_\alpha \theta - d) + e_\theta^\top \Gamma^{-1} \dot{\hat{\theta}},$$

and by substituting  $\tau$  and  $\dot{\hat{\theta}}$ , as well as using  $\|d(\cdot)\| \leq \bar{d}$ ,

$$\begin{aligned}\dot{V} \leq & -e_{tr}^\top K_{tr} e_{tr} - \eta_r^\top K_r \eta_r - \frac{1}{\underline{k}_v} e_{v_q}^\top K_v e_{v_q} + \frac{1}{\underline{k}_v} e_{v_q}^\top Y_\alpha e_\theta + \frac{1}{\underline{k}_v} \|e_{v_q}\| \bar{d} - \\ & \frac{1}{\underline{k}_v} e_\theta^\top Y_\alpha^\top e_{v_q} - \sigma_\theta e_\theta^\top \hat{\theta} \\ \leq & -e_{tr}^\top K_{tr} e_{tr} - \eta_r^\top K_r \eta_r - \frac{1}{\underline{k}_v} e_{v_q}^\top K_v e_{v_q} + \frac{1}{\underline{k}_v} \|e_{v_q}\| \bar{d} - \sigma_\theta e_\theta^\top \hat{\theta} \\ \leq & -\underline{k}_{tr} \|e_{tr}\|^2 - \underline{k}_r \|\eta_r\|^2 - \|e_{v_q}\|^2 + \frac{1}{\underline{k}_v} \|e_{v_q}\| \bar{d} - \sigma_\theta \|e_\theta\|^2 - \sigma_\theta e_\theta^\top \hat{\theta}.\end{aligned}$$

Next, by using the identity  $\alpha\beta \leq \frac{1}{2}\alpha^2 + \frac{1}{2}\beta^2$ ,  $\forall \alpha, \beta \in \mathbb{R}$ , we obtain

$$\begin{aligned}\dot{V} \leq & -\underline{k}_{tr} \|e_{tr}\|^2 - \underline{k}_r \|\eta_r\|^2 - \frac{1}{2} \|e_{v_q}\|^2 - \frac{\sigma_\theta}{2} \|e_\theta\|^2 + \frac{\bar{d}^2}{2\underline{k}_v^2} + \frac{\sigma_\theta}{2} \|\theta\|^2 \\ \leq & -\underline{k}_{tr} \|e_{tr}\|^2 - \underline{k}_r \|\eta_r\|^2 - \frac{1}{2} \|e_{v_q}\|^2 - \frac{\sigma_\theta}{2} \|e_\theta\|^2 + d_x.\end{aligned}$$

Therefore,  $\dot{V}$  is negative when  $\|e_{tr}\| \geq \sqrt{\frac{d_x}{\underline{k}_{tr}}}$ , or  $\|\eta_r\| \geq \sqrt{\frac{d_x}{\underline{k}_r}}$ , or  $\|e_{v_q}\| \geq \sqrt{2d_x}$ , or  $\|e_\theta\| \geq \sqrt{\frac{2d_x}{\sigma_\theta}}$  and hence we conclude that there exists a finite  $T$  such that the state is ultimately bounded as

$$x_R(t) \in \Omega_x := \left\{ x_R \in \mathbb{R}^{2n+\ell} : \|e_{tr}(t)\| \leq \sqrt{\frac{d_x}{\underline{k}_{tr}}}, \quad \|\eta_r(t)\| \leq \sqrt{\frac{d_x}{\underline{k}_r}}, \quad \|e_{v_q}(t)\| \leq \sqrt{2d_x}, \quad \|e_\theta(t)\| \leq \sqrt{\frac{2d_x}{\sigma_\theta}} \right\}$$

for all  $t \geq T$ . Since, outside  $\Omega_x$  it holds that  $\dot{V} < 0$ , we obtain that  $x \notin \Omega_x \Rightarrow V(x(t)) \leq V_0 := V(x(0))$ ,  $\forall t \geq 0$ , i.e.,  $\|e_{tr}(t)\| \leq 2V_0$ ,  $\|\eta_r(t)\| \leq$

$V_0$ ,  $\forall t \geq 0$ . Therefore, we conclude that  $\|e_{tr}(t)\| \leq \max\{2V_0, \sqrt{d_x/\underline{k}_{tr}}\}$ ,  $\|\eta_{tr}(t)\| \leq \max\{2V_0, \sqrt{d_x/\underline{k}_r}\}$ ,  $\forall t \geq 0$ . Finally, since  $V(t)$  remains bounded  $\forall t \in [0, t_f]$ , we conclude that all the closed loop signals remain bounded and  $\cos(e_{r_j}(t)) \neq \pi$ ,  $\forall t \in [0, t_f]$ ,  $j \in \{1, \dots, n_r\}$ .  $\square$

Note that the disturbance term  $d(\cdot)$  prohibits the system from achieving asymptotic convergence, i.e.,  $\lim_{t \rightarrow \infty} (q(t) - q_d(t)) = 0$ . Nevertheless, Theorem 6.1 establishes a funnel around the desired trajectory  $q_d$  where the state  $q(t)$  will evolve in. This funnel will be used as clearance in the motion planner of the subsequent section to derive a collision-free path to the goal region. Note however, that this funnel cannot be accurately known by the user/designer, since  $V_0$  cannot be accurately known (the terms  $B(q(0))$  and  $e_\theta(0)$  contain the unknown terms  $\theta$ ). Lower and upper bounds of  $\theta$  can be obtained, however, since these involve mass and moments of inertia, which can be estimated by the geometry and the material of the links/motors. Hence, one can obtain an upper bound on  $V_0$ . On the same note, a conservative estimate of  $d_x$ , appearing in (6.9), can be obtained by estimating an upper bound of  $d(\cdot)$  (e.g., by testing suitable trajectories on the robot) and using the aforementioned upper bound of  $\theta$ . Therefore, we can obtain an overestimate of the bounds in (6.9), which will be used in the motion planner of the next section. These bounds can be tightened by appropriate tuning of the gain constants, as elaborated in the next remark.

**Remark 6.2.** The collision-free geometric trajectory  $q_d$  of the motion planner will connect the initial condition  $q(0)$  to the goal and hence it is reasonable to enforce  $q_a(0) = q(0)$ . By also reasonably assuming that  $\dot{q}(0) = 0$ ,  $V_0$  from Theorem 6.1 becomes  $V_0 = \frac{1}{2\underline{k}_v} \dot{q}_d(0)B(q(0))\dot{q}_d(0) + \frac{1}{2}e_\theta(0)^\top \Gamma^{-1}e_\theta(0)$ , which can be rendered arbitrarily small by choosing large values for the control gains  $\underline{k}_v$  and  $\Gamma$ . In the same vein, choosing large values for  $\underline{k}_v$ ,  $\underline{k}_{tr}$ , and  $\underline{k}_r$  shrinks the constants  $\sqrt{\frac{d_x}{\underline{k}_{tr}}}$  and  $\sqrt{\frac{d_x}{\underline{k}_r}}$ , respectively. Therefore, the size of the funnel dictated by (6.9) can become smaller by appropriate gain tuning. This will lead to less conservative solutions for the motion planner of the next section, as will be clarified in the next subsection. Nevertheless, it should be noted that too large gains might result in excessive control inputs that cannot be realized by the actuators in realistic systems. Finally, note that the incorporation of  $\underline{k}_v$  in the adaptation law (6.8), which is not common in standard adaptive control techniques, has been included to create an extra degree of freedom for reducing the value of  $V_0$ . This, along with the tracking using the chordal metric  $d_C$  for the rotation part, constitute the differences of the proposed control scheme with respect to standard adaptive control for uncertain robotic systems.

### 6.3.3 Motion Planner

We describe here the construction of the sampling-based motion planner, referred to as Bounded-RRT or B-RRT, that drives the robot from an initial state to the goal, which follows similar steps as the standard geometric RRT algorithm. Before presenting the algorithm, we define the extended-free space, which will be used to integrate the results from the feedback control of the previous subsection. In order to do that, we define first the open polyhedron as

$$\mathcal{P}(q, \delta) := \{y = [y_{tr}^\top, y_r^\top]^\top \in \mathbb{T} : \|y_{tr} - q_{tr}\| < \delta_{tr}, \|\mathsf{H}(y_r, q_r)\| < \delta_r\}, \quad (6.10)$$

for  $q = [q_{tr}^\top, q_r^\top]^\top \in \mathbb{T}$  and  $\delta = (\delta_{tr}, \delta_r) \in \mathbb{R}^2$ , where  $\mathsf{H}(\cdot)$  is the metric introduced in (6.5). We define now the  $\delta$ -extended free space  $\bar{\mathcal{A}}_{\text{free}}(\delta) := \{q \in \mathbb{T} : \bar{\mathcal{A}}(q, \delta) \cap \mathcal{O} = \emptyset\}$ , where  $\bar{\mathcal{A}}(q, \delta) := \bigcup_{x \in \mathcal{P}(q, \delta)} \mathcal{A}(x)$ . Note that  $\bar{\mathcal{A}}_{\text{free}}((\delta_{1_{tr}}, \delta_{1_r})) \subseteq \bar{\mathcal{A}}_{\text{free}}((\delta_{2_{tr}}, \delta_{2_r}))$  if  $\delta_{1_{tr}} \geq \delta_{2_{tr}}$  and/or  $\delta_{1_r} \geq \delta_{2_r}$ .

**Remark 6.3.** Since  $\mathcal{A}_{\text{free}}$  is open, there exist positive constants  $\delta_{tr}, \delta_r$  such that  $Q_g \subset \bar{\mathcal{A}}_{\text{free}}((\delta_{tr}, \delta_r))$  and the feasible path  $\mathbf{q}_p$  from Assumption 6.2 satisfies  $\mathbf{q}_p(\nu) \in \bar{\mathcal{A}}_{\text{free}}((\delta_{tr}, \delta_r)), \forall \nu \in [0, \sigma]$ .

The control scheme of the previous subsection guarantees that the robot can track a trajectory within the bounds (6.9). In other words, given a desired trajectory signal  $q_d : [t_0, t_f] \rightarrow \mathbb{T}$ , the control algorithm (6.7) - (6.8) guarantees that  $q(t) \in \bar{\mathcal{A}}_{\text{free}}((\bar{e}_{tr}, \bar{\eta}_r)), \forall t \in [t_0, t_f]$ , with  $\bar{e}_{tr}, \bar{\eta}_r$  as defined in (6.9). Hence, the motion planner developed here takes that into account by producing trajectories that belong to the extended free space  $\bar{\mathcal{A}}_{\text{free}}((\bar{e}_{tr}, \bar{\eta}_r))$ <sup>2</sup>. The respective algorithm is presented in Algorithm 2. It is a variant of the standard RRT algorithm. The main difference, which constitutes the key point of the algorithm, is the procedure that aims to find a collision-free trajectory from a node on the tree towards the sampled point. In particular, the sampling of new nodes-points as well as the collision checker of the path between two nodes are carried out with respect to the extended free space  $\bar{\mathcal{A}}_{\text{free}}((\bar{e}_{tr}, \bar{\eta}_r))$ . Moreover, the motion planner does not need to integrate the system dynamics (6.3) with sampled input values in order to design a feasible and collision-free robot trajectory. Instead, we use the established evolution funnel to design a collision-free trajectory for the robot, without involving the dynamics. This implies that the motion planner is purely geometrical.

The functions that appear in Algorithm 2 are the following:

- **Sample( $i$ ):** Samples  $q_{\text{rand}}$  from a uniform distribution in the extended free space  $\bar{\mathcal{A}}_{\text{free}}((\bar{e}_{tr}, \bar{\eta}_r))$ , where  $\bar{e}_{tr}, \bar{\eta}_r$  are the constants from (6.9)

---

<sup>2</sup>We keep the same notation  $(\bar{e}_{tr}, \bar{\eta}_r)$ , although only upper bounds of these values can be actually estimated and hence used by the planner

**Algorithm 2** B-RRT

---

```

1: procedure TREE
2:    $\mathcal{V} \leftarrow \{q(0)\}; \mathcal{E} \leftarrow \emptyset; i \leftarrow 0$ 
3:   while  $i < N_s$  do
4:      $\mathcal{G} \leftarrow (\mathcal{V}, \mathcal{E});$ 
5:      $q_{\text{rand}} \leftarrow \text{Sample}(i); i \leftarrow i + 1;$ 
6:      $q_{\text{nearest}} \leftarrow \text{Nearest}(\mathcal{G}, q_{\text{rand}});$ 
7:      $q_{\text{new}} \leftarrow \text{Steer}(q_{\text{nearest}}, q_{\text{rand}});$ 
8:     if  $\text{ObstacleFree}(q_{\text{nearest}}, q_{\text{new}})$  then
9:        $V \leftarrow V \cup \{q_{\text{new}}\}; \mathcal{E} \leftarrow \mathcal{E} \cup \{(q_{\text{nearest}}, q_{\text{new}})\};$ 

```

---

that define the funnel polyhedron  $\mathcal{P}(q_d(t), (\bar{e}_{tr}, \bar{\eta}_r))$  (see (6.10)) around a reference trajectory  $q_d(t)$  that  $q(t)$  can evolve in.

- $\text{Nearest}(\mathcal{G}, q)$ : Finds the node  $q_{\text{nearest}}$  in the tree such that  $d_{\mathbb{T}}(q_{\text{nearest}}, q) = \min_{z \in \mathcal{V}} d_{\mathbb{T}}(z, q)$ .
- $\text{Steer}(q, z)$ : Computes a point  $q_{\text{new}}$  lying on the straight line from  $z$  to  $q$  such that  $d_{\mathbb{T}}(q, q_{\text{new}}) = \epsilon$ , where  $\epsilon$  is a tuning constant that represents the incremental distance from  $q$  to  $q_{\text{new}}$ .
- $\text{ObstacleFree}(q, z)$ : Checks whether the path  $X_{\text{Line}} : [0, \sigma] \rightarrow \mathbb{T}$ , for some positive  $\sigma$ , from  $q$  to  $z$  is collision free with respect to the extended free space, i.e., check whether  $q' \in \bar{\mathcal{A}}_{\text{free}}((\bar{e}_{tr}, \bar{\eta}_r)), \forall q' \in X_{\text{Line}}$ .

The difference hence of B-RRT with respect to the standard RRT algorithm is the use of the extended free space  $\bar{\mathcal{A}}_{\text{free}}((\bar{e}_{tr}, \bar{\eta}_r))$  in the procedures of sampling new points (function `Sample`) and checking collisions of the path between two nodes (function `ObstacleFree`). As stated before, this stems from the control design of the previous section, which guarantees that the robot trajectory will evolve in  $\bar{\mathcal{A}}_{\text{free}}((\bar{e}_{tr}, \bar{\eta}_r))$  with respect to a desired trajectory  $q_d$ .

We briefly describe now the B-RRT algorithm. The tree  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  to be constructed is initialized in line 2, with the node set  $\mathcal{V}$  initialized to the system's initial configuration  $q(0)$ , and the respective edge set  $\mathcal{E}$  to the empty set. The algorithm samples then a point  $q_{\text{rand}}$  in the extended free space  $\bar{\mathcal{A}}_{\text{free}}((\bar{e}_{tr}, \bar{\eta}_r))$ . Then the nearest neighbor  $q_{\text{nearest}}$ , in terms of  $d_{\mathbb{T}}$ , is found in the tree (line 7), and a new point  $q_{\text{new}}$  on the line between  $q_{\text{nearest}}$  and  $q_{\text{rand}}$  is computed;  $q_{\text{new}}$  can be chosen such that  $d_{\mathbb{T}}(q_{\text{nearest}}, q_{\text{new}}) = \epsilon$ , according to a predefined incremental distance  $\epsilon$  ([?]). If the line segment between  $q_{\text{rand}}$  and  $q_{\text{new}}$  belongs to the extended free space  $\bar{\mathcal{A}}_{\text{free}}((\bar{e}_{tr}, \bar{\eta}_r))$ , then the respective node  $q_{\text{new}}$  and edge  $\{q_{\text{nearest}}, q_{\text{new}}\}$  are added to the tree (lines 9-11). After the execution of the algorithm, a standard search

algorithm can be employed to find the sequence of edges that lead from  $q(0)$  to  $Q_g$ . Moreover, note that the bounds in (6.9) concern (at least twice) continuously differentiable trajectories. Therefore, the resulting solution path, which is formed by the concatenation of the respective edges in the tree, has to be converted to a such a trajectory. This procedure might modify the initial path that was checked for collisions, and hence the smooth version should be re-checked for collisions in  $\bar{\mathcal{A}}_{\text{free}}((\bar{e}_{tr}, \bar{\eta}_r))$ . Subsequently, the resulting smooth (at least twice cont. different.) path is endowed with time constraints to derive a timed trajectory  $q_d : [0, t_f] \rightarrow \bar{\mathcal{A}}_{\text{free}}((\bar{e}_{tr}, \bar{\eta}_r))$ , for some  $t_f > 0$ , which is given as the desired trajectory input to the control protocol designed in the previous section. The actual trajectory of the system  $q(t)$  is guaranteed to track  $q_d(t)$  in the funnel defined by  $\bar{e}_{tr}, \bar{\eta}_r$ . Since these bounds are taken into account in the design of the trajectory  $q_d$  by Algorithm 2, the system will remain collision free. Note also that  $t_f$  and hence the velocity of the formed trajectory  $q_d$  is chosen by the user. Therefore, the robot can execute the respective path in a predefined time interval.

The probabilistic completeness of the algorithm is stated in the next theorem.

**Theorem 6.2.** *Under Assumption 6.2 and for sufficiently high gains  $k_v, k_{tr}, k_r$ , as introduced in eq. (6.6), (6.7) and (6.8), Algorithm 2 is probabilistically complete.*

*Proof.* Assumption 6.2 and Remark 6.3 imply that there exist positive  $\delta_{tr}$  and  $\delta_r$  and (at least) one twice differentiable path  $q_p : [0, \sigma] \rightarrow \bar{\mathcal{A}}_{\text{free}}((\delta_{tr}, \delta_r))$  connecting  $q_0$  and  $Q_g$ . As stated in Remark 6.2, by increasing the values of the control gains  $k_v, k_{tr}, k_r$ , one can decrease the constants  $\bar{e}_{tr}, \bar{\eta}_{tr}$  from eq. (6.9) such that  $\bar{e}_{tr} < \delta_{tr}, \bar{\eta}_r < \delta_r$ . Hence,  $Q_g$  satisfies  $Q_g \subset \bar{\mathcal{A}}_{\text{free}}((\delta_{tr}, \delta_r)) \subset \bar{\mathcal{A}}_{\text{free}}((\bar{e}_{tr}, \bar{\eta}_r))$  and the feasible path satisfies  $q_p(\nu) \in \bar{\mathcal{A}}_{\text{free}}((\delta_{tr}, \delta_r)) \subset \bar{\mathcal{A}}_{\text{free}}((\bar{e}_{tr}, \bar{\eta}_r)), \forall \nu \in [0, \sigma]$ , which guarantees the feasibility of Algorithm 2.

Next, by following similar arguments with Lemma 2 of [?], one can prove that for any  $q \in \bar{\mathcal{A}}_{\text{free}}((\bar{e}_{tr}, \bar{\eta}_r))$  and  $\epsilon > 0$ , it holds that  $\lim_{i \rightarrow \infty} \mathbb{P}(D_{i,q} < \epsilon)$ , where  $D_{i,q}$  is the random variable associated with the minimum distance of the tree  $\mathcal{G}$  to the point  $q$  (in terms of  $d_{\mathbb{T}}$ ) after iteration  $i$ , and  $\mathbb{P}$  denotes the probability. Hence, the vertices  $\mathcal{V}$  of  $\mathcal{G}$  converge to the sampling distribution in  $\bar{\mathcal{A}}_{\text{free}}((\bar{e}_{tr}, \bar{\eta}_r))$ , which is assumed to be uniform. Therefore, since  $q_p$  lies in  $\bar{\mathcal{A}}_{\text{free}}((\bar{e}_{tr}, \bar{\eta}_r))$ , a subset of  $\mathcal{V}$  converges to it and the proof follows.  $\square$

### 6.3.4 Collision Checking in $\bar{\mathcal{A}}_{\text{free}}(\bar{e}_{tr}, \bar{\eta}_r)$

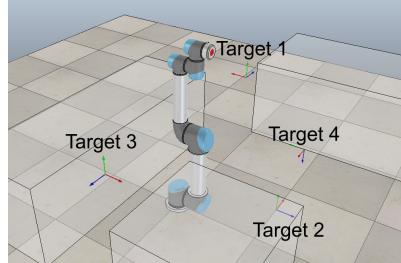
Collision checking for points  $q \in \mathbb{T}$  in variants of the standard RRT algorithm is performed by checking whether  $q$  belongs to  $\mathcal{A}_{\text{free}}$  or not. For a line segment

$X_{\text{Line}}$  connecting two nodes of  $\mathcal{V}$ , the latter is usually discretized into a finite set of points, which are checked separately for collision (e.g., [? ?]). Another approach is to consider an over-approximation of the convex hull of the points that form  $X_{\text{Line}}$  ([? ]). In our case, however, we are interested in checking for collisions in the *extended* free space  $\bar{\mathcal{A}}_{\text{free}}(\bar{e}_{tr}, \bar{\eta}_r)$ . Recall that the proposed feedback control scheme guarantees that  $q(t) \in \mathcal{P}(q_d(t), (\bar{e}_{tr}, \bar{\eta}_r))$  for any trajectory  $q_d(t)$ , formed by the several line segments  $X_{\text{Line}}$  that connect the nodes in  $\mathcal{V}$  sampled in Algorithm 2. Therefore, checking whether the points  $q_s \in X_{\text{Line}}$  belong to  $\mathcal{A}_{\text{free}}$  is not sufficient. That is, for each such point  $q_s \in X_{\text{Line}}$ , one must check whether  $z \in \mathcal{A}_{\text{free}}, \forall z \in \mathcal{P}(q_s, (\bar{e}_{tr}, \bar{\eta}_r))$ , which is equivalent to checking if  $q_s \in \bar{\mathcal{A}}_{\text{free}}(\bar{e}_{tr}, \bar{\eta}_r)$ . There are two procedures that one can use for that. Firstly, for each  $q_s$ , a finite number of points  $z$  can be sampled from a uniform distribution in  $\mathcal{P}(q_s, (\bar{e}_{tr}, \bar{\eta}_r))$  and separately checked for collision. Then, for a sufficiently high number of such samples, and assuming a certain “fat”-structure of the workspace obstacles (e.g., there are no long and skinny obstacles such as wires, cables and tree branches, etc., see ([? ]) for more details), this approach can be considered to be complete, i.e., the resulting path will belong to the extended free space  $\bar{\mathcal{A}}_{\text{free}}$ . Secondly, we calculate the limit poses of each link of the robot, based on the lower and upper bounds by the joints that affect it, as defined by  $(\bar{e}_{tr}, \bar{\eta}_r)$ . Subsequently, we compute the convex hull of these limit poses, which is expanded by an appropriate constant to yield an over-approximation of the swept volume of the potential motion of the link, as described in [? ]. The resulting shape is then checked for collisions for each link separately.

### 6.3.5 Experimental Results

This section presents experimental results for a UR5 robot, which consists of 6 rotational degrees of freedom (see Fig. 6.3), using the V-REP environment ([? ]). We assume that the first joint is free to move on the unit circle, i.e.,  $q_{r_1} \in [0, 2\pi]$ , whereas the rest of the joints are restricted to  $[-\pi, \pi]$  to avoid problematic configurations. We consider that the robot end-effector has to sequentially navigate from its initial configuration  $q_0 = [0, 0, 0, 0, 0, 0]^\top$  rad to the following four target points (depicted in Fig. 6.3):

- Target 1:  $T_1 = [-0.15, -0.475, 0.675]^\top$  m and orientation  $[\frac{\pi}{2}, 0, 0]^\top$  rad, which yields the configuration  $q_1 = [-0.07, -1.05, 0.45, 2.3, 1.37, -1.33]^\top$  rad.
- Target 2:  $T_2 = [-0.6, 0, 2.5]^\top$  m and orientation  $[0, -\frac{\pi}{2}, -\frac{\pi}{2}]^\top$  rad, which yields the configuration  $q_2 = [1.28, 0.35, 1.75, 0.03, 0.1, -1.22]^\top$  rad.



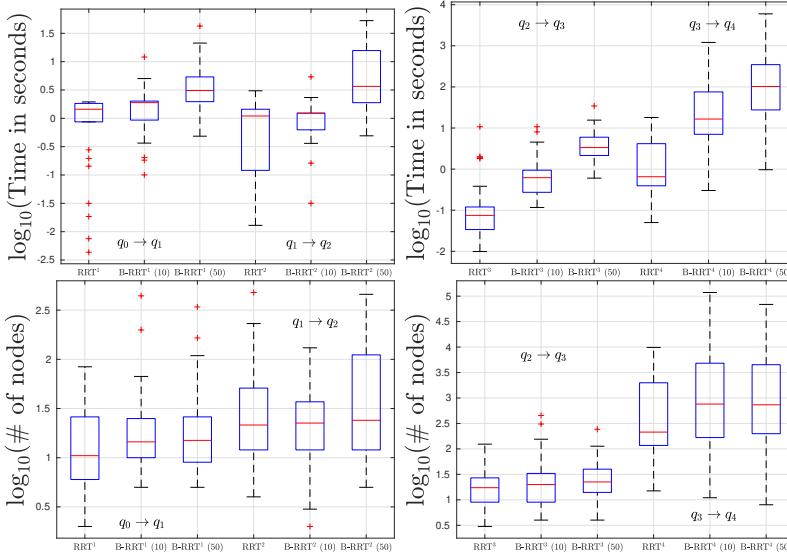
**Figure 6.3:** A UR5 robotic arm in an obstacle-cluttered environment with 4 targets.

- Target 3:  $T_3 = [-0.025, 0.595, 0.6]^\top$  m and orientation  $[-\frac{\pi}{2}, 0, \pi]^\top$  rad, which yields the configuration  $q_3 = [-0.08, 0.85, -0.23, 2.58, 2.09, -2, 36]^\top$  rad.
- Target 4:  $T_4 = [-0.525, -0.55, 0.28]^\top$  m and orientation  $[\pi, 0, -\frac{\pi}{2}]^\top$  rad, which yields the configuration  $q_4 = [-0.7, -0.76, -1.05, -0.05, -3.08, 2.37]^\top$  rad.

Regarding the collision checking in  $\bar{\mathcal{A}}_{\text{free}}(\bar{e}_{tr}, \bar{\eta}_r)$  of the B-RRT algorithm, we check a finite number of samples around each point of the resulting trajectory  $q_d$  for collision. We run B-RRT with 10 and 50 such samples and we compared the results to a standard geometric RRT algorithm in terms of time per number of nodes. The results for 30 runs of the algorithms are given in Fig. 6.4 for the four paths, in logarithmic scale. One can notice that the average nodes created do not differ significantly among the different algorithms. As expected, however, B-RRT requires more time than the standard geometric RRT algorithm, since it checks the extra samples in  $\bar{\mathcal{A}}_{\text{free}}(\bar{e}_{tr}, \bar{\eta}_r)$  for collision. One can also notice that the time increases with the number of samples. However, more samples imply greater coverage of  $\bar{\mathcal{A}}_{\text{free}}(\bar{e}_{tr}, \bar{\eta}_r)$  and hence the respective solutions are more likely to be complete with respect to collisions.

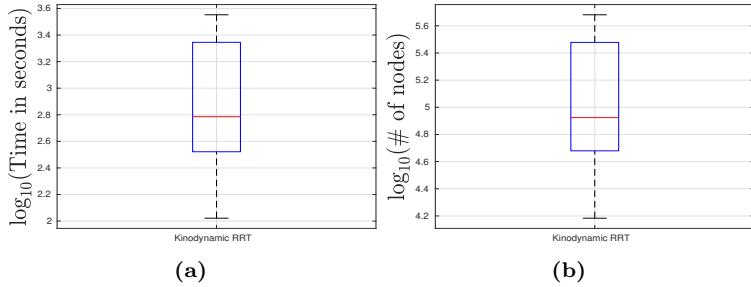
Since, in contrast to the standard geometric RRT, B-RRT implicitly takes into account the robot dynamics through the designed tracking control scheme and the respective extended free space  $\bar{\mathcal{A}}_{\text{free}}(\bar{e}_{tr}, \bar{\eta}_r)$ , we compare the results to a standard kinodynamic RRT algorithm that simulates forward the robot dynamics, assuming known dynamical parameters. In particular, we run the algorithm only for the first two joints, with initial and goal configurations at  $(0, 0)$  and  $(-\frac{\pi}{18}, \frac{\pi}{4})$  rad, respectively, and keep the other joints fixed at 0. For the forward simulation of the respective dynamics we choose a sampling step of  $10^{-3}$  sec and total simulation time 30 sec for each constant control input. The termination threshold distance is set to

0.25 (with respect to the distance  $d_{\mathbb{T}}$ ), i.e., the algorithm terminates when the forward simulation reached a configuration closer than 0.25 units to the goal configuration. The results for 10 runs of the algorithm are depicted in Fig. 6.5, which provides the execution time and number of nodes created in logarithmic scale. Note that, even for this simple case (planning for only two joints), the execution time is comparable to the B-RRT case of 50 samples in the fourth path scenario  $q_3 \rightarrow q_4$ . As pointed out before, this is justified by the fact that the inputs are randomized as well as the complex dynamics of the considered robotic system. Hence, one concludes the necessity of an efficient technique that still takes into the robotic dynamics, which is given by our two-layer framework, combining an appropriately designed RRT planner with an “intelligent” feedback control algorithm that also compensates for the uncertain dynamics.



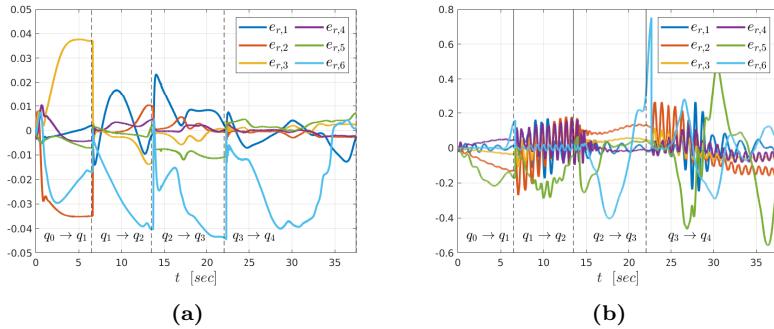
**Figure 6.4:** Box plots showing the execution time (top) and the nodes (bottom) created of the three algorithms (in logarithmic scale) for the four paths (organized in two groups of two (left and right)); ‘+’ indicate the outliers.

Next, we illustrate the motion of the robot through the four target points via the control design of Section 6.3.2. For each sub-path ( $q_i \rightarrow q_{i+1}$ ,  $\forall i \in \{0, 1, 2, 3\}$ ) we fit a smooth timed trajectory  $q_d^i(t)$ ,  $\forall i \in \{0, \dots, 3\}$  on the generated nodes, whose total time duration depends on the distance between successive nodes. The estimates of the masses and inertias of the robot links and rotors, composing  $\theta$ , were initialized at 60% of the actual values.



**Figure 6.5:** Box plots showing the execution time (a) and number of nodes (b) created for the kinodynamic RRT in logarithmic scale (for the first two joints and the path  $(0, 0) \rightarrow (-\frac{\pi}{18}, \frac{\pi}{4})$ ).

Moreover, in view of (6.9), we aim to impose an upper bound of 0.1 rad for each  $|q_{r_j} - q_{d,j}^i|$ ,  $\forall j \in \{1, \dots, 6\}$ ,  $i \in \{0, \dots, 3\}$ . To that end, we choose the control gains as  $k_{r_1} = \dots = k_{r_6} = 0.005$ ,  $K_v = \text{diag}\{[35, 65, 45, 20, 10, 0.5]\}$ , and  $\Gamma = 50\text{diag}\{\hat{\theta}(0)\}$ . The results are depicted in Fig. 6.6(a), which shows the error values  $e_{r_j}(t) = e_{r_j}(t) - q_{d,r_j}^i(t)$ ,  $\forall j \in \{1, \dots, 6\}$ , and all paths  $i \in \{0, \dots, 3\}$ . One can verify that the error values stay always bounded in the region  $(-0.1, 0.1)$  rad, achieving thus the desired performance. For



**Figure 6.6:** The error values  $e_{r_j}(t) = e_{r_j}(t) - q_{d,r_j}^i(t)$  for the adaptive controller (a) and the PID one (b).

comparison purposes, we also simulate a PID controller of the form

$$\tau = -K_1 e_x - K_2(\dot{q} - \dot{q}_d) - K_3 \int e_x(\nu) d\nu,$$

where  $K_1 = \text{diag}\{100, 1000, 1000, 100, 1, 1\}$ , and  $K_2 = K_3 = I_6$  are positive definite gain matrices. The errors  $e_{r_j}(t) = e_{r_j}(t) - q_{d,r_j}^i(t)$ ,  $\forall j \in \{1, \dots, 6\}$ , for the four paths  $i \in \{0, \dots, 3\}$  are shown in Fig. 6.6(b). Note that they

exceed the interval  $(-0.1, 0.1)$ , which defined the clearance in the B-RRT algorithm, jeopardizing hence the actual trajectory of the robot. A video illustrating the robot trajectory using the two control laws can be found here: <https://youtu.be/y7bCoUoTlPA>.

## 6.4 Asymptotic Tracking of Nonsmooth Feedback Stabilizable Unknown Systems with Prescribed Transient Response

Finally, inspired by funnel-based techniques and the PPC methodology [?], we propose in this section a novel control scheme that achieves asymptotic stability while complying with funnel constraints, for a 2nd-order control affine uncertain and possibly non-smooth system.

### 6.4.1 Problem Formulation

We consider the asymptotic tracking control problem subject to transient constraints imposed by a predefined funnel. The consider systems are MIMO systems of the form

$$\dot{x}_1 = x_2, \quad (6.11a)$$

$$\dot{z} = F_z(x, z, t), \quad (6.11b)$$

$$\dot{x}_2 = F(x, z, t) + G(x, z, t)u, \quad y = x_1 \quad (6.11c)$$

where  $z \in \mathbb{R}^{n_z}$ ,  $x := [x_1^\top, x_2^\top]^\top \in \mathbb{R}^{2n}$ , with  $x_j := [x_{j_1}, \dots, x_{j_n}]^\top \in \mathbb{R}^n$ ,  $\forall j \in \{1, 2\}$ , are the system's states,  $y := [y_1, \dots, y_n]^\top \in \mathbb{R}^n$  is the system's output, which is required to track a desired trajectory  $y_d(t)$ , and  $F : \mathbb{R}^{2n+n_z} \times [t_0, \infty) \rightarrow \mathbb{R}^n$ ,  $F_z : \mathbb{R}^{2n+n_z} \times [t_0, \infty) \rightarrow \mathbb{R}^{n_z}$ ,  $G : \mathbb{R}^{2n+n_z} \times [t_0, \infty) \rightarrow \mathbb{R}^{n \times n}$  are unknown vector fields, not necessarily continuous everywhere. We assume that  $x$  is available for measurement, whereas  $z$  is not. In fact, the dynamics governing  $z$  is called dynamic uncertainty and represents unmodeled dynamic phenomena that potentially affect the closed-loop response. The assumptions on the system dynamics are restricted to local essential boundedness and measurability as well as controllability conditions on  $G$  and internal stability of  $z$ , without considering any uniform boundedness/growth condition or model approximation:

**Assumption 6.3.** The maps  $(x, z) \mapsto F(x, z, t) : \mathbb{R}^{2n+n_z} \rightarrow \mathbb{R}^n$ ,  $(x, z) \mapsto G(x, z, t) : \mathbb{R}^{2n+n_z} \rightarrow \mathbb{R}^n$ ,  $(x, z) \mapsto F_z(x, z, t) : \mathbb{R}^{2n+n_z} \rightarrow \mathbb{R}^{n_z}$  are Lebesgue measurable and locally essentially bounded for each fixed  $t \in [t_0, \infty)$ , uniformly in  $t$ , and the maps  $t \mapsto F(x, z, t) : [t_0, \infty) \rightarrow \mathbb{R}^n$  and  $t \mapsto G(x, z, t) :$

$[t_0, \infty) \rightarrow \mathbb{R}^n$  are Lebesgue measurable and uniformly bounded for each fixed  $(x, z) \in \mathbb{R}^{2n+n_z}$ , by unknown bounds.

**Assumption 6.4.** The matrix

$$\tilde{G}(x, z, t) := G(x, z, t) + G(x, z, t)^\top$$

is positive definite,  $\forall (x, z, t) \in \mathbb{R}^{2n+n_z} \times [t_0, \infty)$ , i.e.,  $\lambda_{\min}(\tilde{G}(x, z, t)) > 0$ , where  $\lambda_{\min}(\tilde{G}(x, z, t))$  is its *unknown* minimum eigenvalue.

**Assumption 6.5.** There exists a sufficiently smooth function  $U_z : \mathbb{R}_{\geq 0}^{n_z} \rightarrow \mathbb{R}_{\geq 0}$  and class  $\mathcal{K}_\infty$  functions  $\underline{\gamma}_z(\cdot), \bar{\gamma}_z(\cdot), \gamma_z(\cdot)$  such that  $\underline{\gamma}_z(\|z\|) \leq U_z(z) \leq \bar{\gamma}_z(\|z\|)$ , and

$$\left( \frac{\partial U_z}{\partial z} \right)^\top F_z(x, z, t) \leq -\gamma_z(\|z\|) + \pi_z(x, t),$$

where  $x \mapsto \pi_z(x, t) : \mathbb{R}^{2n} \rightarrow \mathbb{R}_{\geq 0}$  is continuous and class  $\mathcal{K}_\infty$  for each fixed  $t \in [t_0, \infty)$ , and  $t \mapsto \pi_z(x, t) : [t_0, \infty) \rightarrow \mathbb{R}_{\geq 0}$  is uniformly bounded for each fixed  $x \in \mathbb{R}^{2n}$ .

**Assumption 6.6.** The state  $x$  is available for measurement.

**Assumption 6.7.** The desired trajectory and its derivatives are bounded by finite and unknown constants  $\bar{y}_{d,0}, \bar{y}_{d,1} > 0$ , i.e.,  $\|y_d(t)\| < \bar{y}_{d,0} \leq \bar{y}_d$ ,  $\|y_d(t)\| < \bar{y}_{d,1} \leq \bar{y}_d$ ,  $\forall t \in [t_0, \infty)$ , where  $\bar{y}_d := \max\{\bar{y}_{d,0}, \bar{y}_{d,1}\}$ .

Note that Assumption 6.4 is a sufficient controllability condition and Assumption 6.5 suggests that  $z$  is input-to-state practically stable with respect to  $x, t$  implying stable zero (internal) dynamics [? ].

### 6.4.2 Problem Solution

The control objective is the *asymptotic* output tracking of a desired bounded trajectory  $y_d := [y_{1,d}, \dots, y_{n,d}] : [t_0, \infty) \rightarrow \mathbb{R}^n$ , with bounded derivatives, as stated in Assumption 6.7. Moreover, as discussed before, we aim at imposing a certain predefined behavior for the transient response of the system. More specifically, motivated by funnel control techniques (Appendix B, [? ?]), given  $n$  predefined funnels, described by the smooth functions  $\rho_{p_i} : [t_0, \infty) \rightarrow [\underline{\rho}_{p_i}, \bar{\rho}_{p_i}] \subset \mathbb{R}_{>0}$ , where  $\underline{\rho}_{p_i}, \bar{\rho}_{p_i} \in \mathbb{R}_{>0}$  are positive lower and upper bounds, respectively, we aim at guaranteeing that<sup>3</sup>  $-\rho_{p_i}(t) > y_i(t) - y_{i,d}(t) > \rho_{p_i}(t)$ ,  $\forall t \in [t_0, \infty)$ , given that  $-\rho_{p_i}(t_0) > y_i(t_0) - y_{i,d}(t_0) > \rho_{p_i}(t_0)$ ,  $\forall i \in \{1, \dots, n\}$ . These functions can encode maximum overshoot or

---

<sup>3</sup>The analysis can be extended to non-symmetric funnels.

convergence rate properties. Note that, compared to the majority of the related works on funnel control (e.g., [? ? ? ],[?]), we do not require arbitrarily small final values  $\lim_{t \rightarrow \infty} \rho_{p_i}(t)$ , which would achieve convergence of  $y_i(t) - y_{i,d}(t)$  arbitrarily close to zero, since one of the objectives is actual *asymptotic stability*. In this section, the problem statement is as follows:

**Problem 6.3.** Consider the system (6.11) and let a desired trajectory  $y_d : [t_0, \infty) \rightarrow \mathbb{R}^n$  as well as  $n$  prescribed funnels, described by  $\rho_{p_i} : [t_0, \infty) \rightarrow [\underline{\rho}_{p_i}, \bar{\rho}_{p_i}]$ ,  $\forall i \in \{1, \dots, n\}$ . Design a control protocol  $u \in \mathbb{R}^n$  such that

1.  $\lim_{t \rightarrow \infty} (y_i(t) - y_{i,d}(t)) = 0, \forall i \in \{1, \dots, n\}$
2.  $-\rho_{p_i}(t) > y_i(t) - y_{i,d}(t) > \rho_{p_i}(t), \forall i \in \{1, \dots, n\}, t \in [t_0, \infty)$ ,

and all closed loop signals remain bounded.

Our solution to Problem 6.3 is based on the PPC error transformation, which converts the constrained error behavior  $-\rho_{p_i}(t) > y_i(t) - y_{i,d}(t) > \rho_{p_i}(t)$  to an unconstrained one. More specifically, we define the errors

$$e_p := [e_{p_1}, \dots, e_{p_n}]^\top := y - y_d, \quad (6.12)$$

as well as the error transformations  $\varepsilon_{p_i} \in \mathbb{R}$  according to:

$$e_{p_i} = \rho_{p_i} T(\varepsilon_{p_i}), \quad \forall i \in \{1, \dots, n\}, \quad (6.13)$$

where  $T : \mathbb{R} \rightarrow (-1, 1)$  is a smooth, strictly increasing analytic function, with  $T(0) = 0$ . Since  $T$  is increasing, the inverse mapping  $T^{-1} : (-1, 1) \rightarrow \mathbb{R}$  is well-defined, and it holds that

$$\lim_{\zeta \rightarrow -\infty} T(\zeta) = -1, \quad \lim_{\zeta \rightarrow +\infty} T(\zeta) = 1 \quad (6.14a)$$

and hence, if  $\varepsilon_{p_i}$  remains bounded in a compact set, the desired funnel objective  $-\rho_{p_i}(t) < e_{p_i}(t) < \rho_{p_i}(t)$  is achieved,  $\forall i \in \{1, \dots, n\}$ . We further require that

$$|\zeta| < \left| \frac{\partial T^{-1}(\zeta)}{\partial \zeta} T^{-1}(\zeta) \right|, \quad \forall \zeta \in (-1, 1). \quad (6.15)$$

A possible choice that satisfies the aforementioned specifications is  $T(\zeta) = \frac{\exp(\zeta) - 1}{\exp(\zeta) + 1}$ .

From (6.13), we obtain

$$\varepsilon_{p_i} = T^{-1} \left( \frac{e_{p_i}}{\rho_{p_i}} \right), \quad (6.16)$$

which, after differentiation, becomes

$$\dot{\varepsilon}_{p_i} = \frac{r_{p_i}}{\rho_{p_i}} \left( x_{2_i} - \dot{y}_{i,d} - \frac{\dot{\rho}_{p_i} e_{p_i}}{\rho_{p_i}} \right),$$

or, in stack vector form,

$$\dot{\varepsilon}_p = r_p \rho_p^{-1} (x_2 - \dot{y}_d - \dot{\rho}_p \rho_p^{-1} e_p), \quad (6.17)$$

where  $\varepsilon_p := [\varepsilon_{p_1}, \dots, \varepsilon_{p_n}]^\top$ ,  $r_{p_i} := \frac{\partial T^{-1}(\zeta)}{\partial \zeta} \Big|_{\zeta=\frac{e_{p_i}}{\rho_{p_i}}}$ ,  $r_p := \text{diag}\{r_{p_1}, \dots, r_{p_n}\}$ , and  $\rho_p := \text{diag}\{\rho_{p_1}, \dots, \rho_{p_n}\}$ . Due to the increasing property of  $T(\cdot)$ , it holds that  $r_p$  is positive definite, and thus in order to render  $\dot{\varepsilon}_p$  negative a straightforward choice for a desired value for  $x_2$  is

$$x_{2,d} := \dot{y}_d + \dot{\rho}_p \rho_p^{-1} e_p - k_p r_p \varepsilon_p, \quad (6.18)$$

where  $k_p \in \mathbb{R}_{>0}$  is a positive and constant scalar gain. Since, however,  $x_2$  is not the system's input, we follow a backstepping-like methodology and define the error

$$e_v := [e_{v_1}, \dots, e_{v_n}]^\top := x_2 - x_{2,d}. \quad (6.19)$$

Next, we proceed in a similar manner and define a funnel for each  $e_{v_i}$ ,  $i \in \{1, \dots, n\}$ , described by the functions  $\rho_{v_i} : [t_0, \infty) \rightarrow [\underline{\rho}_{v_i}, \bar{\rho}_{v_i}] \subset \mathbb{R}_{>0}$ , where  $\underline{\rho}_{v_i}, \bar{\rho}_{v_i} \in \mathbb{R}_{>0}$  are the positive lower and upper bounds, respectively, with the constraint  $\rho_{v_i}(t_0) > |e_{v_i}(t_0)|$ ,  $i \in \{1, \dots, n\}$ . Note that  $e_{v_i}(t_0) = x_2(t_0) - x_{2,d}(t_0)$  can be calculated at  $t = t_0$  since it is a function of the state, the funnel functions and the desired trajectory profile. Then, we define the open set

$$\mathcal{D}_{u,t} := \{(x, t) \in \mathbb{R}^{2n} \times [t_0, \infty) : \rho_p(t)^{-1} e_p \in (-1, 1)^n, \rho_v(t)^{-1} e_v \in (-1, 1)^n\}, \quad (6.20)$$

and design the control law  $u : \mathcal{D}_{u,t} \rightarrow \mathbb{R}^n$  as

$$u = -k_{v_2} \rho_v^{-1} \left( k_{v_3} \|r_p \varepsilon_p\| + k_{v_4} \hat{d} \right) s_v - k_{v_1} \rho_v^{-1} r_v \varepsilon_v \quad (6.21)$$

where

$$s_v := \begin{cases} \frac{r_v \varepsilon_v}{\|r_v \varepsilon_v\|}, & \text{if } \|r_v \varepsilon_v\| \neq 0, \\ 0, & \text{otherwise,} \end{cases}$$

$\rho_v := \text{diag}\{\rho_{v_1}, \dots, \rho_{v_n}\}$ ,  $\varepsilon_v := [\varepsilon_{v_1}, \dots, \varepsilon_{v_n}]^\top$ ,  $\varepsilon_{v_i} := T^{-1} \left( \frac{e_{v_i}}{\rho_{v_i}} \right)$ ,  $r_v := \text{diag}\{r_{v_1}, \dots, r_{v_n}\}$ ,  $r_{v_i} := \frac{\partial T^{-1}(\zeta)}{\partial \zeta} \Big|_{\zeta=\frac{e_{v_i}}{\rho_{v_i}}}$ ,  $k_{v_i} \in \mathbb{R}_{>0}$ ,  $i \in \{1, \dots, 4\}$  are positive constant scalar gains, and  $\hat{d}$  is an adaptive variable gain, subject to the constraint  $\hat{d}(t_0) \geq 0$ , and dynamics

$$\dot{\hat{d}} = \gamma_d \|r_v \varepsilon_v\|, \quad (6.22)$$

where  $\gamma_d \in \mathbb{R}_{>0}$  is a positive constant gain.

**Remark 6.4.** The control design procedure follows closely the prescribed performance backstepping-like methodology of Section 2.2.4 [3.2] introduced in [? ]. The desired signals and control laws there consist only of proportional terms with respect to the transformed errors  $\varepsilon_p$ ,  $\varepsilon_v$ , i.e.,  $-k_p r_p \varepsilon_p$  and  $-k_{v_1} \rho_v^{-1} r_v \varepsilon_v$  in (6.18) and (6.21), respectively, which are guaranteed to be ultimately bounded. In this work, we incorporate (a) the extra terms in (6.18) that would render (6.17) exponentially stable, and (b) the discontinuous term in (6.21), which, as will be shown in the sequel, enforces convergence of the transformed errors to zero, guaranteeing thus asymptotic stability. This is achieved without requiring the funnel functions to converge to zero. However, one can still set the prescribed funnel to converge arbitrarily close to zero, achieving thus a predefined convergence rate.

**Remark 6.5.** Note that no information regarding the dynamic model is incorporated in the control protocol (6.12)-(6.22). All the necessary signals consist of the funnel terms  $\rho_p$ ,  $\rho_v$  and of known functions of the state and the desired trajectory  $y_d$ . Furthermore, no a-priori gain tuning is needed and, as the next theorem states, the solution of Problem 6.3 is guaranteed from all initial conditions that satisfy  $-\rho_{p_i}(t_0) > y_i(t_0) - y_{i,d}(t_0) > \rho_{p_i}(t_0)$ ,  $\forall i \in \{1, \dots, n\}$ . As will be revealed subsequently, the adaptive gain  $\hat{d}$  compensates the unknown dynamic terms, which are proven to be bounded due to the confinement of the state in the prescribed funnels.

The correctness of the control protocol (6.12)-(6.22) is shown in the next theorem.

**Theorem 6.3.** Consider a system subject to the dynamics (6.11), Assumptions 6.3-6.7, as well as a desired trajectory  $y_d$  and funnels as described in Problem 6.3 satisfying  $-\rho_{p_i}(t_0) > y_i(t_0) - y_{i,d}(t_0) > \rho_{p_i}(t_0)$ ,  $\forall i \in \{1, \dots, n\}$ . Then the control protocol (6.12)-(6.22) guarantees the existence of at least one local Filippov solution of the closed-loop system (6.11)-(6.21) that solves Problem 6.3. Moreover, every such local solution can be extended to a global solution and all closed-loop signals remain bounded, for all  $t \geq t_0$ .

*Proof.* The intuition of the subsequent proof is as follows: We first show the existence of at least one Filippov solution of the closed loop system in  $\mathcal{D}_{u,t}$  for a time interval  $I_t \subseteq [t_0, \infty)$ . Next, we prove that for any of these solutions, the state remains bounded in  $I$  by bounds independent of the endpoint of  $I$ . Hence, the dynamic terms of (6.11) are also upper bounded by a term, which we aim to compensate via the adaptation gain  $\hat{d}$ .

We start by defining some terms that will be used in the subsequent analysis:

$$\begin{aligned}
 M_p &:= \max_{i \in \{1, \dots, n\}} \{\bar{\rho}_{p_i}\} & m_p &:= \min_{i \in \{1, \dots, n\}} \{\underline{\rho}_{p_i}\} \\
 M_{\dot{p}} &:= \max_{i \in \{1, \dots, n\}} \left\{ \sup_{t \geq t_0} \{|\dot{\rho}_{p_i}| \} \right\} & M_v &:= \max_{i \in \{1, \dots, n\}} \{\bar{\rho}_{v_i}\} \\
 m_v &:= \min_{i \in \{1, \dots, n\}} \{\underline{\rho}_{v_i}\} & \lambda &:= \lambda_{\min} \left( \rho_v^{-1} \tilde{G}(x, z, t) \rho_v^{-1} \right) \\
 \bar{\beta} &:= (k_{v_2} k_{v_4} \lambda)^{-1} & r_p &:= \inf_{\zeta \in (-1, 1)} \frac{\partial T^{-1}(\zeta)}{\partial \zeta}.
 \end{aligned}$$

Note that all the aforementioned terms are strictly positive. In particular,  $\lambda$  is strictly positive due to the definition of the funnels  $\rho_v$  and Assumption 6.4, and  $r_p$  is strictly positive due to the strictly increasing property of  $T(\cdot)$  and hence of  $T^{-1}(\cdot)$ . Moreover, in view of (6.14), it holds that  $\arg \inf_{\zeta \in (-1, 1)} \frac{\partial T^{-1}(\zeta)}{\partial \zeta} \in (-1, 1)$ .

By employing (6.21), (6.22), we can write the closed loop system

$$\dot{x}_1 = x_2, \tag{6.23a}$$

$$\dot{z} \in K[F_z](x, z, t), \tag{6.23b}$$

$$\dot{x}_2 \in K[F](x, z, t) + K[G](x, z, t)K[u](x, t), \tag{6.23c}$$

$$\dot{\hat{d}} = \gamma_d \|r_v \varepsilon_v\|, \tag{6.23d}$$

where  $K[F](x, z, t)$ ,  $K[G](x, z, t)$ ,  $K[u](x, t)$  are the Filippov regularizations (see (A.5)) of the respective terms. For  $u$  specifically,  $K[u](x, t)$  is formed by substituting the term  $s_v$  with its regularized term, which is  $S_v = \frac{r_v \varepsilon_v}{\|r_v \varepsilon_v\|}$  if  $\|r_v \varepsilon_v\| \neq 0$ , and  $S_v \in (-1, 1)^n$  otherwise. Note that, in any case, it holds that  $(r_v \varepsilon_v)^\top S_v = \|r_v \varepsilon_v\|$ . Define now  $\tilde{x} := [x^\top, z^\top, \hat{d}]^\top \in \mathbb{R}^{2n+n_z+1}$  and consider the open set  $\mathcal{D}_c := \{(\tilde{x}, t) \in \mathbb{R}^{2n+n_z+1} \times [t_0, \infty) : (x, t) \in \mathcal{D}_{u,t}\}$ . Since  $\rho_{p_i}(t_0) > |e_{p_i}(t_0)|$  and  $\rho_{v_i}(t_0) > |e_{v_i}(t_0)|$ ,  $\forall i \in \{1, \dots, n\}$ , the set  $\mathcal{D}_c$  is nonempty. Moreover, since  $T(\cdot)$ , and hence its derivative, are analytic, their zero sets have zero measure [?] and thus the right hand-side of (6.23) is Lebesgue measurable and locally essentially bounded in  $\tilde{x}$  over the set  $\{\tilde{x} : (\tilde{x}, t) \in \mathcal{D}_c\}$ , and Lebesgue measurable in  $t$  over the set  $\{t : (\tilde{x}, t) \in \mathcal{D}_c\}$ . Hence, according to Prop. A.1 of Appendix A, for each initial condition  $(\tilde{x}(t_0), t_0) \in \mathcal{D}_c$ , there exists at least one Filippov solution  $\tilde{x}(t)$  of (6.23), defined in  $I_t := [t_0, t_{\max})$ , where  $t_{\max} > t_0$  such that  $(\tilde{x}(t), t) \in \mathcal{D}_c$ ,  $\forall t \in I_t$ . By applying (6.16), we conclude the existence of the respective Filippov solutions  $\varepsilon_p(t), \varepsilon_v(t) \in \mathbb{R}^n$ ,  $\forall t \in I_t$ . Let now  $\tilde{x}(t_0)$  denote the initial condition of the system (6.23) satisfying  $(\tilde{x}(t_0), t_0) \in \mathcal{D}_c$  and consider the family of Filippov solutions starting from  $\tilde{x}(t_0)$  denoted by the set  $\mathfrak{X}$ . Note that,

although not explicitly stated,  $t_{\max}$  and  $I_t$  might be different for each solution in  $\mathfrak{X}$ . We aim to prove that all  $\varepsilon_p(t), \varepsilon_v(t)$  are bounded and converge to zero, for all  $\tilde{x}(t) \in \mathfrak{X}$ .

In view of the definition of  $\mathcal{D}_c$  (see also (6.20)), for all  $\tilde{x}(t) \in \mathfrak{X}$  it holds that

$$|e_{p_i}(t)| < \bar{\rho}_{p_i}, \quad (6.24a)$$

$$|e_{v_i}(t)| < \bar{\rho}_{v_i}, \quad (6.24b)$$

$\forall t \in I_t$ , where  $\bar{\rho}_{p_i}$  and  $\bar{\rho}_{v_i}$  are the upper bounds of  $\rho_{p_i}(t)$  and  $\rho_{v_i}(t)$ , respectively,  $\forall i \in \{1, \dots, n\}$ . Consider now the Lyapunov function  $V_p := \frac{1}{2} \|\varepsilon_p\|^2$ , for which it holds, in view of (6.17), (6.18), (6.19), and (6.24b)

$$\begin{aligned} \dot{V}_p &= \varepsilon_p^\top r_p \rho_p^{-1} (x_2 - \dot{y}_d - \dot{\rho}_p \rho_p^{-1} e_p) \\ &= -k_p \varepsilon_p^\top r_p \rho_p^{-1} r_p \varepsilon_p + \varepsilon_p^\top r_p \rho_p^{-1} e_v < -\frac{k_p}{M_p} \|r_p \varepsilon_p\|^2 + \frac{M_v}{m_p} \|r_p \varepsilon_p\|, \end{aligned}$$

$\forall t \in I_t$ . Hence, we conclude that  $\dot{V}_p < 0$  when  $\|r_p \varepsilon_p\| > \frac{M_v M_p}{k_p m_p r_p}$ . Since  $r_{p_i}$  is positive definite,  $\forall i \in \{1, \dots, n\}$ , the latter is equivalent to  $\|\varepsilon_p\| > \frac{M_v M_p}{k_p m_p r_p} \Rightarrow \dot{V}_p < 0$ . Hence, we conclude that all  $\tilde{x}(t) \in \mathfrak{X}$  satisfy

$$\|\varepsilon_p(t)\| \leq \bar{\varepsilon}_p := \max \left\{ \|\varepsilon_p(t_0)\|, \frac{M_v M_p}{k_p m_p r_p} \right\}.$$

Since  $\bar{\varepsilon}_p$  is finite, it holds that  $T(\bar{\varepsilon}_p) < 1$  and hence  $|T(\varepsilon_{p_i}(t))| \leq T(\bar{\varepsilon}_p) < 1$ ,  $\forall i \in \{1, \dots, n\}, t \in I_t$ . Moreover, since  $T(\cdot)$  and  $T^{-1}(\cdot)$  are smooth, the derivative  $\frac{\partial T^{-1}(\zeta)}{\partial \zeta}$  approaches infinity only when  $\zeta \rightarrow \pm 1$ . Therefore, in view of the definition of  $r_{p_i}$  in (6.17), we conclude the existence of a finite  $\bar{r}_p > 0$  such that  $\|r_p(t)\| \leq \bar{r}_p$ ,  $\forall t \in I_t$ . Next, (6.13) implies that  $\|e_p(t)\| \leq \bar{\varepsilon}_p := M_p T(\bar{\varepsilon}_p) \sqrt{n}$ ,  $\forall t \in I_t$ . Hence, we conclude that  $\|x_{2,d}(t)\| \leq \bar{x}_{2,d} := \bar{y}_d + \frac{M_p}{m_p} \bar{\varepsilon}_p + k_p \bar{r}_p \bar{\varepsilon}_p$ ,  $\forall t \in I_t$ , where  $\bar{y}_d$  is the uniform bound of the desired trajectory, introduced in Assumption 6.7. We also conclude that  $\|x_1(t)\| \leq \bar{x}_1 := \bar{\varepsilon}_p + \bar{y}_d$ ,  $\forall t \in I_t$ . In addition, by employing  $x_2 = e_v + x_{2,d}$  and (6.24b), we conclude that  $\|x_2(t)\| < \tilde{x}_2 := M_v \sqrt{n} + \bar{x}_{2,d}$ ,  $\forall t \in I_t$ . Finally, by differentiating  $x_{2,d}$ , employing the smoothness and boundedness of  $\rho_p$  and its derivatives, the smoothness of  $T(\cdot)$ , the boundedness of  $\dot{y}_d(t)$  as well as the aforementioned bounds, we can conclude the existence of a bound  $\bar{v}_d$  such that  $\|\dot{x}_{2,d}(t)\| \leq \bar{v}_d$ ,  $\forall t \in I_t$ .

Furthermore, the boundedness of  $x(t)$  and Assumption 6.5 imply the existence of a positive finite constant  $\bar{z}$  such that  $\|z(t)\| \leq \bar{z}$ ,  $\forall t \in I_t$ . Hence, since  $F(x, z, t)$  is Lebesgue measurable and locally essentially bounded in  $\mathbb{R}^{2n+n_z}$  and  $\|x_1(t)\| \leq \bar{x}_1 < \infty$ ,  $\|x_2(t)\| < \tilde{x}_2 < \infty$ ,  $\|z(t)\| \leq \bar{z}$ ,  $\forall t \in I_t$ ,

there exists some positive  $\bar{F}$ , such that  $\|F(x(t), z(t), t)\| \leq \bar{F}$ ,  $\forall t \in I_t$ , and hence, for each  $(x, z)$ , since  $K[F]$  is formed by the convex closure of  $F$ , it holds that  $\max_{\zeta \in K[F](x(t), z(t), t)} \{\zeta\} \leq \bar{F}$ ,  $\forall t \in I_t$  and  $\tilde{x}(t) \in \mathfrak{X}$ . Note that, in view of the aforementioned discussion,  $\bar{F}$  depends solely on the initial conditions and the parameters of the funnel functions. Define now the finite constant term  $d_b \in \mathbb{R}_{>0}$  as

$$d_b := \frac{\bar{\beta}}{m_v} (\bar{F} + \bar{v}_d + M_p \sqrt{n}). \quad (6.25)$$

Note that the term in the parenthesis of (6.25) is an upper bound for the term  $\|F(x(t), z(t), t) - \dot{x}_{2,d}(t) - \dot{\rho}_v(t)\rho_v(t)^{-1}e_v(t)\|$ , for all  $\tilde{x}(t) \in \mathfrak{X}$  and almost all  $t \in I_t$ .

Define also the signal  $\tilde{d} := \hat{d} - d_b$ , where  $\hat{d}$  is the adaptive gain introduced in (6.21). Consider now the function

$$V_v(\tilde{\varepsilon}) := \bar{\alpha} V_p + \frac{\bar{\beta}}{2} \|\varepsilon_v\|^2 + \frac{1}{2\gamma_d} \tilde{d}^2,$$

where  $\tilde{\varepsilon} := [\varepsilon_p^\top, \varepsilon_v^\top, \tilde{d}]^\top$ , and  $\bar{\alpha} > 0$  is a positive constant to be defined;  $V_v(\tilde{\varepsilon})$  satisfies  $W_1(\tilde{\varepsilon}) \leq V_v(\tilde{\varepsilon}) \leq W_2(\tilde{\varepsilon})$ , for  $W_1(\tilde{\varepsilon}) := \min \left\{ \frac{\bar{\alpha}}{2}, \frac{\bar{\beta}}{2}, \frac{1}{2\gamma_d} \right\} \|\tilde{\varepsilon}\|^2$  and  $W_2(\tilde{\varepsilon}) := \max \left\{ \frac{\bar{\alpha}}{2}, \frac{\bar{\beta}}{2}, \frac{1}{2\gamma_d} \right\} \|\tilde{\varepsilon}\|^2$ . Then, according to Lemma A.2 of Appendix A,  $\dot{V}_v(\tilde{\varepsilon}(t)) \stackrel{a.e.}{\in} \dot{\tilde{V}}_v(\tilde{\varepsilon}(t))$  with

$$\dot{\tilde{V}}_v := \bigcap_{\xi \in \partial V_v(\tilde{\varepsilon})} \xi^\top K[\dot{\tilde{\varepsilon}}]$$

Since  $V_v(\tilde{\varepsilon})$  is continuously differentiable, its generalized gradient reduces to the standard gradient and thus it holds that  $\dot{\tilde{V}}_v = \nabla V_v^\top K[\dot{\tilde{\varepsilon}}]$ , where  $\nabla V_v = [\bar{\alpha} \varepsilon_p^\top, \bar{\beta} \varepsilon_v^\top, \frac{1}{\gamma_d} \tilde{d}]^\top$ . After using (6.11), (6.21), (6.22), and  $x_2 = x_{2,d} + e_v$ , one obtains

$$\begin{aligned} \dot{\tilde{V}}_v \subset \widetilde{W}_s &:= -\bar{\alpha} k_p \varepsilon_p^\top r_p \rho_p^{-1} r_p \varepsilon_p + \bar{\alpha} \varepsilon_p^\top r_p \rho_p^{-1} e_v - \bar{\beta} k_{v_1} \varepsilon_v^\top r_v \rho_v^{-1} K[G](x, t) \rho_v^{-1} r_v \varepsilon_v \\ &\quad + \tilde{d} \|r_v \varepsilon_v\| + \bar{\beta} \varepsilon_v^\top r_v \rho_v^{-1} (K[F](x, t) - \dot{x}_{2,d} - \dot{\rho}_v \rho_v^{-1} e_v) \\ &\quad - \bar{\beta} k_{v_2} \varepsilon_v^\top r_v \rho_v^{-1} K[G](x, t) \rho_v^{-1} S_v \left( k_{v_3} \|r_p \varepsilon_p\| + k_{v_4} \hat{d} \right). \end{aligned}$$

Note that, since  $\hat{d}(t_0) \geq 0$ , (6.22) implies that  $\hat{d}(t) \geq 0$ ,  $\forall t \in I_t$ . Moreover, since the Filippov regularization (A.5) is defined as a closed set and  $\dot{\tilde{V}}_v \subset \widetilde{W}_s$ , it holds that  $\max_{\zeta \in \dot{\tilde{V}}_v} \{\zeta\} \leq \max_{\zeta \in \widetilde{W}_s} \{\zeta\}$ . By substituting  $G = \frac{G+G^\top}{2} +$

$\frac{G-G^\top}{2}$  and employing the skew-symmetry of the second term, we obtain in view of Assumption 6.4 and the definition of  $d_b$  in (6.25):

$$\begin{aligned} \max_{\zeta \in \tilde{V}_v} \{\zeta\} &\leq \max_{\zeta \in \tilde{W}_s} \{\zeta\} \leq -\bar{\alpha} \frac{k_p}{M_p} \|r_p \varepsilon_p\|^2 - k_{v_1} \bar{\beta} \lambda \|r_v \varepsilon_v\|^2 - k_{v_2} k_{v_4} \bar{\beta} \lambda \|r_v \varepsilon_v\| \hat{d} - \\ & k_{v_2} k_{v_3} \bar{\beta} \lambda \|r_v \varepsilon_v\| \|r_p \varepsilon_p\| + \tilde{d} \|r_v \varepsilon_v\| + \|r_v \varepsilon_v\| d + \bar{\alpha} \|\varepsilon_p^\top r_p \rho_p^{-1} e_v\|, \end{aligned}$$

for all solutions  $\tilde{x}(t) \in \mathfrak{X}$ . By setting  $\zeta = T(\varepsilon_{v_i})$  in (6.15), we obtain  $|T(\varepsilon_{v_i})| \leq |r_{v_i} \varepsilon_{v_i}|$  and hence by employing  $e_{v_i} = \rho_{v_i} T(\varepsilon_{v_i})$ ,  $i \in \{1, \dots, n\}$ , we obtain that

$$\bar{\alpha} \|\varepsilon_p^\top r_p \rho_p^{-1} e_v\| \leq \bar{\alpha} \frac{M_v}{m_p} \|r_p \varepsilon_p\| \|r_v \varepsilon_v\|.$$

Therefore, by setting  $\bar{\alpha} := \frac{k_{v_2} k_{v_3} m_p \bar{\beta} \lambda}{M_v}$ , employing  $d_b = \hat{d} - \tilde{d}$ , and in view of the fact that  $\bar{\beta} = (k_{v_2} k_{v_4} \lambda)^{-1}$ , we obtain

$$\max_{\zeta \in \tilde{V}_v} \{\zeta\} \leq -\bar{\alpha} \frac{k_p}{M_p} \|r_p \varepsilon_p\|^2 - k_{v_1} \bar{\beta} \lambda \|r_v \varepsilon_v\|^2 =: -W(\tilde{\varepsilon}),$$

$\forall t \in I_t$ ,  $\tilde{x}(t) \in \mathfrak{X}$ , where  $W$  is continuous and positive semi-definite on  $\mathbb{R}^{2n+1}$ , since  $r_v$  and  $r_p$  are positive definite. Hence, we conclude that  $\zeta \leq -W(\tilde{\varepsilon})$ ,  $\forall \zeta \in \tilde{V}_v(\tilde{\varepsilon}(t))$ ,  $\forall t \in I_t$  and all  $\tilde{x}(t) \in \mathfrak{X}$ . Choose now any finite  $r_a > 0$  and let  $c_a < \min_{\|\tilde{\varepsilon}\|=r_a} W_1(\tilde{\varepsilon})$ . Note that all the conditions of Theorem A.6 in Appendix A are satisfied and hence, all Filippov solutions starting from  $\tilde{\varepsilon}(t_0) \in \Omega_f := \{\tilde{\varepsilon} \in \mathcal{B}(0, r_a) : W_2(\tilde{\varepsilon}) \leq c_a\}$  are bounded and remain in  $\Omega_f$ ,  $\forall t \in I_t$ . Moreover,  $t_{\max} = \infty$ , implying that  $I_t = [t_0, \infty)$  and it also holds that  $\lim_{t \rightarrow \infty} \|\varepsilon_p(t)\| = 0$  and  $\lim_{t \rightarrow \infty} \|\varepsilon_v(t)\| = 0$ , which, in view of the increasing property of  $T(\cdot)$  and the fact that  $T(0) = 0$ , implies that  $\lim_{t \rightarrow \infty} \|e_p(t)\| = 0$  and  $\lim_{t \rightarrow \infty} \|e_v(t)\| = 0$ . Notice that Zeno behavior is avoided since  $t_{\max} = \infty$ .

Note that  $r_a$ , and hence  $c_a$ , can be arbitrarily large allowing any finite initial condition  $\tilde{\varepsilon}$ , which implies any  $(\tilde{x}(t_0), t_0) \in \mathcal{D}_c$ . In addition, it holds that  $\|\tilde{\varepsilon}\|^2 \leq \tilde{c} := (\max\{\frac{\bar{\alpha}}{2}, \frac{\bar{\beta}}{2}, \frac{1}{2\gamma_d}\})^{-1} c_a$ , which implies the boundedness of  $\|\varepsilon_p\|$ ,  $\|\varepsilon_v\|$  and  $\tilde{d}$  by  $\sqrt{\tilde{c}}$ . Therefore, we conclude that  $\|\hat{d}(t)\| \leq \bar{d} := d_b + \sqrt{\tilde{c}}$ ,  $\forall t \in I_t$ . Moreover, by employing (6.13), we conclude that  $|\rho_{v_i}(t)^{-1} e_{v_i}(t)| \leq T(\sqrt{\tilde{c}}) < 1$ , and hence  $|e_{v_i}(t)| \leq M_v T(\sqrt{\tilde{c}}) \Rightarrow \|x_2(t)\| \leq \bar{x}_2 := M_v T(\sqrt{\tilde{c}}) \sqrt{n} + \bar{x}_{2,d}$ ,  $\forall t \in I_t$ . Therefore, we conclude that all solutions are bounded in compact sets  $\forall t \in I_t$ , which means that  $u$ , and  $\dot{d}$ , as designed in (6.21) and (6.22), respectively, remain also bounded,  $\forall t \in I_t$ .  $\square$

**Remark 6.6.** Note that no boundedness assumptions or growth conditions are needed for the vector fields  $F(x, z, t)$  and  $G(x, z, t)$ . In particular, the

effect of  $F(x, z, t)$  is canceled by the introduced adaptive signal  $\hat{d}$ , which increases according to (6.22). It is proved, nevertheless, that this adaptive signal remains bounded. Moreover, the response of the system is solely determined by the funnel functions  $\rho_{p_i}$  and  $\rho_{v_i}$ , isolated from the system dynamics and the control gains selection. Nevertheless, we note that appropriate gain tuning might be needed to suppress chattering in real life scenarios. Similarly, note that the region of attraction (initial conditions) of  $(\varepsilon_p, \varepsilon_v) = (0, 0)$  is independent from the system dynamics and the control gain selection and depends only on the choice of the funnel functions  $\rho_{p_i}$ ,  $\forall i \in \{1, \dots, n\}$ . In particular, if  $\rho_{p_i}(t_0)$  are design parameters, we can always choose them such that  $-\rho_{p_i}(t_0) < e_{p_i}(t_0) < \rho_{p_i}(t_0)$ ,  $\forall i \in \{1, \dots, n\}$ , which renders the result global. In fact, the choice  $\lim_{t \rightarrow t_0^+} \frac{1}{\rho_{p_i}(t_0)} = 0$ ,  $\forall i \in \{1, \dots, n\}$  [? ] is not excluded from our control scheme and does not restrict the initial condition  $y(t_0)$ . Moreover, noise can be taken into account in the measurement of  $x_2$ , i.e., consider that  $x_2 + n(x, t)$  is available for measurement, where  $n(x, t)$  is an unknown noise signal with appropriate continuity and boundedness properties. By redefining  $e_v = x_2 + n(x, t) - x_{2,d}$  and including the time derivative of  $n(x, t)$  in (6.25), the analysis still holds. Note, however, that in this case it can only be deduced that  $\lim_{t \rightarrow \infty} (x_2(t) + n(x, t) - x_{2,d}(t)) = 0$  and hence  $x_2(t)$  does not necessarily converge to  $x_{2,d}(t)$ .

**Remark 6.7.** Since funnel control traditionally guarantees confinement of the state in the desired funnel, a common practice is to tune the funnel to converge to arbitrarily small values, achieving thus “practical stability”, i.e., the state converging arbitrarily close to zero. Note that, in our case, the funnel functions  $\rho_{p_i}$ ,  $\rho_{v_i}$ , are not required to decrease to values arbitrarily close to zero, yet *asymptotic stability* is still achieved. In fact, the proposed control schemes can be used to achieve merely asymptotic stability results without any funnel constraints, if the latter is not required. More specifically, given the initial errors  $e_{p_i}(t_0)$ , we can use the proposed control protocols by employing any *constant* values  $\rho_{p_i} > |e_{p_i}(t_0)|$ ,  $\forall i \in \{1, \dots, n\}$ . Finally, the proposed control scheme can be extended to systems of the form  $\dot{x}_i = \dot{x}_{i+1}$ ,  $i \in \{1, \dots, k-1\}$ ,  $\dot{x}_k = F(x, z, t) + G(x, z, t)u$  for some  $k > 0$ , where the funnel constraints are set for the combined signal  $\sum_{j \in \{1, \dots, k-1\}} e_p^{(k)}$ .

### 6.4.3 Simulation Results

We consider here the simulation of two inverted pendulum connected by a spring and a damper [?], with dynamics:

$$\begin{aligned} J_1 \ddot{x}_{11} &= g_s \sin(x_{11}) - \frac{1}{4} F_s \cos(x_{11} - \theta_f) - T_{f1} + u_1 \\ J_2 \ddot{x}_{12} &= 1.25 g_s \sin(x_{12}) + \frac{1}{4} F_s \cos(x_{12} - \theta_f) - T_{f2} + \sigma_f(t) u_2, \end{aligned}$$

where  $F_s := 150(d_s - \frac{1}{2}) + \dot{d}_s$  is the force between the connection points of the spring and damper at the pendulums, and

$$d_s := \sqrt{\frac{1}{4} + \frac{1}{4}(\sin(x_{11} - x_{12})) + \frac{1}{8}(1 - \cos(x_{12} - x_{11}))}$$

is the distance between these connection points;  $\theta_f$  is defined as

$$\theta_f := \tan^{-1} \left( \frac{\frac{1}{4}(\cos(x_{12}) - \cos(x_{11}))}{\frac{1}{2} + \frac{1}{4}(\sin(x_{11}) - \sin(x_{12}))} \right)$$

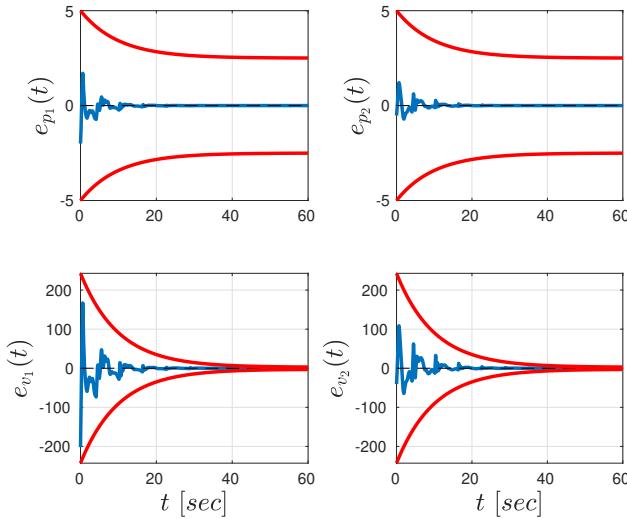
and  $T_{f1}, T_{f2}$  are friction terms on the motors evolving according to  $T_{f_i} := \tau_{f_i} + \dot{\tau}_{f_i} + \dot{x}_{1i}$ , with

$$\dot{\tau}_{f_i} = \dot{x}_{1i} - \frac{|\dot{x}_{1i}|}{1 + \exp\left(-\left|\frac{\dot{x}_{1i}}{0.1}\right|^2\right)}$$

The time varying signal  $\sigma_f(t)$  is taken as:

$$\sigma_f(t) = \begin{cases} 1 & \text{if } t \in [0, 3) \cup [3.5, \infty), \\ 0.5 & \text{if } t \in [3, 3.5) \end{cases}$$

modeling a loss of effectiveness of the second motor when  $t \in [3, 3.5]$ . We also choose  $g_s = 9.81$  as the gravity constant and  $J_1 = 0.5, J_2 = 0.625$ . The initial conditions are  $t_0 = 0, x(0) = [0, 0, 0, 0]^\top$  (rad, rad/s),  $\tau_{f1}(0) = \tau_{f2}(0) = 0$  and the desired trajectory  $y_d = [2 \cos(t), \frac{\pi}{2} - 2 \sin(t)]^\top$  rad. The prescribed funnel functions are chosen as  $\rho_{p_i}(t) = 2.5 \exp(-0.1t) + 2.5, \forall i \in \{1, 2\}$ , which converge to 2.5. We also choose  $\rho_{v_i}(t) = (\|e_v(0)\|_1 - 2) \exp(-0.1t) + 2.5$ , as well as the gains  $k_p = 10, k_{v1} = 2 \cdot 10^3, k_{v2} = 0.1, k_{v3} = 0.025, k_{v4} = 0.05$ , and  $\gamma_d = 50$ . The simulation results are depicted in Figs. 6.7-6.10 for  $t \in [0, 60]$  sec. More specifically, Fig. 6.7 depicts the errors  $e_p(t), e_v(t)$  along with the performance functions  $\rho_p(t), \rho_v(t)$ . One can conclude that  $e_p(t)$  and  $e_v(t)$  not only respect their imposed funnels but also converge asymptotically to zero, without the need of arbitrarily small values for  $\lim_{t \rightarrow \infty} \rho_p(t)$  and  $\lim_{t \rightarrow \infty} \rho_v(t)$ . This can be verified also by Fig. 6.8, which

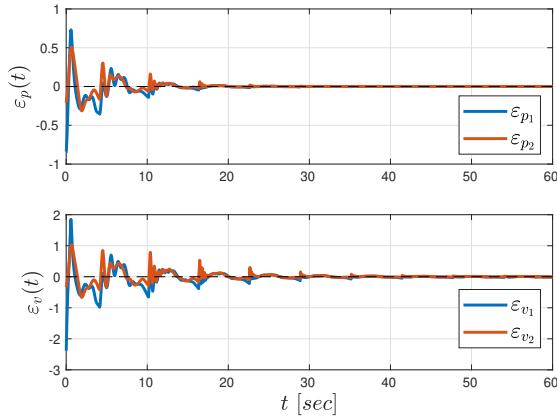


**Figure 6.7:** The evolution of the errors  $e_p(t)$  (top),  $e_v(t)$  (bottom), depicted with blue, along with the performance functions  $\rho_p(t)$ ,  $\rho_v(t)$ , depicted with red,  $\forall t \in [0, 60]$  sec.

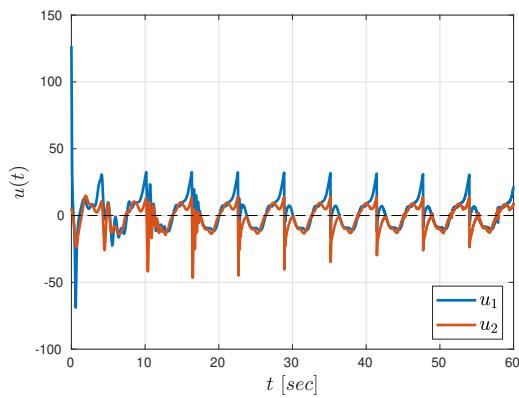
depicts the evolution of the transformed errors  $\varepsilon_p(t)$ ,  $\varepsilon_v(t)$ ,  $\forall t \in [0, 60]$  sec, and shows their asymptotic convergence to zero. Finally, Figs. 6.9 and 6.10 illustrate the inputs  $u(t)$  as well as the adaptation signal  $\hat{d}(t)$ ,  $\forall t \in [0, 60]$  sec. One can conclude the convergence of  $\hat{d}(t)$  to a constant value as well as the boundedness of the control input  $u(t)$ , as was proved in the theoretical analysis.

## 6.5 Conclusion

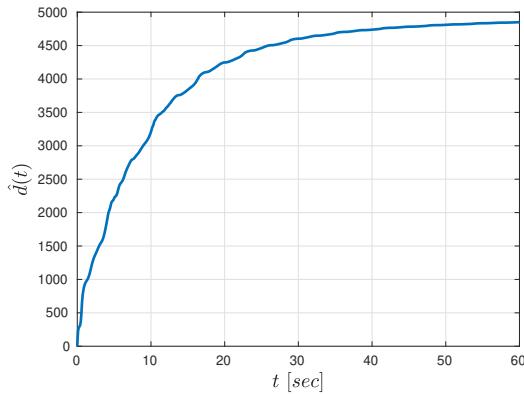
This chapter presented planning and control algorithms for single-agent systems. Firstly, we developed a hybrid algorithm for the planning of a robotic system under timed temporal logic formulas in an obstacle-cluttered workspace. By using previous results, we guaranteed the collision-free timed navigation leading to a timed abstraction of the system. A high-level planner and a novel optimization technique provided the timed path that satisfy the specification and is asymptotically optimal via reconfiguration. Secondly, we turned our attention to the motion planning of high-dimensional uncertain systems (e.g., robotic manipulators). We developed a two-layer framework by integrating adaptive control techniques and sampling-based motion planning. The closed-loop system provably navigated to a desired goal while avoiding collisions and compensating for the dynamic uncertainties.



**Figure 6.8:** The evolution of the transformed errors  $\varepsilon_p(t)$ ,  $\varepsilon_v(t)$ ,  $\forall t \in [0, 60]$  sec.



**Figure 6.9:** The evolution of the control inputs  $u(t) = [u_1(t), u_2(t)]^\top$ ,  $\forall t \in [0, 60]$  sec.



**Figure 6.10:** The evolution of the adaptation signal  $\hat{d}(t)$ ,  $\forall t \in [0, 60]$  sec.

Finally, we developed a control scheme that guarantees asymptotic stability for an uncertain 2nd-order system, while complying to funnel constraints, by integrating adaptive and discontinuous control methodologies.



# **Chapter 7**

# **Summary and Future Research Directions**

This chapter summarizes the content of the thesis and provides potential future research directions.

## **7.1 Summary**

In Chapter 2 we proposed a set of control algorithms for the cooperative manipulation of rigid objects. We tackled first the case of rigid grasping contacts, and developed closed-form adaptive control algorithms, compensating for uncertainty in the dynamic parameters of the object and the agents, and Nonlinear Model Predictive Control schemes, taking into account constraints such as obstacle avoidance and input saturation. A Prescribed Performance Control methodology has been also developed to achieve prescribed transient and steady-state response for the object. Secondly, we considered the case of rolling contacts, for which we developed novel centralized and decentralized control algorithms that guarantee agents-object contact maintenance, along with object reference tracking.

In Chapter 3, we tackled the problem of multi-agent formation and its relation to rigid cooperative manipulation. Firstly, we developed a robust model-free decentralized control scheme for the formation control of a tree-graph multi-agent system with prescribed transient and steady-state response, subject to collision and connectivity constraints. Secondly, we associated rigid cooperative manipulation schemes to multi-agent rigidity theory. We related the grasp matrix of the former to the rigidity matrix of the latter and we used that to derive novel conditions for the internal force-free cooperative manipulation.

In Chapter 4 we considered the problems of multi-agent navigation and leader-follower coordination subject to collision and/or connectivity constraints as well as uncertain dynamics. We first developed an adaptive

control protocol for the problem of single-agent navigation in an obstacle-cluttered environment with uncertain dynamics under almost all initial conditions. This was extended to a prioritization-based decentralized scheme for multi-agent systems. Secondly, we proposed a novel adaptive control scheme for the leader-follower coordination, that is, navigation of a leader agent to a predefined pose, while guaranteeing collision avoidance and connectivity maintenance. Finally, we designed an adaptive control protocol for the collision avoidance among ellipsoidal agents using a novel distance metric for 3D ellipsoids.

In Chapter 5 we used previous continuous control schemes to derive appropriate discrete abstractions and synthesize controllers for the satisfaction of complex tasks expressed as temporal logic formulas. We first considered local tasks for multi-agent systems, such as UAVs and robotic manipulators, and then we focused on cases where unactuated objects have their own specifications. We considered discretizations both using predefined regions of interest as well as a full workspace partition. Collision avoidance was taken into account to define safe transitions among the discrete states. Linear and Metric Interval Temporal Logic formulas were applied and control synthesis was performed using standard automata-based formal verification techniques.

In Chapter 6 we developed extension algorithms for single-agent problems. Firstly, we considered the motion planning problem of a single agent in an obstacle-cluttered environment under timed temporal tasks. By using previous results on safe timed navigation, we developed an algorithm that guarantees the satisfaction of the timed specification as well as asymptotically optimal performance in terms of energy efficiency. Secondly, we addressed the problem of motion planning of high-dimensional complex systems in obstacle-cluttered environments with uncertain dynamics. We integrated adaptive control techniques with sampling-based motion planning algorithms to develop a two-layer framework that guarantees the safe navigation of the system to its goal. Finally, we developed a novel model-free adaptive control scheme that guarantees asymptotic stability of a class of nonlinear systems while respecting predefined funnel constraints.

## 7.2 Future Research Directions

Regarding the cooperative manipulation schemes of Chapter 2, a strong assumption is that the agents operate away from kinematic singularities (except for the NMPC frameworks). Future directions can aim at addressing this issue and guarantee singularity avoidance. Moreover, grasp reconfiguration that allows more modular schemes is a promising direction, as well as fault-tolerant extensions of the current schemes. Finally, real-time experi-

ments with robotic agents with soft fingertip-type end-effectors should be attempted.

Regarding Chapter 3, future efforts can aim at extending the developed formation control scheme to more general multi-agent graph structures as well as taking into account collisions among all the agents. Moreover, one can notice that the control law for provably achieving zero internal forces in the rigid cooperative manipulation scheme, depending on the multi-agent rigidity matrix, is centralized. Therefore, one might aim at extending the proposed algorithm to a *decentralized* scheme as well as considering compliant grasping contacts. Real-time experiments should be also conducted to showcase the validity of the proposed conditions.

Regarding the multi-agent coordination of Chapter 4, future works can address less conservative solutions for the multi-agent collision-free navigation. This can be also attempted by using the proposed leader-follower scheme and appropriate prioritization. Sampled inter-agent communication should be also used to resemble more realistic scenarios. Regarding the proposed ellipsoidal collision avoidance problem, one drawback that is required to be tackled in future works is the fact that the scheme interferes with the main assigned tasks, potentially causing them to fail (local minima scenarios).

Regarding Chapter 5, future directions are needed towards the generalization of the proposed multi-agent schemes to incorporate timed temporal specifications. Regarding the multi-agent-object hybrid scheme that is based on region-of-interest discretization, a decentralized extension must be considered. Moreover, failure of transition executions can be taken into account via plan reconfiguration. Finally, future works should also focus on real-time experiments to further validate the proposed frameworks.

Finally, regarding Chapter 6, future works are required to focus mainly at extending the proposed frameworks to multi-agent schemes, as well as incorporation of input saturation constraints.

As a final remark, another interesting topic of research is the consideration of delays in the multi-agent communication, as well as the sampled feedback and control realization, which have been neglected in this thesis.



# Appendix A

## Dynamical Systems

This Appendix provides preliminary background on the theory of dynamical systems. We consider both smooth and non-smooth systems.

### A.1 Lipschitz Continuous Systems

We start with defining standard results on the existence of solutions of ODEs.

Consider the initial value problem:

$$\dot{x} = h(x, t), x(t_0) \in \Omega, \quad (\text{A.1})$$

with  $h : \Omega \times [t_0, \infty) \rightarrow \mathbb{R}^n$  where  $\Omega \subset \mathbb{R}^n$  is a non-empty open set containing the origin, and  $t_0 \in \mathbb{R}_{\geq 0}$ .

**Definition A.1.** [?] A solution  $x(t)$  of the initial value problem (A.1) is maximal if it has no proper right extension that is also a solution of (A.1).

**Theorem A.1.** [?] Let  $h : \bar{\Omega} := \Omega \times [t_0, \infty) \rightarrow \mathbb{R}^n$  from (A.1) satisfy the following conditions:

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

$$\begin{aligned} \|h(x, t)\| &\leq C_K, \\ \|h(x, t) - h(y, t)\| &\leq L_K \|x - y\|, \end{aligned}$$

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

Then the initial value problem (A.1) with  $h : \bar{\Omega} \rightarrow \mathbb{R}^n$  and some  $x_0 \in \Omega$ , has a unique and maximal solution  $x : [t_0, t_{\max}) \rightarrow \mathbb{R}^n$ , with  $t_{\max} > t_0$  and  $(x(t), t) \in \bar{\Omega}, \forall t \in [t_0, t_{\max})$ .

Note that the second condition imposed in the aforementioned theorem is a locally Lipschitz condition.

**Theorem A.2.** [?] Let the conditions of Theorem A.1 hold in  $\bar{\Omega}$  and let  $x(t), t \in [t_0, t_{\max})$  be a maximal solution of the initial value problem (A.1). Then, either  $t_{\max} = \infty$  or

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

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

**Definition A.2.** The origin  $x = 0$  is the equilibrium point for (A.1) if

$$h(0, t) = 0, \forall t \in [t_0, \infty)$$

We next provide the comparison function definitions, necessary for the stability classification of the equilibrium point.

**Definition A.3.** ([? ?]) A continuous function  $\alpha : [0, a) \rightarrow \mathbb{R}_{\geq 0}$  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  $\lim_{r \rightarrow \infty} \alpha(r) = \infty$ .

**Definition A.4.** ([? ?]) A continuous function  $\beta : [0, a) \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$  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  $\lim_{s \rightarrow \infty} \beta(r, s) = 0$ .

It is said to belong to class  $\mathcal{KL}_\infty$  if, in addition, for each fixed  $s$ , the mapping  $\beta(r, s)$  belongs to class  $\mathcal{K}_\infty$  with respect to  $r$ .

Now we can characterize the equilibrium point of (A.1) with respect to its stability.

**Definition A.5.** ([? ?]) The equilibrium point  $x = 0$  of (A.1) is

- uniformly stable, if there exists a class  $\mathcal{K}$  function  $\gamma(\cdot)$  and a positive constant  $c$  independent of  $t_0$ , such that

$$\|x(t)\| \leq \gamma(\|x(t_0)\|), \forall t \geq t_0, \|x(t_0)\| < c, \quad (\text{A.2})$$

- uniformly asymptotically stable, if there exists a class  $\mathcal{KL}$  function  $\beta(\cdot, \cdot)$  and a positive constant  $c$  independent of  $t_0$ , such that

$$\|x(t)\| \leq \beta(\|x(t_0)\|, t - t_0), \forall t \geq t_0, \|x(t_0)\| < c, \quad (\text{A.3})$$

- exponentially stable, if (A.3) is satisfied with  $\beta(r, s) = kr \exp(-\alpha s)$ ,  $k, \alpha \in \mathbb{R}_{\geq 0}$ ,
- globally uniformly stable, if (A.2) is satisfied with  $\gamma \in \mathcal{K}_\infty$  for any initial state  $x(t_0)$  and  $\Omega = \mathbb{R}^n$ ,
- globally uniformly asymptotically stable, if (A.3) is satisfied with  $\beta \in \mathcal{KL}_\infty$  for any initial state  $x(t_0)$ ,
- globally exponentially stable, if (A.3) is satisfied for any initial state  $x(t_0)$  and with  $\beta(r, s) = kr \exp(-\alpha s)$ ,  $k, \alpha \in \mathbb{R}_{\geq 0}$ .

The main Lyapunov stability theorem is then formulated as follows:

**Theorem A.3.** (*[? ? ]*) Let  $x = 0$  be an equilibrium point of (A.1). Let  $V : \Omega \times [t_0, \infty) \rightarrow \mathbb{R}_{\geq 0}$  be a continuously differentiable function such that,  $\forall t \geq t_0$ ,  $x \in \Omega$ ,

$$\begin{aligned}\gamma_1(\|x\|) &\leq V(x, t) \leq \gamma_2(\|x\|), \\ \frac{\partial V}{\partial t} + \frac{\partial V}{\partial x} h(x, t) &\leq -\gamma_3(\|x\|).\end{aligned}$$

Let  $r \in \mathbb{R}$  such that  $B(0, r) \subset \Omega$ . Then, the equilibrium point  $x = 0$  of (A.1) is

- uniformly stable, if  $\gamma_1$  and  $\gamma_2$  are class  $\mathcal{K}$  functions on  $[0, r)$  and  $\gamma_3(\cdot) \geq 0$  on  $[0, r)$ ,
- uniformly asymptotically stable, if  $\gamma_1$ ,  $\gamma_2$ , and  $\gamma_3$  are class  $\mathcal{K}$  functions on  $[0, r)$ ,
- exponentially stable, if  $\gamma_i(\rho) = k_i \rho^\alpha$  on  $[0, r)$ ,  $k_i, \alpha \in \mathbb{R}_{>0}$ ,  $\forall i \in \{1, 2, 3\}$ ,
- globally uniformly stable, if  $\Omega = \mathbb{R}^n$ ,  $\gamma_1$  and  $\gamma_2$  are class  $\mathcal{K}_\infty$  functions, and  $\gamma_3(\cdot) \geq 0$  on  $\mathbb{R}_{\geq 0}$ ,
- globally uniformly asymptotically stable if  $\Omega = \mathbb{R}^n$ ,  $\gamma_1$  and  $\gamma_2$  are class  $\mathcal{K}_\infty$  functions, and  $\gamma_3$  is a class  $\mathcal{K}$  function on  $\mathbb{R}_{\geq 0}$ ,
- globally exponentially stable, if  $\Omega = \mathbb{R}^n$ ,  $\gamma_i(\rho) = k_i \rho^\alpha$  on  $\mathbb{R}_{\geq 0}$ ,  $k_i \in \mathbb{R}_{>0}$ ,  $\forall i \in \{1, 2, 3\}$ ,

We provide next standard invariance results for time-invariant and time-varying systems.

**Theorem A.4.** (LaSalle [? ? ]) Let  $\Omega \subset \mathbb{R}^n$  be a positive invariant non-empty set of the time-invariant ODE  $\dot{x} = h_I(x)$ , where  $h_I : \Omega \rightarrow \mathbb{R}^n$  is continuous and satisfies condition 2 of Theorem (A.1). Let  $V : \Omega \rightarrow \mathbb{R}_{\geq 0}$  be a continuously differentiable function  $V(x)$  such that  $\dot{V}(x) \leq 0$ ,  $\forall x \in \Omega$ . Let  $E := \{x \in \Omega : \dot{V}(x) = 0\}$ , and let  $M$  be the largest invariant set contained in  $E$ . Then, every bounded solution  $x(t)$  starting in  $\Omega$  converges to  $M$  as  $t \rightarrow \infty$ .

**Lemma A.1.** (Barbalat [? ? ]) Let  $\phi : \mathbb{R} \rightarrow \mathbb{R}$  be a uniformly continuous function on  $[0, \infty)$ . Suppose that  $\lim_{t \rightarrow \infty} \int_0^t \phi(\tau) d\tau$  exists and is finite. Then,

$$\lim_{t \rightarrow \infty} \phi(t) = 0.$$

We conclude the results for smooth systems with the standard ultimate boundedness theorem.

**Theorem A.5.** (?) Let  $x = 0$  be an equilibrium point of (A.1). Let  $V : \Omega \times [t_0, \infty) \rightarrow \mathbb{R}$  be a continuously differentiable function such that

$$\begin{aligned} \gamma_1(\|x\|) &\leq V(x) \leq \gamma_2(\|x\|) \\ \frac{\partial V}{\partial t} + \frac{\partial V}{\partial x} h(x, t) &\leq -W(x), \forall \|x\| \geq \mu > 0, \end{aligned}$$

$\forall t \geq 0$ ,  $x \in \Omega$ , where  $\gamma_1$ ,  $\gamma_2$  are class  $\mathcal{K}$  functions and  $W$  is a continuous positive definite function. Take  $r > 0$  such that  $B(0, r) \subseteq \Omega$  and suppose that  $\mu < \gamma_2^{-1}(\gamma_1(r))$ . Then, there exist a class  $\mathcal{K}_\infty$  function  $\gamma_3$  and for every initial state  $x(t_0)$  satisfying  $\|x(t_0)\| \leq \gamma_2^{-1}(\gamma_1(r))$ , there exists a  $T \geq 0$  such that

$$\begin{aligned} \|x(t)\| &\leq \gamma_3(\|x(t_0)\|), \forall t_0 \leq t \leq T, \\ \|x(t)\| &\leq \gamma_1^{-1}(\gamma_2(\mu)), \forall t > T. \end{aligned}$$

Moreover, if  $\Omega = \mathbb{R}^n$  and  $\gamma_1$  belongs to class  $\mathcal{K}_\infty$ , then the aforementioned result holds for any initial state  $x(t_0)$ , with no restriction on how large  $\mu$  is.

Note that the aforementioned results also apply for the case where  $x$  evolves in a manifold, by changing the  $\|\cdot\|$  metric to the respective manifold one.

## A.2 Systems with Discontinuous Right-Hand-Side

This section provides some equivalent results for non-smooth systems.

Consider the following differential equation with a discontinuous right-hand side:

$$\dot{x} = h(x, t), \quad (\text{A.4})$$

where  $h : \Omega \times [t_0, \infty) \rightarrow \mathbb{R}^n$ ,  $\Omega \subset \mathbb{R}^n$ , is Lebesgue measurable and locally essentially bounded, uniformly in  $t$ . The Filippov regularization of  $f$  is defined as [?]

$$\mathsf{K}[f](x, t) := \bigcap_{\delta > 0} \bigcap_{\mu(\bar{N})=0} \overline{\text{co}}(f(\mathcal{B}(x, \delta) \setminus \bar{N}), t), \quad (\text{A.5})$$

where  $\bigcap_{\mu(\bar{N})=0}$  is the intersection over all sets  $\bar{N}$  of Lebesgue measure zero, and  $\overline{\text{co}}(E)$  is the convex closure of a set  $E$ . We are interested in the Filippov solutions of (A.4):

**Definition A.6** ([?]). A function  $x : [t_0, t_1] \rightarrow \mathbb{R}^n$ , with  $t_1 > t_0$ , is called a Filippov solution of (A.4) on  $[t_0, t_1]$  if  $x(t)$  is absolutely continuous and if, for almost all  $t \in [t_0, t_1]$ , it satisfies  $\dot{x} \in \mathsf{K}[h](x, t)$ , where  $\mathsf{K}[h](x, t)$  is the Filippov regularization of  $h(x, t)$ .

The existence of Filippov solutions is given next.

**Proposition A.1** ([?]). Let  $\dot{x} \in \mathsf{K}[h](x, t)$ , where  $\mathsf{K}[h](x, t)$  is the Filippov regularization of  $h(x, t)$ . Let also  $h(x, t)$  be measurable and locally essentially bounded in  $x$  over  $\Omega$ , and measurable in  $t$  over  $[t_0, \infty)$ . Then, there exists a Filippov solution  $x : [t_0, t_1] \rightarrow \mathbb{R}^n$  of (A.4).

We next provide the definitions for regular functions and generalized gradients.

**Definition A.7** ([?]). Given a function  $h : \mathbb{R}^m \rightarrow \mathbb{R}^n$ , the right directional derivative of  $h$  at  $x \in \mathbb{R}^m$  in the direction of  $v \in \mathbb{R}^m$  is defined as

$$f'(x, v) := \lim_{t \rightarrow 0^+} \frac{f(x + tv) - f(x)}{t}.$$

Additionally, the generalized directional derivative of  $h$  at  $x$  in the direction of  $v$  is defined as

$$f^o(x, v) := \lim_{y \rightarrow x} \sup_{t \rightarrow 0^+} \frac{f(x + tv) - f(y)}{t}.$$

**Definition A.8** ([?]). A function  $h : \mathbb{R}^m \rightarrow \mathbb{R}^n$  is said to be regular at  $x \in \mathbb{R}^m$  if for all  $v \in \mathbb{R}^m$ , the right directional derivative of  $h$  at  $x$  in the direction of  $v$  exists and  $f'(x, v) = f^o(x, v)$ .

**Definition A.9** ([? ]). For a function  $V : \mathbb{R}^n \times [t_0, \infty) \rightarrow \mathbb{R}$  that is locally Lipschitz in  $(x, t)$ , define the generalized gradient of  $V$  at  $(x, t)$  by

$$\partial V(x, t) := \overline{\text{co}} \left\{ \lim \nabla V(x, t) : (x_i, t_i) \rightarrow (x, t), (x_i, t_i) \notin \Omega_V \right\},$$

where  $\Omega_V$  is the set of measure zero where the gradient of  $V$  is not defined.

**Lemma A.2** ([? ]). Let  $x(t)$  be a Filippov solution of (A.4) and  $V : \Omega \times [t_0, t_1] \rightarrow \mathbb{R}$  be a locally Lipschitz, regular function. Then  $V(x(t), t)$  is absolutely continuous,  $\dot{V}(x(t), t) = \frac{\partial}{\partial t} V(x(t), t)$  exists almost everywhere (a.e.), i.e., for almost all  $t \in [t_0, t_1]$ , and  $\dot{V}(x(t), t) \stackrel{a.e.}{\in} \dot{\tilde{V}}(x(t), t)$ , where

$$\dot{\tilde{V}} := \bigcap_{\xi \in \partial V(x, t)} \xi^\top \begin{bmatrix} K[f](x, t) \\ 1 \end{bmatrix}.$$

Finally, we provide the main invariance and stability result for the non-smooth type (A.4).

**Theorem A.6** ([? ]). For the system given in (A.4), let  $\Omega \subset \mathbb{R}^n$  be an open and connected set containing  $x = 0$  and suppose that  $f$  is Lebesgue measurable and  $x \mapsto f(x, t)$  is essentially locally bounded, uniformly in  $t$ . Let  $V : \Omega \times [t_0, t_1] \rightarrow \mathbb{R}$  be locally Lipschitz and regular such that  $W_1(x) \leq V(x, t) \leq W_2(x)$ ,  $\forall t \in [t_0, t_1]$ ,  $x \in \Omega$ , and

$$z \leq -W(x(t)), \quad \forall z \in \dot{\tilde{V}}(x(t), t), \quad t \in [t_0, t_1], \quad x \in \Omega,$$

where  $W_1$  and  $W_2$  are continuous positive definite functions and  $W$  is a continuous positive semi-definite on  $\Omega$ . Choose  $r > 0$  and  $c > 0$  such that  $\bar{\mathcal{B}}(0, r) \subset \Omega$  and  $c < \min_{\|x\|=r} W_1(x)$ . Then for all Filippov solutions  $x : [t_0, t_1] \rightarrow \mathbb{R}^n$  of (A.4), with  $x(t_0) \in \mathbb{D} := \{x \in \bar{\mathcal{B}}(0, r) : W_2(x) \leq c\}$ , it holds that  $t_1 = \infty$ ,  $x(t) \in \Omega$ ,  $\forall t \in [t_0, \infty)$ , and  $\lim_{t \rightarrow \infty} W(x(t)) = 0$ .

### A.3 Reduction Principle

Consider now the autonomous time-invariant version of (A.1)

$$\dot{x} = h_I(x), \tag{A.6}$$

where  $h_I : \Omega \rightarrow \mathbb{R}^n$  is sufficiently smooth, and  $x = 0$  is an equilibrium point, i.e.,  $h_I(0) = 0$ . Let  $A := \frac{\partial h_I}{\partial x} \Big|_{x=0}$  be the respective Jacobian matrix with  $n_+$  eigenvalues with positive real part,  $n_-$  eigenvalues with negative real part, and  $n_0$  eigenvalues with zero real part. Let  $T^c$  denote the linear (generalized) eigenspace of  $A$  corresponding to the union of the  $n_0$  eigenvalues on the imaginary axis. The next theorem provides the center manifold existence around the equilibrium point.

**Theorem A.7** ([? ]). *There is a locally defined smooth  $n_0$ -dimensional invariant manifold  $W^c(0)$  of (A.6) that is tangent to  $T^c$  at  $x = 0$ .*

System (A.6) can be written as

$$\dot{u} = Bu + g(u, v), \quad (\text{A.7a})$$

$$\dot{v} = Cv + h(u, v), \quad (\text{A.7b})$$

where  $u \in \mathbb{R}^{n_0}$ ,  $v \in \mathbb{R}^{n_++n_0}$ ,  $B \in \mathbb{R}^{n_0 \times n_0}$  is a matrix with all its  $n_0$  eigenvalues on the imaginary axis, while  $C \in \mathbb{R}^{(n_++n_i) \times (n_++n_i)}$  has no eigenvalue on the imaginary axis. The functions  $g$  and  $h$  have Taylor expansions starting with at least quadratic terms. The center manifold  $W^c$  of system (A.7) can be locally represented as a graph of a smooth function  $V(\cdot)$  as  $W^c = \{(u, v) : v = V(u)\}$ .

**Theorem A.8** ([? ]). *System (A.7) is locally topologically equivalent near the origin to the system*

$$\dot{u} = Bu + g(u, V(u)),$$

$$\dot{v} = Cv$$



## Appendix B

# Funnel Control

This Appendix provides preliminary background on funnel control and in particular, Prescribed Performance Control (PPC).

Funnel control describes the behavior where the output of the system is confined to a predefined funnel, as depicted in Fig. B.1. A special instance of funnel control, which this thesis focuses on, is Prescribed Performance Control, proposed in [?], and describes the behavior where a tracking error  $e : [t_0, \infty) \rightarrow \mathbb{R}$ , with  $t_0 \in \mathbb{R}_{\geq 0}$ , 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 inequalities:

$$-\rho_L(t) < e(t) < \rho_U(t), \quad \forall t \geq t_0,$$

where  $\rho_L(t), \rho_U(t)$  are smooth and bounded functions of time satisfying  $\lim_{t \rightarrow \infty} \rho_L(t) > 0$  and  $\lim_{t \rightarrow \infty} \rho_U(t) > 0$ , called performance functions.

Although the functions  $\rho_L(t), \rho_U(t)$  can be any bounded functions, it is usually preferred to be decaying functions. A particular interesting instance is the choice of the exponential performance functions  $\rho_i(t) := (\rho_{i,0} - \rho_{i,\infty}) \exp(-l_i t) + \rho_{i,\infty}$ , 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  $\rho_{L,\infty} := \lim_{t \rightarrow \infty} \rho_L(t), \rho_{U,\infty} := \lim_{t \rightarrow \infty} \rho_U(t)$  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  $e(t)$ .

The systems considered are general nonlinear systems of the form

$$\dot{x} = f(x, t) + g(x, t)u$$

where  $f, g : \mathbb{R}^n \times [t_0, \infty) \rightarrow \mathbb{R}^n$  are unknown functions satisfying certain continuity and controllability properties. We describe briefly the control design as well as the intuition behind it. Firstly, given a desired signal  $x_d$  and an error  $e = x - x_d \in \mathbb{R}^n$ , as well as a choice of performance functions  $\rho_j(t)$ , for each component  $j \in \{1, \dots, n\}$ , we define the normalized error

$$\xi := [\xi_1, \dots, \xi_n]^\top := P^{-1}e,$$

where  $P := \text{diag}\{\{\rho_i\}_{i \in \{1, \dots, n\}}\}$ , as well as the transformation  $\varepsilon : (-1, 1)^n \rightarrow \mathbb{R}^n$ , with

$$\varepsilon := \left[ \ln\left(\frac{1+\xi_1}{1-\xi_1}\right), \dots, \ln\left(\frac{1+\xi_n}{1-\xi_n}\right) \right]^\top.$$

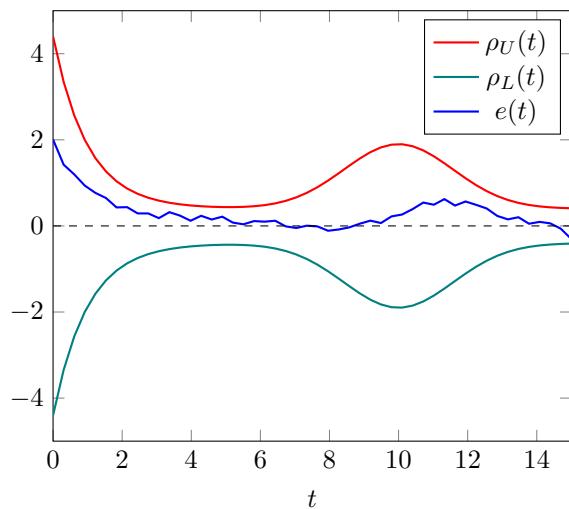
Intuitively, in order to guarantee  $\xi \in (-1, 1)^n$ , we need to guarantee that  $\varepsilon$  is bounded. This is attempted via the choice of control law

$$u = -kP^{-1}\frac{\partial \varepsilon}{\partial \xi}\varepsilon,$$

which acts as a barrier function, i.e., it increases to infinity as  $\xi$  approaches the boundary of  $(-1, 1)^n$ . Local existence of solutions guarantees that there exists a maximal time  $\tau > t_0$  such that  $\xi(t) \in (-1, 1)^n, \forall t \in [t_0, \tau]$ . Differentiating thus the well-defined (for  $t \in [t_0, \tau)$ ) Lyapunov function candidate  $V := \frac{1}{2}\|\varepsilon\|^2$  yields

$$\begin{aligned} \dot{V} &= \varepsilon^\top \frac{\partial \varepsilon}{\partial \xi}^\top P^{-1}(f(x, t) - kg(x, t)P^{-1}\frac{\partial \varepsilon}{\partial \xi}\varepsilon - \dot{x}_d - \dot{P}\xi) \\ &\leq -k\varepsilon^\top \frac{\partial \varepsilon}{\partial \xi}^\top P^{-1}g(x, t)P^{-1}\frac{\partial \varepsilon}{\partial \xi}\varepsilon + \left\| P^{-1}\frac{\partial \varepsilon}{\partial \xi}\varepsilon \right\| \|f(x, t) - \dot{x}_d - \dot{P}\xi\|. \end{aligned}$$

By assuming Lipschitz  $f(\cdot, t)$ , uniform bounded  $f(x, \cdot)$ , and positive definite  $g(x, t)$ , we use the boundedness of  $\xi(t) \in (-1, 1)^n, \forall t \in [t_0, \tau)$  to conclude that the second term above is bounded by a constant  $\bar{F}, \forall t \in [t_0, \tau)$ . Hence it turns out that  $\dot{V} < 0$  when  $\left\| \frac{\partial \varepsilon}{\partial \xi}\varepsilon \right\| > \frac{F}{k\lambda_{\min}(g(x, t))}$ , from which we can conclude the ultimate boundedness of  $\varepsilon(t)$  in a compact set, and hence that  $\tau = \infty$ .



**Figure B.1:** Illustration of funnel control, where the error  $e(t)$  is confined in the prescribed funnel defined by the functions  $\rho_L(t)$ ,  $\rho_U(t)$ .



## Appendix C

# Navigation Functions

This Appendix provides preliminary background on navigation functions.

Navigation functions, initially proposed in [? ] 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 versions introduced in [? ] and [? ].

### C.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.,  $\bar{\mathcal{B}}(q_i, r_i)$ ,  $i \in \mathcal{N}$ , operating in an open spherical workspace  $\mathcal{W} := \mathcal{B}(0, r_0)$  of radius  $r_0 \in \mathbb{R}_{>0}$ . Each robot has a destination point  $q_{d_i} \in \mathbb{R}^n$ ,  $i \in \mathcal{N}$ , and  $q_d := [q_{d_1}^\top, \dots, q_{d_N}^\top]^\top$ . Let  $\mathcal{F} \subset \mathbb{R}^n$  be a compact connected analytic manifold with boundary. A map  $\varphi : \mathcal{F} \rightarrow [0, 1]$  is a Multirobot Navigation Function (MRNF) if

1. It is analytic on  $\mathcal{F}$ ,
2. It has only one minimum at  $q_d \in \text{Int}(\mathcal{F})$ ,
3. Its Hessian at all critical points is full rank,
4.  $\lim_{q \rightarrow \partial \mathcal{F}} = 1 > \varphi(q')$ ,  $\forall q' \in \text{Int}(\mathcal{F})$ ,

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  $\gamma(q) := \|q - q_d\|^2$  is the goal function,  $G(q)$  is the obstacle function, and  $\kappa$  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 [? ]. 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  $(b_{R_j})_k := \sum_{(i,m) \in (R_j)_k} \beta_{i,m}$ , 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  $B_{R_j^C} := \prod_{m \in R_j^C} b_m$ . 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  $G_0 := \prod_{i \in \mathcal{N}} \{(r_0 - r_i)^2 - \|q_i\|^2\}$ , 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  $\kappa$  large enough, the negated gradient field  $-\nabla_q \varphi(q)$  leads to the destination configuration  $q_d$ , from almost all initial conditions [? ].

## C.2 Decentralized Navigation Functions (DNFs)

Consider now the class of *decentralized navigation functions*, which has the form  $\varphi_i : \mathcal{F}_i \rightarrow [0, 1]$ , with  $\varphi_i(q) := \frac{\gamma_i(q_i) + f_i(G_i)}{(\gamma_i(q_i)^{\lambda_i} + G_i(q))^{1/\kappa_i}}$ . The key difference in this case is the term  $G_i : \mathbb{R}^{3N} \rightarrow \mathbb{R}$  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 [?]:  $\beta_{ij} : \mathbb{R}^3 \times \mathbb{R}^3 \rightarrow \mathbb{R}$  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  $f_i : \mathbb{R} \rightarrow \mathbb{R}$  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  $\varphi_i$  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 [? ]. With the aforementioned tools, the control law for agent  $i$  is  $u_i = -k_i \frac{\partial \varphi_i(p)}{\partial p_i}$ , which, as shown in [? ], drives all agents to their goal positions and guarantees inter-agent collision-avoidance.



## Appendix D

# Nonlinear Model Predictive Control

This Appendix provides preliminary background on Nonlinear Model Predictive Control (NMPC).

Nonlinear Model Predictive Control has become an attractive feedback control strategy in applications where, except for stabilization or trajectory tracking of a system, certain state and input constraints must be also satisfied [? ? ? ]. In general, the NMPC is formulated as solving at each sampling time step an online Finite Horizon Optimal Control Problem (FHOCP) subject to system dynamics and state and input constraints. Based on measurements obtained at each sampling time step, the controller predicts the dynamic behavior of the system over a predictive horizon in the future and determines the input such that a predetermined open-loop performance objective is minimized. In order to incorporate feedback, the optimal open-loop input is implemented only until the next sampling time step. Using the new system state at the next sampling time step, the whole procedure (prediction and optimization) is repeated, moving the control and prediction horizon forward. Summarizing, a standard NMPC scheme works as follows

1. Obtain estimates of the states of the system.
2. Calculate a constraint-conforming optimal input minimizing the desired cost function over the prediction horizon using the system model and the current state estimate for prediction.
3. Implement the first part of the optimal input until the next sampling time step.
4. Go to step 1.

More technically, consider the stabilization problem for a class of systems described by the following nonlinear dynamical system:

$$\dot{x} = f(x, u), \quad (\text{D.1})$$

with  $f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$  is locally Lipschitz continuous and satisfies  $f(0, 0) = 0$ . The system is subject to the following input and state constraints:

$$u \in \mathcal{U} \subset \mathbb{R}^m, x \in \mathcal{X} \subset \mathbb{R}^n,$$

respectively, where  $\mathcal{U}$  is compact and  $\mathcal{X}$  is connected, and  $(0, 0) \in \mathcal{X} \times \mathcal{U}$ . Denote by  $h \in \mathbb{R}_{>0}$ ,  $T_p > h$  the sampling step and the finite prediction horizon, respectively. Consider a sequence of sampling times  $\{t_j\}, j \in \mathbb{N}$ . Then, at every sampling time step  $t_j$ , the following FHOCP is solved:

$$\min_{\hat{u}(\cdot)} \left\{ E(\hat{x}(t_j + T_p)) + \int_{t_j}^{t_j + T_p} F(\hat{x}(s), \hat{u}(s)) \right\} \quad (\text{D.2a})$$

subject to: (D.2b)

$$\dot{\hat{x}}(s) = f(\hat{x}(s), \hat{u}(s)), \quad \hat{x}(t_j) = x(t_j), \quad (\text{D.2c})$$

$$\hat{x}(s) \in \mathcal{X}, \quad \hat{u}(s) \in \mathcal{U}, \quad s \in [t_j, t_j + T_p], \quad (\text{D.2d})$$

$$\hat{x}(t_j + T_p) \in \mathcal{E}, \quad (\text{D.2e})$$

where  $F$  and  $E$  are running and terminal costs, respectively, usually chosen as

$$\begin{aligned} F(x, u) &:= x^\top Q x + u^\top R u \\ E(x) &:= x^\top P x, \end{aligned}$$

where  $P \in \mathbb{R}^{n \times n}$  and  $R \in \mathbb{R}^{m \times m}$  are positive definite matrix and  $Q \in \mathbb{R}^{n \times n}$  a positive semi-definite weight matrix. The hat  $\hat{\cdot}$  denotes predicted variables (internal to the controller), i.e.,  $\hat{x}(\cdot)$  is the solution of (D.2) driven by the input  $\hat{u}(\cdot) : [t_j, t_j + T_p] \rightarrow \mathcal{U}$  with the initial condition  $x(t_j)$ . Due to the fact that a finite prediction horizon is used, the actual closed-loop input and state trajectories will differ from the predicted open-loop trajectories, even if no model plant mismatch and no disturbances are present. This is the key difference between standard control strategies, where the feedback control law is obtained a priori, and NMPC, where the feedback control law is obtained online. Since a finite horizon approach is used, the terminal set  $\mathcal{E}$  is introduced and is appropriately designed as described in [? ?] in order to guarantee the stability of the closed-loop system. The solution to FHOCP (D.2) is denoted by  $\hat{u}^*(\cdot, x(t_j))$ . It defines the open-loop input that is applied to the system until the next sampling time step  $t_{j+1}$  as

$$u(s; x(t_j)) = \hat{u}^*(s; x(t_j)), \quad s \in [t_j, t_{j+1}).$$

The control input  $u(s; x(t_j))$  is a feedback control law, since it is re-calculated at each sampling instant using the new state information. The solution of

(D.1) from an initial state  $x(t_j)$ , applying an input  $u : [t_j, t_{j+1}] \rightarrow \mathbb{R}^m$  is denoted by  $x(s; u(\cdot), x(t_j))$ ,  $s \in [t_j, t_{j+1}]$ . We next define the admissible control input.

**Definition D.1.** A control input  $u : [0, T_p] \rightarrow \mathbb{R}^m$  for a state  $x_0$  is called *admissible*, if

1. it is piecewise continuous;
2.  $u(s) \in \mathcal{U}, \forall s \in [0, T_p]$ ;
3.  $x(s; u(\cdot), x_0) \in \mathcal{X}, \forall s \in [0, T_p]$ ;
4.  $x(T_p; u(\cdot), x_0) \in \mathcal{E}$ ;

The following theorem states the stability results for NMPC.

**Theorem D.1** ([? ]). Consider system (D.1) and suppose that

- the NMPC open-loop optimal control problem (D.2) is feasible at  $t = 0$
- the terminal set  $\mathcal{E} \subseteq \mathcal{X}$  is closed with  $0 \in \mathcal{E}$  and the terminal cost  $E(x)$  is cont. differentiable and positive definite
- the terminal set and terminal cost are chosen such that,  $\forall x \in \mathcal{E}$ , there exists an (admissible) input  $u_{\mathcal{E}} : [0, \delta] \rightarrow \mathcal{U}$  such that  $x(s) \in \mathcal{E}$ ,  $\forall s \in [0, \delta]$ , and

$$\frac{\partial E}{\partial x} f(x(s), u_{\mathcal{E}}(s)) + F(x(s), u_{\mathcal{E}}(s)) \leq 0, \forall s \in [0, \delta]$$

Then, the closed-loop system is asymptotically stable with the region of attraction being the sets of states for which the an admissible input exists.



# Appendix E

# Graph Theory and Rigid Frameworks

This Appendix provides preliminary background on graph theory and rigid frameworks.

## E.1 Graph Theory

An *undirected graph*  $\mathcal{G}$  is a pair  $(\mathcal{N}, \mathcal{E})$ , where  $\mathcal{N} := \{1, \dots, N\}$  is a finite set of  $N \in \mathbb{N}$  nodes, representing a team of agents, and  $\mathcal{E} \subseteq \{(i, j) \in \mathcal{N}^2 : i < j\}$ , 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  $\mathcal{N}_i := \{j \in \mathcal{N} \text{ s.t. } (i, j) \in \mathcal{E}\}$ . A *directed* graph is formed by the edges being  $\mathcal{E} \subseteq \{(i, j) \in \mathcal{N}^2 : i \neq j\}$ , i.e.,  $(i, j) \in \mathcal{E}$  does not necessarily imply  $(j, i) \in \mathcal{E}$ . The complete graph of  $N$  nodes,  $N > 1$ , is denoted by  $\mathcal{K}_N$ .

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  $k \in \mathcal{K} := \{1, \dots, K\}$  of the graph's edge set, we define the *incidence matrix*  $D(\mathcal{G}) := [d_{ik}] \in \mathbb{R}^{N \times M}$  as

$$d_{ik} = \begin{cases} 1, & \text{if } i \text{ is the head of edge } k, \\ -1, & \text{if } i \text{ is the tail of edge } k, \\ 0, & \text{otherwise.} \end{cases}$$

Moreover, the incidence matrix satisfies the following important property.

**Property E.1.** Consider the incidence matrix  $D(\mathcal{G})$  of an arbitrary graph  $\mathcal{G}$  as  $D(\mathcal{G}) = [d_1, \dots, d_N]^\top \in \mathbb{R}^{N \times M}$ , where  $d_i^\top \in \mathbb{R}^M$  are the rows of  $D$ . Then it holds that  $\sum_{i \in \{1, \dots, N\}} d_i = 0$ , i.e., the rows of the incidence matrix sum up to zero.

**Lemma E.1.** [?, 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  $\Delta \in \mathbb{R}^{N \times N}$ .

## E.2 Rigidity Theory

We review here some necessary notions on distance and bearing rigidity.

### E.2.1 Distance Rigidity in $\mathbb{R}^n$

Let an undirected graph  $(\mathcal{G}, \mathcal{E})$  and  $p_i \in \mathbb{R}^n$  be the position of node  $i \in \mathcal{N}$ . Let also  $p := [p_1^\top, \dots, p_N^\top]^\top \in \mathbb{R}^{nN}$ . A *framework* in  $\mathbb{R}^n$  is the tuple  $(\mathcal{G}, p)$ . We assume that the nodes are not collinear in  $\mathbb{R}^n$  or occupy the exact same position.

**Definition E.1.** Two frameworks  $(\mathcal{G}, p)$  and  $(\mathcal{G}, p')$  are *distance equivalent* if  $\|p_i - p_j\| = \|p'_i - p'_j\|$ ,  $\forall (i, j) \in \mathcal{E}$ .

**Definition E.2.** Two frameworks  $(\mathcal{G}, p)$  and  $(\mathcal{G}, p')$  are *distance congruent* if  $\|p_i - p_j\| = \|p'_i - p'_j\|$ ,  $\forall i, j \in \mathcal{N}$ .

**Definition E.3.** A framework  $(\mathcal{G}, p)$  is *distance rigid* if there exists a constant  $\epsilon \in \mathbb{R}_{>0}$  such that any framework  $(\mathcal{G}, p')$  that is distance equivalent to  $(\mathcal{G}, p)$  and satisfies  $\|p' - p\| < \epsilon$  is also distance congruent to it.

**Definition E.4.** A framework  $(\mathcal{G}, p)$  is *globally distance rigid* if an arbitrary framework that is distance equivalent to  $(\mathcal{G}, p)$  is also distance congruent to it.

Consider an orientation assigned to  $\mathcal{G}$ , as described in the previous section, as well as a numbering  $\mathcal{K} := \{1, \dots, K\}$ ,  $K := |\mathcal{E}|$ . Let the interneighbor distances be expressed by  $p_k := p_i - p_j$ ,  $\forall k = (i, j) \in \mathcal{E}$ . Define the distance function  $F_D : \mathbb{R}^{Nn} \rightarrow \mathbb{R}^{Kn}$  as

$$F_D(p) := \frac{1}{2} [\|p_1\|^2, \dots, \|p_K\|]^\top$$

The *distance rigidity matrix* is defined as the respective Jacobian

$$\mathcal{R}_D(p) := \frac{\partial F_D(p)}{\partial p} \in \mathbb{R}^{K \times Nn}$$

Let now  $\delta p \in \mathbb{R}^N n$  be a variation of the configuration  $p$ . If  $\mathcal{R}_D \delta p = 0$ , then  $\delta p$  is an *infinitesimal distance motion* of  $(\mathcal{G}, p)$ . An infinitesimal distance motion is *trivial* if it corresponds only to a translation and a rotation of the entire framework.

**Definition E.5.** A framework is *infinitesimally rigid* if all the infinitesimal distance motions are trivial.

**Lemma E.2.** *A framework  $(\mathcal{G}, p)$  is infinitesimally rigid if*

- $\text{rank}(\mathcal{R}_D(p)) \geq 2N - 3$ , if  $n = 2$
- $\text{rank}(\mathcal{R}_D(p)) \geq 3N - 6$ , if  $n = 3$

### E.2.2 Bearing Rigidity in $\mathbb{SE}(3)$

Let a directed graph  $(\mathcal{G}, \mathcal{E})$  and now  $x_i := (p_i, R_i) \in \mathbb{SE}(3)$  be the *pose* of node  $i \in \mathcal{N}$ . Let also  $x := (x_1, \dots, x_N) \in \mathbb{SE}^{3N}$ . A *framework* in  $\mathbb{SE}(3)$  is the tuple  $(\mathcal{G}, x)$ . We assume that the nodes are not collinear in  $\mathbb{R}^n$  or occupy the exact same position.

Let  $b_k : \mathbb{SE}(3)^2 \rightarrow \mathbb{S}^2$  be the relative bearings, with

$$b_k(x_i, x_j) := R_i^\top \frac{p_i - p_j}{\|p_i - p_j\|}, \forall k = (i, j) \in \mathcal{E}$$

as well as  $b_{\mathcal{G}}(x) := [b_1^\top, \dots, b_N^\top]^\top \in \mathbb{S}^{2N}$ .

**Definition E.6.** Two frameworks  $(\mathcal{G}, x)$  and  $(\mathcal{G}, x')$  are bearing equivalent if

$$R_i^\top \frac{p_i - p_j}{\|p_i - p_j\|} = R_i'^\top \frac{p'_i - p'_j}{\|p'_i - p'_j\|}, \forall (i, j) \in \mathcal{E}$$

**Definition E.7.** Two frameworks  $(\mathcal{G}, x)$  and  $(\mathcal{G}, x')$  are bearing congruent if

$$R_i^\top \frac{p_i - p_j}{\|p_i - p_j\|} = R_i'^\top \frac{p'_i - p'_j}{\|p'_i - p'_j\|}, \forall i, j \in \mathcal{N}$$

**Definition E.8.** A framework  $(\mathcal{G}, x)$  is globally bearing rigid in  $\mathbb{SE}(3)$  if every framework which is equivalent to it is also congruent to it.

**Definition E.9.** A framework  $(\mathcal{G}, x)$  is bearing rigid in  $\mathbb{SE}(3)$  if there exists a neighborhood  $S \subset \mathbb{SE}(3)^N$  of  $x$  such that

$$b_{\mathcal{K}_N}^{-1}(b_{\mathcal{K}_N}(x)) \cap S = b_{\mathcal{G}}^{-1}(b_{\mathcal{G}}(x)) \cap S$$

Similar to distance rigidity, the respective rigidity matrix is defined as  $\mathcal{R}_{\mathcal{G}}(x) := \frac{\partial b_{\mathcal{G}}(x)}{\partial x}$ , whose null-space describes the infinitesimal bearing motions of  $(\mathcal{G}, x)$  in  $\mathbb{SE}(3)$ . An infinitesimal bearing motion in  $\mathbb{SE}(3)$  is trivial if it corresponds only to a translation, rotation, or dilation of the framework. The infinitesimal bearing rigidity in  $\mathbb{SE}(3)$  follows from Definition E.5.

**Lemma E.3.** *A framework  $(\mathcal{G}, x)$  is infinitesimally bearing rigid in  $\mathbb{SE}(3)$  if and only if  $\text{null}(\mathcal{R}_{\mathcal{G}}(x)) = \text{null}(\mathcal{R}_{\mathcal{K}_N}(x))$ , or, equivalently,  $\text{rank}(\mathcal{R}_{\mathcal{G}}(x)) = 6N - 7$ .*

## Appendix F

# Temporal Logics-based Task Specification

This Appendix provides preliminary background on specifications expressed as temporal logic formulas.

## F.1 Task Specification in LTL

**Definition F.1.** A transition system  $(\mathcal{TS})$  is a tuple  $\mathcal{T} := (\Pi, \Pi_0, \rightarrow, \Psi, \mathcal{L})$ , where  $\Pi$  is a discrete finite set of states,  $\Pi_0$  is a discrete finite set of initial states,  $\rightarrow \subseteq \Pi \times \Pi$  is a transition relation,  $\Psi$  is a discrete set of atomic propositions<sup>1</sup>, 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 F.2.** A *run* of a  $\mathcal{TS}$  is an infinite sequence

$$r_{\mathcal{TS}} := \pi_1 \pi_2 \pi_3 \dots,$$

with  $\pi_1 \in \Pi_0$ ,  $\pi_i \in \Pi$ ,  $\forall i \in \mathbb{N}$ .

**Definition F.3.** A *word*  $w_{\mathcal{TS}}$  of a run  $r_{\mathcal{TS}}$  is the infinite sequence

$$w_{\mathcal{TS}}(r_{\mathcal{TS}}) = w_1 w_2 w_3 \dots,$$

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

We focus on the task specification  $\Phi$  given as a Linear Temporal Logic (LTL) formula. The basic ingredients of a LTL formula are a set of atomic propositions  $\Psi$  and several boolean and temporal operators. LTL formulas are formed according to the following grammar [?]:  $\Phi ::= \text{true} \mid a \mid \Phi_1 \wedge \Phi_2 \mid \neg \Phi \mid \bigcirc \Phi \mid \Phi_1 \cup \Phi_2$ , where  $a \in \Psi$ ,  $\Phi_1$  and  $\Phi_2$  are LTL formulas and  $\bigcirc$ ,  $\cup$  are the *next* and *until* operators, respectively. Definitions of other useful

---

<sup>1</sup>boolean variables that are either true or false in a given state

operators like  $\square$  (*always*),  $\diamond$  (*eventually*) and  $\Rightarrow$  (*implication*) are omitted and can be found at [? ]. The semantics of LTL are defined over infinite words over  $2^\Psi$ . Intuitively, an atomic proposition  $\psi \in \Psi$  is satisfied on a word  $w_{\mathcal{TS}} = w_1 w_2 w_3 \dots$  if it holds at its first position  $w_1$ , i.e.  $\psi \in w_1$ , denoted as  $w_{\mathcal{TS}} \models \Phi$ . Formula  $\bigcirc \Phi$  holds true if  $\Phi$  is satisfied on the word suffix that begins in the next position  $w_2$ , whereas  $\Phi_1 \cup \Phi_2$  states that  $\Phi_1$  has to be true until  $\Phi_2$  becomes true. Finally,  $\diamond \Phi$  and  $\square \Phi$  holds on  $w_{\mathcal{TS}}$  eventually and always, respectively. For a full definition of the LTL semantics, the reader is referred to [? ].

A LTL formula  $\Phi$  over a set of atomic propositions  $\Psi$  can be translated to a Büchi Automaton  $\mathcal{A}_\Phi$  [? ]. Then, by calculating the product of the transition system  $\mathcal{TS} = (\Pi, \Pi_0, \rightarrow, \Psi, \mathcal{L})$  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  $\Phi$ . These runs can then be projected back to  $\mathcal{TS}$ , providing paths over  $\Pi$  that satisfy  $\Phi$ . More details regarding the technique can be found in [? ].

## F.2 Task Specification in MITL

**Definition F.4.** A *Weighted Transition System* ( $\mathcal{WTS}$ ) is a tuple

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

where  $\Pi$  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,  $\Psi$  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 F.5.** [? ] The time sequence  $t_1 t_2 t_3 \dots$  is an infinite sequence of time values  $t_j \in \mathbb{R}_{\geq 0}, \forall j \in \mathbb{N}$ , 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 F.6.** Let  $\Psi$  be a finite set of atomic propositions. A timed word  $w$  over  $\Psi$  is an infinite sequence

$$w = (w_1, t_1)(w_2, t_2), \dots,$$

where  $w_1 w_2 \dots$  is an infinite word over  $2^\Psi$  and  $t_1 t_2 \dots$  is a time sequence according to Definition F.5.

**Definition F.7.** A *timed run* of a  $\mathcal{WTS}$  is an infinite sequence

$$r_{\mathcal{WTS}} = (r_1, t_1)(r_2, t_2) \dots,$$

such that  $r_1 \in \Pi_0$ , 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  $r_{\mathcal{WTS}}$  generates the timed word

$$w_{\mathcal{WTS}}(r_{\mathcal{WTS}}) = w_1(r_1)w_2(r_2)\dots = (\mathcal{L}(r_1), t_1)(\mathcal{L}(r_2), t_2)\dots$$

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

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

$$\Phi := p \mid \neg\Phi \mid \Phi_1 \wedge \Phi_2 \mid \bigcirc_I \Phi \mid \diamond_I \Phi \mid \square_I \Phi \mid \Phi_1 \mathcal{U}_I \Phi_2,$$

where  $p \in \Psi$ , and  $\bigcirc, \diamond, \square$  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  $\mathcal{WTS}$ .

**Definition F.8.** [? ?] Given a run  $r_{\mathcal{WTS}} = (r_1, t_1)(r_2, t_2)\dots$  of a WTS and a MITL formula  $\Phi$ , we define  $(r_{\mathcal{WTS}}, j) \models \phi, j \in \mathbb{N}$  ( $r_{\mathcal{WTS}}$  satisfies  $\Phi$  at  $j$ ) as follows:

$$\begin{aligned} (r_{\mathcal{WTS}}, j) \models p &\Leftrightarrow p \in \mathcal{L}(r_j), \\ (r_{\mathcal{WTS}}, j) \models \neg\Phi &\Leftrightarrow (r_{\mathcal{WTS}}, j) \not\models \Phi \\ (r_{\mathcal{WTS}}, j) \models \Phi_1 \wedge \Phi_2 &\Leftrightarrow (r_{\mathcal{WTS}}, j) \models \Phi_1 \text{ and } (r_{\mathcal{WTS}}, j) \models \Phi_2 \\ (r_{\mathcal{WTS}}, j) \models \bigcirc_I \Phi &\Leftrightarrow (r_{\mathcal{WTS}}, j+1) \models \Phi \text{ and } t_{j+1} - t_j \in I \\ (r_{\mathcal{WTS}}, j) \models \Phi_1 \mathcal{U}_I \Phi_2 &\Leftrightarrow \exists k, j, \text{ with } j \leq k, \text{s.t. } (r_{\mathcal{WTS}}, k) \models \Phi_2, t_k - t_j \in I \\ &\quad \text{and } (r_{\mathcal{WTS}}, m) \models \Phi_1, \forall m \in \{j, \dots, k\} \end{aligned}$$

Also,  $\diamond_I \Phi = \top \mathcal{U}_I \Phi$  and  $\square_I \Phi = \neg \diamond_I \neg \Phi$ . The sequence  $r_{\mathcal{WTS}}$  satisfies  $\Phi$ , denoted as  $r_{\mathcal{WTS}} \models \Phi$ , if and only if  $(r_{\mathcal{WTS}}, 1) \models \Phi$ .

We provide next a description of *Timed Büchi Automata (TBA)*, originally proposed in [? ]. Let  $\mathbf{CL} := \{\mathbf{cl}_1, \dots, \mathbf{cl}_{|\mathbf{CL}|}\}$  be a finite set of *clocks*. The set of *clock constraints*  $\Phi(\mathbf{CL})$  is defined by the grammar:

$$\phi ::= \top \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid \mathbf{cl} \bowtie \psi,$$

where  $\text{cl} \in \text{CL}$  is a clock,  $\psi \in \mathbb{Q}$  is a clock constraint, and  $\bowtie \in \{<, >, \geq, \leq, =\}$ . A *clock valuation* is a mapping  $v : \text{CL} \rightarrow \mathbb{R}$  that assigns a value to each clock. A clock  $\text{cl}_i$  has valuation  $v_i$  for  $i \in \{1, \dots, |\text{CL}|\}$ . Given  $v := (v_1, \dots, v_{|\text{CL}|})$  and  $t \in \mathbb{R}_{\geq 0}$ , we denote by  $v \models \phi$  and  $t \models \phi$  the fact that the valuation  $v$  and the time instant  $t$ , respectively, satisfy the clock constraint  $\phi$ .

**Definition F.9.** A *Timed Büchi Automaton* is a tuple

$$\mathcal{A}_t := (Q, Q_0, \text{CL}, \Psi, E, F),$$

where  $Q$  is a finite set of locations,  $Q_0 \subseteq Q$  is the set of initial locations,  $\text{CL}$  is a finite set of clocks,  $\Psi$  is a finite set of atomic propositions that defines the input alphabet  $2^\Psi$ ,  $E \subset Q \times \Phi(\text{CL}) \times 2^{\text{CL}} \times 2^\Psi \times Q$  gives the set of edges of the form  $e = (q, g, R, \alpha, q')$ , where  $q, q'$  are the source and target locations,  $g$  is the guard of edge,  $R$  is a set of clocks to be reset upon executing the edge, and  $\alpha$  is an input string; finally,  $F \subseteq Q$  is a set of accepting locations.

A state of  $\mathcal{A}_t$  is a pair  $(q, v) \in \mathbb{R} \times \mathbb{R}^{|\text{CL}|}$ . The initial state of  $\mathcal{A}_t$  is  $(q_0, 0_{|\text{CL}|})$ , with  $q_0 \in Q_0$ . Given two states  $(q, v)$ ,  $(q', v')$ , and an edge  $e = (q, g, R, \alpha, q')$ , there exists a *discrete transition*  $(q, v) \xrightarrow{e} (q', v')$  if  $v \models g$ . Moreover,  $v'_i = 0$ ,  $\forall \text{cl}_i \in R$ , and  $v'_i = v_i$ ,  $\forall \text{cl}_i \notin R$ . Given  $\delta \in \mathbb{R}$ , there exists a *time transition*  $(q, v) \xrightarrow{\delta} (q', v')$  if  $q = q'$  and  $v' = v + \delta$  (component-wise summation). We write  $(q, v) \xrightarrow{\delta} \xrightarrow{e} (q', v')$  if there exists  $q'', v''$  such that  $(q, v) \xrightarrow{\delta} (q'', v'')$  and  $(q'', v'') \xrightarrow{e} (q', v')$ , with  $q'' = q$ .

An infinite run of  $\mathcal{A}_t$  starting at a state  $(q_1, v_1)$  is an infinite sequence of time and discrete transitions  $(q_1, v_1) \xrightarrow{\delta_1} (q'_1, v'_1) \xrightarrow{e_1} (q_2, v_2) \xrightarrow{\delta_2} (q'_2, v'_2) \dots$ , where  $e_i = (q_i, g_i, R_i, \sigma_i, q'_i)$ ,  $\forall i \in \mathbb{N}$ . This run corresponds to the timed word  $w_t = (\sigma_1, \tau_1)(\sigma_2, \tau_2)\dots$ , with  $\tau_{i+1} = \tau_i + \delta_i$ ,  $\forall i \in \mathbb{N}$ . The run is called accepting if  $q_j \in F$  for infinitely many  $j \in \mathbb{N}$ . A timed word is called *accepting* if there exists an accepting run associated with it. The problem of deciding the language emptiness of a given TBA is PSPACE-complete [? ]. In other words, an accepting run of a given TBA can be synthesized, if one exists. Any timed formula  $\Phi$  over  $\Psi$  originating from the decidable fragment of timed logics (*e.g.*, MITL, discrete-time MTL, finite MTL<sup>2</sup>, coFlat-MTL, Bounded-MTL [? ? ]) can be algorithmically translated into a TBA with input alphabet  $2^\Psi$ , such that the language of timed words that satisfy  $\varphi$  is the language of timed words produced by the TBA.

---

<sup>2</sup>In this case, the generated Timed Automaton will have finite accepting runs.

## Appendix G

# Useful Properties

This appendix provides some additional technical properties that are useful throughout the thesis.

**Proposition G.1.** Let  $f : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}$ , 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,  $f(x) \geq f(0) = 0, \forall x \in \mathbb{R}_{\geq 0}$ .  $\square$

**Proposition G.2.** [?] Let  $R_1, R_2 \in \mathbb{SO}(3)$ , and  $e_R := S^{-1}(R_1^\top R_2 - R_2^\top R_1)$ . Then  $\|e_R\|^2 := \|R_1 - R_2\|_{\mathbb{F}}^2 \left(1 - \frac{1}{8}\|R_1 - R_2\|_{\mathbb{F}}^2\right)$ .

**Proposition G.3.** Let  $R_1, R_2 \in \mathbb{SO}(3)$ . Then, for the rotation matrix  $R_2^\top R_1 \in \mathbb{SO}(3)$  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.

Let  $x, y \in \mathbb{R}^3$ ,  $R \in \mathbb{SO}(3)$ , and  $A \in \mathbb{R}^{3 \times 3}$ . Then the following hold [?]:

- $x^\top S(y)x = 0$ ;
- $S(Rx) = RS(x)R^\top$ ;
- $-\frac{1}{2}\text{tr}[S(x)S(y)] = x^\top y$ ;
- $\text{tr}[AS(x)] = \frac{1}{2}\text{tr}[S(x)(A - A^\top)] = -x^\top S^{-1}(A - A^\top)$ .

We provide next some useful properties of linear algebra.

**Definition G.1.** A matrix  $A \in \mathbb{R}^{n \times m}$  is *left equivalent* (or *row equivalent*) to a matrix  $B \in \mathbb{R}^{n \times m}$  if and only if there exists an invertible matrix  $P \in \mathbb{R}^{n \times n}$  such that  $A = PB$ .

The following propositions can be proved:

**Proposition G.4.** Let  $A, B \in \mathbb{R}^{n \times m}$ . Then  $A$  and  $B$  are left equivalent if and only if they have the same nullspace  $\text{null}(A) = \text{null}(B)$ , i.e.,  $Ax = 0 \Leftrightarrow Bx = 0$ , for any  $x \in \mathbb{R}^m$ .

**Proposition G.5.** Let  $A \in \mathbb{R}^{n \times m}$ , and  $B := KA$ , where  $K \in \mathbb{R}^{n \times n}$  is an invertible matrix. Then it holds that

$$A^\dagger A = B^\dagger B.$$

**Proposition G.6.** Let  $A, B \in \mathbb{R}^{n \times m}$  such that  $\text{range}(A^\top) = \text{null}(B)$ . Then it holds that

$$A^\dagger A + B^\dagger B = I_m.$$

**Proposition G.7.** 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  $\lambda_1, \lambda_2, \lambda_3$ , 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 G.8.** [?] Consider two planar ellipsoids

$$\mathcal{A} = \{z \in \mathbb{R}^3 \text{ s.t. } z^\top A(t)z \leq 0\}, \quad \mathcal{B} = \{z \in \mathbb{R}^3 \text{ s.t. } z^\top B(t)z \leq 0\},$$

with  $z = [p^\top 1]^\top$ ,  $p \in \mathbb{R}^2$ , and  $A, B : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^{3 \times 3}$  terms that describe their motion in 2D space. Given their characteristic polynomial  $f(\lambda) = \det(\lambda A - B)$ , which has degree 3, the following hold:

- (i)  $\exists \lambda^* > 0$  s.t.  $f(\lambda^*) = 0$ , i.e, the polynomial  $f(\lambda)$  has always one positive real root,
- (ii)  $\mathcal{A} \cap \mathcal{B} = \emptyset$  if and only if the characteristic equation  $f(\lambda) = 0$  has two distinct negative roots, i.e.,  $\exists \lambda_1^*, \lambda_2^* < 0$ , with  $\lambda_1^* \neq \lambda_2^*$ , and  $f(\lambda_1^*) = f(\lambda_2^*) = 0$ .
- (iii)  $\mathcal{A} \cap \mathcal{B} \neq \emptyset$  and  $\text{Int}\mathcal{A} \cap \text{Int}\mathcal{B} = \emptyset$ , i.e.,  $\mathcal{A}$  and  $\mathcal{B}$  touch externally, if and only if and only if  $f(\lambda) = 0$  has a negative root with multiplicity 2.

# 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] M. Krstic, I. Kanellakopoulos, and P. Kokotovic. Nonlinear and Adaptive Control Design. *Publisher: Wiley New York*, 1995.
- [13] Christos K Verginis, Matteo Mastellaro, and Dimos V Dimarogonas. Robust quaternion-based cooperative manipulation without force/torque information. *IFAC-PapersOnLine*, 50(1):1754–1759, 2017.
- [14] Christos K Verginis, Matteo Mastellaro, and Dimos V Dimarogonas. Robust cooperative manipulation without force/torque measurements: Control design and experiments. *IEEE Transactions on Control Systems Technology*, 28(3):713–729, 2020.
- [15] A. Nikou, C. K. Verginis, S. Heshmati-alamdar, 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.
- [16] Christos K Verginis, Alexandros Nikou, and Dimos V Dimarogonas. Communication-based decentralized cooperative object transportation using nonlinear model predictive control. *European control conference (ECC)*, pages 733–738, 2018.
- [17] C. K. Verginis, W. S. Cortez, and D. V. Dimarogonas. Adaptive cooperative manipulation with rolling contacts. *American Control Conference (ACC), Denver, Colardo, USA*, 2020.
- [18] C. K. Verginis, W. S. Cortez, and D. V. Dimarogonas. Decentralized adaptive control for cooperative manipulation with rolling contacts. *Under preparation*, 2020.
- [19] Christos K Verginis, Alexandros Nikou, and Dimos V Dimarogonas. Robust formation control in  $se(3)$  for tree-graph structures with prescribed transient and steady state performance. *Automatica*, 103:538–548, 2019.
- [20] C. K. Verginis and D. V. Dimarogonas. Energy-optimal cooperative manipulation via provable internal-force regulation. *IEEE International Conference on Robotics and Automation (ICRA), Paris, France*, 2020.

- [21] Christos K. Verginis, Daniel Zelazo, and Dimos V. Dimarogonas. Cooperative manipulation via internal force regulation: A rigidity theory perspective. *To appear in the IEEE Transactions on Control of Network Systems*, 2022.
- [22] C. K. Verginis and D. V. Dimarogonas. Adaptive robot navigation with collision avoidance subject to 2nd-order uncertain dynamics. *Under Review*, 2019.
- [23] C. K. Verginis and D. V. Dimarogonas. Adaptive leader-follower coordination of lagrangian multi-agentsystems under transient constraints. *IEEE Conference on Decision and Control (CDC)*, pages 3833–3838, 2020.
- [24] C. K. Verginis and D. V. Dimarogonas. Closed-form barrier functions for multi-agent ellipsoidal systems with uncertain lagrangian dynamics. *IEEE Control Systems Letters (LCSS)*, pages 727–732, 2019.
- [25] 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.
- [26] 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*, pages 2222–2227, 2017.
- [27] C. K. Verginis and D. V. Dimarogonas. Distributed cooperative manipulation under timed temporal specifications. *American Control Conference (ACC), Seattle, WA, USA*, pages 1358–1363, 2017.
- [28] Christos K Verginis and Dimos V Dimarogonas. Timed abstractions for distributed cooperative manipulation. *Autonomous Robots*, 42(4):781–799, 2018.
- [29] Christos K Verginis and Dimos V Dimarogonas. Multi-agent motion planning and object transportation under high level goals. *IFAC-PapersOnLine*, 50(1):15816–15821, 2017.
- [30] C. K. Verginis and D. V. Dimarogonas. Motion and cooperative transportation planning for multi-agent systems under temporal logic formulas. *BOSCH AI Conference, Arxiv Link: <https://arxiv.org/pdf/1803.01579.pdf>*, 2018.

- [31] Christos K Verginis, Constantinos Vrohidis, Charalampos P Bechlioulis, Kostas J Kyriakopoulos, and Dimos V Dimarogonas. Reconfigurable motion planning and control in obstacle cluttered environments under timed temporal tasks. *IEEE International Conference on Robotics and Automation (ICRA)*, pages 951–957, 2019.
- [32] C. K. Verginis, D. V. Dimarogonas, and L. E. Kavraki. Sampling-based motion planning for uncertain high-dimensional systems via adaptive control. *To appear in the Workshop on the Algorithmic Foundations of Robotics (WAFR), Oulu, Finland*, 2020.
- [33] Christos K Verginis and Dimos V Dimarogonas. Asymptotic stability of uncertain lagrangian systems with prescribed transient response. *IEEE Conference on Decision and Control (CDC)*, pages 7037–7042, 2019.
- [34] C. K. Verginis and D. V. Dimarogonas. Asymptotic tracking of second-order nonsmooth feedback stabilizable unknown systems with prescribed transient response. *under review*, 2019.
- [35] Alexandros Nikou, Christos K Verginis, and Dimos V Dimarogonas. Robust distance-based formation control of multiple rigid bodies with orientation alignment. *IFAC-PapersOnLine*, 50(1):15458–15463, 2017.
- [36] Christos K Verginis, Alexandras Nikou, and Dimos V Dimarogonas. Position and orientation based formation control of multiple rigid bodies with collision avoidance and connectivity maintenance. *IEEE Conference on Decision and Control (CDC)*, pages 411–416, 2017.
- [37] Lars Lindemann, Christos K Verginis, and Dimos V Dimarogonas. Prescribed performance control for signal temporal logic specifications. *IEEE Conference on Decision and Control (CDC)*, pages 2997–3002, 2017.
- [38] A. Nikou, S. Heshmati-alamdar, C. K. Verginis, and D. V. Dimarogonas. Decentralized abstractions and timed constrained planning of a general class of coupled multi-agent systems. *IEEE Conference on Decision and Control (CDC), Melbourne, Australia*, pages 990–995, 2017.
- [39] Jieqiang Wei, Christos Verginis, Junfeng Wu, Dimos V Dimarogonas, Henrik Sandberg, and Karl H Johansson. Asymptotic and finite-time almost global attitude tracking: representations free approach. *2018 European Control Conference (ECC)*, pages 3126–3131, 2018.

- [40] T. Pan, C. K. Verginis, A. M. Wells, D. V. Dimarogonas, and L. E. Kavraki. Augmenting control policies with motion planning for robust and safe multi-robot navigation. *submitted to the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020.
- [41] N. Lissandrini, C. K. Verginis, P. Roque, A. Cenedese, and D. V. Dimarogonas. Decentralized nonlinear mpc for robust cooperative manipulation by heterogeneous aerial-ground robots. *submitted to the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020.
- [42] 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.
- [43] T. G. Sugar and V. Kumar. Control of cooperating mobile manipulators. *IEEE Transactions on robotics and automation*, 18(1):94–103, 2002.
- [44] O. Khatib, K. Yokoi, K. Chang, D. Ruspini, R. Holmberg, and A. Casal. Decentralized cooperation between multiple manipulators. *IEEE International Workshop on Robot and Human Communication*, pages 183–188, 1996.
- [45] 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.
- [46] 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.
- [47] 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.
- [48] 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.
- [49] J. T. Wen and K. Kreutz-Delgado. Motion and force control of multiple robotic manipulators. *Automatica*, 28(4):729–743, 1992.
- [50] 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.
- [51] C. D. Kopf. Dynamic two arm hybrid position/force control. *Robotics and Autonomous Systems*, 5(4):369–376, 1989.
  - [52] F. Caccavale, P. Chiacchio, and S. Chiaverini. Task-space regulation of cooperative manipulators. *Automatica*, 36(6):879–887, 2000.
  - [53] 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.
  - [54] 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.
  - [55] 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.
  - [56] 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.
  - [57] 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.
  - [58] J. Szewczyk, F. Plumet, and P. Bidaud. Planning and controlling cooperating robots through distributed impedance. *Journal of Robotic Systems*, 19(6):283–297, 2002.
  - [59] A. Tsiamis, C. K. Verginis, 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.
  - [60] 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.

- [61] A.-N. Ponce-Hinestrosa, J.-A. Castro-Castro, H.-I. Guerrero-Reyes, V. Parra-Vega, and E. Olguín-Díaz. 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.
- [62] 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.
- [63] 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.
- [64] 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.
- [65] Alessandro Marino. Distributed adaptive control of networked cooperative mobile manipulators. *IEEE Transactions on Control Systems Technology*, 26(5):1646–1660, 2017.
- [66] 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.
- [67] 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.
- [68] 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.
- [69] 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.
- [70] Alessandro Marino and Francesco Pierri. A two stage approach for distributed cooperative manipulation of an unknown object without explicit communication and unknown number of robots. *Robotics and Autonomous Systems*, 103:122–133, 2018.

- [71] Alessandro Marino. A decentralized adaptive control for tightly connected networked lagrangian systems. *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 4656–4661, 2017.
- [72] Davide Ortenzi, Rajkumar Muthusamy, Alessandro Freddi, Andrea Monterù, and Ville Kyrki. Dual-arm cooperative manipulation under joint limit constraints. *Robotics and Autonomous Systems*, 99:110–120, 2018.
- [73] 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.
- [74] 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.
- [75] 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.
- [76] 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.
- [77] 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.
- [78] H. Bai and J. T. Wen. Cooperative load transport: A formation-control perspective. *IEEE Transactions on Robotics*, 26(4):742–750, 2010.
- [79] Dominik Sieber and Sandra Hirche. Human-guided multirobot cooperative manipulation. *IEEE Transactions on Control Systems Technology*, 27(4):1492–1509, 2018.
- [80] 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.
- [81] Martina Lippi and Alessandro Marino. Cooperative object transportation by multiple ground and aerial vehicles: Modeling and planning. *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1084–1090, 2018.

- [82] Dario Sanalitro, Heitor J Savino, Marco Tognon, Juan Cortés, and Antonio Franchi. Full-pose manipulation control of a cable-suspended load with multiple uavs under uncertainties. *IEEE Robotics and Automation Letters*, 5(2):2185–2191, 2020.
- [83] Chiara Gabellieri, Marco Tognon, Dario Sanalitro, Lucia Pallottino, and Antonio Franchi. A study on force-based collaboration in swarms. *Swarm Intelligence*, pages 1–26, 2019.
- [84] D. Koditschek and E. Rimon. Robot navigation functions on manifolds with boundary. *Advances in Applied Mathematics*, 11(4):412–442, 1990.
- [85] B. Siciliano L. Sciavicco, L. Villani, and G. Oriolo. Robotics: modelling, planning and control. *Springer Science & Business Media*, 2010.
- [86] K. Oliveira and M. Morari. Contractive model predictive control for constrained nonlinear systems. *IEEE Transactions on Automatic Control*, 45(6):1053–1071, 2000.
- [87] 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.
- [88] H. Chen and F. Allgöwer. A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. *Automatica*, 34(10):1205–1217, 1998.
- [89] 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.
- [90] F. Fontes. A general framework to design stabilizing nonlinear model predictive controllers. *Systems and Control Letters*, 42(2):127–143, 2001.
- [91] L. Grüne and J. Pannek. Nonlinear Model Predictive Control. *Springer London*, 2011.
- [92] 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.
- [93] B. Kouvaritakis and M. Cannon. Nonlinear predictive control: Theory and practice. *Iet*, (61), 2001.

- [94] Janick V Frasch, Andrew Gray, Mario Zanon, Hans Joachim Ferreau, Sebastian Sager, Francesco Borrelli, and Moritz Diehl. An auto-generated nonlinear mpc algorithm for real-time obstacle avoidance of ground vehicles. *2013 European Control Conference (ECC)*, pages 4136–4141, 2013.
- [95] 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.
- [96] Ryuta Ozawa and Kenji Tahara. Grasp and dexterous manipulation of multi-fingered robotic hands: a review from a control view point. *Advanced Robotics*, 31(19-20):1030–1050, 2017.
- [97] Jeffrey Kerr and Bernard Roth. Analysis of multifingered hands. *The International Journal of Robotics Research*, 4(4):3–17, 1986.
- [98] Arlene BA Cole, John Edmond Hauser, and Sosale Shankara Sastry. Kinematics and control of multifingered hands with rolling contact. *IEEE Transactions on Automatic Control*, 34(4):398–404, 1989.
- [99] Yongxiang Fan, Liting Sun, Minghui Zheng, Wei Gao, and Masayoshi Tomizuka. Robust dexterous manipulation under object dynamics uncertainties. *IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, pages 613–619, 2017.
- [100] A. Caldas, A. Micaelli, M. Grossard, M. Makarov, P. Rodriguez-Ayerbe, and D. Dumur. Object-level impedance control for dexterous manipulation with contact uncertainties using an LMI-based approach. *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3668–3674, 2015.
- [101] T Wimböck, C Ott, A Albu-Schäffer, and G Hirzinger. Comparison of object-level grasp controllers for dynamic dexterous manipulation. *The International Journal of Robotics Research*, 31(1):3–23, 2012.
- [102] Kenji Tahara, Suguru Arimoto, and Morio Yoshida. Dynamic object manipulation using a virtual frame by a triple soft-fingered robotic hand. *IEEE International Conference on Robotics and Automation (ICRA)*, pages 4322–4327, 2010.
- [103] Akihiro Kawamura, Kenji Tahara, Ryo Kurazume, and Tsutomu Hasegawa. Dynamic grasping of an arbitrary polyhedral object. *Robotica*, 31(4):511–523, 2013.

- [104] Wenceslao Shaw-Cortez, Denny Oetomo, Chris Manzie, and Peter Choong. Tactile-based blind grasping: Trajectory tracking and disturbance rejection for in-hand manipulation of unknown objects. *American Control Conference (ACC)*, pages 693–698, 2019.
- [105] S. Ueki, H. Kawasaki, and T. Mouri. Adaptive control of multi-fingered robot hand using quaternion. *17th IFAC World Congress*, pages 6757–6762, 2008.
- [106] S. Ueki, H. Kawasaki, T. Mouri, and A. Kaneshige. Object manipulation based on robust and adaptive control by hemispherical soft fingertips. *18th IFAC World Congress*, 18:14654–14659, 2011.
- [107] C.C. Cheah, H.Y. Han, S. Kawamura, and S. Arimoto. Grasping and position control for multi-fingered robot hands with uncertain jacobian matrices. *IEEE International Conference on Robotics and Automation (ICRA)*, 3:2403–2408, 1998.
- [108] Papat Fungtammasan and Tetsuyou Watanabe. Grasp input optimization taking contact position and object information uncertainties into consideration. *IEEE Transactions on Robotics*, 28(5):1170–1177, 2012.
- [109] 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.
- [110] E. Lavretsky and K. Wise. Robust and Adaptive Control: With Aerospace Applications. *Springer Science and Business Media*, 2012.
- [111] 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.
- [112] 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.
- [113] 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.
- [114] 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.

- [115] C. Bechlioulis and G. Rovithakis. Robust Adaptive Control of Feedback Linearizable MIMO Nonlinear Systems with Prescribed Performance. *IEEE Transactions on Automatic Control (TAC)*, 53(9):2090–2099, 2008.
- [116] Alexandros Nikou and Dimos V Dimarogonas. Decentralized tube-based model predictive control of uncertain nonlinear multiagent systems. *International Journal of Robust and Nonlinear Control*, 29(10):2799–2818, 2019.
- [117] Richard M Murray, Zexiang Li, and S Shankar Sastry. *A mathematical introduction to robotic manipulation*. CRC press, 1994.
- [118] Kaiyu Hang, Miao Li, Johannes A. Stork, Yasemin Bekiroglu, Florian T. Pokorny, Aude Billard, and Danica Kragic. Hierarchical fingertip space: A unified framework for grasp planning and in-hand grasp adaptation. *IEEE Transactions on robotics*, 32(4):960–972, 2016.
- [119] Wenceslao Shaw-Cortez, Denny Oetomo, Chris Manzie, and Peter Choong. Grasp constraint satisfaction for object manipulation using robotic hands. *2018 IEEE Conference on Decision and Control (CDC)*, pages 415–420, 2018.
- [120] W. Shaw-Cortez, D. Oetomo, C. Manzie, and P. Choong. Tactile-based blind grasping: A discrete-time object manipulation controller for robotic hands. *IEEE Robot. Autom. Lett.*, 3(2), 2018.
- [121] Romeo Ortega, Antonio Loria, Per Johan Nicklasson, and Hebertt Sira-Ramirez. *Passivity-Based Control of Euler-Lagrange Systems: Mechanical, Electrical, and Electromechanical Applications*. Communications and Control Engineering. Springer, 1998.
- [122] 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.
- [123] 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.
- [124] 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.

- [125] 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.
- [126] 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.
- [127] B. Anderson, C. Yu, B. Fidan, and J. Hendrickx. Rigid Graph Control Architectures for Autonomous Formations. *IEEE Control Systems*, 28:48–63, 2008.
- [128] 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.
- [129] 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.
- [130] 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.
- [131] 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.
- [132] 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.
- [133] K. Oh and H. Ahn. Formation control of mobile agents based on inter-agent distance dynamics. *Automatica*, 47(10):2306–2312, 2011.
- [134] 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.
- [135] 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.

- [136] 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.
- [137] 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.
- [138] 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.
- [139] 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.
- [140] Tolga Eren. Formation shape control based on bearing rigidity. *International Journal of Control (IJC)*, 85(9):1361–1379, 2012.
- [141] 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.
- [142] 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.
- [143] 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.
- [144] 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.
- [145] Hector Garcia De Marina, Bayu Jayawardhana, and Ming Cao. Distributed rotational and translational maneuvering of rigid formations and their applications. *IEEE Transactions on Robotics*, 32(3):684–697, 2016.
- [146] Zhiyong Sun. Distributed stabilization control of rigid formations with prescribed orientations. *Cooperative Coordination and Formation Control for Multi-agent Systems*, pages 81–99, 2018.

- [147] Xudong Chen, Mohamed-Ali Belabbas, and Tamer Başar. Global stabilization of triangulated formations. *SIAM Journal on Control and Optimization*, 55(1):172–199, 2017.
- [148] Shaoshuai Mou, Mohamed-Ali Belabbas, A Stephen Morse, Zhiyong Sun, and Brian DO Anderson. Undirected rigid formations are problematic. *IEEE Transactions on Automatic Control*, 61(10):2821–2836, 2016.
- [149] Daniel Zelazo, Antonio Franchi, Heinrich H Bülfhoff, and Paolo Robuffo Giordano. Decentralized rigidity maintenance control with range measurements for multi-robot systems. *The International Journal of Robotics Research*, 34(1):105–128, 2015.
- [150] Yu-Ping Tian and Qin Wang. Global stabilization of rigid formations in the plane. *Automatica*, 49(5):1436–1441, 2013.
- [151] Kwang-Kyo Oh and Hyo-Sung Ahn. Distance-based undirected formations of single-integrator and double-integrator modeled agents in n-dimensional space. *International Journal of Robust and Nonlinear Control*, 24(12):1809–1820, 2014.
- [152] Laura Krick, Mireille E Broucke, and Bruce A Francis. Stabilisation of infinitesimally rigid formations of multi-robot networks. *International Journal of control*, 82(3):423–439, 2009.
- [153] Brian DO Anderson, Changbin Yu, Baris Fidan, and Julien M Hendrickx. Rigid graph control architectures for autonomous formations. *IEEE Control Systems Magazine*, 28(6):48–63, 2008.
- [154] Eren Togla, O. K. Goldenberg, W. Whiteley, Y. R. Yang, A. S. Morse, B. D. O. Anderson, and P. N. Belhumeur. Rigidity, computation, and randomization in network localization. *IEEE INFOCOM*, 4:2673–2684, 2014.
- [155] G. Mao, B. Fidan, and B. D. O. Anderson. Wireless sensor network localization techniques. *Computer networks*, 51(10):2529–2553, 2007.
- [156] J Aspnes, T. Eren, D. K. Goldenberg, A. S. Morse, W. Whiteley, Y. R. Yang, , B. D. O. Anderson, and P. N. Belhumeur. A theory of network localization. *IEEE Transactions on Mobile Computing*, 5(12):1663–1678, 2006.
- [157] S. Zhao and D. Zelazo. Bearing rigidity theory and its applications for control and estimation of network systems: Life beyond distance rigidity. *IEEE Control Systems Magazine*, 39(2):66–83, April 2019.

- [158] S. Zhao and D. Zelazo. Bearing rigidity and almost global bearing-only formation stabilization. *IEEE Transactions on Automatic Control*, 61(5):1255–1268, May 2016.
- [159] R. Tron, L. Carbone, F. Dellaert, and K. Daniilidis. Rigid components identification and rigidity control in bearing-only localization using the graph cycle basis. *American Control Conference (ACC)*, pages 3911–3918, July 2015.
- [160] Tolga Eren. Formation shape control based on bearing rigidity. *International Journal of Control*, 85(9):1361–1379, 2012.
- [161] A. Bishop, I. Shames, and B. Anderson. Stabilization of rigid formations with direction-only constraints. *Proceedings of the IEEE Conference on Decision and Control (CDC), Orlando, FL, USA*, pages 746–752, 2011.
- [162] D. Zelazo, A. Franchi, and P. R. Giordano. Rigidity theory in  $\text{se}(2)$  for unscaled relative position estimation using only bearing measurements. *2014 European Control Conference (ECC)*, pages 2703–2708, June 2014.
- [163] G. Michieletto, A. Cenedese, and A. Franchi. Bearing rigidity theory in  $\text{se}(3)$ . *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 5950–5955, Dec 2016.
- [164] Ian D Walker, Robert A Freeman, and Steven I Marcus. Analysis of motion and internal loading of objects grasped by multiple cooperating manipulators. *The International journal of robotics research*, 10(4):396–409, 1991.
- [165] D. Williams and O. Khatib. The virtual linkage: a model for internal forces in multi-grasp manipulation. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 1:1025–1030, 1993.
- [166] J. H. Chung, B.-Y. Yi W. K., and Kim. Analysis of internal loading at multiple robotic systems. *Journal of mechanical science and technology*, 19(8):1554–1567, 2005.
- [167] 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.

- [168] Farhad Mehdifar, Charalampos P. Bechlioulis, Farzad Hashemzadeh, and Mahdi Baradarannia. Prescribed performance distance-based formation control of multi-agent systems (extended version), 2019.
- [169] 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.
- [170] 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.
- [171] Leonard Asimow and Ben Roth. The rigidity of graphs. *Transactions of the American Mathematical Society*, 245:279–289, 1978.
- [172] Hyo-Sung Ahn. *Formation Control: Approaches to Distributed Agents*. Springer International Publishing, 2020.
- [173] Shiyu Zhao and Daniel Zelazo. Localizability and distributed protocols for bearing-based network localization in arbitrary dimensions. *Automatica*, 69:334–341, 2016.
- [174] Giulia Michieletto, Daniel Zelazo, and Angelo Cenedese. A unified dissertation on bearing rigidity theory. *CoRR*, abs/1902.03101, 2019.
- [175] R E. Kalaba and Firdaus Udwadia. Equations of motion for nonholonomic, constrained dynamical systems via gauss's principle. *ASME. J. Appl. Mech*, 60(3):662–668, 1993.
- [176] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.
- [177] Firdaus Udwadia and R E. Kalaba. A new perspective on constrained motion. *Proceedings of The Royal Society A: Mathematical, Physical and Engineering Sciences*, 439:407–410, 1992.
- [178] Arthur Albert. *Regression and the Moore-Penrose Pseudoinverse*. Elsevier, 1972.
- [179] Nicholas IM Gould. On practical conditions for the existence and uniqueness of solutions to the general equality quadratic programming problem. *Mathematical Programming*, 32(1):90–99, 1985.
- [180] R. Eric, P. N. S. Surya F., and Marc. Vrep: a versatile and scalable robot simulation framework. *Proceedings of The International Conference on Intelligent Robots and Systems (IROS)*, 2013.

- [181] Li Wang, Aaron D Ames, and Magnus Egerstedt. Safety barrier certificates for collisions-free multirobot systems. *IEEE Transactions on Robotics*, 33(3):661–674, 2017.
- [182] Lorenzo Sabattini, Cristian Secchi, and Nikhil Chopra. Decentralized connectivity maintenance for networked lagrangian dynamical systems with collision avoidance. *Asian Journal of Control*, 17(1), 2015.
- [183] Daniel Claes and Karl Tuyls. Multi robot collision avoidance in a shared workspace. *Autonomous Robots*, pages 1–22, 2018.
- [184] Chengtao Cai, Chunsheng Yang, Qidan Zhu, and Yanhua Liang. Collision avoidance in multi-robot systems. *International Conference on Mechatronics and Automation (ICMA)*, pages 2795–2800, 2007.
- [185] Shihua Li and Xiangyu Wang. Finite-time consensus and collision avoidance control algorithms for multiple auvs. *Automatica*, 49(11):3359–3367, 2013.
- [186] Michael M Zavlanos, Magnus B Egerstedt, and George J Pappas. Graph-theoretic connectivity control of mobile robot networks. *Proceedings of the IEEE*, 99(9):1525–1540, 2011.
- [187] Lorenzo Sabattini, Cristian Secchi, Nikhil Chopra, and Andrea Gasparri. Distributed control of multirobot systems with global connectivity maintenance. *IEEE Transactions on Robotics*, 29(5):1326–1332, 2013.
- [188] Meng Ji and Magnus Egerstedt. Distributed coordination control of multiagent systems while preserving connectedness. *IEEE Transactions on Robotics*, 23(4):693–703, 2007.
- [189] Chao Sun, Guoqiang Hu, Lihua Xie, and Magnus Egerstedt. Robust finite-time connectivity preserving coordination of second-order multiagent systems. *Automatica*, 89:21–27, 2018.
- [190] Yiannis Kantaros and Michael M Zavlanos. Global planning for multirobot communication networks in complex environments. *IEEE Transactions on Robotics*, 32(5):1045–1061, 2016.
- [191] Matthew Turpin, Nathan Michael, and Vijay Kumar. Capt: Concurrent assignment and planning of trajectories for multiple robots. *The International Journal of Robotics Research*, 33(1):98–112, 2014.
- [192] Michael M Zavlanos and George J Pappas. Distributed connectivity control of mobile networks. *IEEE Transactions on Robotics*, 24(6):1416–1428, 2008.

- [193] Michael M Zavlanos and George J Pappas. Potential fields for maintaining connectivity of mobile networks. *IEEE Transactions on Robotics*, 23(4):812–816, 2007.
- [194] Jean-Claude Latombe. *Robot motion planning*, volume 124. Springer Science & Business Media, 2012.
- [195] Howie M Choset, Seth Hutchinson, Kevin M Lynch, George Kantor, Wolfram Burgard, Lydia E Kavraki, and Sebastian Thrun. *Principles of robot motion: theory, algorithms, and implementation*. MIT press, 2005.
- [196] Steven M LaValle. *Planning algorithms*. Cambridge university press, 2006.
- [197] Elon Rimon and Daniel E Koditschek. Exact robot navigation using artificial potential functions. *IEEE Transactions on Robotics and Automation*, 8(5):501–518, 1992.
- [198] J-O Kim and Pradeep K Khosla. Real-time obstacle avoidance using harmonic potential functions. *IEEE Transactions on Robotics and Automation*, 8(3):338–349, 1992.
- [199] Ioannis Filippidis and Kostas J Kyriakopoulos. Adjustable navigation functions for unknown sphere worlds. *2011 50th IEEE Conference on Decision and Control and European Control Conference*, pages 4276–4281, 2011.
- [200] Ioannis F Filippidis and Kostas J Kyriakopoulos. Navigation functions for everywhere partially sufficiently curved worlds. *2012 IEEE International Conference on Robotics and Automation*, pages 2115–2120, 2012.
- [201] Savvas G Loizou and Kostas J Kyriakopoulos. Closed loop navigation for multiple holonomic vehicles. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 3:2861–2866, 2002.
- [202] Dimos V Dimarogonas, Savvas G Loizou, Kostas J Kyriakopoulos, and Michael M Zavlanos. A feedback stabilization and collision avoidance scheme for multiple independent non-point agents. *Automatica*, 42(2):229–243, 2006.
- [203] Giannis Roussos and Kostas J Kyriakopoulos. Decentralized and prioritized navigation and collision avoidance for multiple mobile robots. *Distributed Autonomous Robotic Systems*, pages 189–202, 2013.

- [204] Santiago Paternain, Daniel E Koditschek, and Alejandro Ribeiro. Navigation functions for convex potentials in a space with convex obstacles. *IEEE Transactions on Automatic Control*, 63(9):2944–2959, 2017.
- [205] Herbert G Tanner and Amit Kumar. Towards decentralization of multi-robot navigation functions. *IEEE International Conference on Robotics and Automation (ICRA)*, 4:4132, 2005.
- [206] Savvas G Loizou, Herbert G Tanner, Vijay Kumar, and Kostas J Kyriakopoulos. Closed loop navigation for mobile agents in dynamic environments. *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)(Cat. No. 03CH37453)*, 4:3769–3774, 2003.
- [207] 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.
- [208] Savvas G Loizou. Closed form navigation functions based on harmonic potentials. *2011 50th IEEE Conference on Decision and Control and European Control Conference*, pages 6361–6366, 2011.
- [209] S. G. Loizou. The navigation transformation. *IEEE Transactions on Robotics*, 33(6):1516–1523, 2017.
- [210] Panagiotis Vlantis, Constantinos Vrohidis, Charalampos P Bechlioulis, and Kostas J Kyriakopoulos. Robot navigation in complex workspaces using harmonic maps. *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1726–1731, 2018.
- [211] Stephen Waydo and Richard M Murray. Vehicle motion planning using stream functions. *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)*, 2:2484–2491, 2003.
- [212] Paweł Szulczyński, Dariusz Pazderski, and Krzysztof Kozłowski. Real-time obstacle avoidance using harmonic potential functions. *Journal of Automation Mobile Robotics and Intelligent Systems*, 5:59–66, 2011.
- [213] Savvas G Loizou. The multi-agent navigation transformation: Tuning-free multi-robot navigation. *Robotics: Science and Systems*, 6:1516–1523, 2014.
- [214] Constantinos Vrohidis, Panagiotis Vlantis, Charalampos P Bechlioulis, and Kostas J Kyriakopoulos. Prescribed time scale robot navigation. *IEEE Robotics and Automation Letters*, 3(2):1191–1198, 2018.

- [215] Alexandros Filotheou, Alexandros Nikou, and Dimos V Dimarogonas. Decentralized control of uncertain multi-agent systems with connectivity maintenance and collision avoidance. *European Control Conference*, 2018.
- [216] José M Mendes Filho, Eric Lucet, and David Filliat. Real-time distributed receding horizon motion planning and control for mobile multi-robot dynamic systems. *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 657–663, 2017.
- [217] Jur Van Den Berg, Jamie Snape, Stephen J Guy, and Dinesh Manocha. Reciprocal collision avoidance with acceleration-velocity obstacles. *IEEE International Conference on Robotics and Automation (ICRA)*, 2011.
- [218] Steven Roelofsen, Denis Gillet, and Alcherio Martinoli. Collision avoidance with limited field of view sensing: A velocity obstacle approach. *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1922–1927, 2017.
- [219] Erick J Rodríguez-Seda, Dušan M Stipanović, and Mark W Spong. Collision avoidance control with sensing uncertainties. *Proceedings of the 2011 American Control Conference*, pages 3363–3368, 2011.
- [220] Sotiris Stavridis, Dimitrios Papageorgiou, and Zoe Doulgeri. Dynamical system based robotic motion generation with obstacle avoidance. *IEEE Robotics and Automation Letters*, 2(2):712–718, 2017.
- [221] Victoria Grushkovskaya and Alexander Zuyev. Obstacle avoidance problem for second degree nonholonomic systems. *IEEE Conference on Decision and Control (CDC)*, pages 1500–1505, 2018.
- [222] Jan Maximilian Montenbruck, Mathias Bürger, and Frank Allgöwer. Navigation and obstacle avoidance via backstepping for mechanical systems with drift in the closed loop. *2015 American Control Conference (ACC)*, pages 625–630, 2015.
- [223] Omur Arslan and Daniel E Koditschek. Exact robot navigation using power diagrams. *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1–8, 2016.
- [224] Omur Arslan, Dan P Guralnik, and Daniel E Koditschek. Coordinated robot navigation via hierarchical clustering. *IEEE Transactions on Robotics*, 32(2):352–371, 2016.

- [225] Soulaimane Berkane, Andrea Bisoffi, and Dimos V Dimarogonas. A hybrid controller for obstacle avoidance in an n-dimensional euclidean space. *2019 European Control Conference (ECC)*, 2019.
- [226] Muhamad Mujahed and Barbel Mertsching. The admissible gap (ag) method for reactive collision avoidance. *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1916–1921, 2017.
- [227] Lukas Huber, Aude Billard, and Jean-Jacques Slotine. Avoidance of convex and concave obstacles with convergence ensured through contraction. *2019 IEEE International Conference on Robotics and Automation*, 2019.
- [228] Daniel E Koditschek. The control of natural motion in mechanical systems. *Journal of dynamic systems, measurement, and control*, 113(4):547–551, 1991.
- [229] Omur Arslan and Daniel E Koditschek. Smooth extensions of feedback motion planners via reference governors. *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4414–4421, 2017.
- [230] Jiangping Hu and Gang Feng. Distributed tracking control of leader–follower multi-agent systems under noisy measurement. *Automatica*, 46(8):1382–1387, 2010.
- [231] Xianfu Zhang, Lu Liu, and Gang Feng. Leader–follower consensus of time-varying nonlinear multi-agent systems. *Automatica*, 52:8–14, 2015.
- [232] Zhongkui Li, Xiangdong Liu, Wei Ren, and Lihua Xie. Distributed tracking control for linear multiagent systems with a leader of bounded unknown input. *IEEE Transactions on Automatic Control*, 58(2), 2013.
- [233] Wei Liu and Jie Huang. Adaptive leader-following consensus for a class of higher-order nonlinear multi-agent systems with directed switching networks. *Automatica*, 79:84–92, 2017.
- [234] Meng Guo, Jana Tumova, and Dimos V Dimarogonas. Communication-free multi-agent control under local temporal tasks and relative-distance constraints. *IEEE Transactions on Automatic Control*, 61(12):3948–3962, 2016.

- [235] Jie Mei, Wei Ren, and Guangfu Ma. Distributed coordinated tracking with a dynamic leader for multiple euler-lagrange systems. *IEEE Transactions on Automatic Control*, 56(6):1415–1421, 2011.
- [236] Tove Gustavi, Dimos V Dimarogonas, Magnus Egerstedt, and Xiaoming Hu. Sufficient conditions for connectivity maintenance and rendezvous in leader–follower networks. *Automatica*, 46(1):133–139, 2010.
- [237] Christos K Verginis, Charalampos P Bechlioulis, Dimos V Dimarogonas, and Kostas J Kyriakopoulos. Decentralized 2-d control of vehicular platoons under limited visual feedback. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3566–3571, 2015.
- [238] Christos K Verginis, Charalampos P Bechlioulis, Dimos V Dimarogonas, and Kostas J Kyriakopoulos. Robust distributed control protocols for large vehicular platoons with prescribed transient and steady-state performance. *IEEE Transactions on Control Systems Technology*, 26(1):299–304, 2018.
- [239] Sadra Sadraddini, S Sivarajani, Vijay Gupta, and Calin Belta. Provably safe cruise control of vehicular platoons. *IEEE Control Systems Letters*, 1(2):262–267, 2017.
- [240] Wei Ren, Randal W Beard, and Ella M Atkins. A survey of consensus problems in multi-agent coordination. *American Control Conference*, pages 1859–1864, 2005.
- [241] Kwang-Kyo Oh, Myoung-Chul Park, and Hyo-Sung Ahn. A survey of multi-agent formation control. *Automatica*, 53:424–440, 2015.
- [242] Magnus Egerstedt and Xiaoming Hu. Formation constrained multi-agent control. *IEEE transactions on robotics and automation*, 17(6):947–951, 2001.
- [243] Lorenzo Sabattini, Cristian Secchi, and Cesare Fantuzzi. Arbitrarily shaped formations of mobile robots: artificial potential fields and coordinate transformation. *Autonomous Robots*, 30(4):385, 2011.
- [244] Silvia Mastellone, Dušan M Stipanović, Christopher R Graunke, Koji A Intlekofer, and Mark W Spong. Formation control and collision avoidance for multi-agent non-holonomic systems: Theory and experiments. *The International Journal of Robotics Research*, 27(1):107–126, 2008.

- [245] Yan Yan and Gregory S Chirikjian. Closed-form characterization of the minkowski sum and difference of two ellipsoids. *Geometriae Dedicata*, 177(1):103–128, 2015.
- [246] Elon Rimon and Stephen P Boyd. Obstacle collision detection using best ellipsoid fit. *Journ. Intell. Rob. Syst.*, 18(2):105–126, 1997.
- [247] Andrew Best, Sahil Narang, and Dinesh Manocha. Real-time reciprocal collision avoidance with elliptical agents. *IEEE International Conference on Robotics and Automation*, pages 298–305, 2016.
- [248] Muhammad Zakiullah Romdlony and Bayu Jayawardhana. Stabilization with guaranteed safety using control lyapunov–barrier function. *Automatica*, 66:39–47, 2016.
- [249] Xiangru Xu. Constrained control of input–output linearizable systems using control sharing barrier functions. *Automatica*, 87:195–201, 2018.
- [250] Dimitra Panagou, Dušan M Stipanović, and Petros G Voulgaris. Distributed coordination control for multi-robot networks using lyapunov-like barrier functions. *IEEE Transactions on Automatic Control*, 61(3):617–632, 2015.
- [251] Lars Lindemann and Dimos V Dimarogonas. Control barrier functions for signal temporal logic tasks. *IEEE control systems letters*, 3(1):96–101, 2019.
- [252] C Canudas De Wit, Hans Olsson, Karl Johan Astrom, and Pablo Lischinsky. A new model for control of systems with friction. *IEEE Transactions on automatic control*, 40(3):419–425, 1995.
- [253] C Makkar, WE Dixon, WG Sawyer, and G Hu. A new continuously differentiable friction model for control systems design. *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics.*, pages 600–605, 2005.
- [254] Nathan P Koenig and Andrew Howard. Design and use paradigms for gazebo, an open-source multi-robot simulator. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 4, 2004.
- [255] Rafael Kelly, Victor Santibáñez Davila, and Julio Antonio Loría Perez. *Control of robot manipulators in joint space*. Springer-Verlag, 2005.
- [256] H. Kress-Gazit, G. Fainekos, and G. J. Pappas. Temporal-Logic-Based Reactive Mission and Motion Planning. *IEEE Transactions on Robotics (TRO)*, 25(6):1370–1381, 2009.

- [257] 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.
- [258] 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.
- [259] M. Kloetzer and C. Belta. Automatic Deployment of Distributed Teams of Robots From Temporal Motion Specifications. *IEEE Transactions on Robotics (TRO)*, 26(1):48–61, 2010.
- [260] C. Baier, J.P. Katoen, and K. G. Larsen. Principles of model checking. *MIT Press*, 2008.
- [261] 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.
- [262] 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.
- [263] 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.
- [264] 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.
- [265] 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.
- [266] 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.
- [267] 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.
- [268] G. Fainekos, A. Girard, K.G. Hadas, and G. J. Pappas. Temporal Logic Motion Planning for Dynamic Robots. *Automatica*, 45(2):343–352, 2009.
- [269] 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.
- [270] 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.
- [271] 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.
- [272] 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.
- [273] 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.
- [274] 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.
- [275] 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.
- [276] S. Liu, C. Wang, G. Liang, H. Chen, and X. Wu. Formation control strategy for a group of quadrotors. *Proceedings of the IEEE International Conference on Information and Automation*, pages 2504–2511, 2015.

- [277] 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.
- [278] 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.
- [279] 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.
- [280] 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.
- [281] 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.
- [282] 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.
- [283] 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.
- [284] S. Ulrich. Nonlinear passivity-based adaptive control of spacecraft formation flying. *American Control Conference (ACC), Boston, MA, USA*, pages 7432–7437, 2016.
- [285] 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.
- [286] 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.
- [287] 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.
- [288] 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.
  - [289] Sina Sharif Mansouri, George Nikolakopoulos, and Thomas Gustafsson. Distributed model predictive control for unmanned aerial vehicles. *2015 Workshop on Research, Education and Development of Unmanned Aerial Systems (RED-UAS)*, pages 152–161, 2015.
  - [290] 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.
  - [291] Javier Alonso-Mora, Tobias Naegeli, Roland Siegwart, and Paul Beard-sley. Collision avoidance for aerial vehicles in multi-agent scenarios. *Autonomous Robots*, 39(1):101–121, 2015.
  - [292] 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.
  - [293] 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.
  - [294] 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.
  - [295] 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.
  - [296] 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.
  - [297] C. Belta and L. C. Habets. Controlling a class of nonlinear systems on rectangles. *IEEE Transactions on Automatic Control*, 51(11):1749–1759, 2006.

- [298] G. Reissig. Computing abstractions of nonlinear systems. *IEEE Transactions on Automatic Control*, 56(11):2583–2598, 2011.
- [299] A. Tiwari. Abstractions for hybrid systems. *Formal Methods in System Design*, 32(1):57–83, 2008.
- [300] 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.
- [301] 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.
- [302] C. Belta and V. Kumar. Abstraction and control for groups of robots. *IEEE Transactions on robotics*, 20(5):865–875, 2004.
- [303] R. Alur, T. Feder, and T. A. Henzinger. The benefits of relaxing punctuality. *Journal of the ACM (JACM)*, 43(1):116–146, 1996.
- [304] 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.
- [305] R. Alur and D. L. Dill. A theory of timed automata. *Theoretical computer science*, 126(2):183–235, 1994.
- [306] 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.
- [307] 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.
- [308] Lars Lindemann and Dimos V Dimarogonas. Decentralized robust control of coupled multi-agent systems under local signal temporal logic tasks. *2018 Annual American Control Conference (ACC)*, pages 1567–1573, 2018.
- [309] 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.

- [310] Paul Gastin and Denis Oddoux. Fast ltl to büchi automata translation. *International Conference on Computer Aided Verification*, pages 53–65, 2001, <http://www.lsv.ens-cachan.fr/gastin/ltl2ba/>.
- [311] B. Siciliano and O. Khatib. *Springer handbook of robotics*. Springer Science & Business Media, 2008.
- [312] 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.
- [313] E. Rimon and D. Koditschek. Exact robot navigation using artificial potential functions. *IEEE Transactions on Robotics and Automation (TRA)*, 8(5):501–518, 1992.
- [314] S. G. Loizou and K. J. Kyriakopoulos. A feedback-based multiagent navigation framework. *International Journal of Systems Science*, 37(6):377–384, 2006.
- [315] S. G. Loizou. The multi-agent navigation transformation: Tuning-free multi-robot navigation. *Proceedings of Robotics: Science and Systems*, 2014.
- [316] D. Z. Chen. Sphere packing problem. *Encyclopedia of Algorithms*, pages 1–99, 2008.
- [317] M. R. Cutkosky. *Robotic grasping and fine manipulation*, volume 6. Springer Science & Business Media, 2012.
- [318] 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.
- [319] H. K. Khalil. Nonlinear Systems. *Prentice Hall*, 2002.
- [320] Lydia E Kavraki, Petr Svestka, J-C Latombe, and Mark H Overmars. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE transactions on Robotics and Automation*, 12(4):566–580, 1996.
- [321] David Hsu, J-C Latombe, and Rajeev Motwani. Path planning in expansive configuration spaces. *Proceedings of International Conference on Robotics and Automation*, 3:2719–2726, 1997.
- [322] Steven M LaValle. Rapidly-exploring random trees: A new tool for path planning. 1998.

- [323] Steven M LaValle and James J Kuffner Jr. Randomized kinodynamic planning. *The international journal of robotics research*, 20(5):378–400, 2001.
- [324] Ioan A Şucan and Lydia E Kavraki. Kinodynamic motion planning by interior-exterior cell exploration. *Algorithmic Foundation of Robotics VIII*, pages 449–464, 2009.
- [325] Eduard Vidal, Mark Moll, Narcís Palomeras, Juan David Hernández, Marc Carreras, and Lydia E Kavraki. Online multilayered motion planning with dynamic constraints for autonomous underwater vehicles. *2019 International Conference on Robotics and Automation (ICRA)*, pages 8936–8942, 2019.
- [326] Russ Tedrake, Ian R Manchester, Mark Tobenkin, and John W Roberts. Lqr-trees: Feedback motion planning via sums-of-squares verification. *The International Journal of Robotics Research*, 29(8):1038–1052, 2010.
- [327] Philipp Reist, Pascal Preiswerk, and Russ Tedrake. Feedback-motion-planning with simulation-based lqr-trees. *The International Journal of Robotics Research*, 35(11):1393–1416, 2016.
- [328] Albert Wu, Sadra Sadraddini, and Russ Tedrake. R3t: Rapidly-exploring random reachable set tree for optimal kinodynamic planning of nonlinear hybrid systems. 2019.
- [329] Anirudha Majumdar and Russ Tedrake. Funnel libraries for real-time robust feedback motion planning. *The International Journal of Robotics Research*, 36(8):947–982, 2017.
- [330] Noel E Du Toit and Joel W Burdick. Probabilistic collision checking with chance constraints. *IEEE Transactions on Robotics*, 27(4):809–815, 2011.
- [331] Èric Pairet, Juan David Hernández, Morteza Lahijanian, and Marc Carreras. Uncertainty-based online mapping and motion planning for marine robotics guidance. *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2367–2374, 2018.
- [332] Adam Bry and Nicholas Roy. Rapidly-exploring random belief trees for motion planning under uncertainty. *2011 IEEE international conference on robotics and automation*, pages 723–730, 2011.
- [333] Ali-Akbar Agha-Mohammadi, Suman Chakravorty, and Nancy M Amato. Firm: Sampling-based feedback motion-planning under motion uncertainty and imperfect measurements. *The International Journal of Robotics Research*, 33(2):268–304, 2014.

- [334] Jerome Le Ny and George J Pappas. Sequential composition of robust controller specifications. *2012 IEEE International Conference on Robotics and Automation*, pages 5190–5195, 2012.
- [335] Brandon D Luders, Sertac Karaman, Emilio Frazzoli, and Jonathan P How. Bounds on tracking error using closed-loop rapidly-exploring random trees. *Proceedings of the 2010 American Control Conference*, pages 5406–5412, 2010.
- [336] Jay A Farrell and Marios M Polycarpou. *Adaptive approximation based control: unifying neural, fuzzy and traditional adaptive approximation approaches*, volume 48. John Wiley & Sons, 2006.
- [337] Zhongle Wu, Jian Chen, Chengshuai Wu, and Kaixiang Zhang. Adaptive tracking for mimo nonlinear systems with unknown fast time-varying parameters. *International Journal of Robust and Nonlinear Control*, 28(13):4058–4074, 2018.
- [338] Jian Chen, Aman Behal, and Darren M Dawson. Robust feedback control for a class of uncertain mimo nonlinear systems. *IEEE Transactions on Automatic Control*, 53(2):591–596, 2008.
- [339] Bin Xian, Darren M Dawson, Marcio S de Queiroz, and Jian Chen. A continuous asymptotic tracking control strategy for uncertain nonlinear systems. *IEEE Transactions on Automatic Control*, 49(7):1206–1211, 2004.
- [340] Lei Wang, Romeo Ortega, Hongye Su, and Zhitao Liu. Stabilization of nonlinear systems nonlinearly depending on fast time-varying parameters: an immersion and invariance approach. *IEEE Transactions on Automatic Control*, 60(2):559–564, 2015.
- [341] Håvard Fjær Grip, Tor A Johansen, Lars Imsland, and Glenn-Ole Kaasa. Parameter estimation and compensation in systems with nonlinearly parameterized perturbations. *Automatica*, 46(1):19–28, 2010.
- [342] Alessandro Astolfi and Romeo Ortega. Immersion and invariance: a new tool for stabilization and adaptive control of nonlinear systems. *IEEE Transactions on Automatic control*, 48(4):590–606, 2003.
- [343] K Zhang, Z Wang, and A Behal. A continuous asymptotic tracking control strategy for a class of uncertain mimo nonlinear systems. *Decision and Control (CDC), 2015 IEEE 54th Annual Conference on*, pages 6209–6214, 2015.

- [344] Baris Bidikli, Enver Tatlicioglu, Erkan Zergeroglu, and Alper Bayrak. An asymptotically stable continuous robust controller for a class of uncertain mimo nonlinear systems. *arXiv preprint arXiv:1301.5483*, 2013.
- [345] Riccardo Marino and Patrizio Tomei. An adaptive output feedback control for a class of nonlinear systems with time-varying parameters. *IEEE Transactions on Automatic Control*, 44(11):2190–2194, 1999.
- [346] Shuzhi Sam Ge and Cong Wang. Adaptive nn control of uncertain nonlinear pure-feedback systems. *Automatica*, 38(4):671–682, 2002.
- [347] Dan Wang and Jie Huang. Adaptive neural network control for a class of uncertain nonlinear systems in pure-feedback form. *Automatica*, 38(8):1365–1372, 2002.
- [348] Bong-Jun Yang and Anthony J Calise. Adaptive control of a class of nonaffine systems using neural networks. *IEEE Transactions on Neural Networks*, 18(4):1149–1159, 2007.
- [349] Wei-Yen Wang, Yi-Hsing Chien, Yih-Guang Leu, and Tsu-Tian Lee. Adaptive t-s fuzzy-neural modeling and control for general mimo unknown nonaffine nonlinear systems using projection update laws. *Automatica*, 46(5):852–863, 2010.
- [350] Charalampos P Bechlioulis and George A Rovithakis. Prescribed performance adaptive control for multi-input multi-output affine in the control nonlinear systems. *IEEE Transactions on Automatic Control*, 55(5):1220–1226, 2010.
- [351] Achilles Theodorakopoulos and George A Rovithakis. Low-complexity prescribed performance control of uncertain mimo feedback linearizable systems. *IEEE Transactions on Automatic Control*, 61(7):1946–1952, 2016.
- [352] Thomas Berger, Huy Hoàng Lê, and Timo Reis. Funnel control for nonlinear systems with known strict relative degree. *Automatica*, 87:345–357, 2018.
- [353] Achim Ilchmann and Stephan Trenn. Input constrained funnel control with applications to chemical reactor models. *Systems & control letters*, 53(5):361–375, 2004.
- [354] Yiannis Karayannidis and Zoe Doulgeri. Model-free robot joint position regulation and tracking with prescribed performance guarantees. *Robotics and Autonomous Systems*, 60(2):214–226, 2012.

- [355] Hashim A Hashim, Sami El-Ferik, and Frank L Lewis. Adaptive synchronisation of unknown nonlinear networked systems with prescribed performance. *International Journal of Systems Science*, 48(4):885–898, 2017.
- [356] Charalampos P Bechlioulis and George A Rovithakis. Decentralized robust synchronization of unknown high order nonlinear multi-agent systems with prescribed transient and steady state performance. *IEEE Transactions on Automatic Control*, 62(1):123–134, 2017.
- [357] Luca Macellari, Yiannis Karayannidis, and Dimos V Dimarogonas. Multi-agent second order average consensus with prescribed transient behavior. *IEEE Transactions on Automatic Control*, 62(10):5282–5288, 2017.
- [358] Achim Ilchmann and Eugene P Ryan. Asymptotic tracking with prescribed transient behaviour for linear systems. *International Journal of Control*, 79(8):910–917, 2006.
- [359] Achim Ilchmann, Eugene P Ryan, and Philip Townsend. Tracking with prescribed transient behavior for nonlinear systems of known relative degree. *SIAM Journal on Control and Optimization*, 46(1):210–230, 2007.
- [360] Daniel Liberzon and Stephan Trenn. The bang-bang funnel controller for uncertain nonlinear systems with arbitrary relative degree. *IEEE Transactions on Automatic Control*, 58(12):3126–3141, 2013.
- [361] Martijn Dresscher and Bayu Jayawardhana. Prescribing transient and asymptotic behaviour of lti systems with stochastic initial conditions. *IFAC-PapersOnLine*, 50(1):1822–1827, 2017.
- [362] Seong Ik Han. Prescribed consensus and formation error constrained finite-time sliding mode control for multi-agent mobile robot systems. *IET Control Theory & Applications*, 12(2):282–290, 2017.
- [363] C. Vrohidis, P. Vlantis, C. P. Bechlioulis, and K. J. Kyriakopoulos. Prescribed time scale robot navigation. *IEEE Robotics and Automation Letters*, 3(2):1191–1198, April 2018.
- [364] Thomas Brihaye, Gilles Geeraerts, Hsi-Ming Ho, and Benjamin Monmege. MightyL: A Compositional Translation from MITL to Timed Automata. *29th International Conference on Computer Aided Verification (CAV'17)*, 10426:421–440, July 2017.

- [365] Meng Guo, Karl H Johansson, and Dimos V Dimarogonas. Motion and action planning under ltl specifications using navigation functions and action description language. *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pages 240–245, 2013.
- [366] Patrizio Tomei. Robust adaptive control of robots with arbitrary transient performance and disturbance attenuation. *IEEE transactions on automatic control*, 44(3):654–658, 1999.
- [367] Jean-Jacques E Slotine and Weiping Li. On the adaptive control of robot manipulators. *The international journal of robotics research*, 6(3):49–59, 1987.
- [368] James J Kuffner and Steven M LaValle. Rrt-connect: An efficient approach to single-query path planning. *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, 2:995–1001, 2000.
- [369] Sertac Karaman and Emilio Frazzoli. Incremental sampling-based algorithms for optimal motion planning. *Robotics Science and Systems VI*, 104(2), 2010.
- [370] John Schulman, Yan Duan, Jonathan Ho, Alex Lee, Ibrahim Awwal, Henry Bradlow, Jia Pan, Sachin Patil, Ken Goldberg, and Pieter Abbeel. Motion planning with sequential convex optimization and convex collision checking. *The International Journal of Robotics Research*, 33(9):1251–1270, 2014.
- [371] A Frank van der Stappen, Dan Halperin, and Mark H Overmars. The complexity of the free space for a robot moving amidst fat obstacles. *Computational Geometry*, 3(6):353–373, 1993.
- [372] Charalampos P Bechlioulis and George A Rovithakis. Robust partial-state feedback prescribed performance control of cascade systems with unknown nonlinearities. *IEEE Transactions on Automatic Control*, 56(9):2224–2230, 2011.
- [373] Jin Gyu Lee and Stephan Trenn. Asymptotic tracking via funnel control. *IEEE Conference on Decision and Control*, 2019.
- [374] Boris Mityagin. The zero set of a real analytic function. *arXiv:1512.07276v1*, 2015.
- [375] E. D. Sontag. Mathematical control theory: deterministic finite dimensional systems. *Springer Science & Business Media*, 6, 2013.

- 
- [376] A. Bressan and B. Piccoli. Introduction to the mathematical theory of control. *American institute of mathematical sciences Springfield*, 2, 2007.
  - [377] Brad Paden and Shankar Sastry. A calculus for computing filippov's differential inclusion with application to the variable structure control of robot manipulators. *IEEE transactions on circuits and systems*, 34(1):73–82, 1987.
  - [378] Nicholas Fischer, Rushikesh Kamalapurkar, and Warren E Dixon. Lasalle-yoshizawa corollaries for nonsmooth systems. *IEEE Transactions on Automatic Control*, 58(9):2333–2338, 2013.
  - [379] Jorge Cortes. Discontinuous dynamical systems. *IEEE control Systems*, 28(3), 2008.
  - [380] Yuri A Kuznetsov. *Elements of applied bifurcation theory*, volume 112. Springer Science & Business Media, 2013.
  - [381] Rolf Findeisen and Frank Allgöwer. An introduction to nonlinear model predictive control. *21st Benelux meeting on systems and control*, 11:119–141, 2002.
  - [382] 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.
  - [383] Patricia Bouyer, Nicolas Markey, Joël Ouaknine, and James Worrell. The cost of punctuality. *Proceedings of the 22nd Annual IEEE Symposium on Logic in Computer Science (LICS'07)*, pages 109–118, 2007.
  - [384] Yi-King Choi, Wenping Wang, Liu Yang, and Myung-Soo Kim. Continuous collision detection for two moving elliptic disks. *IEEE Transactions on Robotics*, 22(2), 2006.