

# University of Groningen

## Formation control in the port-Hamiltonian framework

Vos, Ewoud

**IMPORTANT NOTE:** You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

*Document Version*

Publisher's PDF, also known as Version of record

*Publication date:*

2015

[Link to publication in University of Groningen/UMCG research database](#)

*Citation for published version (APA):*

Vos, E. (2015). *Formation control in the port-Hamiltonian framework*. [S.n.].

**Copyright**

Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

**Take-down policy**

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

*Downloaded from the University of Groningen/UMCG research database (Pure): <http://www.rug.nl/research/portal>. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.*

# **Formation control in the port-Hamiltonian framework**

Ewoud Vos



The research described in this thesis has been carried out at the Faculty of Mathematics and Natural Sciences, University of Groningen, The Netherlands, within collaboration between the ENgineering and TEchnology institute Groningen (ENTE) and the Johann Bernoulli Institute for Mathematics and Computer Science (JBI).

# disc

This thesis has been completed in partial fulfillment of the requirements of the Dutch Institute of Systems and Control (DISC) for graduate study.



Enabling new technology

The research reported in this thesis has been supported by the Dutch Technology Foundation STW via the ROSE (Energy-efficient design and control of mobile RObotic SEnsor networks) project under the Autonomous Sensor SYStems (ASSYS) program.

Printed by Ipkamp Drukkers  
Enschede, The Netherlands

Cover design by Bianca Pijl  
Groningen, The Netherlands

ISBN (book): 978-90-367-7604-2  
ISBN (e-book): 978-90-367-7603-5



university of  
groningen

# Formation control in the port-Hamiltonian framework

PhD thesis

to obtain the degree of PhD at the  
University of Groningen  
on the authority of the  
Rector Magnificus Prof. E. Sterken  
and in accordance with  
the decision by the College of Deans.

This thesis will be defended in public on  
Friday 20 February 2015 at 11:00 hours

by

**Ewoud Vos**

born on 2 April 1986  
in Peize

**Supervisors**

Prof. J.M.A. Scherpen  
Prof. A.J. van der Schaft

**Assessment Committee**

Prof. M. Arcak  
Prof. M. Cao  
Prof. H. Nijmeijer

*To Elze,*

*and my family,  
Berend, Grietje, and Judith*



---

# Contents

<b>About the author</b>	<b>xi</b>
<b>Acknowledgments</b>	<b>xii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Theoretical background . . . . .	1
1.1.1 Port-Hamiltonian systems theory . . . . .	2
1.1.2 Formation control of networks . . . . .	4
1.2 Application areas . . . . .	6
1.2.1 ROSE project . . . . .	6
1.2.2 Dike inspection using robotic sensor networks . . . . .	7
1.2.3 Formation flying of satellite constellations . . . . .	9
1.3 Contributions . . . . .	10
1.4 Thesis outline . . . . .	11
1.5 Publications . . . . .	12
1.6 Notation . . . . .	14
<b>2 Preliminaries</b>	<b>17</b>
2.1 Stability of time-invariant systems . . . . .	17
2.2 Graph theory . . . . .	19
2.3 Port-Hamiltonian systems . . . . .	21
2.3.1 Input-state-output port-Hamiltonian systems . . . . .	22
2.3.2 Mechanical port-Hamiltonian systems with constraints . . . . .	24
2.3.3 Generalized canonical transformations . . . . .	25
2.3.4 Port-Hamiltonian systems on graphs . . . . .	27
2.4 Non-smooth analysis . . . . .	29
2.5 Concluding remarks . . . . .	31

<b>3 Formation control of fully actuated systems</b>	<b>33</b>
3.1 Introduction . . . . .	33
3.2 Dynamical model . . . . .	35
3.3 Formation control . . . . .	37
3.3.1 Formation control using virtual couplings . . . . .	38
3.3.2 Simulation and experimental results . . . . .	40
3.4 Formation control in the presence of Coulomb friction . . . . .	42
3.4.1 Model of ideal Coulomb friction . . . . .	42
3.4.2 Formation control using continuous virtual springs . . . . .	46
3.4.3 Formation control using discontinuous virtual springs . . . . .	50
3.4.4 Simulation results . . . . .	53
3.5 Deployment . . . . .	54
3.5.1 Deployment using virtual couplings and virtual walls . . . . .	56
3.5.2 Simulation and experimental results . . . . .	59
3.6 Concluding remarks . . . . .	62
<b>4 Formation control of nonholonomic wheeled robots</b>	<b>63</b>
4.1 Introduction . . . . .	63
4.2 Dynamical model of the wheeled robot . . . . .	65
4.3 Formation control . . . . .	70
4.3.1 Formation control of wheeled robots using virtual springs . .	70
4.3.2 Simulation and experimental results . . . . .	73
4.4 Formation control with velocity tracking . . . . .	75
4.4.1 Velocity tracking control and formation control using virtual couplings . . . . .	77
4.4.2 Simulation and experimental results . . . . .	84
4.5 Formation control in the presence of matched input disturbances .	89
4.5.1 Matched input disturbance rejection using an internal-model- based approach . . . . .	89
4.5.2 Simulation results . . . . .	95
4.6 Concluding remarks . . . . .	100
<b>5 Orbital phasing of satellites on circular orbits</b>	<b>101</b>
5.1 Introduction . . . . .	101
5.2 Dynamical model . . . . .	103
5.2.1 Derivation of the error dynamics with respect to a circular orbit using generalized canonical transformations . . . . .	105
5.3 Orbital phasing on circular orbits . . . . .	108
5.3.1 Internal control system . . . . .	108
5.3.2 External control system . . . . .	109
5.3.3 Closed-loop analysis . . . . .	110

## Contents

---

5.3.4	Simulation results . . . . .	113
5.4	Concluding remarks . . . . .	117
<b>6</b>	<b>Conclusions and future research</b>	<b>119</b>
6.1	Conclusions . . . . .	119
6.2	Recommendations for future research . . . . .	120
<b>A</b>	<b>Background on dike inspection</b>	<b>123</b>
A.1	Dike inspection in The Netherlands . . . . .	124
A.2	Dike inspection using sensor technology . . . . .	125
<b>B</b>	<b>ROSE project partners and utilization</b>	<b>127</b>
<b>C</b>	<b>Experimental setup e-pucks</b>	<b>133</b>
C.1	The e-puck robot . . . . .	133
C.2	Experimental setup . . . . .	134
<b>D</b>	<b>Complementary simulation and experimental data</b>	<b>137</b>
<b>Bibliography</b>		<b>145</b>
<b>Summary</b>		<b>155</b>
<b>Samenvatting</b>		<b>157</b>



---

## About the author



Ewoud Vos was born in Peize, The Netherlands on 2 April 1986. In 2004 he received his VWO diploma with profiles *Nature and Technics* and *Nature and Health* at the Willem Lodewijk Gymnasium in Groningen. The same year he started the study of Applied Physics at the Faculty of Mathematics and Natural Sciences of the University of Groningen. One year later he switched to Industrial Engineering and Management at the same faculty, where he received the B.Sc. and M.Sc. degree (cum laude) respectively in 2008 and 2010. In his master thesis he designed a nonlinear controller for the VDO measurement machine of Irmato, which is used for sampling shaver heads of the nearby production line of Philips.

In 2010 Ewoud started as a PhD researcher in a joint project of the research groups Discrete Technology and Production Automation (DTPA) and Systems, Control and Applied Analysis (SCAA) under the supervision of Jacquelin Scherpen and Arjan van der Schaft. During four years he presented his work at many national and international conferences. In 2012 he won the Best Junior Presentation award at the Benelux Meeting on Systems and Control, for which he was nominated again in 2014. In 2013 he was on the short list for the Simon Stevin Leerling Prize at the STW Annual Congress. The results of his PhD research are presented in this thesis.



---

## Acknowledgments

It is hard to imagine four years of PhD research, without the help and support of many. Both technical help during the struggle with research problems and emotional support during mentally challenging periods.

First of all, I want to thank my supervisors Jacquelin and Arjan. They gave me the freedom to explore my interests and stimulated me to keep developing myself as a researcher, as an engineer, and as a person. I also greatly appreciate the opportunity they provided me to travel around the world for conferences, workshops, courses and robotic fish water polo tournaments.

Second, I want to thank Ming, Henk Nijmeijer, and Murat Arcak for their efforts in assessing my thesis and providing me with valuable comments which have improved my thesis significantly.

Third, I want to thank Matin, Claudio, Shaik, Geoff Stacey, Steven Engelen, Ate, Eelke and Mohammed for their contribution in several collaborations during the course of this research. Also a big thank you to all my colleagues at the research groups *Discrete Technology and Production Automation* (DTPA), *Systems, Control and Applied Analysis* (SCAA) and *Smart Manufacturing Systems* (SMS), for making the Zernike campus a nice place to work. This is actually quite an achievement, Zernike campus being the quite boring place that it (still) is.

Fourth, I want to thank technology foundation STW and the members of the user committee of the ROSE project for the financial support, sharing of information and new ideas which evolved during the user committee meetings. With many agendas the meetings were usually hard to schedule, but they provided me with inspiration for new research and useful feedback on existing work. A special thanks to Douwe and Stefano for the nice collaboration during the project.

Fifth, I want to point out my gratitude to three colleagues and friends in particular: Nima, Gunn and Shuo. Nima and Gunn, at the time of writing I have not defended yet, but as as paranympths I know you will be the best support possible during my defense. Nima, *mamnoon* for your company during all the conferences

we visited together. I will never forget our trip to the MTNS 2012 because of all the fun we had, great dinners, and “doegie doegie” at Harbour bridge. Gunn, *takk* for being a great friend. I am proud of what the two of us have achieved with setting up the DTPA/SMS group meeting. Shuo, *Xièxiè* for all the laughter and serious stuff we shared in room 5117.0120. All three of you are great persons and I know we will keep in touch in the future.

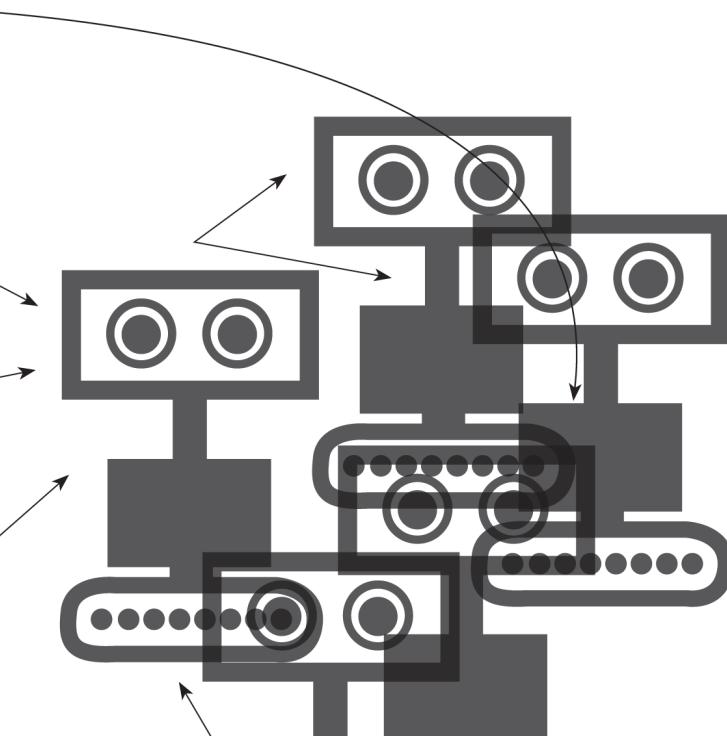
Last, but certainly not least, thanks to Elze, Berend, and Grietje for their unconditional support and trust during the past four years. Elze, as little you could help me on the technical side, so much have you personally supported me. Without you, I would probably have turned into the subject of my PhD: a robot. Pap, your no-nonsense attitude and personal advice is always a beacon of tranquility and very valuable for me. Mam, having a PhD yourself, in my personal environment you know most of what doing research is about. Thank you for stimulating me to go for a PhD and all the (unsolicited) advice you gave me.

Ewoud Vos  
Groningen, January 2015



# Chapter 1

## Introduction





# Chapter 1

---

## Introduction

Formation control is a specific control problem within the broad class of coordination control. The main challenge in coordination control is to achieve a prescribed group behavior for a group of agents employing local feedback rules, rather than centralized controllers [3]. For formation control the prescribed group behavior is to achieve a geometrical shape for the network of agents (i.e., to achieve a formation) [58, 83, 115]. Two other well known types of coordination control which are outside the scope of this thesis are consensus (or agreement) [55, 84, 93, 95] and synchronization [23, 80, 81, 84].

All systems in this thesis are modeled as port-Hamiltonian systems. Port-Hamiltonian systems theory is an energy-based modeling framework [86] which provides powerful tools for the analysis and design of controllers by exploiting the physical structure of the system. The port-Hamiltonian framework considers systems as the interconnection of energy storing and energy dissipating components which exchange energy through power ports. External power ports enable the system to interact with external systems, such as controllers and the environment [39, 100]. Furthermore, the framework provides insight into the energy supplied by the controllers.

This chapter starts with a theoretical background on port-Hamiltonian systems theory and formation control (Section 1.1). Section 1.2 continues with the two main application areas for the results in this thesis. The main contributions, thesis outline, and publications are given in respectively Sections 1.3, 1.4 and 1.5. Finally Section 1.6 introduces the (mathematical) notation for the subsequent chapters.

### 1.1 Theoretical background

Formation control has received a wide interest from different perspectives in recent years. Two major topics include formation stability (i.e., to achieve and maintain a formation) [30, 55, 72, 83], and changing formation patterns (i.e., to change from one formation to another) [30, 67]. Another distinction can be made according to the systems under investigation (e.g. nonholonomic robots [30, 34, 40, 67, 72] and satellites [2, 23, 94, 102]). The passivity-based design tool for group coordination in [3] provides a starting point for the formation control algorithms in this thesis

and is further explored in Section 1.1.2.

In this thesis the physical structure of the network is exploited for the analysis and design of formation control algorithms. The physical structure not only refers to the *real* physical structure of the agents in the network, but also to the *virtual* structure, which is part of the controller design. Here, formation control is achieved by assigning virtual couplings between real agents (i.e., robots and satellites) in the network [43, 70]. Each virtual coupling consist of a spring and an optional damper in parallel. Virtual spring achieve formation control by shaping the energy function of the network, while virtual dampers guarantee stability by injecting damping.

The integration of real physical systems with virtual couplings is a perfect example of a *cyber-physical system*. Cyber-physical systems deal with the integration of physical processes and computation and have become a hot topic in different disciplines in recent years [7, 69, 103]. The ability of cyber-physical systems to interact with and expand the capabilities of the physical world through computation, communication and control is a key enabler for future technology developments [7].

All cyber-physical networks in this thesis are modeled as port-Hamiltonian systems, which is an energy-based modeling framework. The remainder of this section provides the background on port-Hamiltonian systems (Section 1.1.1), followed by the formation control (Section 1.1.2). Section 1.1.2 also provides the intuition behind the algorithms in Chapters 3, 4, and 5.

### 1.1.1 Port-Hamiltonian systems theory

Port-Hamiltonian systems theory was introduced in the early 90s by Maschke and Van der Schaft [73] as an energy-based modeling framework for the modeling of a large class of multi-domain physical systems. The theory brings together the traditions of port-based modeling [12, 88], geometric mechanics [14, 71], and systems and control theory [99]. Energy concepts are well-known to practitioners and port-Hamiltonian models may therefore serve as a *lingua franca* amongst engineers, by interpreting the models and control actions using physical intuitions [86]. The Euler-Lagrange and Brayton-Moser modeling frameworks provide similar advantages, but are outside the scope of this thesis. The interested reader is referred to e.g. [31, 61, 85, 86] for more information on these frameworks.

Not only does port-Hamiltonian systems theory enable systematic and intuitive modeling and analysis of multi-domain systems, it also provides a starting point for control design. It is widely recognized that physical properties of systems should be exploited in the design of robust and physically interpretable control systems. Exploiting the physical structure for design and control inherits robustness and fault-tolerance for the design from the intrinsic physical robustness. Furthermore, port-Hamiltonian systems theory offers powerful tools and concepts for doing this

(e.g. [19, 38, 45, 62, 87, 98, 99]) and has proven successful in many applications (e.g. [39, 41, 86, 106, 107] and references therein).

Control of port-Hamiltonian systems is achieved by interconnecting the system to be controlled to a controller using *power ports*. A power port has two corresponding port variables, whose product has the dimension of power. Usually the port variables are the input  $u$  and output  $y$  of the system, implying that the product of the input and output  $y^T u$  equals the external power supplied through the port.

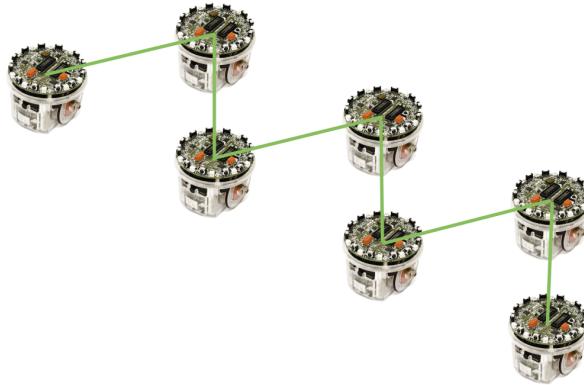
In addition the framework provides insight into the energy consumption of the controller. Consider a control input  $u = -k(z - z^*) - d\dot{z}$  for simple set-point regulation problem, with controller gains  $k, d$ , position  $z$ , set-point  $z^*$ . This simplified control law is closely related to the formation control algorithms in this thesis. The two terms of the control input  $u$  are interpretable as respectively a virtual spring and damping force. The energy supplied by the controller equals the integral of the power supplied through the power port and is given by

$$\underbrace{\int_0^T y^T u \, dt}_{\text{'control energy'}} = \underbrace{\frac{1}{2} k(z - z^*)^2 \Big|_0 - \frac{1}{2} k(z - z^*)^2 \Big|_T}_{\text{'added energy'}} - \underbrace{\int_0^T d \dot{z}^2 \, dt}_{\text{'extracted energy'}}. \quad (1.1)$$

Note that this energy is the real energy supplied by the controller, which differs from the usually considered measure for energy consumption  $\int u^2 \, dt$ . Analyzing (1.1) for different values of  $k$  and  $d$  facilitates energy-efficient controller designs.

The Hamiltonian  $H$  is the total energy stored in the system (i.e.,  $H$  equals the sum of the kinetic and potential energy). For the mechanical systems in this thesis, the physical structure ensures that  $\dot{H} \leq y^T u$  (see Section 2.3). In other words the energy flow in the system  $\dot{H}$  is always equal or smaller than the external power supplied  $y^T u$  through the power port  $(u, y)$ . This implies that port-Hamiltonian systems are passive from the input  $u$  to the output  $y$ . Section 2.3 provides more technical details on port-Hamiltonian systems.

Port-Hamiltonian systems theory can also be used to model complex networks of dynamical agents [100]. The theory of *port-Hamiltonian systems on graphs* is used for the modeling, analysis and design of formation control algorithms in the remainder. In this approach both the agents and the virtual couplings are modeled as port-Hamiltonian systems and a power-continuous interconnection structure is imposed on the network. This interconnection structure is derived from the graph structure modeling the network. In this thesis, all networks are modeled as undirected graphs, where agent dynamics are assigned to the nodes and virtual coupling dynamics are assigned to the edges of the graph [100]. The next section continues with a more detailed background on the formation control problem.



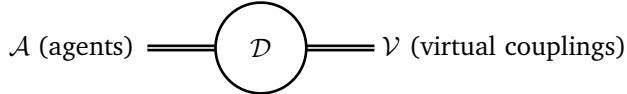
**Figure 1.1:** Geometrical shape for a network of robots corresponding to a zig-zag formation.

### 1.1.2 Formation control of networks

Coordination control is about achieving prescribed group behaviors for a group of agents (e.g. robots, satellites) with local feedback rules, rather than with centralized controllers [3]. Formation control is a specific type of coordination problem where the prescribed group behavior is to achieve prescribed displacements between the agents in the network. (see Figure 1.1). The approach to formation control in this thesis consists of three components: the agents  $\mathcal{A}$  in the network, the virtual couplings  $\mathcal{V}$  to achieve formation control, and the interconnection structure  $\mathcal{D}$  to interconnected the first two components (Figure 1.2). For example, consider Figure 1.1, where  $\mathcal{A}$ ,  $\mathcal{V}$ , and  $\mathcal{D}$  correspond to respectively the seven robots, six green lines, and the way in which robots are interconnected by lines.

Starting point for the formation control algorithms in this thesis is the passivity-based design tool for group coordination in [3, 9]. The approach consists of an internal and an external feedback. The internal feedback is a local feedback which renders agents in the network passive with respect to an error output, to track reference velocities. The external feedback achieves group coordination by rendering target sets invariant and asymptotically stable by interconnecting agents in the network. The target sets correspond to group coordination tasks such as consensus, synchronization, and formation control. The symmetry in the interconnection structure presented in [3] is also exploited in this thesis.

The dynamics of the agents  $\mathcal{A}$  in Figure 1.2 play an important role in the formation control problem. Various classes of dynamic agents have been considered e.g. [9, 81, 93, 100]. The dynamics considered here include fully actuated agents (Chapter 3), nonholonomic wheeled robots (Chapter 4), and satellites (Chapter 5), which are all modeled as (mechanical) port-Hamiltonian systems. Dissipation due to friction and damping plays an important role in the stability analysis of networks



**Figure 1.2:** General setup for formation control with agents  $\mathcal{A}$ , virtual couplings  $\mathcal{V}$ , and interconnection structure  $\mathcal{D}$ .

of dynamical agents (e.g. [9, 57]). Different types of dissipation are considered here, ranging from discontinuous Coulomb friction to virtual dampers in between the agents.

For the application areas in this thesis (see Section 1.2.2), formation control alone is not enough. The whole network also has to move by tracking a prescribed reference velocity (i.e., it has to sweep [22, 66] or cover [25, 82] a surface). Velocity tracking is achieved by rendering the agent dynamics passive from the input to the velocity error output [3]. Generalized canonical transformations enable velocity tracking for port-Hamiltonian systems [45], by deriving and stabilizing the error dynamics preserving the port-Hamiltonian structure.

Another, more practical, challenge is the presence of disturbances, which distort the formation shape or even render the system unstable. Proportional-integral control with quantized information and time-varying topologies has been proposed for a network of single-integrator robots [120], while adaptive internal model control was studied for a single port-Hamiltonian system [47, 48]. Section 4.5 provides new insights using an internal model controller for disturbance rejection in a network of nonholonomic wheeled robots.

In this thesis, formation control is achieved by assigning virtual couplings ( $\mathcal{V}$  in Figure 1.2) between the agents. Different types of virtual couplings have been considered [43, 70]. Here virtual couplings are interpreted as virtual springs, with an optional virtual damper in parallel. The springs shape the energy function of the whole network, while the dampers inject damping for stabilization. The virtual damper is an extension to the external feedback in [3], which enables only virtual springs. Moreover, the setup fits within the IDA-PBC paradigm [87] and builds upon the theory of port-Hamiltonian systems on graphs [100].

The interconnection structure  $\mathcal{D}$  in Figure 1.2 models the way in which agents are interconnected by virtual couplings. Graph theory [11, 50] provides powerful tools for modeling the interconnection structure of complex networks. Nodes and edges of the graph corresponds to respectively agents and virtual couplings in the network [100]. The topology of the network consists of two layers. The *communication topology* models which agents exchange information with other agents in the network. The *interconnection topology* models the (virtual) physical structure of the network of agents and virtual couplings. Due to the underlying physical structure the interconnection topology is modeled as an undirected graph

[3, 9, 93]. The communication topology on the other hand is often assumed to be directed [81, 100]. In this work the communication topology is assumed to be equal to the interconnection topology and therefore only undirected graphs are considered.

All undirected graphs in this thesis are assumed to be connected and acyclic, except for the cycle graphs in Chapter 5. The graph being connected prevents agents in the network to drift away from others [3, 9, 81, 93, 100]. Considering acyclic graphs has pros and cons. An advantage of acyclic graphs is that they impose the communication network to be loop-free, which prevents error accumulation during parallel distributed signal processing at the robots. On the other hand, cyclic graphs provide robustness to link failures. Excluding cyclic graphs in the approach yields less general results, while on the other hand cyclic graphs require an additional (restrictive) condition on the desired relative displacement of the formation [3, 100] (see also Remark 4.3). Furthermore, for some types of virtual springs undesired equilibria may arise [3, 8, 9] when the graph has cycles.

In the remainder, the graph topology is a design freedom and considering only acyclic graphs is therefore not restrictive. More technical details on graph theory are provided in Section 2.2.

## 1.2 Application areas

The formation control algorithms developed in this thesis concern a broad class of applications (see e.g. [2, 23, 30, 34, 40, 67, 72, 94, 102, 104, 105] and references therein). The two main application areas for this thesis both originate from the ROSE project, which is introduced in Section 1.2.1. Section 1.2.2 continues with the first application area, being the inspection of dikes using robotic sensor networks. The second application area concerns the formation flying of satellite constellations, which is elaborated in Section 1.2.3.

### 1.2.1 ROSE project

ROSE is the acronym for “*Energy-efficient design and control of mobile RObotic SEnsor networks*”. Funded by Dutch Technology Foundation STW, ROSE is a collaboration between scientists (academia) and users (industry) (see Appendix B for an overview of the partners). ROSE started in 2010 and is part of the Autonomous Sensor Systems (ASSYS) program of STW under project 10550.

ROSE can be divided into two parts. The first part concerns the energy-efficient design of autonomous and mobile sensor-integrated robotic devices for data acquisition on dikes. This part is carried out by the University of Twente, where the focus is on the design of a mobile robotic sensor which acquires data



**Figure 1.3:** Experimental setup of the All-in-one/sensor validation experiment (Source: Stichting IJkdijk).

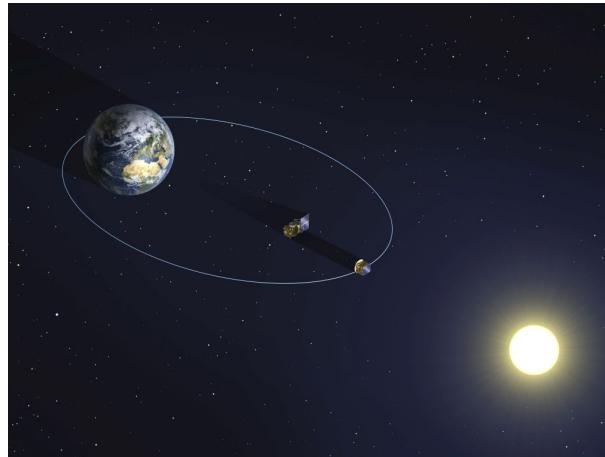
on dike composition, consistency and condition in an autonomous way [36]. The novelty of the design is a new concept for the locomotion of the robot, using Continuously Variable Transmissions [37].

The second part concerns the design of algorithms for coordinating a network of such robotic sensors, using ideas from passive systems [3] and port-Hamiltonian system theory [39, 99]. Exploiting the physical structure enables the generalization of the dike inspection application to other application areas such as the formation flying of satellites. The results in this thesis encompass this part of the ROSE project. The following two sections elaborate on the two application areas.

### 1.2.2 Dike inspection using robotic sensor networks

The first application area aligns with the algorithms developed in Chapters 3 and 4 and concerns the inspection of dikes. Recent dike breaches show that current inspection methods are not sufficient to guarantee dike safety (see Section A.1 for more background information on dike inspection in The Netherlands). Sensor technology has shown great potential for improving dike safety during several ground breaking experiments of the Stichting FloodControl IJkdijk (see Section A.2 for more background information on dike inspection using sensor technology). Figure 1.3 shows the experimental setup of one of these experiments, which was carried out by Stichting IJkdijk in 2012.

The results in this thesis enable the next step in the use of sensor technology for dike inspection, by developing formation control algorithms for robotic sensor networks. A robotic sensor network is a group (or network) of mobile sensor-



**Figure 1.4:** Proba-3 is a double satellite mission investigating close formation flying techniques (Source: ESA - P. Carril, 2013).

equipped robots, which gathers measurements in a coordinated fashion. Recent advances in sensor-equipped autonomous mobile robots enable the use of robotic sensor networks for a wide range of applications [21]. In addition, the experiments executed at the IJkdijk test facility show that sensor technology provides crucial information not attainable by traditional visual inspection methods.

Formation control is a crucial aspect for the use of robotic sensor networks in the dike inspection application. Acquiring high resolution measurements on the dike interior imposes a prescribed *acquisition geometry* to the sensor network [109], which corresponds to achieving prescribed relative displacements between the robotic sensors. Furthermore, the robots need to cover the whole dike surface which requires coverage or deployment control. At this moment there is no definition yet on the optimal formation shape for dike monitoring. Therefore the algorithms developed here are able to deal with different types of formations, based on different interconnection topologies.

Three application areas are foreseen for the use of robotic sensor networks within the dike inspection application. First of all, for existing dikes it is infeasible to install static sensors everywhere. The number of kilometers of dike and the costs for positioning static sensors is simply too high. A mobile robotic sensor network is more flexible and often less costly for the coordinated monitoring of dikes. Second, the profile and substance (interior) of dikes is in many cases unknown. Mobile robotic sensors can be used for the exploration of both. Finally, robotic sensors can be used for the guidance of drilling during the installation of existing static sensors.



**Figure 1.5:** Galileo satellite constellation (Source: ESA - J. Huart, 2002).

### 1.2.3 Formation flying of satellite constellations

The second application area concerns the results of Chapter 5 and deals with formation flying of satellites. Formation flying of satellites is usually divided according to the dynamic environment in which the satellites operate. For *deep space* formation flying the satellite translational dynamics are approximated as (double) integrators, while for *planetary orbital* formation flying the satellites are subject to significant gravitational dynamics and other environmental disturbances [101, 102]. Along the same line, two application areas are identified.

For deep space applications formation flying aims to coordinate several (or many) small satellites, such that the network acts as one big instrument. The dimensions of such a setup enable specifications which outperform single satellite setups. Other advantages of using a network of satellites are the ability to reconfigure formations, adapt baselines and acquire targets. Accurate control of the relative displacements amongst the satellites is important to achieve sufficiently high resolutions. Two missions which involve formation flying in deep space are the *Far-InfraRed Interferometry* (FIRI) mission of SRON Netherlands Institute for Space Research and the Proba-3 mission of ESA. FIRI aims to provide very large baselines and thus a high spatial resolution for the measuring of wavelengths between 25 and 300 microns. Proba-3 (see Figure 1.4) is a two satellite setup, which forms a 150 m long solar coronagraph to study the Sun's faint corona closer to the solar rim than has ever before been achieved (source: [sci.esa.int/sre-ft/37936-formation-flying](http://sci.esa.int/sre-ft/37936-formation-flying)).

The second application area considers formation flying on planetary orbits, which is related to coverage. Coverage here refers to the fact that each point on the planetary surface is always covered by one (or more) satellite(s). Well-known

examples are *Global Navigation Satellite Systems* (GNSS) like the Global Positioning System (GPS) and Galileo (Figure 1.5). For GNSSs it is of the utmost importance to phase satellites on the orbits, to ensure that there are always at least four satellites covering the object to be localized. The algorithms in Chapter 5 are generalizations of the algorithms in Chapters 3 and 4, dealing with the highly complex satellite dynamics.

## 1.3 Contributions

The contributions of this thesis are summarized as follows:

- This thesis provides an extension of the passivity-based design tool for network coordination in [3] to port-Hamiltonian systems. Coordination of the network is achieved using virtual couplings [58–60, 110–116], which consist of a virtual springs and dampers in parallel. The springs shape the energy function of the network, while the dampers inject damping for stabilization. In addition to standard formation control, several related control problems are considered (see next bullet).
- In addition to standard formation control, this thesis addresses formation control in the presence of Coulomb friction, deployment, velocity tracking, disturbance rejection for nonholonomic systems, and orbital phasing for satellites. Coulomb friction renders the agent dynamics non-smooth and rigorous stability proofs using tools from non-smooth analysis are given for both continuous virtual springs and their discontinuous counterpart [58, 59]. Deployment is achieved using a combination of virtual couplings and virtual walls, which are positioned at prescribed reference points [110]. Velocity tracking is achieved by stabilizing the error dynamics with respect to the reference velocity, which are obtained using generalized canonical transformations [113, 114]. Internal model control is able to counteract the effect of harmonic matched input disturbances for networks of nonholonomic wheeled robots [60, 116], while for constant disturbances only stability is proven [60]. Finally, the versatility of virtual couplings is shown, by their application to orbital phasing of satellites [114].
- The concept of virtual couplings is used for the formation control of three different classes of systems. For each system the virtual couplings are tailored to the system under consideration. Chapter 3 starts with continuous springs and develops a discontinuous counterpart to achieve formation control in the presence of Coulomb friction [58, 59]. For the nonholonomic wheeled robots in Chapter 4, the virtual couplings are assigned to the front ends of

the robots rather than the center of mass [60, 113, 115, 116]. This change of point of action is required to deal with the nonholonomic constraint on the wheel axle of the robot. For the satellites in Chapter 5, both translational and rotational couplings are considered [111, 112, 114]. The translational couplings guarantee convergence to the desired altitude, while the rotational couplings achieve orbital phasing and convergence to the desired angular velocity.

- Virtual couplings provides a clear physical interpretation to practitioners thereby facilitating implementation of the algorithms [86]. The physical structure of the network of agents interconnected by virtual couplings is exploited for the analysis and design of the control algorithms. Since the algorithms only exchange local information amongst agents in the network all algorithms are distributed and therefore easily scalable.
- The rigorous stability proofs for formation control in the presence of Coulomb friction require several tools for non-smooth dynamics. The use of the Krasovskii notion of solution, generalized Clarke gradient, and a non-smooth version of LaSalle’s invariance principle fits completely into port-Hamiltonian systems theory. These powerful tools enable the generalization of existing non-smooth theory to port-Hamiltonian systems.
- The algorithms and theory in this thesis are illustrated and validated using simulations and experiments. Using a testbed of e-puck wheeled robots experimental results are provided for formation control and deployment of fully actuated systems (Sections 3.3 and 3.5), and for formation control and velocity tracking of wheeled robots [113, 115]. For all other systems, extensive simulation results are provided [58–60, 110–112, 114, 116].

## 1.4 Thesis outline

The outline of the remainder of this thesis is as follows. Chapter 2 starts with the preliminaries on stability theory, graph theory, port-Hamiltonian systems, and non-smooth analysis. Subsequent Chapters 3, 4, and 5 deal with formation control of three types of systems.

Chapter 3 starts with formation control and deployment of fully actuated systems. Special attention is devoted to agents in the presence of Coulomb friction, which renders the agent dynamics non-smooth and requires tools from non-smooth systems theory for the stability analysis. Chapter 4 starts with the derivation of a dynamical model of the wheeled robot in the port-Hamiltonian framework. Then algorithms are designed for the problems of formation control, velocity tracking and matched input disturbance rejection. Chapter 5 deals with orbital phasing of

a network of satellites. A local controller keeps each satellite on the orbit, while a distributed controller equally distributes the satellites on the orbit. Concluding remarks and recommendations for future research are presented in Chapter 6.

The appendices provide more background information on dike inspection in The Netherlands (Appendix A), partners of the ROSE project (Appendix B), the experimental setup used in Chapters 3 and 4 (Appendix C), and complementary simulation and experimental results (Appendix D).

## 1.5 Publications

All publications contributing to this thesis are enlisted below, divided into journal papers, conference papers, conference abstracts, posters, and graduation projects.

### Journal papers

- E. Vos, J.M.A. Scherpen, and A.J. van der Schaft. Equal distribution of satellite constellations on circular target orbits. *Automatica*, 50(10):2641–2647, 2014
- E. Vos, A.J. van der Schaft, and J.M.A. Scherpen. Formation control and velocity tracking of wheeled robots, 2014. *Submitted*
- M. Jafarian, E. Vos, C. De Persis, A.J. van der Schaft, and J.M.A. Scherpen. Formation control of a multi-agent system subject to ideal Coulomb friction, 2014. *Submitted*
- M. Jafarian, E. Vos, C. De Persis, J.M.A. Scherpen, and A.J. van der Schaft. Disturbance rejection in formation keeping control of nonholonomic wheeled robots, 2015. *In preparation*

### Conference papers

- E. Vos, S. Fiaz, J.M.A. Scherpen, and A.J. van der Schaft. Control for formation flying of satellites: port-Hamiltonian approach. In *International Workshop on Satellite Constellations and Formation Flying*, Lisbon, Portugal, 2013. Paper ID: IWSCFF-2013-02-02
- E. Vos, J.M.A. Scherpen, and A.J. van der Schaft. Spatial distribution of satellite constellations on circular orbits. In *IEEE Conference on Decision and Control*, pages 829–834, Florence, Italy, 2013
- E. Vos, J.M.A. Scherpen, A.J. van der Schaft, and A. Postma. Formation control of wheeled robots in the port-Hamiltonian framework. In *World Congress of*

*the International Federation of Automatic Control*, pages 6662–6667, Cape Town, South Africa, 2014

- E. Vos, M. Jafarian, C. De Persis, J.M.A. Scherpen, and A.J. van der Schaft. Formation control of nonholonomic wheeled robots in the presence of matched input disturbances, 2015. *Submitted*

## Conference abstracts

- E. Vos, J.M.A. Scherpen, and A.J. van der Schaft. Port-Hamiltonian approach to deployment. In *International Symposium on Mathematical Theory of Networks and Systems*, Melbourne, Australia, 2012
- M. Jafarian, E. Vos, C. De Persis, A.J. van der Schaft, and J.M.A. Scherpen. On formation control of agents subject to ideal Coulomb friction. In *International Symposium on Mathematical Theory of Networks and Systems*, pages 1736–1739, Groningen, The Netherlands, 2014
- E. Vos, J.M.A. Scherpen, and A.J. van der Schaft. “Port-Hamiltonian approach to deployment on a line”. *Benelux Meeting on Systems and Control*. Heijden, The Netherlands, 2012
- E. Vos, J.M.A. Scherpen, and A.J. van der Schaft. “Formation flying of satellites: a port-Hamiltonian approach”. *Benelux Meeting on Systems and Control*. Houffalize, Belgium, 2013
- M. Jafarian, E. Vos, C. De Persis, J.M.A. Scherpen, and A.J. van der Schaft. “Formation keeping control: a quantized port-Hamiltonian approach”. *Benelux Meeting on Systems and Control*. Houffalize, Belgium, 2013
- E. Vos, M. Jafarian, C. De Persis, J.M.A. Scherpen, and A.J. van der Schaft. “Formation control in the presence of ideal Coulomb friction”. *Benelux Meeting on Systems and Control*. Heijden, The Netherlands, 2014

## Posters

- E. Vos, J.M.A. Scherpen, and A.J. van der Schaft. *Port-Hamiltonian approach to deployment of a robotic sensor network*. Poster presented at Sense of Contact, April 11, Kontakt der Kontinenten, Soesterberg, The Netherlands. 2012
- E. Vos, J.M.A. Scherpen, and A.J. van der Schaft. *Coordination of a Robotic Sensor Network*. Poster presented at Vision, Robotics & Mechatronics en de RoboNED Conference, May 22–23, NH Conference Centre Koningshof, Veldhoven, The Netherlands. 2012

- E. Vos, J.M.A. Scherpen, and A.J. van der Schaft. *Equal Distribution of Satellite Constellations on Circular Orbits*. Poster presented at Sense of Contact, April 10, Kontakt der Kontinenten, Soesterberg, The Netherlands. 2013
- E. Vos, J.M.A. Scherpen, and A.J. van der Schaft. *ROSE: Energy-efficient design and control of mobile robotic sensor networks*. Poster presented at the STW Annual Congress, October 3, NBC, Nieuwegein, The Netherlands. 2013
- E. Vos, J.M.A. Scherpen, and A.J. van der Schaft. *Formation Control of Wheeled Robots in an Energy-Based Setting*. Poster presented at Sense of Contact, April 2, Kontakt der Kontinenten, Soesterberg, The Netherlands. 2014

## Graduation projects

The following graduation projects to obtain the Master or Bachelor degree in Industrial Engineering and Management contributed directly or indirectly to this thesis:

- T. Hoogerwerf. Dijken verkennen door groepen autonome robots. Bachelor's thesis, University of Groningen, 2011
- Ö. Arslan. Design of a collaboration strategy for autonomous multi-robot systems. an application in plastic marine debris cleanup. Bachelor's thesis, University of Groningen, 2012
- R. Kamps. Reinforcement learning: As a strategy in a robotic fish waterpolo competition, as a proof of concept for the enhancement of order picking performance. Bachelor's thesis, University of Groningen, 2012
- A. Postma. Port-Hamiltonian based deployment of dike robots. Master's thesis, University of Groningen, 2013
- E. Halbertsma. Potential field based obstacle avoidance with multiple robotic agents in the port-Hamiltonian framework. Master's thesis, University of Groningen, 2013
- M. Chahbari. Port-Hamiltonian modeling and control of a mobile multi robot system for borehole radar dike inspections. Master's thesis, University of Groningen, 2014

## 1.6 Notation

Throughout the remainder of this thesis, the following notion is adopted.

- Let  $n$  denote the dimension of the state space under consideration and let  $\mathbb{R}, \mathbb{R}^n, \mathbb{R}^{n \times n}$  denote respectively the set of real numbers, the set of  $n$ -dimensional column vectors of real numbers, and the set of  $n \times n$  matrices of real numbers. Moreover,  $\mathbf{1}_n$  denotes the  $n$ -dimensional column vector of ones, given by  $\mathbf{1}_n = (1, \dots, 1)^T$ . Unless otherwise stated 0 is always assumed to be appropriately dimensioned.
- For two vectors  $a, b \in \mathbb{R}^n$ , their *duality product* is denoted by  $\langle a | b \rangle$ .
- For a scalar function  $H(x) : \mathbb{R}^n \rightarrow \mathbb{R}$  with  $x \in \mathbb{R}^n$  the column vector of partial derivatives  $\frac{\partial H}{\partial x}(x)$  and the  $n \times n$  *Hessian matrix*  $\frac{\partial^2 H}{\partial x^2}(x)$  are defined respectively as

$$\frac{\partial H}{\partial x}(x) = \begin{pmatrix} \frac{\partial H}{\partial x_1}(x) \\ \vdots \\ \frac{\partial H}{\partial x_n}(x) \end{pmatrix}, \quad \frac{\partial^2 H}{\partial x^2}(x) = \begin{pmatrix} \frac{\partial^2 H}{\partial x_1^2}(x) & \cdots & \frac{\partial^2 H}{\partial x_1 \partial x_n}(x) \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 H}{\partial x_n \partial x_1}(x) & \cdots & \frac{\partial^2 H}{\partial x_n^2}(x) \end{pmatrix}.$$

For a vector function  $v(x) = (v_1(x), \dots, v_m(x))^T$ , with scalar functions  $v_i(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ , the  $m \times n$  *Jacobian matrix*  $\frac{\partial v}{\partial x}(x)$  is defined as

$$\frac{\partial v}{\partial x}(x) = \begin{pmatrix} \frac{\partial v_1}{\partial x_1}(x) & \cdots & \frac{\partial v_1}{\partial x_n}(x) \\ \vdots & \ddots & \vdots \\ \frac{\partial v_m}{\partial x_1}(x) & \cdots & \frac{\partial v_m}{\partial x_n}(x) \end{pmatrix}.$$

- Consider two column vectors  $S_i(x), S_j(x) \in \mathbb{R}^n$  with  $x \in \mathbb{R}^n$ . Then, the *Lie-bracket* denoted by  $[S_i, S_j](x)$  is defined as

$$[S_i, S_j](x) = \frac{\partial S_j}{\partial x}(x)S_i(x) - \frac{\partial S_i}{\partial x}(x)S_j(x),$$

with  $\frac{\partial S_j}{\partial x}(x)$  the Jacobian matrix.

- For graph  $\mathcal{G}(\mathcal{V}, \mathcal{E})$  let  $N = |\mathcal{V}|$  denote the number of nodes and let  $E = |\mathcal{E}|$  denote the number of edges. Moreover, let  $N_i = |\mathcal{V}_i|$  denote the number of *internal nodes* and let  $N_b = |\mathcal{V}_b|$  denote the number of *boundary nodes* (see Section 2.3.4).
- For two matrices  $A \in \mathbb{R}^{m \times n}$  and  $B \in \mathbb{R}^{p \times q}$  the *Kronecker product*  $A \otimes B \in \mathbb{R}^{mp \times nq}$  is defined as

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

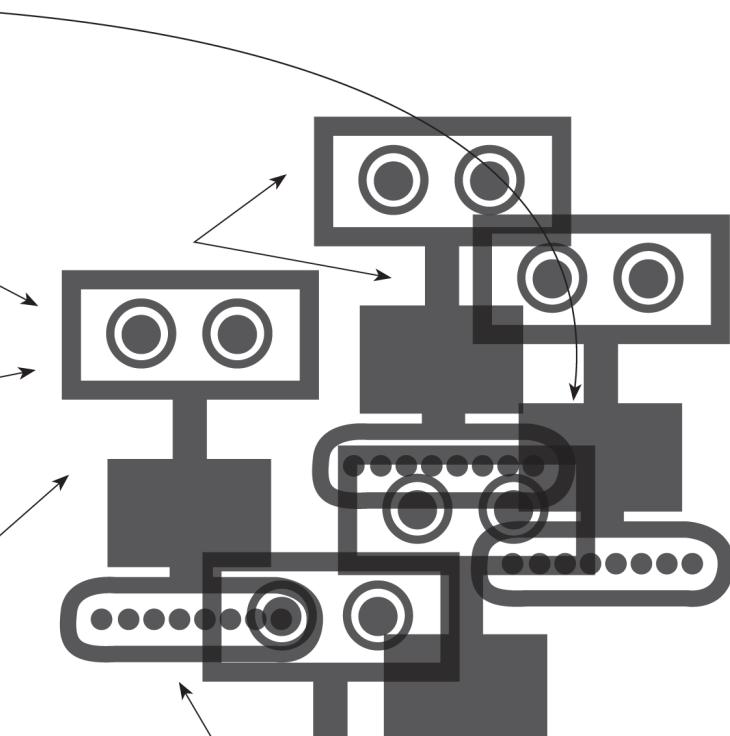
Furthermore, let  $\ker A$  denote the *kernel* of matrix  $A$  and  $\text{im } A$  the *image*.

- The *Krasovskii operator* acting on some (possibly discontinuous) function  $f(x)$  is denoted by  $\mathcal{K}f(x)$  (see Section 2.4).
- The *convex hull* is denoted by  $\text{co}$ , while the *convex closure* is denoted by  $\overline{\text{co}}$ . A *disc* of radius  $r$  centered at  $x \in \mathbb{R}^n$  is denoted by  $B(x, r)$  (see Section 3.4).



## Chapter 2

### Preliminaries





# Chapter 2

---

## Preliminaries

This chapter provides the preliminaries for the remainder of this thesis on stability of time-invariant systems, graph theory, port-Hamiltonian systems, and non-smooth analysis. The fundamental Lyapunov stability theorems for time-invariant systems without inputs (Section 2.1) are used to assess the stability for networks of fully actuated point masses (Chapter 3), nonholonomic wheeled robots (Chapter 4), and satellites (Chapter 5).

Throughout this thesis graph theory (Section 2.2) is used to model the interconnection topology of the network under consideration. The nodes of the graph correspond to the agents, while the edges of the graph correspond to the virtual couplings interconnecting the agents.

All systems are modeled within the port-Hamiltonian framework, which is introduced in Section 2.3. Fully actuated systems (Chapter 3) and satellites (Chapter 5) are modeled as (mechanical) input-state-output port-Hamiltonian systems, while nonholonomic wheeled robots (Chapter 4) are modeled as mechanical port-Hamiltonian systems with constraints. Generalized canonical transformations are used to derive error dynamics in Chapters 4 and 5. The recently introduced theory of port-Hamiltonian systems on graphs (Section 2.3.4) provides tools for the modeling and analysis of networks.

Finally, Section 2.4 recalls some essential tools from non-smooth analysis, which are used in the analysis of the discontinuous systems considered in Chapter 3. Concluding remarks of the chapter are given in Section 2.5.

### 2.1 Stability of time-invariant systems

This section presents a brief recall of Lyapunov stability theory for time-invariant systems (see [64] for more details). Let  $x \in \mathbb{R}^n$  denote the state of the system of interest and consider the time-invariant system

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

where  $f : \mathcal{D} \rightarrow \mathbb{R}^n$  is a locally Lipschitz map from a domain  $\mathcal{D} \subset \mathbb{R}^n$  into  $\mathbb{R}^n$ . Suppose that  $x^*$  is an equilibrium point of (2.1), i.e.,  $f(x^*) = 0$ . Without loss of

generality, assume that  $x^* = 0$ . The stability of the origin  $x^* = 0$  is defined as follows.

**Definition 2.1** (Stability [64]). The equilibrium point  $x^* = 0$  of (2.1) is

- **stable** if, for each  $\epsilon > 0$  there exists a  $\delta_1 = \delta_1(\epsilon) > 0$  such that

$$\|x(0)\| < \delta_1 \Rightarrow \|x(t)\| < \epsilon, \quad \forall t \geq 0.$$

- **unstable** if it is not stable.

- **asymptotically stable** if it is stable and additionally  $\delta_2$  can be chosen such that

$$\|x(0)\| < \delta_2 \Rightarrow \lim_{t \rightarrow \infty} x(t) = 0.$$

Stability of system (2.1) in the sense of Definition 2.1 can be assessed using Lyapunov's well-known stability theorem, which is stated next.

**Theorem 2.2** (Lyapunov's direct method [64]). Let  $x = 0$  be an equilibrium point for (2.1) and  $\mathcal{D} \subset \mathbb{R}^n$  be an open subset containing  $x = 0$ . Let  $V : \mathcal{D} \rightarrow \mathbb{R}$  be a continuously differentiable function such that

$$V(0) = 0 \text{ and } V(x) > 0 \text{ in } \mathcal{D} - \{0\}, \quad (2.2)$$

$$\dot{V}(x) \leq 0 \text{ in } \mathcal{D}. \quad (2.3)$$

Then,  $x = 0$  is stable. Moreover, if

$$\dot{V}(x) < 0 \text{ in } \mathcal{D} - \{0\}, \quad (2.4)$$

then  $x = 0$  is asymptotically stable.

A continuously differentiable function  $V(x)$  satisfying (2.2) and (2.3) is called a *Lyapunov function*. Many (physical) systems fail to meet condition (2.4), because  $\dot{V}$  is only negative semi-definite ( $\dot{V} \leq 0$ ) in  $\mathcal{D} - \{0\}$ . In this case, LaSalle's invariance principle can be invoked to assess the stability of (2.1). In order to state LaSalle's invariance principle, first the definition of a positively invariant set is needed.

**Definition 2.3** (Positively invariant set [64]). A set  $\mathcal{M}$  is said to be a *positively invariant* set if

$$x(0) \in \mathcal{M} \Rightarrow x(t) \in \mathcal{M}, \quad \forall t \geq 0.$$

Using Definition 2.3 LaSalle's invariance principle is now stated as follows.

**Theorem 2.4** (LaSalle's invariance principle [64]). Let  $\Omega \subset \mathcal{D}$  be a compact set that is positively invariant with respect to (2.1). Let  $V : \mathcal{D} \rightarrow \mathbb{R}$  be a  $C^1$  function satisfying (2.3). Let  $\mathcal{F}$  be the set of all points in  $\mathcal{D}$  where  $\dot{V}(x) = 0$ . Let  $\mathcal{M}$  be the largest invariant set in  $\mathcal{F}$ . Then every solution starting in  $\Omega$  approaches  $\mathcal{M}$  as  $t \rightarrow \infty$ .

## 2.2 Graph theory

This section provides some essentials from the field of graph theory. Graphs are (mathematical) structures to model the pairwise interaction between objects. They are used to model many types of systems in physical, biological, social, and information systems. Examples are atomic structures, spreading of diseases amongst animal populations, rumor spreading, and the structure of websites. Graph theory provide powerful tools for the modeling, analysis, and design of complex networks.

In this thesis, graphs are used to model the interaction amongst a group of agents. The nodes of the graph corresponds to the agents, while the edges corresponds to the fact whether or not two agents interact. Dynamics are assigned to the edges in the form of virtual couplings, which consist of springs and dampers in parallel. Therefore, the graph can be interpreted as the physical interconnection structure of the agents and virtual couplings. The definitions below are distilled from [11, 50, 78].

A graph is a pair  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  consisting of a *node set*  $\mathcal{V}$  and an *edge set*  $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ . The node set  $\mathcal{V} = \{n_1, \dots, n_N\}$  has  $N = |\mathcal{V}|$  elements, while the edge set  $\mathcal{E} = \{e_1, \dots, e_E\}$  has  $E = |\mathcal{E}|$  elements. As said before, the nodes of the graph correspond to agents, while the edges correspond to virtual couplings.

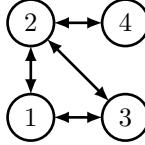
There exists an edge  $(n_i, n_j) \in \mathcal{E}$  if and only if agent  $i$  can interact with agent  $j$ . Since the graph models the *physical* interconnection structure, only *undirected* graphs are considered throughout this thesis. For undirected graphs the edge set is an unordered pair of nodes (i.e.,  $(n_i, n_j) \in \mathcal{E}$  if and only if  $(n_j, n_i) \in \mathcal{E}$ ). Furthermore, self-loops are not considered (i.e.,  $(n_i, n_i) \notin \mathcal{E}$ ). For an undirected graph, one can arbitrarily assign an orientation to each edge [78], by assigning a positive sign to one end (the *head*) and a negative sign to the other end (the *tail*).

For graph  $\mathcal{G}$ , there exists an *undirected path* from node  $n_i$  to node  $n_j$  if there exists a sequence of distinct edges  $e_1, \dots, e_K$  such that  $e_k \in \mathcal{E}$  for  $k = 1, \dots, K$  and  $e_1 = (n_i, *), e_K = (*, n_j)$ . A graph is *connected* if there exists a path from every node to every other node. A *cycle* is a path that starts and ends with the same node.

Graphs admit a straightforward representation in terms of matrices. For an undirected graph  $\mathcal{G}$ , let  $d(n_i)$  denote the *degree* of node  $n_i$ , which equals the number of nodes which are adjacent to  $n_i$  in  $\mathcal{G}$ . The *degree matrix*  $\Delta \in \mathbb{R}^{N \times N}$  is defined as  $\Delta = \text{diag}(d(n_1), \dots, d(n_N))$ . The *adjacency matrix*  $A \in \mathbb{R}^{N \times N}$  encodes the adjacency relationships in  $\mathcal{G}$  [78], and is defined as

$$a_{ij} = \begin{cases} 1 & \text{if } (n_i, n_j) \in \mathcal{E}, \\ 0 & \text{if } (n_i, n_j) \notin \mathcal{E}. \end{cases}$$

The adjacency matrix  $A$  is symmetric, since  $\mathcal{G}$  is undirected, and  $a_{ii} = 0$  since self



**Figure 2.1:** Graph for Example 2.1.

loops are not considered here. The *incidence matrix*  $B \in \mathbb{R}^{N \times E}$  associated to  $\mathcal{G}$  is defined as

$$b_{ik} = \begin{cases} +1 & \text{if node } n_i \text{ is the head of edge } e_k, \\ -1 & \text{if node } n_i \text{ is the tail edge } e_k, \\ 0 & \text{otherwise.} \end{cases}$$

In contrast with the adjacency matrix  $A$ , the incidence matrix  $B$  not only models the adjacency relationships, but also the orientation of the graph. The  $N$  rows of  $B$  correspond to the nodes of  $\mathcal{G}$ , while the  $E$  columns correspond to the edges of  $\mathcal{G}$ . The last matrix considered here is the *Laplacian matrix*  $L \in \mathbb{R}^{N \times N}$ , which is defined as

$$\ell_{ij} = \begin{cases} \sum_{j=1}^N a_{ij} & \text{if } i = j, \\ -a_{ij} & \text{if } i \neq j. \end{cases}$$

The Laplacian matrix  $L$  can also be expressed as  $L = BB^T$  or  $L = \Delta - A$  [78].

Chapters 3 and 4 deal with a specific subclass of graphs, the so-called *tree graphs*. A graph  $\mathcal{T}(\mathcal{V}, \mathcal{E})$  is called a tree graph if  $\mathcal{T}$  is an undirected graph in which any two nodes are connected by exactly one unique path. This definition is equivalent to  $\mathcal{T}$  being connected and acyclic [50]. For any tree graph  $\mathcal{T}$  the nodes of degree 1 (i.e.,  $\{n \in \mathcal{V} | d(n) = 1\}$ ) are called *terminal nodes*. In other words, a terminal node is a node with exactly one edge incident to it. Furthermore, we define *terminal edges* as those edges which are incident to terminal nodes.

Consider a tree graph  $\mathcal{T}$ . Since it is acyclic and connected,  $\mathcal{T}$  has at least two terminal nodes. Hence, the incidence matrix  $B$  associated to  $\mathcal{T}$  has at least two rows with exactly one nonzero entry. Removing a terminal node from  $\mathcal{T}$  results in a new tree graph  $\mathcal{T}'$ , since  $\mathcal{T}'$  remains connected and acyclic.

**Example 2.1.** Consider the undirected graph  $\mathcal{G}(\mathcal{V}, \mathcal{E})$  shown in Figure 2.1. The node set is given by  $\mathcal{V} = \{n_1, n_2, n_3, n_4\}$  and the edge set is given by  $\mathcal{E} = \{(n_1, n_2), (n_1, n_3), (n_2, n_3), (n_2, n_4)\}$ . Since there is a path from every node to every other node the graph is connected. However, since the graph has a cycle with corresponding path  $(n_1, n_2, n_3, n_1)$ , graph  $\mathcal{G}$  is not a tree graph. The degree matrix  $\Delta$ , adjacency matrix  $A$ , incidence matrix  $B$ , and Laplacian matrix  $L$  associated to  $\mathcal{G}$

are given by

$$\Delta = \begin{pmatrix} 2 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad A = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix},$$

$$B = \begin{pmatrix} -1 & -1 & 0 & 0 \\ 1 & 0 & 1 & -1 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad L = \begin{pmatrix} 2 & -1 & -1 & 0 \\ -1 & 3 & -1 & -1 \\ -1 & -1 & 2 & 0 \\ 0 & -1 & 0 & 1 \end{pmatrix}.$$

## 2.3 Port-Hamiltonian systems

The *port-Hamiltonian* framework was introduced in [73] as an energy-based framework for modeling (nonlinear) systems in different domains (mechanical, electrical, etc.). The name comes from two key ingredients of the framework: *power ports* to interconnect (sub)systems and the *Hamiltonian*, which is the total energy stored in the system.

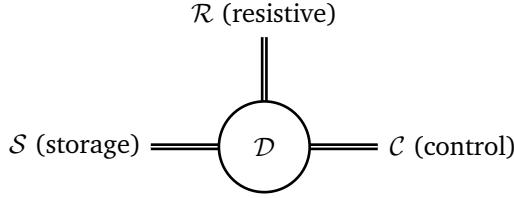
Power ports provide an interface for the sub-models within the model to interact with each other. In physical systems, these interactions are related to the exchange (or flow) of energy (i.e., power) [39]. Each power port has two power-conjugate variables called *flow* and *effort*. The flow vector  $f$  belongs to the flow space  $\mathcal{F}$ , while the effort vector  $e$  belongs to the effort space  $\mathcal{E}$  which is the dual of the flow space  $\mathcal{E} := \mathcal{F}^*$ . The total space of flows and effort  $\mathcal{F} \times \mathcal{F}^*$  is called the space of port-variables. On the total space of port-variables power is defined as  $P = \langle e | f \rangle$ , where  $\langle e | f \rangle$  denotes the duality product (i.e., the product of flows and efforts has the dimension of power). In the remainder  $\mathcal{F} = \mathbb{R}^n$  and  $f$  and  $e$  are written as column vectors, which implies that  $P = \langle e | f \rangle = e^T f$ .

Central to the port-Hamiltonian framework is the *Dirac structure*  $\mathcal{D}$ , which interconnects sub-systems in a power conserving manner (see Figure 2.2). A formal definition of a Dirac structure is given in the following definition.

**Definition 2.5** (Dirac structure [99]). Consider a finite-dimensional linear space  $\mathcal{F}$  with  $\mathcal{E} = \mathcal{F}^*$ . A subspace  $\mathcal{D} \subset \mathcal{F} \times \mathcal{E}$  is a *Dirac structure* if

1.  $\langle e | f \rangle = 0$  for all  $(f, e) \in \mathcal{D}$ ,
2.  $\dim \mathcal{D} = \dim \mathcal{F}$ .

Property (1) of Definition 2.5 corresponds to power-conservation and expresses the fact that the total power entering (or leaving) a Dirac structure is zero. Another crucial property of the Dirac structure, is that the interconnection of two Dirac structures results in another Dirac structure. As shown in Figure 2.2, there are three



**Figure 2.2:** Port-Hamiltonian system with Dirac structure  $\mathcal{D}$ .

ports entering the Dirac structure. Port  $S$  corresponds to internal energy-storage, port  $\mathcal{R}$  corresponds to the internal energy-dissipation, and port  $C$  corresponds to an external control port. A fourth port corresponding to interaction with the environment is omitted here for simplicity.

This section continues with the description of three representation of port-Hamiltonian systems: input-state-output port-Hamiltonian systems, mechanical port-Hamiltonian systems with constraints, and port-Hamiltonian systems on graphs.

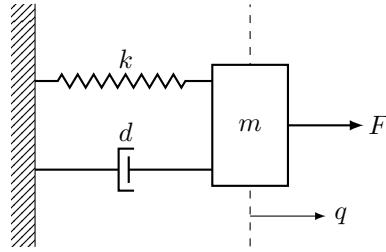
### 2.3.1 Input-state-output port-Hamiltonian systems

This section is based on Chapter 4 in [99]. Let  $x \in \mathcal{X}$  denote the state of the system to be modeled, where  $\mathcal{X}$  denotes the state space (often  $\mathcal{X} = \mathbb{R}^n$ , with  $n = 1, 2, \dots$ ). Under certain assumptions on the Dirac structure and the resistive relation the port-Hamiltonian system can be written into the following form:

$$\begin{aligned}\dot{x} &= (J(x) - R(x)) \frac{\partial H}{\partial x}(x) + g(x)u, \\ y &= g^T(x) \frac{\partial H}{\partial x}(x),\end{aligned}\tag{2.5}$$

with skew-symmetric interconnection matrix  $J(x) = -J^T(x)$ , positive semi-definite dissipation matrix  $R(x) = R^T(x) \geq 0$ , input matrix  $g(x)$ , and Hamiltonian  $H(x)$ . A port-Hamiltonian system of the form (2.5) is called an *input-state-out port-Hamiltonian system*. Port-variables  $(u, y)$  correspond to the control port  $C$  in Figure 2.2. The dissipation matrix  $R(x)$  may be replaced by adding an additional *resistive port* with port-variables  $(f_R, e_R)$  (see Section 3.4 where such a resistive port is used to model discontinuous friction). Since the product of two port variables has the dimension of power, for the resistive port  $(f_R, e_R)$  it immediately follows that  $e_R^T f_R \leq 0$  (i.e., resistive elements dissipate energy).

It is easily verified that (2.5) is always passive [98] with respect to port-variables



**Figure 2.3:** Mass-spring-damper system of Example 2.2.

$(u, y)$  with storage function  $H(x)$  since

$$\begin{aligned}\dot{H} &= \frac{\partial^T H}{\partial x} \dot{x} \\ &= \frac{\partial^T H}{\partial x} (J(x) - R(x)) \frac{\partial H}{\partial x}(x) + \frac{\partial^T H}{\partial x} g(x) u \\ &= -\frac{\partial^T H}{\partial x} R(x) \frac{\partial H}{\partial x}(x) + y^T u \\ &\leqslant y^T u.\end{aligned}\tag{2.6}$$

The interpretation of (2.6) is that the increase in stored energy  $\dot{H}$  is always smaller than or equal to the power  $y^T u$  supplied through the control port  $\mathcal{C}$ . Passivity is often used to prove the stability of the closed-loop system.

A feed-through term may be added to (2.5) (see Section 4.2 in [99]). Port-Hamiltonian systems with feed-through terms show up when the virtual couplings include dampers (Chapters 3, 4, and 5).

**Example 2.2** (Mass-spring-damper system). Consider a simple mass-spring-damper system, where all components are assumed to be linear (see Figure 2.3). Let  $q$  denote the position of the mass and  $p = mq$  the corresponding momentum. The mass  $m$  is assumed to be at rest and is subject to a damping force  $F_d = d\dot{q}$ , a spring force  $F_s = kq$  and an input force  $F$ . The port-Hamiltonian dynamics of such a system can be written as

$$\begin{pmatrix} \dot{q} \\ \dot{p} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -1 & -d \end{pmatrix} \begin{pmatrix} \frac{\partial H}{\partial q}(q, p) \\ \frac{\partial H}{\partial p}(q, p) \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} F,\tag{2.7}$$

with Hamiltonian

$$H(q, p) = \frac{1}{2m}p^2 + \frac{1}{2}kq^2.$$

The first term of the Hamiltonian is the kinetic energy stored in the mass, while the second term is the potential energy stored in the spring. Note for this example that

the corresponding output  $\frac{\partial H}{\partial p} = p/m$  is simply the velocity  $\dot{q}$  of the mass. The time derivative of the Hamiltonian is given by

$$\dot{H} = \frac{\partial H}{\partial q}\dot{q} + \frac{\partial H}{\partial p}\dot{p} = -d\left(\frac{p}{m}\right)^2 \leq 0,$$

which verifies that the mass-spring-damper system is a passive system.

### 2.3.2 Mechanical port-Hamiltonian systems with constraints

All models considered in this thesis are defined in the mechanical domain. In the mechanical domain (2.5) often takes a special form, called a *standard mechanical port-Hamiltonian system*. Let  $q \in \mathbb{R}^n$  denote the position and  $p = M(q)\dot{q} \in \mathbb{R}^n$  the corresponding momentum. The input  $u \in \mathbb{R}^m$  is a force, while the corresponding output  $y \in \mathbb{R}^m$  is a velocity. The dynamics for such a system are given by

$$\begin{aligned} \begin{pmatrix} \dot{q} \\ \dot{p} \end{pmatrix} &= \begin{pmatrix} 0 & I \\ -I & -D(q, p) \end{pmatrix} \begin{pmatrix} \frac{\partial H}{\partial q}(q, p) \\ \frac{\partial H}{\partial p}(q, p) \end{pmatrix} + \begin{pmatrix} 0 \\ B(q) \end{pmatrix} u, \\ y &= B^T(q) \frac{\partial H}{\partial p}(q, p), \end{aligned} \quad (2.8)$$

with dissipation matrix  $D(q, p)$  and input matrix  $B(q)$ . The Hamiltonian is given by  $H(q, p) = \frac{1}{2}p^T M(q)^{-1}p + V(q)$ , where  $\frac{1}{2}p^T M(q)^{-1}p$  denotes the kinetic energy and  $V(q)$  the potential energy.

**Example 2.2 (continued).** The mass-spring-damper system (2.7) of Example 2.2 can be rewritten in the form (2.8), by taking  $D(q, p) = d, M(q) = m, V(q) = \frac{1}{2}kq^2, B(q) = 1$ .

Chapter 4 deals with the modeling and control of nonholonomic wheeled robots. These robots are of the form (2.8) with an additional algebraic constraint. Set  $D(q, p) = 0$  for simplicity, then the dynamics of such a constrained system are given by [99]

$$\begin{aligned} \begin{pmatrix} \dot{q} \\ \dot{p} \end{pmatrix} &= \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix} \begin{pmatrix} \frac{\partial H}{\partial q}(q, p) \\ \frac{\partial H}{\partial p}(q, p) \end{pmatrix} + \begin{pmatrix} 0 \\ A(q) \end{pmatrix} \lambda + \begin{pmatrix} 0 \\ B(q) \end{pmatrix} u, \\ y &= B^T \frac{\partial H}{\partial p}(q, p), \\ 0 &= A^T \frac{\partial H}{\partial p}(q, p), \end{aligned} \quad (2.9)$$

with Lagrange multiplier  $\lambda$ . Here  $A(q)\lambda$  corresponds to the constraint forces. In

general the algebraic constraint  $A^T(x) \frac{\partial H}{\partial p}(q, p) = 0$  in (2.9) constrains the state variables  $q, p$ . Under the assumption that  $A(q)$  has full column-rank, there exist a matrix  $S(q)$  such that  $\tilde{p}^1 = S^T(q)p$ ,  $\tilde{p}^2 = A^T(q)p$  defines a coordinate transformation (see Section 8.4 in [99] for more details). Using the coordinate transformation  $(q, p) \rightarrow (q, \tilde{p}^1, \tilde{p}^2)$  the constrained dynamics are completely determined by the dynamics of  $q$  and  $\tilde{p}^1$ . The dynamics on the constrained state space are then given by

$$\begin{pmatrix} \dot{q} \\ \dot{\tilde{p}}^1 \end{pmatrix} = \begin{pmatrix} 0 & S(q) \\ -S^T(q) & (-p^T [S_i, S_j](q))_{i,j} \end{pmatrix} \begin{pmatrix} \frac{\partial H_c}{\partial q}(q, \tilde{p}^1) \\ \frac{\partial H_c}{\partial \tilde{p}^1}(q, \tilde{p}^1) \end{pmatrix} + \begin{pmatrix} 0 \\ B_c(q) \end{pmatrix} u, \\ y = B_c^T \frac{\partial H}{\partial \tilde{p}^1}(q, \tilde{p}^1) \quad (2.10)$$

where  $B_c(q) = S^T(q)B(q)$  and  $p$  is expressed as a function of  $q, \tilde{p}^1$ . Here  $S_i$  denotes the  $i$ -th column of  $S(q)$  and  $[S_i, S_j]$  is the Lie-bracket of  $S_i$  and  $S_j$  defined as

$$[S_i, S_j](q) = \frac{\partial S_j}{\partial q}(q)S_i(q) - \frac{\partial S_i}{\partial q}(q)S_j(q).$$

### 2.3.3 Generalized canonical transformations

Canonical transformations are widely used for the analysis of the dynamics of mechanical systems in classical mechanics [1]. In [44] the authors introduced *generalized canonical transformations* for port-Hamiltonian systems. The key idea behind generalized canonical transformations is to preserve the port-Hamiltonian structure under time-varying coordinate and feedback transformations.

Generalized canonical transformations can be used for stabilization of port-Hamiltonian systems with respect to a (possibly time-varying) trajectory [45]. In Chapter 4 these ideas are used in the analysis of a group of nonholonomic wheeled robots tracking of a reference velocity, while in Chapter 5 they are used in the analysis of a satellite constellation tracking of a circular target orbit. A generalized canonical transformation is formally defined in the following definition.

**Definition 2.6** (Generalized canonical transformation). A generalized canonical transformation is defined as a set of transformations

$$\begin{aligned} \bar{x} &= \Phi(x, t), \\ \bar{H} &= H(x, t) + U(x, t), \\ \bar{y} &= y + \alpha(x, t), \\ \bar{u} &= u + \beta(x, t), \end{aligned} \quad (2.11)$$

which preserves the port-Hamiltonian structure from (2.5). I.e., a generalized

canonical transformation provides the *error dynamics* for a specific choice of  $\Phi$  as

$$\begin{aligned}\dot{\bar{x}} &= (\bar{J}(\bar{x}, t) - \bar{R}(\bar{x}, t)) \frac{\partial \bar{H}}{\partial \bar{x}}(\bar{x}, t) + \bar{g}(\bar{x}, t)\bar{u}, \\ \bar{y} &= \bar{g}^T(\bar{x}, t) \frac{\partial \bar{H}}{\partial \bar{x}}(\bar{x}, t),\end{aligned}\tag{2.12}$$

with  $\bar{x}, \bar{J}, \bar{R}, \bar{H}, \bar{g}, \bar{u}, \bar{y}$  respectively the new state, structure matrix, dissipation matrix, Hamiltonian, input matrix, input, and output.

Generalized canonical transformations provide a framework for constructing error systems, which describes the dynamics of the tracking error, in the port-Hamiltonian framework. The key idea is to embed the desired trajectory into the Hamiltonian function of the original system. Stabilizing the error system using passivity based approaches then achieves the trajectory tracking objective [45]. The two most important results from [45] are now presented in Theorems 2.7 and 2.8 below.

**Theorem 2.7** (Generalized canonical transformation [45]). *Consider system (2.5). For any functions  $U(x, t) \in \mathbb{R}$  and  $\beta(x, t) \in \mathbb{R}^m$ , there exists a pair of functions  $\Phi(x, t) \in \mathbb{R}^n$  and  $\alpha(x, t) \in \mathbb{R}^m$  such that set (2.11) yields a generalized canonical transformation. The quadruple  $(\Phi, U, \alpha, \beta)$  is a generalized canonical transformation if and only if  $\Phi, U, \alpha = g^T \frac{\partial U}{\partial x}, \beta$  satisfy the partial differential equation*

$$\frac{\partial \Phi}{\partial(x, t)} \begin{pmatrix} (J - R) \frac{\partial U}{\partial x} + (K - S) \frac{\partial(H+U)}{\partial x} + g\beta \\ -1 \end{pmatrix} = 0,\tag{2.13}$$

for a skew-symmetric matrix  $K(x, t) \in \mathbb{R}^{n \times n}$  and a symmetric matrix  $S(x, t) \in \mathbb{R}^{n \times n}$  satisfying  $R + S \geq 0$ . The new system matrices  $\bar{J}$ ,  $\bar{R}$  and  $\bar{g}$  in (2.12) are given by

$$\begin{aligned}\bar{g} &= \frac{\partial^T \Phi}{\partial x} g, \\ \bar{J} &= \frac{\partial^T \Phi}{\partial x} (J + K) \frac{\partial \Phi}{\partial x}, \\ \bar{R} &= \frac{\partial^T \Phi}{\partial x} (R + S) \frac{\partial \Phi}{\partial x}.\end{aligned}\tag{2.14}$$

The following theorem provides another partial differential equation to check whether (2.12) is passive with respect to the new port-variables  $(\bar{u}, \bar{y})$  with the new Hamiltonian  $\bar{H}$  as storage function.

**Theorem 2.8** (Passivity and asymptotic stability [45]). *Consider the system (2.5) transformed by the generalized canonical transformation with  $u$ ,  $\beta$  and  $S$  such that  $H + U \geq 0$ , then the new input-output mapping  $\bar{u} \mapsto \bar{y}$  is passive with the storage*

function  $\bar{H}$  if and only if

$$\frac{\partial(H+U)}{\partial(x,t)} \begin{pmatrix} (J-R)\frac{\partial U}{\partial x} - S\frac{\partial H+U}{\partial x} + g\beta \\ -1 \end{pmatrix} \geq 0. \quad (2.15)$$

Suppose moreover that (2.15) holds, that  $H+U$  is positive definite, and that the system is zero-state detectable (see Definition 2.9 below). Then the feedback  $\bar{u} = C(\bar{x},t)\bar{y}$  with  $C(x,t) \geq \epsilon I > 0 \in \mathbb{R}^{m \times m}$  renders the system asymptotically stable.

**Definition 2.9** (Zero-state detectability [98]). System (2.5) is zero-state detectable if  $u(t) = 0, y(t) = 0$  for all  $t \geq 0$ , implies  $\lim_{t \rightarrow \infty} x(t) = 0$ .

### 2.3.4 Port-Hamiltonian systems on graphs

The theory of *port-Hamiltonian systems on graphs* was recently introduced in [100] to model complex systems composed of many subsystems using graphs. Port-Hamiltonian dynamics are assigned to both the nodes and the edges of the graph. Let  $\mathcal{V}$  and  $\mathcal{E}$  denote respectively the node-set and edge-set defined of a graph  $\mathcal{G}$  (see Section 2.2) and define the node space  $\Lambda_0$  as all functions  $\mathcal{V} \rightarrow \mathcal{R}$ , with dual space  $\Lambda^0$ . Furthermore define the edge space  $\Lambda_1$  as all function  $\mathcal{E} \rightarrow \mathcal{R}$ , with dual space  $\Lambda^1$ . The *incidence operator*  $\hat{B}$  is defined as  $\hat{B} : \Lambda_1 \rightarrow \Lambda_0$ , with corresponding *coincidence operator*  $\hat{B}^* : \Lambda^0 \rightarrow \Lambda^1$ . If  $\mathcal{R} = \mathbb{R}^n$ , with  $n = 1, 2, \dots$ , the incidence and coincidence operator simplify to the linear map  $\hat{B} = B \otimes I_n$  and  $\hat{B}^* = B^T \otimes I_n$  respectively.

Each node of the graph has a corresponding flow  $f_0 \in \Lambda_0$  and effort  $e^0 \in \Lambda^0$ , while each edge has a flow  $f_1 \in \Lambda_1$  and effort  $e^1 \in \Lambda^1$ . Some of the nodes are open to interconnection with an external system. These so-called *boundary nodes* are obtained by identifying a subset  $\mathcal{V}_b \subset \mathcal{V}$  with  $N_b$  boundary nodes. The remaining  $N_i = N - N_b$  nodes in the subset  $\mathcal{V}_i := \mathcal{V} - \mathcal{V}_b$  are called the *internal nodes* of the graph. The splitting of the nodes into internal and boundary nodes induces a splitting of the node space  $\Lambda_0$  and its dual  $\Lambda^0$ , given by

$$\begin{aligned} \Lambda_0 &= \Lambda_{0i} \oplus \Lambda_{0b}, \\ \Lambda^0 &= \Lambda^{0i} \oplus \Lambda^{0b}, \end{aligned}$$

where  $\Lambda_{0b}$  ( $\Lambda_{0i}$ ) is the node space corresponding to the boundary (internal) nodes. Consequently the incidence matrix  $B$  splits into

$$B = B_i \oplus B_b,$$

where  $B_i : \Lambda_1 \rightarrow \Lambda_{0i}$  and  $B_b : \Lambda_1 \rightarrow \Lambda_{0b}$  (i.e.,  $B_i$  describes which edges are incident to internal nodes, while  $B_b$  describes which edges are incident to boundary nodes).

For  $\mathcal{R} = \mathbb{R}^n$ , with  $n = 1, 2, \dots$ , the splitting of the incidence matrix simplifies to

$$B = \begin{pmatrix} B_i \\ B_b \end{pmatrix}.$$

Finally, define the boundary space  $\Lambda_b$  and dual space  $\Lambda^b$  corresponding to the external port of the boundary nodes.

There is a distinction between two types of port-Hamiltonian systems on graphs: so-called *effort-continuous graphs* and *flow-continuous graphs*. The corresponding Dirac structures (see Definition 2.5) are defined below.

**Definition 2.10** (Effort- and flow-continuous Dirac structures [100]). Consider an open graph  $\mathcal{G}$  with node, edge, and boundary spaces, incidence matrix  $B$ , which splits into  $B = B_i \oplus B_b$ . The effort-continuous graph Dirac structure  $\mathcal{D}_e(\mathcal{G})$  is defined as

$$\begin{aligned} \mathcal{D}_e(\mathcal{G}) &:= \{(f_1, e^1, f_0, e^0, f_b, e^b) \\ &\in \Lambda_1 \times \Lambda^1 \times \Lambda_0 \times \Lambda^0 \times \Lambda_b \times \Lambda^b \mid (B_i \otimes I_n)f_1 = f_{0i}, \\ &(B_b \otimes I_n)f_1 = f_{0b} + f_b, e^1 = -(B^T \otimes I_n)e^0, e^b = e^{0b}\}. \end{aligned} \quad (2.16)$$

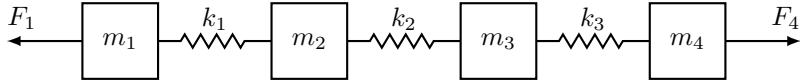
The flow-continuous graph Dirac structure  $\mathcal{D}_f(\mathcal{G})$  is defined as

$$\begin{aligned} \mathcal{D}_f(\mathcal{G}) &:= \{(f_1, e^1, f_{0i}, e^{0i}, f_b, e^b) \\ &\in \Lambda_1 \times \Lambda^1 \times \Lambda_{0i} \times \Lambda^{0i} \times \Lambda_b \times \Lambda^b \mid (B_i \otimes I_n)f_1 = f_{0i}, \\ &(B_i \otimes I_n)f_1 = f_{0i}, f_1 = f_{0i}, (B_b \otimes I_n)f_1 = f_{0i}, f_1 = f_b, \\ &e^1 = -(B_i^T \otimes I_n)e^{0i} - (B_b^T \otimes I_n)e^b\}. \end{aligned} \quad (2.17)$$

For mechanical systems an effort-continuous graph considers to the case where boundary nodes correspond to boundary masses, while for flow-continuous graphs, boundary nodes are massless.

**Example 2.3** (Four masses connected by three springs). Consider a mass-spring system with four masses interconnected by three springs (see Figure 2.4). The corresponding *path graph* has four nodes and three edges. The nodes correspond to the masses, while the edges correspond to the springs. The two outer masses  $m_1, m_4$  are affected by external forces  $F_1, F_4$  and are therefore considered as boundary nodes, while mass  $m_2$  and  $m_3$  are internal nodes. The corresponding incidence matrix  $B$  is given by

$$B = \begin{pmatrix} B_i \\ B_b \end{pmatrix} = \begin{pmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$



**Figure 2.4:** Four masses connected by three springs system of Example 2.3.

In this example, the boundary nodes are boundary masses and therefore the corresponding Dirac structure is effort continuous. The flows in this example are the spring forces  $f_1$ , resulting forces on the internal masses  $f_{0i}$ , resulting forces on the boundary masses  $f_{0b}$ , and external forces  $f_b$ . Correspondingly, the efforts are the elongation velocities of the springs  $e^1$ , velocities of the internal masses  $e^{0i}$ , velocities of the boundary masses  $e^{0b}$ , and boundary velocities  $e^b$ .

Assume for simplicity that  $n = 1$ , then the force and velocity balance follow directly from (2.16) and are given by

$$\begin{aligned} \begin{pmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \end{pmatrix} \begin{pmatrix} F_1^s \\ F_2^s \\ F_3^s \end{pmatrix} &= \begin{pmatrix} F_2^m \\ F_3^m \end{pmatrix}, \\ \begin{pmatrix} -1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} F_1^s \\ F_2^s \\ F_3^s \end{pmatrix} &= \begin{pmatrix} F_1^m \\ F_4^m \end{pmatrix} + \begin{pmatrix} F_1 \\ F_4 \end{pmatrix}, \\ \begin{pmatrix} 1 & 0 & -1 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & -1 & 0 & 1 \end{pmatrix} \begin{pmatrix} v_1^s \\ v_2^s \\ v_3^s \end{pmatrix} &= \begin{pmatrix} v_1^m \\ v_2^m \\ v_3^m \end{pmatrix}, \quad \begin{pmatrix} v_1 \\ v_4 \end{pmatrix} = \begin{pmatrix} v_1^m \\ v_4^m \end{pmatrix}, \end{aligned}$$

where  $F_i^m$ ,  $F_i^s$  denote respectively the resulting force on mass  $m_i$  and the spring force of spring  $k_i$  and  $v_i^m$ ,  $v_i^s$  denote the corresponding velocities.

## 2.4 Non-smooth analysis

This section presents several tools from non-smooth analysis, which are used in Chapter 3 for analyzing systems with discontinuities. Consider the following (possibly discontinuous) differential equation

$$\dot{x} = f(t, x). \quad (2.18)$$

To analyze (2.18) there are several notions of solutions available. In Chapter 3 the *Krasovskii notion of solution* is adopted. Another notion of solution common in literature is the Filippov solution [6]. Every Filippov solution is also a Krasovskii

solution, implying that all results in Chapter 3 also apply to Filippov solutions [42].

**Definition 2.11** (Krasovskii solution [42]). Given  $f : \mathbb{R}_+ \times \mathbb{R}^n \rightarrow \mathbb{R}^n$ , we say that  $x : \mathcal{T} \rightarrow \mathbb{R}^n$  is a solution to (2.18) in the Krasovskii sense if  $x(\cdot)$  is absolutely continuous and for almost every time  $t$  in the interval  $\mathcal{T} \subset \mathbb{R}_+$  satisfies the differential inclusion  $\dot{x}(t) \in \mathcal{K}f(x(t))$ , where

$$\mathcal{K}f(t, x) = \bigcap_{\delta > 0} \overline{\text{co}} f(t, B(x, \delta)),$$

with  $\overline{\text{co}}$  denoting the convex closure and  $B(x, \delta)$  the disc with radius  $\delta$  centered at  $x$ .

Here and elsewhere in this thesis, “almost every” means “except for a set of zero Lebesgue measure” [42]. In the remainder the Krasovskii operator  $\mathcal{K}$  is only applied to time-invariant systems of the form  $\dot{x} = f(x)$ .

Now that a notion of solution is defined, one would like to assess whether the non-smooth system is stable. Consider the following *differential inclusion*

$$\dot{x} \in F(x), \quad (2.19)$$

where  $F(x)$  is a set-valued map. The stability of (2.19) can be assessed using a non-smooth version of LaSalle’s invariance principle (Theorem 2.4). But first the notion of a *set-valued derivative* and the *generalized Clarke gradient* are required. The set-valued derivative of the locally Lipschitz function  $V(x) : \mathbb{R}^n \rightarrow \mathbb{R}$  with respect to differential inclusion (2.19) is given by [6]

$$\dot{\bar{V}}(x) = \{a \in \mathbb{R} \mid \exists v \in F(x) \text{ such that } p \cdot v = a \text{ for all } p \in \partial V(x)\}, \quad (2.20)$$

where  $\partial V(x)$  denotes the generalized Clarke gradient. This non-smooth notion of a gradient is defined as [6]

$$\partial V(x) = \text{co} \left\{ \lim_{i \rightarrow +\infty} \nabla V(x_i) \mid x_i \rightarrow x, x_i \notin \mathcal{S}, x_i \notin \mathcal{N} \right\}, \quad (2.21)$$

where  $\mathcal{N}$  is the set of measure zero where the gradient of  $V$  does not exist,  $\mathcal{S}$  is any set of zero measure in  $\mathbb{R}^n$ , and  $\text{co}\{\cdot\}$  denotes the convex hull. Before stating a non-smooth version of LaSalle’s invariance principle Theorem 2.4, the notion of a weakly invariant set and a regular function are defined below.

**Definition 2.12** (Weakly invariant set [24]). A set  $\mathcal{M}$  is weakly invariant for (2.18) if for each  $x_0 \in \mathcal{M}$ ,  $\mathcal{M}$  contains a maximal solution of (2.18). A maximal solution is a Krasovskii solution (see (2.11)) whose domain of existence is maximal, i.e., cannot be extended any further.

**Definition 2.13** (Regular function [75]). A function  $V$  is called regular if and only if it is differentiable at every point and single-valued throughout a region  $\mathcal{R}$ .

Now a non-smooth version of LaSalle's invariance principle is given as follows.

**Theorem 2.14** (Non-smooth LaSalle's invariance principle [24]). *Let  $V : \mathbb{R}^n \rightarrow \mathbb{R}$  be a locally Lipschitz and regular function. Let  $x_0 \in \mathcal{S} \subset \mathbb{R}^n$ , with  $\mathcal{S}$  compact and strongly invariant for (2.18). Assume that either  $\max \dot{\bar{V}}(x) \leq 0$  or  $\dot{\bar{V}}(x) = \emptyset$  for all  $x \in \mathcal{S}$ . Let  $\mathcal{Z}_{f,V} = \{x \in \mathbb{R}^n \mid 0 \in \dot{\bar{V}}(x)\}$ . Then, any solution  $x : [t_0, +\infty) \rightarrow \mathbb{R}^n$  of (2.18) starting from  $x_0$  at  $t_0$  converges to the largest weakly invariant set  $\mathcal{M}$  (see Definition 2.12) contained in  $\bar{\mathcal{Z}}_{f,V} \cap \mathcal{S}$ . Moreover, if the set  $\mathcal{M}$  is a finite collection of points, then the limit of all solutions starting at  $x_0$  exists and equals one of them.*

## 2.5 Concluding remarks

This chapter presents the preliminaries on stability of time-invariant systems, graph theory, port-Hamiltonian systems and non-smooth analysis. LaSalle's invariance principle is invoked throughout the subsequent chapters to prove converge to desired formations (Sections 3.5, 4.3, 4.4, 4.5, and 5.3), while for the discontinuous systems in Section 3.4 a non-smooth version of LaSalle's invariance principle applies.

Graph theory is used to model the interconnection topology of the networks investigated. The nodes of the graphs correspond to the systems of interest (i.e., fully actuated agents (Chapter 3), wheeled robots (Chapter 4), or satellites (Chapter 5)), while the edges correspond to virtual couplings which are designed to achieve the formation control objectives. The matrices associated to graphs are used to characterize the invariant sets when invoking LaSalle's invariance principle.

All systems in this thesis are modeled as input-state-output port-Hamiltonian systems in the mechanical domain. The theory of mechanical port-Hamiltonian systems with constraints is used to derive dynamics on constrained state spaces for the problem of deployment (Section 3.5) and the dynamical modeling of the wheeled robot (Section 4.2). Generalized canonical transformations are used to derive the error dynamics with respect to reference velocities (Section 4.4) and circular target orbits (Section 5.2).

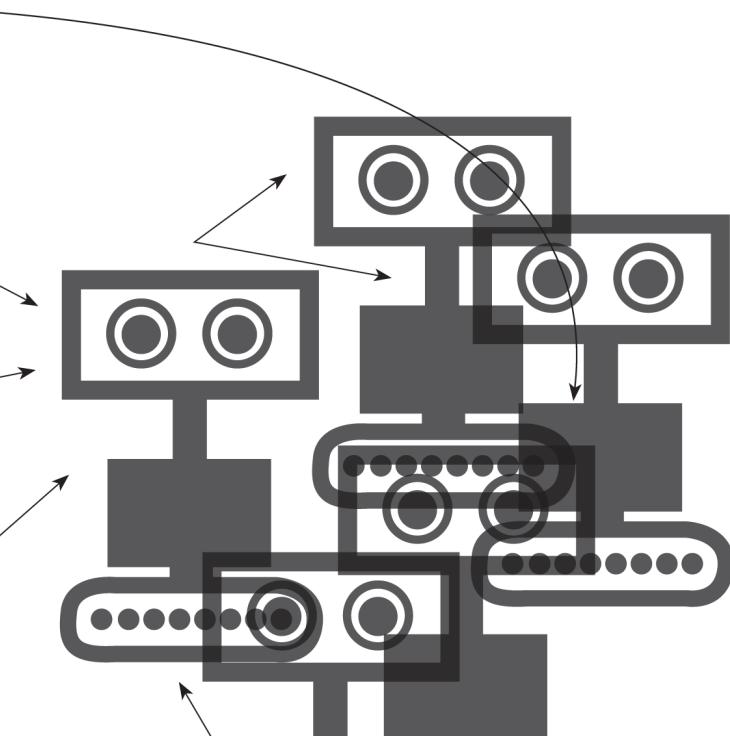
The subsequent three chapters present a port-Hamiltonian approach to formation control and related control problems for three types of mechanical systems. Chapter 3 deals with fully actuated systems and considers the problems of formation control, formation control in the presence of Coulomb friction, and deployment. Chapter 4 continues with the problems of formation control, velocity tracking, and matched input disturbance rejection for nonholonomic wheeled robots. Finally, Chapter 5 studies the problem of orbital phasing of satellites on circular orbits.





# Chapter 3

Formation control of fully actuated systems





# Chapter 3

---

## Formation control of fully actuated systems

This chapter deals with three formation control (related) problems for a network of agents which are *fully actuated*. Fully actuated agents are agents, for which the number of inputs equals the degrees of freedom. The chapter starts with a background on the three problems considered here, respectively formation control, formation control in the presence of Coulomb friction, and deployment (Section 3.1).

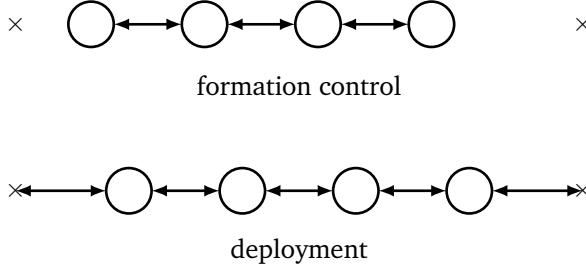
Section 3.2 continues with the agents dynamics, followed by a new interpretation of [100] in the context of formation control in Section 3.3. Section 3.4 extends the results in Section 3.3 for a network of agents subject to Coulomb friction, which render the agent dynamics non-smooth. Another extension is presented in Section 3.5, which deals with the deployment problem. The highlights and concluding remarks of the chapter are given in Section 3.6.

The results in this chapter are published in [58, 59, 110]. The results in Section 3.4 are based on a collaboration with Matin Jafarian and Claudio De Persis.

### 3.1 Introduction

Formation control aims to achieve a prescribed geometrical shape for a network of agents using only local feedback controllers [3], as was already stated in Section 1.1.2. The controllers in this chapter are all based on assigning virtual couplings between the agents and they provide a new interpretation of the theory of port-Hamiltonian systems on graphs [100]. Each virtual coupling consists of a virtual spring with an optional virtual damper in parallel. The springs shape the energy of the network to achieve formation control, while the dampers inject damping to guarantee stability. The virtual couplings correspond to the external feedback of the passivity-based approach for group coordination in [3, 9].

The agent dynamics play an important role in the problem of formation control. In this regard, different classes of dynamic agents have been considered e.g. [9, 81, 93, 100]. For dynamic agents, the dissipation due to friction forces plays an important role in stability analysis of the whole network e.g. [9, 57, 110]. To the author's best knowledge, only continuous friction forces have been considered in the context of formation control. A physical phenomenon as Coulomb friction is therefore neglected, due to its discontinuous nature.



**Figure 3.1:** Illustration of the difference between formation control and deployment: Formation control only considers the inter-agent displacements (top), while deployment considers inter-agent displacements with respect to reference points (bottom).

Coulomb friction is a quantification of the friction force that exists between two (dry) surfaces in contact with each other. The set-valued Coulomb friction model considered in this chapter is a simplification of reality in the sense that each friction surface is considered to be either in a sticking phase or in a slipping phase. Transitions between the two phases are modeled to be instantaneous. The concept of *stiction* is properly modeled, since the friction force is allowed to be non-zero at zero relative velocity [118].

Since Coulomb friction is a discontinuous friction law it renders the networked system non-smooth. The analysis of such non-smooth systems requires tools from non-smooth systems theory (Section 2.4). Non-smooth stability theory has been considered in formation keeping control before, e.g. in finite-time consensus algorithms [24], and in quantized coordination [18, 28, 57]. However, none of the non-smooth tools applied in Section 3.4 have been considered in the port-Hamiltonian framework before. The only effort considering non-smooth Hamiltonians for the control of nonholonomic systems [46] does not use any of these tools.

Another problem which is closely related to formation control, is the problem of deployment. The difference between the two is illustrated in Figure 3.1. Formation control only considers the relative displacements between the agents (Figure 3.1 (top)), while deployment also considers the relative displacements with respect to prescribed reference points (Figure 3.1 (bottom)).

Deployment is a crucial problem in the context of robotic sensor networks [15, 25, 77]. Robotic sensor networks use a large number of relatively simple robots to obtain high quality measurements over a (possibly) large area. Deployment aims to spatially distribute the robotic sensors in some optimal sense in the environment of interest [15]. The problem of deployment is thereby closely related to the *coverage* problem, where a network of sensors needs to cover an environment of interest [25, 82].

The interconnection topology in this chapter (i.e., the way in which virtual

couplings are assigned between the agents) is modeled by a graph. Here, all graphs are assumed to be *tree graphs* (i.e., graphs which are undirected, connected and acyclic), except for Section 3.5. The way in which the couplings are defined complies with the position-based formation control setting described in [3, 9].

The contribution of this chapter is summarized as follows. Section 3.3 provides a new interpretation of [100] in the context of formation control and provides new simulation and experimental results to illustrate the effectiveness of the approach. Section 3.4 extends the result of Section 3.3 by providing rigorous stability analyses using tools from non-smooth systems theory for network of agents subject to Coulomb friction. The use of these non-smooth tools is completely new in the port-Hamiltonian framework. It is shown that standard continuous virtual springs can not achieve exact formations, while their discontinuous counterparts can under a natural condition on their controller gain. Finally, Section 3.5 extends the formation control algorithms in Section 3.3 to the deployment problem by introducing the concept of virtual walls. The port-Hamiltonian framework provides a clear physical interpretation of the models and algorithms. Furthermore, applying virtual springs and dampers as control systems fits naturally into this energy-based framework.

### Utilization within the ROSE project

The legged robotic sensor for dike inspection developed by the ROSE project (Section 1.2.2) is able to freely move along all directions (i.e., forward, backward, sideward). Therefore, fully actuated systems provide an appropriate representation of the legged robot on a high abstraction level. Often locomotion along some directions (forward/backward) is more energy-efficient than others (sideway). This could be incorporated into the models by adjusting the friction coefficients accordingly. Another way is to impose a constraint on the sideway motion, which is considered in the next Chapter 4. The fully actuated model is used to develop formation control algorithms to position the robots relative to each other, while internal control systems (e.g. locomotion control) are omitted in the model.

The next section provides the dynamical modeling of fully actuated systems in the port-Hamiltonian framework.

## 3.2 Dynamical model

The systems considered in this chapter are fully actuated, which means that the number of inputs equals the degrees of freedom of the system. The system of interest is a group (or network) of  $N$  agents, where each agent  $i$  is modeled as a single point mass  $m_i$  moving in  $\mathbb{R}^n$ . Since each agent is modeled as a point mass, the heading is omitted here. In Chapter 4, where agents are modeled as rigid

bodies, the heading is taken into account. The position of agents  $i$  is denoted by  $q_i \in \mathbb{R}^n$  and the corresponding momentum is defined as  $p_i = m_i \dot{q}_i \in \mathbb{R}^n$ . Each agent has a control port  $(u_i, y_i)$  with input  $u_i \in \mathbb{R}^n$  and output  $y_i \in \mathbb{R}^n$ , and resistive port  $(u_i^r, y_i^r)$  with input  $u_i^r \in \mathbb{R}^n$  and output  $y_i^r \in \mathbb{R}^n$ . The dynamics are of the form (2.8) and are given by

$$\begin{aligned} \begin{pmatrix} \dot{q}_i \\ \dot{p}_i \end{pmatrix} &= \begin{pmatrix} 0 & I_n \\ -I_n & -D_i^a(q_i, p_i) \end{pmatrix} \begin{pmatrix} \frac{\partial H_i^a}{\partial q_i}(q_i, p_i) \\ \frac{\partial H_i^a}{\partial p_i}(q_i, p_i) \end{pmatrix} + \begin{pmatrix} 0 \\ I_n \end{pmatrix} u_i + \begin{pmatrix} 0 \\ I_n \end{pmatrix} u_i^r, \\ y_i = y_i^r &= \frac{\partial H_i^a}{\partial p_i}(p_i), \end{aligned} \quad (3.1)$$

with positive semi-definite dissipation matrix  $D_i^a(q_i, p_i) \in \mathbb{R}^{n \times n}$  and Hamiltonian  $H_i^a(q_i, p_i)$ . The Hamiltonian equals the kinetic energy associated to the movement of the mass and is given by

$$H_i^a(p_i) = \frac{1}{2} p_i^T M_i^{-1} p_i,$$

where  $M_i = m_i I_n$ . The dissipation matrix  $D_i^a(q_i, p_i)$  enables the modeling of different types of friction like *linear (viscous) friction* ( $D_i^a(q_i, p_i) = D_i^a$ ) and *nonlinear approximated Coulomb friction* [51].

Due to discontinuity, ideal Coulomb friction can not be modeled using the dissipation matrix  $D_i^a$  and therefore instead a resistive port  $(u_i^r, y_i^r)$  terminated by the ideal Coulomb friction characteristic added (see Section 3.4 for more details). Recall from Section 2.3 that the product of the port variables equals the power supplied to the system. Hence, for the resistive port it holds that  $(y_i^r)^T u_i^r \leq 0$  since resistive elements dissipate energy.

Now consider a group of  $N$  agents of the form (3.1). To compactly denote the agent dynamics denote the collocated vectors  $q = (q_1, \dots, q_N)^T$ ,  $p = (p_1, \dots, p_N)^T$ ,  $u = (u_1, \dots, u_N)^T$ ,  $u^r = (u_1^r, \dots, u_N^r)^T$ ,  $y = (y_1, \dots, y_N)^T$ ,  $y^r = (y_1^r, \dots, y_N^r)^T$  and system matrices  $M = \text{block.diag}(M_1, \dots, M_N)$ ,  $D^a(q, p) = \text{block.diag}(D_1^a(q_1, p_1), \dots, D_N^a(q_N, p_N))$ . Then the dynamics for  $N$  agents of the form (3.1) can be written as

$$\begin{aligned} \begin{pmatrix} \dot{q} \\ \dot{p} \end{pmatrix} &= \begin{pmatrix} 0 & I_{Nn} \\ -I_{Nn} & -D^a(q, p) \end{pmatrix} \begin{pmatrix} \frac{\partial H^a}{\partial q}(p) \\ \frac{\partial H^a}{\partial p}(p) \end{pmatrix} + \begin{pmatrix} 0 \\ I_{Nn} \end{pmatrix} u + \begin{pmatrix} 0 \\ I_{Nn} \end{pmatrix} u^r, \\ y = y^r &= \frac{\partial H^a}{\partial p}(p), \end{aligned} \quad (3.2)$$

with Hamiltonian  $H^a(p) = \sum_{i=1}^N H_i^a(q_i, p_i) = \frac{1}{2} p^T M^{-1} p$ .

The subsequent sections continue with the controller design and analysis for formation control (Section 3.3), formation control in the presence of Coulomb

friction (Section 3.4) and deployment (Section 3.5).

### 3.3 Formation control

This section provides a new interpretation of [100] in the context of formation control of a network of fully actuated systems. The setup described here closely resembles the mass-spring-damper systems considered in Section 3.2 in [100]. The masses correspond to the fully actuated agents, while the springs and dampers correspond to the virtual couplings. The way in which the virtual couplings are defined fits within the position-based formation control design in [3, 9]. Based on the ideas in this section, Section 3.4 and 3.5 elaborate on the formation control problem in the presence of Coulomb friction and the problem of deployment respectively.

Consider a network of  $N$  agents of the form (3.2), where the resistive port  $(u^r, y^r)$  is omitted for simplicity here. Formation control is achieved by assigning virtual couplings in between the agents and the interconnection topology amongst agents via virtual couplings is modeled by a tree graph (see Section 2.2). The  $N$  nodes of the graph correspond to the agents, while the  $E$  edges correspond to the virtual couplings.

For each agent let  $q_i \in \mathbb{R}^n$  denote its position and  $p_i \in \mathbb{R}^n$  the corresponding momentum. Let  $z_j \in \mathbb{R}^n$  denote the relative displacement for two agents interconnected by virtual coupling  $j$  and let  $z_j^* \in \mathbb{R}^n$  denote the desired relative displacement. For  $N$  agents and  $E$  virtual couplings let  $q = (q_1, \dots, q_N)^T$ ,  $p = (p_1, \dots, p_N)^T$ ,  $z = (z_1, \dots, z_E)^T$ ,  $z^* = (z_1^*, \dots, z_E^*)^T$ , then the *formation control* objective can be formally stated as

$$\begin{cases} p \rightarrow 0, \\ z \rightarrow z^*, \end{cases} \quad \text{as } t \rightarrow \infty. \quad (3.3)$$

The second objective is the formation control objective which ensures that all relative displacements  $z$  converge to the desired relative displacements  $z^*$ . The first objective ensures that all agents come to a hold and thereby provides stability for the network as a whole (i.e., the network does not drift). Note that achieving the first objective is not necessary to achieve the second one in (3.3).

The first objective also corresponds to setting  $v(t) = 0$  in the internal feedback in [3]. In contrast, Sections 4.4 and 5.3 provide control designs to track a nonzero reference velocity (see objectives (4.20) and (5.10)).

### 3.3.1 Formation control using virtual couplings

Formation control objectives (3.3) are achieved by assigning virtual couplings in between the agents according to a tree graph. Each virtual coupling consists of a virtual spring and damper in parallel, with spring elongation  $z_j \in \mathbb{R}^n$ , input velocity  $w_j \in \mathbb{R}^n$  and corresponding output force  $\tau_j \in \mathbb{R}^n$ . The dynamics of such a spring-damper system are well-known [39, 99] and are given by

$$\begin{aligned}\dot{z}_j &= w_j, \\ \tau_j &= \frac{\partial H_j^c}{\partial z_j} + D_j^c w_j,\end{aligned}\tag{3.4}$$

with positive semi-definite dissipation matrix  $D_j^c \in \mathbb{R}^{n \times n}$  and Hamiltonian  $H_j^c(z_j)$ . For each virtual coupling  $j$ , the Hamiltonian  $H_j^c$  equals the potential energy in the spring  $j$  and is given by

$$H_j^c(z_j) = \frac{1}{2}(z_j - z_j^*)^T K_j^c(z_j - z_j^*),$$

with rest length  $z_j^*$  and diagonal positive definite spring constant matrix  $K_j^c \in \mathbb{R}^{n \times n}$ .

Now consider  $E$  virtual couplings of the form (3.4). To compactly denote the dynamics, denote the collocated vectors  $z = (z_1, \dots, z_E)^T$ ,  $z^* = (z_1^*, \dots, z_E^*)^T$ ,  $w = (w_1, \dots, w_E)^T$ ,  $\tau = (\tau_1, \dots, \tau_E)^T$  and system matrices  $D^c = \text{block.diag}(D_1^c, \dots, D_E^c)$ ,  $K^c = \text{block.diag}(K_1^c, \dots, K_E^c)$ . Then the dynamics of  $E$  virtual couplings are summarized as

$$\begin{aligned}\dot{z} &= w, \\ \tau &= \frac{\partial H^c}{\partial z} + D^c w,\end{aligned}\tag{3.5}$$

with Hamiltonian  $H^c(z) = \sum_{j=1}^E H_j^c(z_j) = \frac{1}{2}(z - z^*)^T K^c(z - z^*)$ . As said, the interconnection topology is modeled by a tree graph  $\mathcal{G}$ . Let  $B$  denote the incidence matrix associated to  $\mathcal{G}$ , then the coupling of agents on the nodes and virtual couplings at the edges is given by [3, 100]

$$\begin{aligned}u &= -(B \otimes I_n)\tau, \\ w &= (B^T \otimes I_n)y.\end{aligned}\tag{3.6}$$

*Remark 3.1* (External control port). Coupling (3.6) results in an autonomous closed-loop system (see next section). An external control port may be added by labeling some of the nodes as *boundary nodes* (see Section 2.3.4). For simplicity this external port is omitted here. The interested reader is referred to [100, 110] for more details.

The closed-loop dynamics follow directly from (3.2), (3.5), (3.6). The closed-

loop dynamics are decoupled along the dimensions of the state space. Therefore the remainder continues with  $n = 1$  without loss of generality and the closed-loop dynamics are given by

$$\begin{pmatrix} \dot{p} \\ \dot{z} \end{pmatrix} = \begin{pmatrix} -(D^a(p) + B D^c B^T) & -B \\ B^T & 0 \end{pmatrix} \begin{pmatrix} \frac{\partial H}{\partial p} \\ \frac{\partial H}{\partial z} \end{pmatrix}, \quad (3.7)$$

with Hamiltonian  $H(z, p) = \sum_{i=0}^N H_i^a(p_i) + \sum_{j=1}^E H_j^c(z) = \frac{1}{2}p^T M^{-1}p + \frac{1}{2}(z - z^*)^T K^c(z - z^*)$ . The following proposition shows that interconnecting the agents (3.2) using virtual couplings (3.5) via coupling (3.6) achieves the formation control objectives (3.3).

**Proposition 3.2.** *Interconnect agents (3.2) with virtual couplings (3.5) via interconnection constraint (3.6) using a tree graph topology. Then, the solutions of the closed-loop system (3.7) converge to  $p = 0, z = z^*$ , thereby achieving the formation control objectives (3.3).*

*Proof.* Take  $H(z, p)$  as Lyapunov function candidate. It is easily verified that  $H \geq 0$  and the time derivative is given by

$$\dot{H} = -\frac{\partial^T H}{\partial p} (D^a(p) + B D^c B^T) \frac{\partial H}{\partial p} \leq 0.$$

Invoking LaSalle gives that the solutions to (3.7) converge to the largest invariant set where  $p = 0$ . Substituting  $p = 0, \dot{p} = 0$  into (3.7) gives  $-B \frac{\partial H}{\partial z} = -B K^c(z - z^*) = 0$ . Since  $B$  is the incidence matrix associated to an acyclic graph,  $\ker B = 0$  and hence  $z = z^*$ , thereby completing the proof.  $\square$

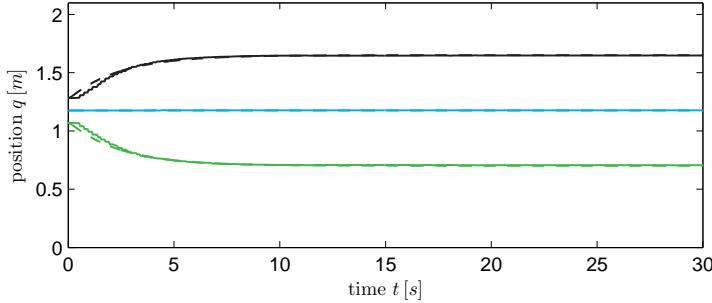
*Remark 3.3* (Control input formation control). The control input for the agents  $u$  follows directly from (3.6) and is given by

$$u = -\underbrace{(B \otimes I_n) K^c(z - z^*)}_{\text{virtual spring force}} - \underbrace{(B \otimes I_n) D^c (B^T \otimes I_n) M^{-1}p}_{\text{virtual damping force}}. \quad (3.8)$$

Control law (3.8) has a clear physical interpretation. The virtual spring force ensures that the formation control objectives (3.3) are achieved, while the virtual damping force can be used to shape the transient response.

Furthermore, note that (3.8) is a distributed control law: Each agent only requires measurements on the relative displacement  $z$  and relative velocity  $\dot{z}$  with respect to its two neighbors. Note that all agents do require knowledge of a global coordinate frame, which can be embedded by a simple compass.

To illustrate Proposition 3.2, new simulation and experimental results are presented in the subsequent section.



**Figure 3.2:** Time evolution of the position during formation control of the e-puck wheeled robots for the simulation (dashed) and experiment (solid).

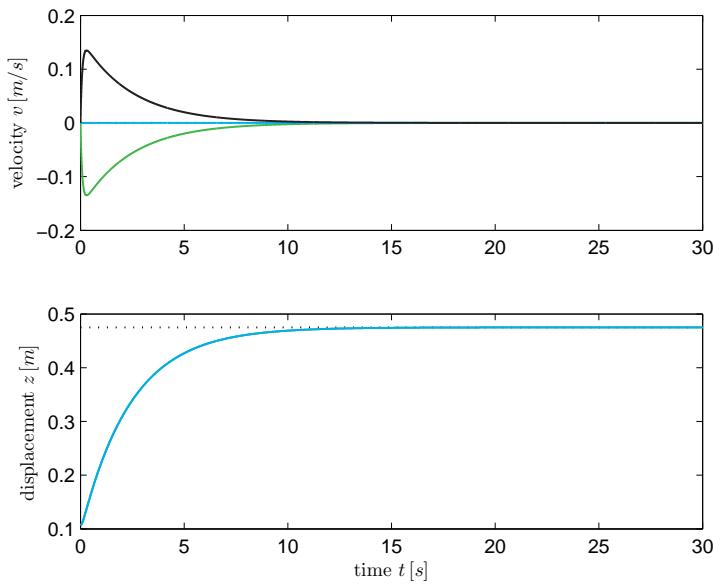
### 3.3.2 Simulation and experimental results

Consider a network with  $N = 3$  agents (3.2) moving in  $\mathbb{R}$ . Each agent  $i$  has a mass  $m_i = 0.167\text{ kg}$  and a friction coefficient  $D_i^a = 2\text{ kg/s}$  for  $i = 1, 2, 3$ . The agents are interconnected using  $E = 2$  homogenous virtual couplings (3.5), with spring constant  $K_j^c = 0.5\text{ kg/s}^2$ , nominal spring length  $z_j^* = 0.475\text{ m}$  and damping coefficient  $D_j^c = 0$ . The nominal spring constant corresponds to the same value as the equilibrium position for the deployment controller presented in Section 3.5.

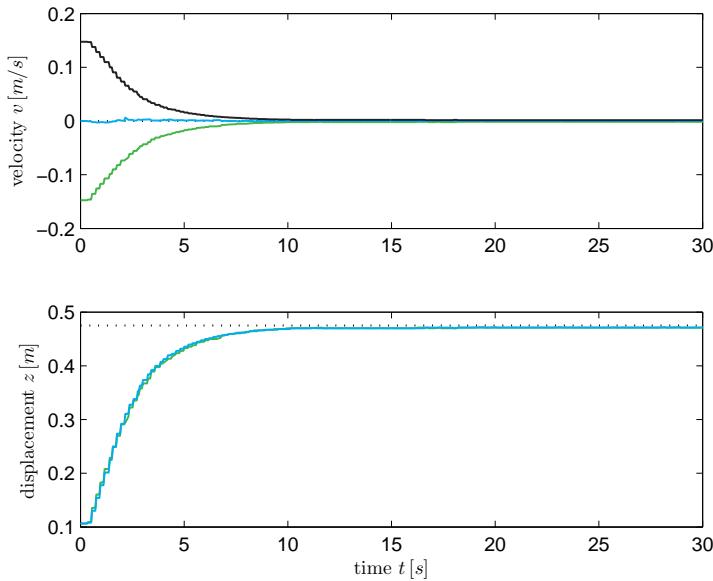
The simulations are run using MATLAB and Simulink. The experimental setup consist of three e-puck wheeled robots, which can only move forward and backward in accordance with the movement in  $\mathbb{R}$ . By constraining the movement of the e-pucks, they act as fully actuated agents along the forward/backward direction since the constraint on the wheel axle is not interfering. More details on the e-puck robot and the experimental setup are given in respectively Sections C.1 and C.2. Both the simulation and the experiment are run for  $t = 30\text{ s}$  and the initial conditions are set at  $q(0) = (1.07, 1.18, 1.28)\text{ m}$ ,  $p(0) = (0, 0, 0)\text{ kg m/s}$ . The results are shown in Figures 3.2, 3.3, and 3.4.

Figure 3.2 show the position of the agents during the simulation and the experiment. The (small) differences are due to localization errors of the vision algorithm, model parameter uncertainty, and a difference in the actuators of the model and the e-puck (the robot dynamics (3.1) do not consider the differential drive of the e-puck). To clarify that the formation control objectives (3.3) are indeed achieved, Figures 3.3 and 3.4 show the time evolution of the velocity  $v$  and relative displacement  $z$  respectively. Both figures show a similar trend and illustrate that  $v \rightarrow 0, z \rightarrow z^*$  as  $t \rightarrow \infty$ .

The next section considers agents of the form (3.2) that are subject to a discontinuous Coulomb friction force, which requires a discontinuous counterpart to the virtual couplings presented in this section.



**Figure 3.3:** Time evolution of the velocity  $v$  and relative displacement  $z$  during formation control (simulation). The dotted lines show the reference values.



**Figure 3.4:** Time evolution of the velocity  $v$  and relative displacement  $z$  during formation control (experiment). The dotted lines show the reference values.

## 3.4 Formation control in the presence of Coulomb friction

The previous Section 3.3 considered formation control of a network of fully actuated agents in  $\mathbb{R}^n$ . This section considers a similar problem setting, but now each agent is subject to (ideal) Coulomb friction. The control objectives remain the same as in (3.3), i.e.,

$$\begin{cases} p \rightarrow 0, \\ z \rightarrow z^*, \end{cases} \quad \text{as } t \rightarrow \infty, \quad (3.9)$$

with agent momentum  $p$ , relative displacement  $z$  and desired relative displacement  $z^*$ . Due to the discontinuity of the friction force, continuous virtual springs like (3.4) are no longer able to achieve exact formations as shown in Section 3.4.2 below. Section 3.4.3 presents a discontinuous counterpart to (3.4), which is able to achieve the formation control objectives (3.9) in the presence of Coulomb friction.

The results of this section are published in [58, 59] and are based on a collaboration with Matin Jafarian and Claudio De Persis. But first, a motivating example provides some intuition on the differences between continuous and discontinuous virtual springs.

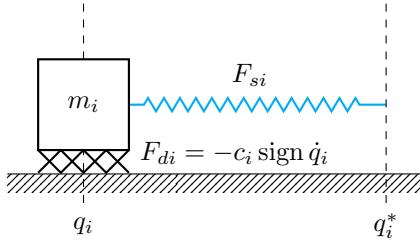
**Example 3.1** (Motivation). Consider a single mass  $m_i$  moving in  $\mathbb{R}$  with position  $q_i \in \mathbb{R}$  (see Figure 3.5). The mass is subject to a Coulomb friction force  $F_{di} = -c_i \operatorname{sign}(q_i)$  [119], with friction coefficient  $c_i$  and velocity  $\dot{q}_i \in \mathbb{R}$ . Assume that the control objective is to move  $m_i$  to a prescribed position  $q_i^*$ . We achieve this by assigning a virtual spring to  $m_i$  with corresponding spring force  $F_{si}$ . We consider two types of springs: a continuous spring where  $F_{si}(q_i) = k_i q_i$  and a discontinuous spring where  $F_{si}(q_i) = k_i \operatorname{sign}(q_i)$ .

In order to get  $m_i$  moving, the virtual spring needs to overcome a *friction threshold* of  $\pm c_i$ . Intuitively, as long as  $q_i - q_i^* > \frac{c_i}{k_i}$  the continuous spring gets  $m_i$  moving, but once  $q_i - q_i^* \leq \frac{c_i}{k_i}$  mass  $m_i$  comes to a hold. Hence the control objective might not be achieved. On the other hand, the discontinuous spring provides a spring force of  $\pm k_i$  as long as  $q_i - q_i^* \neq 0$ . Hence, if  $k_i > c_i$  the control objective is achieved.

The example above provides some intuition why the continuous virtual springs might not achieve (3.9), while the discontinuous counterpart achieves the desired objectives. Before presenting the two formation controller designs, first the agents dynamics subject to Coulomb friction are derived in the port-Hamiltonian framework.

### 3.4.1 Model of ideal Coulomb friction

Consider a point mass  $m_i$  and let  $q_i \in \mathbb{R}$  denote its 1-dimensional position. Assume that the mass is subject to a Coulomb friction force  $F_i(v_i)$ , with  $v_i = \dot{q}_i$  the velocity



**Figure 3.5:** System of Example 3.1: mass  $m_i$  is subject to a Coulomb friction force  $F_{di}$  and is controlled by a virtual spring force  $F_{si}$ .

of the mass. The Coulomb friction force for mass  $m_i$  moving in  $\mathbb{R}$  [59, 118, 119] is given by

$$F_i(v_i) := \begin{cases} \{c_i \operatorname{sign}(v_i)\} & \text{if } v_i \neq 0, \\ [-c_i, c_i] & \text{if } v_i = 0, \end{cases} \quad (3.10)$$

where the function  $\operatorname{sign} : \mathbb{R} \rightarrow \{-1, +1\}$  is defined as  $\operatorname{sign}(v_i) = +1$  if  $v_i \geq 0$  and  $\operatorname{sign}(v_i) = -1$  if  $v_i < 0$ .

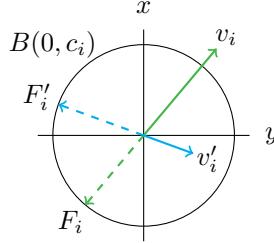
Now consider the same mass  $m_i$  moving in  $\mathbb{R}^2$ . The model of Coulomb friction in  $\mathbb{R}^2$  is more involved. Instead of having two decoupled friction forces along the  $x$  and  $y$  direction, consider a more natural model for the friction force that is defined along the direction of motion (see Figure 3.6). The Coulomb friction force acting on the mass  $m_i$  is modeled as a set-valued map  $F_i : \mathbb{R}^2 \mapsto \mathbb{R}^2$ , where  $F_i(v_i)$  is given by

$$F_i(v_i) := \begin{cases} c_i \frac{v_i}{\|v_i\|} & \text{if } v_i \neq 0, \\ B(0, c_i) & \text{if } v_i = 0, \end{cases} \quad (3.11)$$

with  $c_i \in \mathbb{R}^+$  the friction coefficient and  $B(0, c_i)$  a disc with radius  $c_i$  centered at the origin (see Figure 3.6). Both (3.10) and (3.11) are set-valued maps and their definitions are in the accordance with the definition of the Krasovskii map for  $\operatorname{sign}(v)$  in  $\mathbb{R}$  and  $\mathbb{R}^2$  respectively (see Section 2.4).

*Remark 3.4* (Physical interpretation  $B(0, c_i)$ ). The disc  $B(0, c_i)$  has a clear physical interpretation. When agent  $i$  stands still, the controller needs to overcome a threshold of  $\pm c_i$  before the agent starts moving. The disc  $B(0, c_i)$  represents the physical fact that there is a friction force even though the agent might not be moving (e.g. imagine gently pushing a (heavy) mass which is not moving).

Including Coulomb friction force (3.11) into the agent dynamics (3.1) is straightforward. Let  $q_i = (q_{xi}, q_{yi})^T$  and  $p_i = (p_{xi}, p_{yi})^T$  denote the position and momentum along the  $x$  and  $y$  direction. Furthermore, let  $u_i = (u_{xi}, u_{yi})^T$ ,  $y_i = (y_{xi}, y_{yi})^T$  and  $u_i^r = (u_{xi}^r, u_{yi}^r)^T$ ,  $y_i^r = (y_{xi}^r, y_{yi}^r)^T$  denote the port-variables of the control port and the resistive port respectively. Then the dynamics of agent  $i$  subject to Coulomb



**Figure 3.6:** Illustration of the Coulomb friction force (3.11) in  $\mathbb{R}^2$ . The solid lines represent the velocities, while the dashed lines represent the corresponding Coulomb friction forces.

friction are obtained by taking  $D^a = 0$  and defining the resistive port relation  $u_i^r = -F_i(y_i^r)$ , with  $F_i$  defined in (3.11).

From (3.11) it immediately follows that

$$\begin{cases} y_i^{rT} u_i^r = -c_i \frac{\|y_i^r\|^2}{\|u_i^r\|} < 0 & \text{for } y_i^r \neq 0, \\ y_i^{rT} u_i^r = 0 & \text{for } y_i^r = 0, \end{cases} \quad (3.12)$$

which implies that  $y_i^{rT} u_i^r \leq 0$  (i.e., the resistive element dissipates energy). Hence, it follows that each agent  $i$  is output passive with respect to its velocity.

The dynamics for a network of  $N$  agents of the form subject to Coulomb friction is straightforwardly put into the form (3.2) straightforwardly by setting  $D^a = 0$ , Hamiltonian  $H^a(p) = \sum_{i=1}^N H_i^a(p_i) = \frac{1}{2} p^T M^{-1} p$  and resistive port relation  $u^r = -F(y^r)$  where  $u_i^r = -F_i(y_i^r)$  (see (3.11)).

### Control design and analysis

Two types of controllers to achieve the formation control objectives (3.9) for agents of the form (3.2) are presented here. Each of the controllers acts like a virtual spring which is assigned between the agents. The prescribed relative position  $z_\ell^*$  for  $\ell \in \{1, \dots, E\}$  corresponds to the desired relative position amongst the two agents which are interconnected by spring  $\ell$ . The two types of controller are *continuous* and *discontinuous* virtual springs [59] to achieve the so-called *position-based* formation control in terms of [3, 9].

Before the design and analysis of the continuous (Section 3.4.2) and discontinuous virtual springs (Section 3.4.3) the formal analysis of the motivational Example 3.1 is given to illustrate the non-smooth tools used in the remainder.

**Example 3.2** (Formal analysis of Example 3.1). Example 3.1 considers a mass  $m_i$  subject to a Coulomb friction force  $F_{di}$ , which is controlled by a virtual spring force  $F_{si}$  (see Figure 3.5). For simplicity assume that the control objective is to reach a zero displacement from the wall (i.e.,  $q_i^* = 0$ ).

The port-Hamiltonian dynamics are obtained from (3.1) by taking  $H^a = \frac{1}{2m}p^2$  and are given by

$$\begin{aligned}\dot{q} &= \frac{p}{m}, \\ \dot{p} &= u + u^r, \\ y = y^r &= \frac{p}{m},\end{aligned}\tag{3.13}$$

where  $u$  is the control law (the spring force) and  $u^r$  is the Coulomb friction force defined in (3.10). Hence,  $u^r = F(\frac{p}{m})$  is

$$F\left(\frac{p}{m}\right) = \begin{cases} c\{\text{sign}\left(\frac{p}{m}\right)\} & \text{if } \frac{p}{m} \neq 0, \\ [-c, c] & \text{if } \frac{p}{m} = 0. \end{cases}$$

Considering the above set-valued map, the dynamics (3.13) can be rewritten as the differential inclusion

$$\begin{aligned}\dot{q} &= \frac{p}{m}, \\ \dot{p} &\in -F\left(\frac{p}{m}\right) + u, \\ y = y^r &= \frac{p}{m}.\end{aligned}\tag{3.14}$$

Sections 3.4.2 and 3.4.3 adopt a Krasovskii notion of solution (see Section 2.4) to analyze the solutions of the above differential inclusion.

To control (3.14), first consider a continuous virtual spring of the form

$$u = -\frac{\partial H^c}{\partial q} = -kq,$$

with corresponding potential spring energy  $H^c = \frac{1}{2}kq^2$  and closed-loop Hamiltonian  $H(p, q) = H^a(p) + H^c(q)$ . To analyze the stability and convergence of the solutions of (3.14), take  $H(p, q)$  as candidate Lyapunov function and calculate its set-valued derivative as  $\dot{H} \in -\frac{p}{m}F\left(\frac{p}{m}\right) \subseteq (-\infty, 0]$ . Now, invoking the non-smooth LaSalle's invariance principle (Theorem 2.14), the system converges to the largest weakly invariant set where  $0 \in \dot{H}$ . The latter implies that on the invariant set  $p = 0$ .

Substituting  $p = 0$  in (3.14) gives that  $\dot{q} = 0$  and therefore  $q = q^{eq}$  where  $q^{eq}$  is a constant. Moreover, it follows that  $0 \in [-c, c] - kq^{eq}$  (i.e.,  $-c - kq^{eq} \leq 0 \leq c - kq^{eq}$ ) which implies that  $q^{eq} \in [-\frac{c}{k}, \frac{c}{k}]$ . In other words, the position  $q$  converges to a value in an interval containing the origin. Therefore, convergence of the position to zero cannot be guaranteed and hence the control objective might not be achieved.

Second, consider a discontinuous spring where the potential spring energy is  $H^d = k|q|$ . The control law  $u$  is equal to the Clarke generalized gradient (see

(2.21)) of  $H^d$ , that is

$$u = \begin{cases} -k\{\text{sign}(q)\} & \text{if } q \neq 0, \\ [-k, k] & \text{if } q = 0. \end{cases}$$

The closed-loop Hamiltonian is  $H(p, q) = H^a(p) + H^d(q)$ , which is locally Lipschitz and regular [6]. Same as the previous case, take  $H(p, q)$  as the candidate Lyapunov function and calculate the time derivative  $\dot{H} \in -\frac{p}{m}F(\frac{p}{m}) \subseteq (-\infty, 0]$ . Now, invoking the non-smooth LaSalle's invariance principle (Theorem 2.14) once more, the system converges to the largest weakly invariant set where  $p = 0$ . Substituting  $p = 0$  in (3.14), it follows that  $\dot{q} = 0$  and therefore  $q = q^{eq}$  where  $q^{eq}$  is a constant. If  $q^{eq} > 0$ , it follows that  $0 \in [-c, c] - k$ , while if  $q^{eq} < 0$ , it follows that  $0 \in [-c, c] + k$ .

Now, assume that  $k > c$  (note that  $k$  is a design parameter). Therefore, for  $q^{eq} > 0$  both the lower and upper bounds of the interval  $[-(c+k), c-k]$  are negative, while for  $q^{eq} < 0$  both bounds of  $[-c+k, c+k]$  are positive. Since zero cannot belong to these intervals it follows by contradiction that  $q^{eq}$  is necessarily equal to zero for  $k > c$ , thereby achieving the control objective.

The formal analysis of Example 3.1 above, provides the motivation for the design and analysis of continuous and discontinuous virtual springs to achieve formation control in the presence of Coulomb friction. First continuous virtual springs are discussed, followed by their discontinuous counterpart.

### 3.4.2 Formation control using continuous virtual springs

Position-based formation keeping control aims at achieving a desired shape and orientation for the network of agents [9]. For both types of virtual springs, one spring is assigned along the  $x$  direction and one spring is assigned along the  $y$  direction. For spring  $\ell$  consider the relative position  $z_\ell = (z_{x\ell}, z_{y\ell})^T$ , desired relative position  $z_\ell^* = (z_{x\ell}^*, z_{y\ell}^*)^T$ , spring constants  $K_\ell^c = \text{diag}(k_{x\ell}^c, k_{y\ell}^c)$ . Here the relative position of the spring  $z_\ell$  corresponds to the relative position between two agents in the network. The input to virtual spring  $\ell$  is a relative velocity  $w_\ell = (w_{x\ell}, w_{y\ell})^T$ , while the output is the corresponding spring force  $\tau_\ell = (\tau_{x\ell}, \tau_{y\ell})^T$ .

The continuous spring dynamics are obtained from (3.4) by setting  $D_j^c = 0$ . For  $E$  virtual springs let  $z = (z_1, \dots, z_E)^T$ ,  $z^* = (z_1^*, \dots, z_E^*)^T$ ,  $w = (w_1, \dots, w_E)^T$ ,  $\tau = (\tau_1, \dots, \tau_E)^T$  and  $K^c = \text{block.diag}(K_1^c, \dots, K_E^c)$ . The dynamics of the continuous virtual springs are given by [39, 110]

$$\begin{aligned} \dot{z} &= w, \\ \tau &= \frac{\partial H^c}{\partial z}, \end{aligned} \tag{3.15}$$

with Hamiltonian  $H^c(z) = \frac{1}{2}(z - z^*)^T K^c (z - z^*)$ . Note that the corresponding

partial derivatives along the  $x$  and  $y$  direction are given by  $\frac{\partial H^c}{\partial z_{x\ell}} = k_{x\ell}^c(z_{x\ell} - z_{x\ell}^*)$ ,  $\frac{\partial H^c}{\partial z_{y\ell}} = k_{y\ell}(z_{y\ell} - z_{y\ell}^*)$ . The coupling law to assign the virtual springs between the agents [59] is given by (3.6).

The closed-loop dynamics follows (3.2), (3.6), (3.15) and is given by

$$\begin{aligned}\dot{p} &= -(B \otimes I_2) \frac{\partial H}{\partial z} + u^r, \\ \dot{z} &= (B^T \otimes I_2) \frac{\partial H}{\partial p}.\end{aligned}\tag{3.16}$$

where  $H(p, z)$  is the closed-loop Hamiltonian given by

$$\begin{aligned}H(p, z) &= H^a(p) + H^c(z) \\ &= \frac{1}{2} p^T M^{-1} p + \frac{1}{2} (z - z^*)^T K^c (z - z^*).\end{aligned}\tag{3.17}$$

Since all agents in the network are subject to Coulomb friction, the term  $u^r$  in (3.16) represents the friction and is equal to a set-valued map  $u^r = -F(M^{-1}p)$ , where  $u_i^r = -F_i(M_i^{-1}p_i)$  from (3.11). Hence, the closed loop dynamics of the network is a differential inclusion given by

$$\begin{aligned}\dot{p} &\in -(B \otimes I_2) \frac{\partial H}{\partial z} - F(M^{-1}p), \\ \dot{z} &= (B^T \otimes I_2) \frac{\partial H}{\partial p}.\end{aligned}\tag{3.18}$$

Hence, the closed-loop dynamics can be written in a compact form  $(\dot{p}, \dot{z}) \in \mathcal{K}_1(p, z)$  with

$$\mathcal{K}_1(p, z) = \begin{pmatrix} -(B \otimes I_2)K^c(z - z^*) - F(M^{-1}p) \\ (B^T \otimes I_2)M^{-1}p \end{pmatrix},$$

where  $F(M^{-1}p) = \times_{i=1}^N F_i(M_i^{-1}p_i)$ . The map  $F_i(M_i^{-1}p_i)$  is given by (3.11), i.e.,

$$F_i(M_i^{-1}p_i) = \begin{cases} c_i \frac{M_i^{-1}p_i}{\|M_i^{-1}p_i\|} & \text{if } p_i \neq 0, \\ B(0, c_i) & \text{if } p_i = 0. \end{cases}\tag{3.19}$$

Note that  $M_i^{-1}p_i$  ( $M_i > 0$ ) is equal to the velocity of agent  $i$ . Before presenting the analysis of the closed-loop system controlled with continuous springs, first consider the following definition on the terminal node and edge sets of the graph. The steps correspond to the steps in the proof later on.

**Definition 3.5** (Terminal node and edge set). Let  $\mathcal{V}^s$  denote the set of nodes for step  $s$  ( $s \geq 1$ ), which is defined as  $\mathcal{V}^s = \mathcal{V} - \bigcup_{r=0}^{s-1} \bar{\mathcal{V}}^r$ . Here,  $\bar{\mathcal{V}}^r$  denotes the set of

terminal nodes, which is defined as

$$\bar{\mathcal{V}}^r := \{v_i \in \mathcal{V}^r \mid \deg v_i = 1\},$$

with  $\bar{\mathcal{V}}^0 = \emptyset$ . In a similar way, let  $\mathcal{E}^s$  denote the set of edges for step  $s$ , which is defined as  $\mathcal{E}^s = \mathcal{E} - \bigcup_{r=0}^{s-1} \bar{\mathcal{E}}^r$ . Here,  $\bar{\mathcal{E}}^r$  denotes the set of terminal edges, which is defined as

$$\bar{\mathcal{E}}^r := \{e_k \in \mathcal{E}^r \mid e_k = (v_i, v_j), v_i \in \bar{\mathcal{V}}^r \text{ or } v_j \in \bar{\mathcal{V}}^r\},$$

with  $\bar{\mathcal{E}}^0 = \emptyset$ .

The result for formation control in the presence of Coulomb friction using continuous springs is given in the following Theorem.

**Theorem 3.6** (Continuous virtual springs). *Interconnect agents (3.2) with virtual springs (3.15) via coupling (3.6) using a tree graph topology. Then, the solutions of the closed-loop dynamics (3.18) converge to the largest weakly invariant set where  $p = 0$  and  $(z_\ell - z_\ell^*) \in B(0, \alpha_\ell)$  for all  $\ell \in \mathcal{E}$ , where  $\alpha_\ell$  is a positive constant depending on the spring constants  $K_1^c, \dots, K_E^c$  and Coulomb friction coefficients  $c_1, \dots, c_N$ .*

*Proof.* Take the Hamiltonian  $H(p, z)$  in (3.17) as the Lyapunov function candidate. Since  $H(p, z)$  is continuously differentiable, the set-valued derivative  $\dot{H}(p, z)$  along (3.18) is

$$\dot{H}(p, z) = \{\nabla H(p, z).w, w \in \mathcal{K}_1(p, z)\}.$$

By definition of  $F(M^{-1}p)$  in (3.19), for any  $w \in \mathcal{K}_1(p, z)$  there exists  $w^p \in F(M^{-1}p)$ ,  $w^p = (w_1^p, \dots, w_N^p)^T$ , such that

$$w = \begin{pmatrix} -(B \otimes I_2)K^c(z - z^*) - w^p \\ (B^T \otimes I_2)M^{-1}p \end{pmatrix}.$$

Hence,  $\dot{H}(p, z) = \{a \in \mathbb{R} : a = -p^T M^{-T} w^p\} \subset (-\infty, 0]$ . Therefore, applying the non-smooth LaSalle's invariance principle (Theorem 2.14), the solutions of the closed-loop system converge to the largest weakly invariant set of points  $(p, z)$  where  $p = 0$  and

$$\left\{(0, z) \mid 0 = \begin{pmatrix} -w^p - (B \otimes I_2)K^c(z - z^*) \\ 0 \end{pmatrix}\right\}. \quad (3.20)$$

Consider the graph  $\mathcal{G}(\mathcal{V}, \mathcal{E})$  and define the terminal node-set  $\bar{\mathcal{V}}^1$  and edge-set  $\bar{\mathcal{E}}^1$  accordingly. From (3.20), for all  $v_i \in \bar{\mathcal{V}}^1$  we have

$$0 = -w_i^p + \sum_{\ell \in \bar{\mathcal{E}}^1} b_{i\ell} K_\ell^c (z_\ell - z_\ell^*),$$

where  $w_i^p \in F_i(M_i^{-1}p_i)$ . Note that, similar to  $z_\ell$ ,  $w_i^p$  belongs to  $\mathbb{R}^2$  and it has a vector representation as  $w_i^p = (w_{xi}^p, w_{yi}^p)^T$ . Since  $v_i$  is a terminal node, the above equation simplifies to

$$0 = -w_i^p + b_{i\ell} K_\ell^c (z_\ell - z_\ell^*). \quad (3.21)$$

Since there is only one  $b_{i\ell} \neq 0$  for each  $v_i \in \bar{\mathcal{V}}^1$ . From (3.21) it immediately follows that

$$(z_\ell - z_\ell^*) = b_{i\ell} K_\ell^{c-1} w_i^p, \quad (3.22)$$

for all  $k \in \bar{\mathcal{E}}^1$ . Since  $w_i^p \in B(0, c_i)$ , the equality (3.22) results in

$$(z_\ell - z_\ell^*) \in \left\{ e \mid e = b_{i\ell} K_\ell^{c-1} w_i^p, w_i^p \in B(0, c_i) \right\}.$$

Thus a bound on the size (2-norm) of  $(z_\ell - z_\ell^*)$  is given by

$$\|z_\ell - z_\ell^*\| \in \left[ 0, c_i \sqrt{\text{trace } K_\ell^{c-1} (K_\ell^{c-1})^T} \right].$$

Recall that  $K_\ell^c = \text{diag}(k_{x\ell}^c, k_{y\ell}^c)$  and therefore the above equation implies

$$\|z_\ell - z_\ell^*\| \in \left[ 0, \frac{c_i}{k_{x\ell}^c k_{y\ell}^c} \sqrt{k_{x\ell}^{c-2} + k_{y\ell}^{c-2}} \right].$$

In this a bound is obtained for all springs corresponding to an edge in the terminal edge set  $\bar{\mathcal{E}}^1$ .

Now consider the node set  $\mathcal{V}^2$  and edge set  $\mathcal{E}^2$  (see Definition 3.5) and the corresponding terminal sets  $\bar{\mathcal{V}}^2$  and  $\bar{\mathcal{E}}^2$ . Similar to the previous step, for all  $v_i \in \bar{\mathcal{V}}^2$  it follows that

$$0 = -w_i^p + \sum_{\ell \in \mathcal{E}} b_{i\ell} K_\ell^c (z_\ell - z_\ell^*). \quad (3.23)$$

Rewrite (3.23) as

$$0 = -w_i^p + \sum_{k \in \bar{\mathcal{E}}^1} b_{ik} K_k^c (z_k - z_k^*) + \sum_{\ell \in \bar{\mathcal{E}}^2} b_{i\ell} K_\ell^c (z_\ell - z_\ell^*). \quad (3.24)$$

Similar to the first step, there is exactly one  $b_{i\ell} \neq 0$  for all  $v_i \in \bar{\mathcal{V}}^2, e_\ell \in \bar{\mathcal{E}}^2$ . In addition, for each  $(z_k - z_k^*) \in \bar{\mathcal{E}}^1$  equation (3.22) still holds. Therefore,

$$b_{i\ell} K_\ell^c (z_\ell - z_\ell^*) \in \left\{ e \mid e = w_i^p - \sum_{j \in \bar{\mathcal{V}}^1} w_j^p, w_j^p \in B(0, c_j), w_i^p \in B(0, c_i) \right\}.$$

Thereby obtaining a bound for all springs corresponding to an edge in the terminal

edge set  $\bar{\mathcal{E}}^2$ . Repeating the steps above until  $\bigcup_i \bar{\mathcal{E}}^i = \mathcal{E}$ , a bound is obtained for all springs in the graph. Each bound depends on the friction coefficients  $c_1, \dots, c_N$  and spring constants  $K_1^c, \dots, K_E^c$ , which completes the proof.  $\square$

*Remark 3.7* (Convergence to desired formation). Theorem 3.6 only guarantees that relative displacement error  $z_\ell - z_\ell^*$  converges to the disc  $B(0, \alpha_\ell)$ . It does not guarantee that relative displacement  $z_\ell$  converges to the desired relative displacement  $z_\ell^*$  and therefore objectives (3.9) might not be achieved.

### 3.4.3 Formation control using discontinuous virtual springs

The previous section showed that continuous virtual springs can not always achieve the formation objectives (3.9) exactly. Motivated by Example 3.1 in this section the continuous virtual springs are replaced by their discontinuous counterparts. Using the same notation as in (3.15), the dynamics of  $E$  discontinuous springs are given by

$$\begin{aligned}\dot{z} &= w, \\ \tau &= \partial H^d,\end{aligned}\tag{3.25}$$

with  $\partial H^d$  the generalized Clarke gradient of the locally Lipschitz Hamiltonian  $H^d(z) = \|K^c(z - z^*)\|_1$ , where  $\|K^c(z - z^*)\|_1$  denotes the one-norm. That is  $\|K^c(z - z^*)\|_1 = \sum_{\ell=1}^E (k_{x\ell}^c |z_{x\ell} - z_{x\ell}^*| + k_{y\ell}^c |z_{y\ell} - z_{y\ell}^*|)$ . Since  $H^d$  is not differentiable everywhere, the standard partial derivative is replaced by the Clarke generalized gradient (see (2.21))  $\partial H^d = K^c \mathcal{K} \operatorname{sign}(z - z^*)$  where  $\mathcal{K} \operatorname{sign}(\tilde{z}) = \bigtimes_{\ell=1}^E \mathcal{K} \operatorname{sign}(z_\ell - z_\ell^*)$  with

$$\mathcal{K} \operatorname{sign}(z_\ell - z_\ell^*) = \begin{cases} \left\{ \frac{z_{x\ell} - z_{x\ell}^*}{|z_{x\ell} - z_{x\ell}^*|} \right\} \times \left\{ \frac{z_{y\ell} - z_{y\ell}^*}{|z_{y\ell} - z_{y\ell}^*|} \right\} & \text{if } (z_{x\ell} - z_{x\ell}^*) \neq 0, (z_{y\ell} - z_{y\ell}^*) \neq 0, \\ [-1, +1] \times \left\{ \frac{z_{y\ell} - z_{y\ell}^*}{|z_{y\ell} - z_{y\ell}^*|} \right\} & \text{if } (z_{x\ell} - z_{x\ell}^*) = 0, (z_{y\ell} - z_{y\ell}^*) \neq 0, \\ \left\{ \frac{z_{x\ell} - z_{x\ell}^*}{|z_{x\ell} - z_{x\ell}^*|} \right\} \times [-1, +1] & \text{if } (z_{x\ell} - z_{x\ell}^*) \neq 0, (z_{y\ell} - z_{y\ell}^*) = 0, \\ [-1, +1] \times [-1, +1] & \text{if } (z_{x\ell} - z_{x\ell}^*) = 0, (z_{y\ell} - z_{y\ell}^*) = 0. \end{cases}\tag{3.26}$$

Using the same coupling (3.6) as for the continuous springs, the closed-loop dynamics for the discontinuous springs follows directly from (3.2), (3.6), (3.25) and is given by

$$\begin{aligned}\dot{p} &\in -(B \otimes I_2) \partial_z H - F(M^{-1} p), \\ \dot{z} &= (B^T \otimes I_2) \frac{\partial H}{\partial p},\end{aligned}\tag{3.27}$$

where  $\partial_z H(z) = \partial H^d(z)$  and  $H(p, z)$  is the closed-loop Hamiltonian given by

$$\begin{aligned} H(p, z) &= H^a(p) + H^d(z) \\ &= \frac{1}{2} p^T M^{-1} p + \|K^c(z - z^*)\|_1. \end{aligned} \quad (3.28)$$

Note that (3.27) is a differential inclusion due to the set-valued model of Coulomb friction and the discontinuous spring force  $\partial H^d$ .

*Remark 3.8* (Design choice for discontinuous springs). Discontinuous spring forces (3.26) consider springs along  $x$  and  $y$  separately in accordance with the continuous springs (3.15). This is a design choice that complies with the position-based control design as stated in Section 3.4.2. Instead these “square” discontinuous springs could be replaced by “round” springs similar to (3.11) and similar results can be derived.

Now, write the closed-loop dynamics in a compact form  $(\dot{p}, \dot{z}) \in \mathcal{K}_2(p, \tilde{z})$  with

$$\mathcal{K}_2(p, z) = \begin{pmatrix} -(B \otimes I_2) K^c \mathcal{K} \operatorname{sign}(z - z^*) - F(M^{-1}p) \\ (B^T \otimes I_2) M^{-1} p \end{pmatrix}, \quad (3.29)$$

where  $F(M^{-1}p)$  is defined in (3.19). The main result on formation control in the presence of Coulomb friction using discontinuous virtual springs is stated in the following theorem.

**Theorem 3.9** (Discontinuous springs). *Interconnect agents (3.2) with virtual springs (3.25) via coupling (3.6) using a tree graph topology. Assume that  $\min\{k_{x\ell}, k_{y\ell}\} > \max\{c_i, c_j\}$  for  $e_\ell = (n_i, n_j) \in \mathcal{E}$ . Then the solutions of the closed-loop dynamics (3.27) converge to  $p = 0$ ,  $z = z^*$ , thereby achieving the control objectives (3.9).*

*Proof.* Take the Hamiltonian in (3.28) as candidate Lyapunov function which is a regular and locally Lipschitz function. Since the Hamiltonian is not differentiable everywhere, first calculate its corresponding Clarke generalized gradients  $\partial H(p, z)$  (see (2.21)) as

$$\partial H(p, z) = \left\{ v \mid v = \begin{pmatrix} M^{-1}p \\ K^c v^z \end{pmatrix} \text{ such that } v_k^z \in \mathcal{K} \operatorname{sign}(z_k - z_k^*) \right\}. \quad (3.30)$$

Calculating the set-valued derivative  $\dot{\bar{H}}(p, z)$  along (3.27) gives

$$\dot{\bar{H}}(p, z) = \{a \in \mathbb{R} \mid \exists w \in \mathcal{K}_2(p, z) \text{ such that } a = \langle w, v \rangle \text{ for all } v \in \partial H(p, z)\}.$$

By definition of  $\mathcal{K}_2(p, z)$  in (3.26)-(3.29), for any  $w \in \mathcal{K}_2(p, z)$  there exists  $w^p \in$

$F(M^{-1}p)$  and  $w^z \in \mathcal{K} \text{ sign}(z - z^*)$  such that

$$w = \begin{pmatrix} -(B \otimes I_2)K^c w^z - w^p \\ (B^T \otimes I_2)M^{-1}p \end{pmatrix}.$$

For each  $v \in \partial H(p, z)$ , choose  $w^z = v^z$ . Hence, one obtains  $\dot{H}(p, z) = \{a \in \mathbb{R} : a = -p^T M^{-T} w^p\} \subset (-\infty, 0]$ . Assume that  $\dot{H} \neq \emptyset$ . Therefore, applying the non-smooth version of LaSalle's invariance principle (Theorem 2.14), the solutions of the closed-loop system converge to the largest weakly invariant set of points  $(p, z)$  such that  $p = 0$  and

$$\left\{ (0, z) \mid 0 = \begin{pmatrix} -w^p - (B \otimes I_2) K^c w^z \\ 0 \end{pmatrix} \right\}, \quad (3.31)$$

where  $w^p \in \mathbb{R}^2$  and  $w^z \in \mathbb{R}^2$ . Now, consider the tree graph  $\mathcal{G}(\mathcal{V}, \mathcal{E})$  and define the terminal node-set  $\bar{\mathcal{V}}^1$  and edge-set  $\bar{\mathcal{E}}^1$  accordingly. Based on (3.31), all nodes  $v_i \in \bar{\mathcal{V}}^1$  obey

$$0 = w_i^p + \sum_{\ell \in \bar{\mathcal{E}}} b_{i\ell} K_\ell^c w_\ell^z.$$

Noting that  $v_i$  is a terminal node, the above equation simplifies to

$$0 = w_i^p + b_{i\ell} K_\ell^c w_\ell^z, \quad (3.32)$$

where  $w_i^p = (w_{xi}^p, w_{yi}^p)^T$ ,  $w_i^p \in B(0, c_i)$  and  $w_\ell^z = (w_{x\ell}^z, w_{y\ell}^z)^T$ . Writing (3.32) for each of the components of  $w_i^p$  gives

$$0 = w_{xi}^p + b_{i\ell} k_{x\ell}^c w_{x\ell}^z, \quad (3.33)$$

$$0 = w_{yi}^p + b_{i\ell} k_{y\ell}^c w_{y\ell}^z. \quad (3.34)$$

The remainder of the proof is completed by contradiction. Consider (3.33) and assume that  $\tilde{z}_{x\ell} \neq 0$ , then  $w_{x\ell}^z$  is equal to either  $+1$  or  $-1$ . Moreover, since  $w_i^p \in B(0, c_i)$ , it follows that  $w_{xi}^p \in [-c_i, c_i]$ . Take  $w_{x\ell}^z = +1$ , then from (3.33) it follows that

$$w_{xi}^p + b_{i\ell} k_{x\ell}^c w_{x\ell}^z \in [-c_i + k_{x\ell}^c, c_i + k_{x\ell}^c].$$

By assumption  $k_{x\ell} > c_i$  and hence the upper and lower bounds of the above interval are positive. Hence zero cannot belong to this interval. Similarly, if  $w_{x\ell}^z = -1$ , the bounds of the interval  $[-c_i - k_{x\ell}, c_i - k_{x\ell}]$  are both negative. This result contradicts (3.33). As a result,  $(z_{x\ell} - z_{x\ell}^*)$  is necessarily zero. A similar argument holds for  $y$ -direction (3.34) which results in  $(z_{y\ell} - z_{y\ell}^*) = 0$ . In this way, for all springs corresponding to an edge in the terminal edge set  $\bar{\mathcal{E}}^1$ , the error position  $(z_\ell - z_\ell^*)$  is

equal to zero on the invariant set (3.31).

Now consider the node set  $\mathcal{V}^2$  and edge set  $\mathcal{E}^2$  and the corresponding terminal sets  $\bar{\mathcal{V}}^2$  and  $\bar{\mathcal{E}}^2$ . Using similar arguments as above, it is easily shown that the relative error position is equal to zero for all springs corresponding to an edge in the terminal edge set  $\bar{\mathcal{E}}^2$ . Repeating the steps above until  $\bigcup_i \bar{\mathcal{E}}^i = \mathcal{E}$ , it follows that for each edge in the graph the corresponding relative position error converges zero, thereby completing the proof.  $\square$

Note that in contrast with Theorem 3.6, Theorem 3.9 does guarantee that relative displacement  $z_\ell$  converges to the desired relative displacement  $z_\ell^*$  and therefore objectives (3.9) are achieved.

### 3.4.4 Simulation results

Consider a network of  $N = 5$  robots of the form (3.1) subject to Coulomb friction force (3.11), with model parameters  $m_i = 1\text{ kg}$  and  $c_i = 2\text{ kg/s}$  for  $i = 1, \dots, 5$ . The robots are interconnected using the incidence matrix

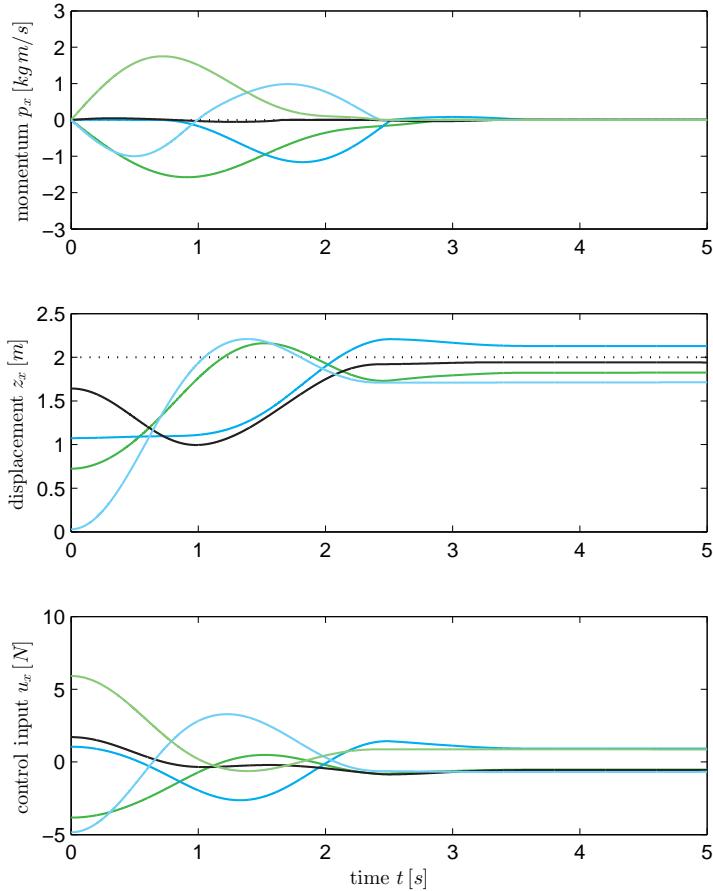
$$B = \begin{pmatrix} -1 & 0 & 0 & 0 \\ +1 & -1 & 0 & 0 \\ 0 & +1 & -1 & 0 \\ 0 & 0 & +1 & -1 \\ 0 & 0 & 0 & +1 \end{pmatrix},$$

which is associated to a *path graph*. The virtual coupling parameters are set at  $k_{x\ell} = k_{y\ell} = 3$  and  $z_{x\ell}^* = z_{y\ell}^* = 2$  for  $\ell = 1, \dots, 4$  (i.e., the condition  $\min\{k_{x\ell}, k_{y\ell}\} > \max\{c_i, c_j\}$  in Theorem 3.9 is satisfied).

The simulations are performed using MATLAB and Simulink and were run for  $t = 5\text{ s}$  starting from the initial conditions  $q_x(0) = (0.3902, 1.1140, 2.1875, 3.8300, 3.8596)\text{ m}$ ,  $q_y(0) = (0.1273, 0.1847, 0.3885, 1.1077, 2.8242)\text{ m}$ ,  $p_x(0) = p_y(0) = 0\text{ kg m/s}$ . The results are shown in Figures 3.7 and 3.8.

Figure 3.7 and Figure 3.8 show the time evolution of the momentum  $p_x$ , displacement  $z_x$ , and control input  $u_x$  along the  $x$  direction for the continuous springs (Theorem 3.6) and the discontinuous springs (Theorem 3.9) respectively. The time evolution of the momentum  $p_y$ , relative displacement  $z_y$  and control input  $u_y$  along the  $y$  direction follow a similar trend and are given in Appendix D.

The top plots in Figure 3.7 and Figure 3.8 show that  $p_x$  converges to zero for both types of springs. Looking at the middle plot in Figure 3.7 shows that the relative displacement  $z_x$  does not converge to the desired one,  $z_x^* = 2$ , using continuous springs. Using discontinuous springs the relative displacement  $z_x$  does converge to the desired prescribed relative displacement  $z_x^*$  (see Figure 3.8, middle).

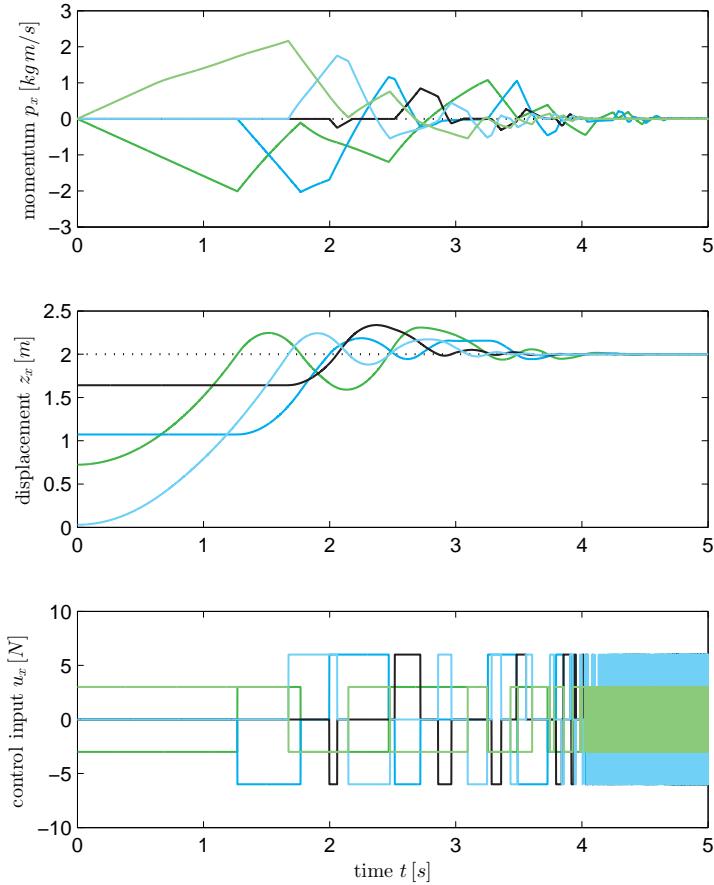


**Figure 3.7:** Time evolution of the momentum  $p_x$ , relative displacement  $z_x$ , and control input  $u_x$  using continuous springs. The dotted lines show the reference values.

Comparing the control action  $u_x$  in (Figure 3.7, bottom) and (Figure 3.8, bottom), the control action related to the discontinuous spring shows a fast switching behavior while  $z_x$  converges to  $z_x^*$ . This switching behavior is a known undesired phenomenon due to the use of a sign-based controller [56, 57].

## 3.5 Deployment

The deployment problem was already briefly introduced in Section 3.1. The objective is to spatially distribute the agents in some optimal sense, usually to cover a certain environment [15]. In this section the objective is to (equally) spatially distribute the agents on a line between two reference points. The approach



**Figure 3.8:** Time evolution of the momentum  $p_x$ , relative displacement  $z_x$ , and control input  $u_x$  using discontinuous springs. The dotted lines show the reference values.

presented here assigns two *virtual walls* at these reference points and deploys the agents (3.2) between these two walls by assigning *virtual couplings* between the agents and the virtual walls. The virtual couplings are the same as the ones considered in Sections 3.3 and 3.4.2 and consists of a virtual spring and a virtual damper.

The interconnection topology (i.e., which agents are interconnected by a virtual coupling) is modeled by a *path graph* (chain graph), which is a special type of tree graph. The outer two nodes of the graph correspond the two virtual walls, while the other  $N$  nodes correspond to the agents. The  $E = N + 1$  edges correspond to the virtual couplings. In a path graph each node is connected to two neighbors, thereby being a natural choice for the deployment on a line problem.

To formally define the deployment objective, let  $z_j \in \mathbb{R}^n$  denote the relative displacement between two agents. The displacement between the two virtual walls is denoted by  $\ell \in \mathbb{R}^n$  and hence  $\sum_{j=1}^{N+1} z_j = \ell$ . Let  $p$  denote the vector of momenta of the agents and let  $z = (z_1, \dots, z_{N+1})^T$ , then the *deployment objective* is formally defined as

$$\begin{cases} p \rightarrow 0, \\ z \rightarrow c, \end{cases} \quad \text{as } t \rightarrow \infty, \quad (3.35)$$

with  $c \in \mathbb{R}^{N+1}$  some arbitrary constant vector with  $\sum_{j=1}^{N+1} c_j = \ell$ .

In many cases, the objective is to *equally* deploy the  $N$  agents on  $\ell$ . In that case (3.35) boils down to the *equal deployment objective*, which is defined as

$$\begin{cases} p \rightarrow 0, \\ z \rightarrow \frac{\ell}{N+1}. \end{cases} \quad \text{as } t \rightarrow \infty. \quad (3.36)$$

The results presented here to achieve (3.35) and (3.36) are published in [110]. In the remainder the resistive port in (3.1) is omitted for simplicity.

### 3.5.1 Deployment using virtual couplings and virtual walls

To position the  $N$  agents, introduce two *virtual walls* positioned at fixed reference points. These two virtual walls are modeled as two additional agents, labeled 0 and  $N+1$ , with position  $q_0$  and  $q_{N+1}$ . Let  $\ell \in \mathbb{R}^n$  denote the displacement between the two walls, which is defined as  $\ell = q_{N+1} - q_0$ .

The outline of this section is as follows. First, the two virtual walls are included into the agent dynamics (3.2). Then, the agents and virtual walls are interconnected using virtual springs. Finally, a kinematic constraint is imposed on the virtual walls which is then eliminated to obtain the dynamics on the constrained state space.

Define the extended state and input vectors as  $\bar{q} = (q_0, q_1, \dots, q_N, q_{N+1})^T$ ,  $\bar{p} = (p_0, p_1, \dots, p_N, p_{N+1})^T$ ,  $\bar{u} = (u_0, u_1, \dots, u_N, u_{N+1})^T$ ,  $\bar{y} = (y_0, y_1, \dots, y_N, y_{N+1})^T$ , and the extended mass and dissipation matrix as  $\bar{M} = \text{block.diag}(M_0, M_1, \dots, M_N, M_{N+1})$ ,  $D^a(q, p) = \text{block.diag}(D_0^a, D_1^a(q_1, p_1), \dots, D_N^a(q_N, p_N), D_{N+1}^a)$ , where  $M_0, M_{N+1}, D_0^a, D_{N+1}^a$  are arbitrary constant diagonal matrices. The agents dynamics including the two virtual walls follow directly from (3.2) and are given by

$$\begin{aligned} \begin{pmatrix} \dot{\bar{q}} \\ \dot{\bar{p}} \end{pmatrix} &= \begin{pmatrix} 0 & I_{(N+2)n} \\ -I_{(N+2)n} & -\bar{D}^a(q, p) \end{pmatrix} \begin{pmatrix} \frac{\partial \bar{H}^a}{\partial \bar{q}}(\bar{p}) \\ \frac{\partial \bar{H}^a}{\partial \bar{p}}(\bar{p}) \end{pmatrix} + \begin{pmatrix} 0 \\ I_{(N+2)n} \end{pmatrix} \bar{u}, \\ \bar{y} &= \frac{\partial \bar{H}^a}{\partial \bar{p}}(\bar{p}), \end{aligned} \quad (3.37)$$

with Hamiltonian  $\bar{H}^a(\bar{p}) = \frac{1}{2} \bar{p}^T \bar{M}^{-1} \bar{p}$ .

The  $N$  agents and two virtual walls (3.37) are interconnected using  $E = N + 1$  virtual couplings (3.5) according to a path graph topology. Let  $B$  denote the incidence matrix associated to the graph, then the coupling of agents and virtual walls on the nodes and virtual couplings at the edges is given by [3, 100]

$$\begin{aligned}\bar{u} &= -(B \otimes I_n)\tau, \\ w &= (B^T \otimes I_n)\bar{y}.\end{aligned}\tag{3.38}$$

Since the closed-loop dynamics are decoupled, the remainder continues with  $n = 1$  without loss of generality. The closed-loop dynamics follow directly from (3.5), (3.37), (3.38) and are given by

$$\begin{pmatrix} \dot{\bar{p}} \\ \dot{z} \end{pmatrix} = \begin{pmatrix} -(\bar{D}^a(p) + B D^c B^T) & -B \\ B^T & 0 \end{pmatrix} \begin{pmatrix} \frac{\partial \bar{H}}{\partial p}(z, \bar{p}) \\ \frac{\partial \bar{H}}{\partial z}(z, \bar{p}) \end{pmatrix},\tag{3.39}$$

with closed-loop Hamiltonian

$$\bar{H}(z, \bar{p}) = \sum_{i=0}^{N+1} \bar{H}_i^a(\bar{p}_i) + \sum_{j=1}^{N+1} H_j^c(z) = \frac{1}{2} \bar{p}^T \bar{M}^{-1} \bar{p} + \frac{1}{2} z^T K z.$$

Note that the dynamics (3.39) include the virtual walls. However, these virtual walls are fixed (i.e.,  $q_0(t) = q_0(0)$ ,  $q_{N+1}(t) = q_{N+1}(0)$ ) which is modeled by the following kinematic constraint

$$\begin{pmatrix} \dot{q}_0 \\ \dot{q}_{N+1} \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 0_{1 \times N} & 0 \\ 0 & 0_{1 \times N} & 1 \end{pmatrix}}_{A^T} \frac{\partial \bar{H}}{\partial \bar{p}}(z, \bar{p}) = 0.\tag{3.40}$$

Since (3.40) is of the form (2.9) one can solve for (3.40) to obtain the dynamics on the constraint state space (see Section 2.3.2). To do so, define matrix  $S$  as

$$S = \begin{pmatrix} 0 \\ I_N \\ 0 \end{pmatrix},$$

such that  $A^T S = 0$  and  $\text{rank } A + \text{rank } S = 2 + N$ . Then, define the coordinate transformation  $p = S^T \bar{p}$  and  $p^w = A^T \bar{p}$ , where  $p^w$  corresponds to the momentum of the two walls (i.e.,  $p^w(t) = 0$  for all  $t$ ). Using this coordinate transformation the  $p^w$  coordinate can be eliminated to obtain the port-Hamiltonian dynamics on the

constrained state space given by

$$\begin{pmatrix} \dot{p} \\ \dot{z} \end{pmatrix} = \begin{pmatrix} -S^T (D^a(p) + B D^c B^T) S & -S^T B \\ B^T S & 0 \end{pmatrix} \begin{pmatrix} \frac{\partial H}{\partial p}(z, p) \\ \frac{\partial H}{\partial z}(z, p) \end{pmatrix}, \quad (3.41)$$

with Hamiltonian

$$H(z, p) = \sum_{i=1}^N H_i^a(p_i) + \sum_{j=1}^{N+1} H_j^c(z_j) = \frac{1}{2} p^T M^{-1} p + \frac{1}{2} z^T K^c z.$$

The main result for deployment is stated in the following proposition.

**Proposition 3.10.** *Interconnect agents and virtual walls (3.37) with virtual couplings (3.5) via coupling (3.6) using a tree graph topology, where the two virtual walls are subject to constraint (3.40). The solutions of the closed-loop system (3.41) converge to  $p = 0, z = c$ , where  $c \in \mathbb{R}^{N+1}$  is a positive constant vector which depends on the inter-wall displacement  $\ell$  and the springs constants  $k_1, \dots, k_{N+1}$ . Thereby achieving deployment objectives (3.35).*

*Proof.* Take  $H(z, p)$  as a candidate Lyapunov function and calculate its time derivative

$$\dot{H}(z, p) = -p^T M^{-T} S^T (D^a(p) + B D^c B^T) S M^{-1} p \leq 0.$$

By invoking LaSalle's invariance principle (Theorem 2.4) it follows that system (3.41) converges to the largest invariant set where  $p = 0$ . On this set

$$S^T B K^c z = 0. \quad (3.42)$$

Writing out  $S^T B$  gives that

$$S^T B = \begin{pmatrix} 1 & -1 & \cdots & 0 & 0 \\ 0 & \ddots & \ddots & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & -1 \end{pmatrix},$$

which equals the transpose of the incidence matrix  $B_p \in \mathbb{R}^{N+1 \times N}$  associated to a path graph with  $N+1$  nodes and  $N$  edges. Since the path graph is a connected graph, it follows that  $\ker S^T B = \ker B_p^T = \alpha \mathbb{1}_{N+1}$ , for some arbitrary constant  $\alpha \in \mathbb{R}$ . Hence from (3.42) it follows that

$$z = \alpha(K^c)^{-1} \mathbb{1}_{N+1}. \quad (3.43)$$

From the problem it also follows that  $\sum_{j=1}^{N+1} z_j = \mathbb{1}_{N+1}^T z = \ell$ , with  $\ell$  the dis-

placement between the two virtual walls (i.e., the spring lengths sum up to the displacement between the walls). Multiplying (3.43) from the left by  $\mathbb{1}_{N+1}^T$  and equating the two terms gives

$$\mathbb{1}_{N+1}^T z = \alpha \mathbb{1}_{N+1}^T (K^c)^{-1} \mathbb{1}_{N+1} = \ell,$$

which implies that  $\alpha = \ell / \sum_{j=1}^{N+1} \frac{1}{k_j^c}$ . Substituting  $\alpha$  into (3.43) results in

$$z_i = \frac{\ell}{k_i^c \sum_{j=1}^{N+1} \frac{1}{k_j^c}} \quad \text{for } i = 1, \dots, N+1, \quad (3.44)$$

thereby completing the proof.  $\square$

Proposition 3.10 provides a result for the general deployment objectives (3.35). For the equal deployment objectives (3.36) consider the following corollary.

**Corollary 3.11** (Equal deployment). *Interconnect agents and virtual walls (3.37) with virtual couplings (3.5) via coupling (3.6) using a tree graph topology, where the two virtual walls are subject to constraint (3.40). Set  $k_1^c = \dots = k_{N+1}^c = k^c$ . Then, the solutions of the closed-loop system (3.41) converge to  $p = 0, z = \frac{\ell}{N+1}$ , thereby achieving the equal deployment objectives (3.36).*

*Proof.* Setting  $k_1^c = \dots = k_{N+1}^c = k^c$  and substituting into (3.44) yields

$$z_i = \frac{\ell}{k^c \sum_{j=1}^{N+1} \frac{1}{k^c}} = \frac{\ell}{N+1} \quad \text{for } i = 1, \dots, N+1. \quad \square$$

*Remark 3.12* (Control input deployment). The control input for the agents  $u$  follows directly from (3.38) and is given by

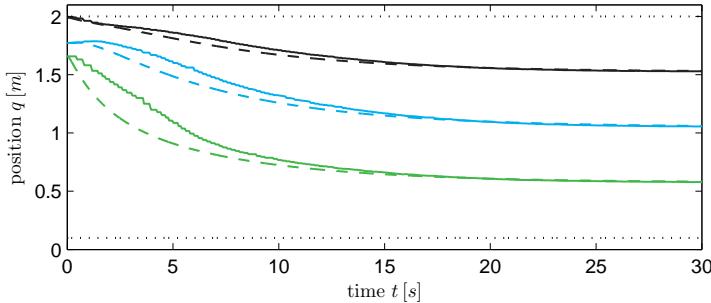
$$u = - \underbrace{S^T (B \otimes I_n) K^c z}_{\text{virtual spring force}} - \underbrace{S^T (B \otimes I_n) D^c (B^T \otimes I_n) S M^{-1} p}_{\text{virtual damping force}}. \quad (3.45)$$

Similar to the control law for formation control (3.8), control law (3.45) has a clear physical interpretation and is a distributed control law.

Proposition 3.10 is illustrated using simulation and experimental results in the next section.

### 3.5.2 Simulation and experimental results

Consider a network with  $N = 3$  agents of the form (3.2) moving in  $\mathbb{R}$ . Each agent  $i$  has a mass  $m_i = 0.167 \text{ kg}$  and a friction coefficient  $D_i^a = 2 \text{ kg/s}$  for  $i = 1, 2, 3$ . The two virtual walls are positioned at  $q_0 = 0.1 \text{ m}, q_4 = 2.0 \text{ m}$  such that



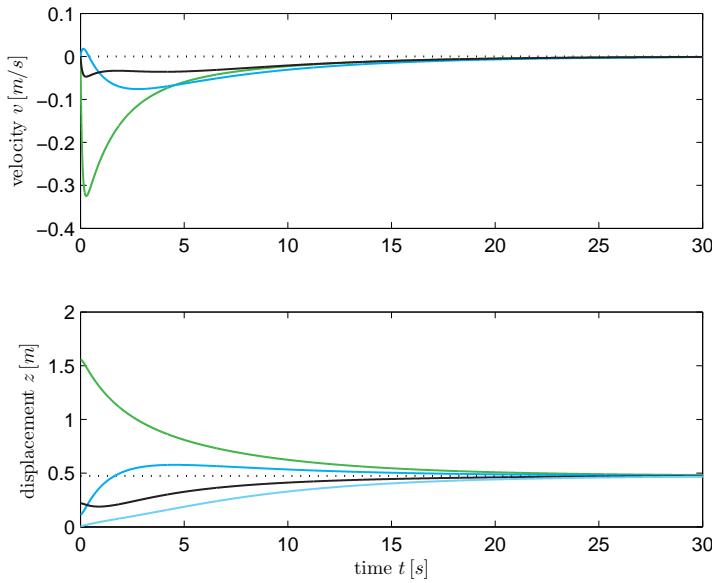
**Figure 3.9:** Time evolution of the position during deployment of the e-puck wheeled robots for the simulations (dashed) and experiments (solid). The black dotted lines represent the two virtual walls.

$\ell = q_4 - q_1 = 1.9 \text{ m}$ . The agents are interconnected using  $E = 4$  homogenous virtual couplings (3.5), with spring constant  $K_j^c = 0.5 \text{ kg/s}^2$  and damping coefficient  $D_j^c = 0$  for  $j = 1, 2, 3, 4$ . Setting all springs constants to the same value corresponds to equal deployment.

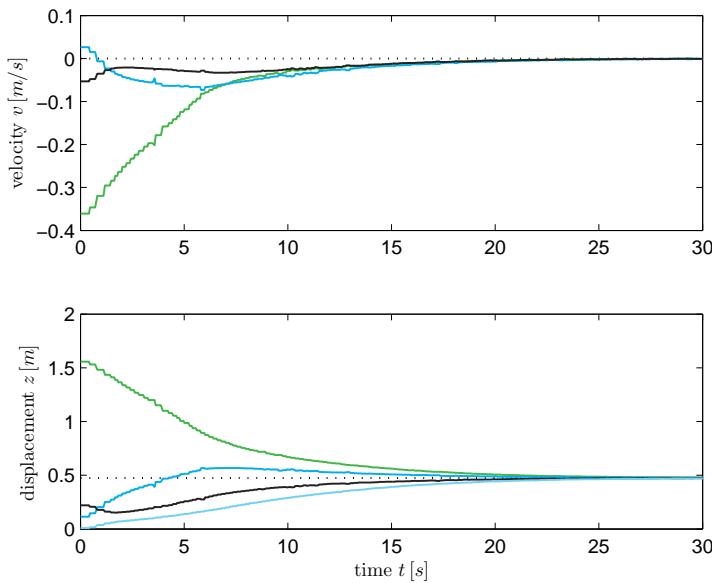
The simulations are run using MATLAB and Simulink. The experimental setup consist of three e-puck wheeled robots, which can only move forward and backward in accordance with the movement in  $\mathbb{R}$ . By constraining the movement of the e-pucks, they act as fully actuated agents along the forward/backward direction since the constraint on the wheel axle is not interfering. More details on the e-puck robot and the experimental setup are given in respectively Sections C.1 and C.2. Both the simulation and the experiment are run for  $t = 30 \text{ s}$  and the initial conditions are set at  $q(0) = (1.66, 1.77, 1.99) \text{ m}$ ,  $p(0) = (0, 0, 0) \text{ kg m/s}$ . The results are shown in Figures 3.9, 3.10, and 3.11.

Figure 3.9 show the position of the agents during the simulation and the experiment. The (small) differences are due to localization errors of the vision algorithm, model parameter uncertainty, and a difference in the actuators of the model and the e-puck (the robot dynamics (3.1) do not consider the differential drive of the e-puck). To clarify that the formation control objectives (3.36) are indeed achieved, Figures 3.10 and 3.11 show the time evolution of the velocity  $v$  and relative displacement  $z$  respectively. Both figures show a similar trend and illustrate that  $v \rightarrow 0, z \rightarrow \frac{\ell}{4} = \text{as } t \rightarrow \infty$ .

Note that the relative displacement  $z$  (Figure 3.10-3.11 (bottom)) converges to the same value as for the formation control simulation and experiment in Section 3.3 (Figure 3.3-3.4 (bottom)). However, due to the two virtual walls, the agents position quite differently during the deployment (Figure 3.9) and formation simulation and experiment (Figure 3.2). This observation nicely aligns with the schematic representation in Figure 3.1.



**Figure 3.10:** Time evolution of the velocity  $v$  and relative displacement  $z$  during deployment (simulation). The dotted lines show the reference values.



**Figure 3.11:** Time evolution of the velocity  $v$  and relative displacement  $z$  during deployment (experiment). The dotted lines show the reference values.

Preliminary results on tuning of the controller gains in an experimental setting may be found in [92]. The performance indicators considered in [92] are the energy consumption, absolute formation error, and percentage of time out of formation.

### 3.6 Concluding remarks

This chapter considers the problems of formation control in the presence of ideal Coulomb friction and deployment for a network of fully actuated systems. First, a brief recall on formation control using virtual couplings in the port-Hamiltonian framework is given, providing a starting point for the two problems considered.

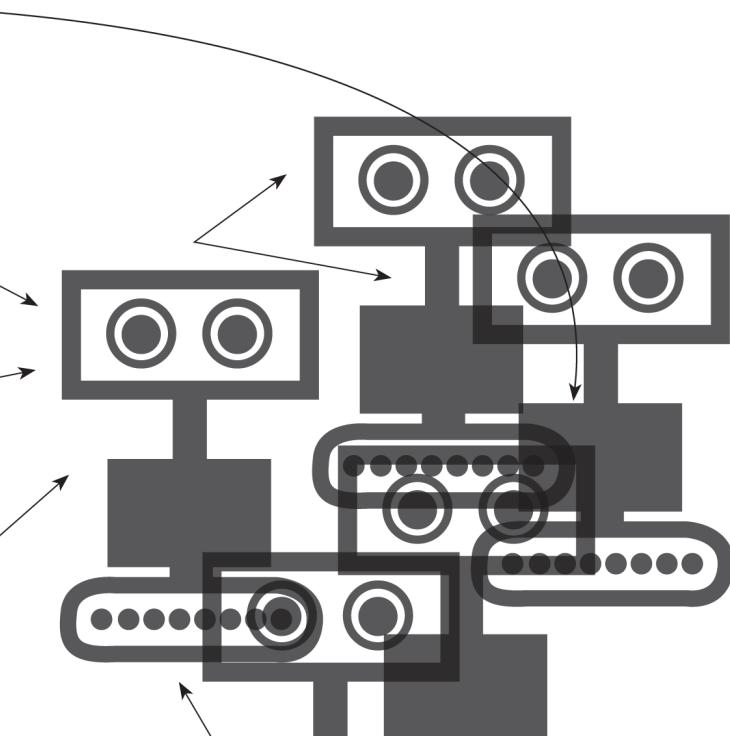
Coulomb friction renders the agent dynamics non-smooth, which prevents standard continuous springs to achieve exact formations. A bound on the formation error is presented, when using continuous virtual springs. This bound depends on the friction coefficients of the agents and the spring constants of the virtual springs. The discontinuous counterpart of the virtual spring does achieve exact formations, under a natural condition on its control gain.

For the deployment problem the concept of a virtual wall is introduced. By interconnecting agents in the network and virtual walls using virtual couplings the deployment objective is achieved. Simulation and experimental results are provided to illustrate the effectiveness of the approach.



# Chapter 4

Formation control of nonholonomic wheeled robots





## Chapter 4

# Formation control of nonholonomic wheeled robots

This chapter deals with formation control of a network of *nonholonomic wheeled robots*, as opposed to the fully actuated point masses considered in the previous Chapter 3. First an introduction to the three problems considered in this chapter is given in Section 4.1. Then, Section 4.2 continues with the derivation of the dynamical model of the wheeled robot in the port-Hamiltonian framework.

The three subsequent sections deal with three related problems: formation control (Section 4.3), formation control with velocity tracking (Section 4.4), and formation control in the presence of matched input disturbances (Section 4.5). Each section provides a formal problem formulation, controller design, and closed-loop analysis. The results are illustrated with simulation and experimental results. The highlights and concluding remarks of the chapter are given in Section 4.6.

The results in this chapter are published in [60, 113, 115, 116]. The results in Section 4.5 are based on a collaboration with Matin Jafarian and Claudio De Persis.

## 4.1 Introduction

This chapter considers three formation control problems for a network of wheeled robots. Starting point is the approach using virtual couplings presented in Chapter 3. Recall that the objective for formation control is to achieve a prescribed geometrical shape, using only local feedback rules. In addition to standard formation control the problems of *velocity tracking* and *disturbance rejection* are considered here.

Control of nonholonomic wheeled robots has received quite some attention in recent years [5, 17, 34, 49, 67, 91, 97]. Due to the nonholonomic constraint on the wheel axle, the wheeled robot does not satisfy Brockett's necessary condition for continuous smooth feedback stabilization [13]. Hence discontinuous control laws [5, 17, 68] and time-varying control laws [91, 97] have been developed to stabilize the robot dynamics. While [5, 17, 91, 97] are devoted to a single robot, more recent work focused on the control of multiple wheeled robots [34, 49, 67].

Another reason why the wheeled robot is of interest, is the availability of many types of wheeled robots for implementation of algorithms in practice. One example of such a robot is the *e-puck* (Figure C.1), which is used throughout this thesis for

the experimental results. The e-puck robot is designed for engineering education at the university level [79] and provides an easy interface to test algorithms.

The algorithms in this chapter achieve formation control by assigning virtual couplings between the robots in the network. This approach is a flexible form of the more rigid virtual structure approach for formation control of mobile robots [67]. As in Chapter 3, each virtual coupling consists of a virtual spring and a virtual damper. The virtual springs determine the formation shape by shaping the energy function of the network, while the virtual dampers shape the transient response by injecting damping. The interconnection topology amongst robots and virtual couplings is as in Chapter 3 described by a tree graph (i.e., an undirected, connected, acyclic graph).

The problems considered in Chapter 3 require the network of robots to come to a hold once the formation is achieved. However, motivated by Section 1.2, the objective is often to track a prescribed reference velocity. Applications include sweeping [22, 66] and deployment for coverage [25, 82]. Velocity tracking for port-Hamiltonian systems is achieved using generalized canonical transformations [45] to derive the error dynamics with respect to the reference velocity. By stabilizing these error dynamics, the network of wheeled robots tracks the reference velocity thereby enabling the movement of the whole network.

Another, more practical challenge in the field of formation keeping is to reach and maintain a desired formation shape despite input disturbances. Two types of disturbances are considered in this chapter: constant and harmonic (or sinusoidal) disturbances. In applications constant and harmonic disturbances correspond to respectively offsets in actuators and sensors, and acoustic disturbances and vibrations in rotating equipment [90]. Output regulation techniques [16, 29] and proportional-integral controllers with quantized information and time-varying topologies [120] have been studied in this context. For port-Hamiltonian systems input disturbance rejection is achieved using an adaptive internal-model-based controller [47, 48]. To the authors best knowledge existing literature does not consider disturbance rejection for a network of nonholonomic port-Hamiltonian systems.

The contributions of this chapter include the derivation of a dynamical model of the wheeled robot in the port-Hamiltonian framework and the design and analysis of three formation control problems. Standard formation control is achieved by assigning virtual couplings between the front ends of the robots. Second, a velocity tracking controller is proposed using generalized canonical transformations, which guarantees that all robots in the network track a constant reference velocity. A local (nonlinear) heading controller ensures that all robots also move along the same desired heading, under a natural assumption on the initial heading of the robots. Third, matched input disturbances are counteracted using an internal-model-based controller. The controller provides stability for constant disturbances and complete

rejection of harmonic disturbances.

### Utilization within the ROSE project

Wheeled robots do not comply with the legged robot design discussed in Section 1.2.2. However, there are two main reasons why studying wheeled robots aligns with the utilization within the ROSE project. First, the constraint on the wheel axle of the wheeled robot prevents the robot from moving sideward. The sideward motion of a legged robot is often less (energy-)efficient than the forward/backward motion. Chapter 3 incorporated this consideration by altering the friction coefficients accordingly, the present chapter does so by imposing a nonholonomic constraint on the sideward motion of the robots.

The second reason concerns the experimental validation of theoretical and simulation results. Since there is no prototype of the new robot design available at the time of writing (let alone a network of them), another type of robot is used for this purpose. The Discrete Technology & Production Automation (DTPA) laboratory of the University of Groningen provides several e-puck robots, which are used for the experimental results in this chapter.

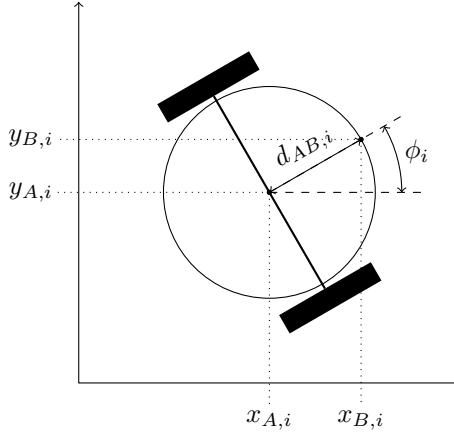
The next section continues with the dynamical modeling of the wheeled robot in the port-Hamiltonian framework.

## 4.2 Dynamical model of the wheeled robot

A schematic figure of wheeled robot  $i$  is depicted in Figure 4.1. Let the point  $(x_{A,i}, y_{A,i})$  denote the center of the wheel axle and  $(x_{B,i}, y_{B,i})$  a point in front of the robot. Here, the center of mass is assumed to be at the center of the wheel axle [67, 68]. The point  $(x_{B,i}, y_{B,i})$  is at a distance  $d_{AB,i} > 0$  from the axle center and is used in the controller design later on. Furthermore, let  $\phi_i$  denote the heading of the robot.

The dynamics of the wheeled robot are derived from the rigid body dynamics with a nonholonomic constraint on the wheel axle. Then, this constraint is solved for, thereby obtaining the dynamics on the constrained state space, which is used in the analysis of Sections 4.3, 4.4, and 4.5. In this section a superscript  $rb$  is added to differentiate between the rigid body dynamics and the dynamics on the constrained state space.

Let  $q_i \in \mathbb{R}^3$  denote the position, and let  $p_i^{rb} \in \mathbb{R}^3$  denote the corresponding momentum of robot  $i$ , where  $q_i = (x_{A,i}, y_{A,i}, \phi_i)^T$  and  $p_i^{rb} = (p_{x,i}^{rb}, p_{y,i}^{rb}, h_i^{rb})^T$  (see Figure 4.1). Note that the position vector  $q_i$  includes the heading of the robot, since it is modeled as a rigid body rather than a point mass (see Section 3.2). The momentum is related to the position by  $p_i^{rb} = M_i^{rb} \dot{q}_i$ , with constant mass-inertia



**Figure 4.1:** Schematic representation of wheeled robot  $i$

matrix  $M_i^{rb} = \text{diag}(m_i, m_i, I_{cm,i})$  where  $m_i$  denotes the robot's mass and  $I_{cm,i}$  denotes the moment of inertia around the center of mass  $(x_{A,i}, y_{A,i})$ . The rigid body dynamics are of the form (2.8) and are given by

$$\begin{pmatrix} \dot{q}_i \\ \dot{p}_i^{rb} \end{pmatrix} = \begin{pmatrix} 0 & I_3 \\ -I_3 & -D_i^{rb} \end{pmatrix} \begin{pmatrix} \frac{\partial H_i^{rb}}{\partial q_i} \\ \frac{\partial H_i^{rb}}{\partial p_i^{rb}} \end{pmatrix} + \begin{pmatrix} 0 \\ G_i(q_i) \end{pmatrix} u_i, \quad (4.1)$$

$$y_i = G^T(q_i) \frac{\partial H_i^{rb}}{\partial p_i^{rb}},$$

with input  $u_i = (u_{f,i}, u_{\phi,i})^T \in \mathbb{R}^2$ , output  $y_i = (y_{f,i}, y_{\phi,i})^T \in \mathbb{R}^2$ . Here  $u_{f,i}, u_{\phi,i}$  correspond to a forward force and torque acting at  $(x_{A,i}, y_{A,i})$  respectively, while  $y_{f,i}, y_{\phi,i}$  refer to the corresponding (angular) velocities. The dissipation matrix  $D_i^{rb}$  and input matrix  $G_i(q_i)$  are given by

$$D_i^{rb} = \begin{pmatrix} d_{f,i} & 0 & 0 \\ 0 & d_{f,i} & 0 \\ 0 & 0 & d_{\phi,i} \end{pmatrix}, \quad G_i(q_i) = \begin{pmatrix} \cos \phi_i & 0 \\ \sin \phi_i & 0 \\ 0 & 1 \end{pmatrix},$$

with  $d_{f,i}^{rb}$  and  $d_{\phi,i}^{rb}$  respectively the forward and angular friction coefficient of the rigid body. The Hamiltonian is the kinetic energy stored in the rigid body and is given by

$$H_i^{rb}(p_i^{rb}) = \frac{1}{2} p_i^{rb T} M_i^{rb -1} p_i^{rb}.$$

In addition to the rigid body dynamics (4.1), physics impose a *nonholonomic constraint* on the wheel axle, which is not able to move sideward. This nonholonomic

constraint is expressed in terms of the axle's center  $(x_{A,i}, y_{A,i})$  as

$$\sin \phi_i \dot{x}_{A,i} - \cos \phi_i \dot{y}_{A,i} = 0, \quad (4.2)$$

which may be rewritten as

$$\underbrace{\begin{pmatrix} \sin \phi_i & -\cos \phi_i & 0 \end{pmatrix}}_{A_i^T(q_i)} \frac{\partial H_i^{rb}}{\partial p_i^{rb}}(p_i^{rb}) = 0. \quad (4.3)$$

Since (4.3) is of the form (2.9) one may solve for (4.3) to obtain the *dynamics on the constrained state space* (Section 2.3.2). First note that  $\text{rank } A_i^T(q_i) = 1$  and define the following matrix

$$S_i(q_i) = \begin{pmatrix} \cos \phi_i & 0 \\ \sin \phi_i & 0 \\ 0 & 1 \end{pmatrix},$$

such that  $A_i^T(q_i)S_i(q_i) = 0$  and  $\text{rank } S_i(q_i) = 2$ . Now define the new momenta coordinates  $p_i = (p_{f,i}, h_i)^T, p_{s,i}$  as

$$\begin{aligned} p_i &:= S_i^T(q_i)p_i^{rb}, \\ p_{s,i} &:= A_i^T(q_i)p_i^{rb}, \end{aligned} \quad p_i \in \mathbb{R}^2, p_{s,i} \in \mathbb{R}. \quad (4.4)$$

Here  $p_{f,i}, p_{s,i}$  refer to the *forward* and *sideward* momenta of robot  $i$ . Clearly  $(q_i, p_i^{rb}) \mapsto (q_i, p_i, p_{s,i})$  defines a coordinate transformation, since the rows of  $S_i(q_i)$  are orthogonal to the rows of  $A_i(q_i)$ . In the new coordinates, the  $p_{s,i}$  dynamics may be eliminated to obtain the dynamics on the constrained state space, which are given by

$$\begin{aligned} \begin{pmatrix} \dot{q}_i \\ \dot{p}_i \end{pmatrix} &= \begin{pmatrix} 0 & S_i(q_i) \\ -S_i^T(q_i) & -D_i^r \end{pmatrix} \begin{pmatrix} \frac{\partial H_i^r}{\partial q_i} \\ \frac{\partial H_i^r}{\partial p_i} \end{pmatrix} + \begin{pmatrix} 0 \\ I_2 \end{pmatrix} u_i, \\ y_i &= \frac{\partial H_i^r}{\partial p_i}, \end{aligned} \quad (4.5)$$

with Hamiltonian  $H_i^r = \frac{1}{2}p_i^T M_i^{r-1} p_i$ , where  $M_i^r = \text{diag}(m_i, I_{cm,i})$ . The new dissipation matrix  $D_i$  is defined as

$$D_i^r = S_i^T(q_i) D_i^{rb} S_i(q_i) = \text{diag}(d_{f,i}, d_{\phi,i}).$$

*Remark 4.1* (Brockett's condition). The well-known paper by Brockett [13] provides a necessary condition under which systems can be asymptotically stabilized using continuous feedback. From Proposition 4.2.14 in [98] it follows that for (4.5) this

condition boils down to

$$\bigcup_{\{x_i ; \|x_i - x_{0,i}\| < \epsilon\}} \left( \underbrace{\text{im} \begin{pmatrix} 0 & S_i(q_i) \\ -S_i^T(q_i) & -D_i^r \end{pmatrix}}_{J_i(q_i) - R_i} + \underbrace{\text{im} \begin{pmatrix} 0 \\ I_2 \end{pmatrix}}_{g_i} \right) = \mathbb{R}^5, \quad (4.6)$$

with  $x_i = (q_i, p_i)$  and thus

$$\text{im} \begin{pmatrix} 0 & S_i(q_i) \\ -S_i^T(q_i) & -D_i^r \end{pmatrix} + \text{im} \begin{pmatrix} 0 \\ I_2 \end{pmatrix} \subset \text{im} \begin{pmatrix} S_i(q_i) \\ 0 \end{pmatrix} + \text{im} \begin{pmatrix} 0 \\ I_2 \end{pmatrix}.$$

By an appropriate change of configuration coordinates  $q_i \rightarrow \tilde{q}_i$ ,  $S(\tilde{q}_i)$  takes the form

$$S_i(\tilde{q}_i) = \begin{pmatrix} \bar{S}_i(\tilde{q}_i) \\ I_2 \end{pmatrix}$$

and vectors of the form  $\begin{pmatrix} * \\ 0 \end{pmatrix}$  can not be in the image of  $S_i(\tilde{q}_i)$  and hence not in  $\text{im}[J_i(q_i) - R_i] + \text{im } g_i$ . Thus (4.5) can not be asymptotically stabilized using continuous feedback. However, the formation control objective is defined in terms of the relative displacement between front ends of the robots. Therefore the formation control objective is not hindered by (4.6).

To compactly write the dynamics for all  $N$  robots denote the stacked vectors  $q = (q_1, \dots, q_N)^T$ ,  $p = (p_1, \dots, p_N)^T$ ,  $u = (u_1, \dots, u_N)^T$ ,  $y = (y_1, \dots, y_N)^T$  and the system matrices  $S(q) = \text{block.diag}(S_1(q_1), \dots, S_N(q_N))$ ,  $D^r = \text{block.diag}(D_1^r, \dots, D_N^r)$ ,  $M^r = \text{block.diag}(M_1^r, \dots, M_N^r)$ . The dynamics (4.5) for all  $i = 1, \dots, N$  robots is now compactly denoted by

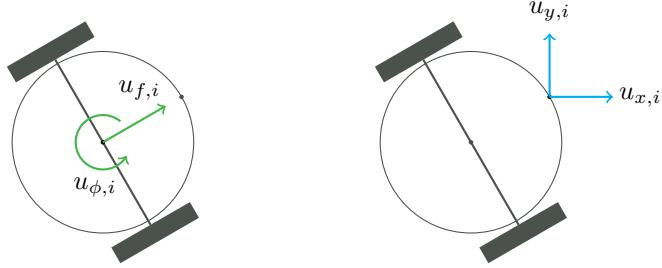
$$\begin{pmatrix} \dot{q} \\ \dot{p} \end{pmatrix} = \begin{pmatrix} 0 & S(q) \\ -S^T(q) & -D^r \end{pmatrix} \begin{pmatrix} \frac{\partial H^r}{\partial q} \\ \frac{\partial H^r}{\partial p} \end{pmatrix} + \begin{pmatrix} 0 \\ I_{2N} \end{pmatrix} \bar{u}, \quad (4.7)$$

$$y = \frac{\partial H^r}{\partial p},$$

with Hamiltonian  $H^r = \sum_{i=1}^N H_i^r(p_i) = \frac{1}{2} p^T M^r^{-1} p$ .

### Change of inputs

For control purposes, Sections 4.3, 4.4.1 and 4.5 require a change of inputs for the wheeled robots. By assigning the virtual couplings to the front end of the robot (point  $(x_{B,i}, y_{B,i})$  in Figure 4.1) rather than the center of the wheel axle (point  $(x_{A,i}, y_{A,i})$  in Figure 4.1), the formation control is not hindered by the constraint on the wheel axle. Furthermore, sensors and end effectors are usually positioned



**Figure 4.2:** Illustration of the original input  $u_i = (u_{f,i}, u_{\phi,i})^T$  (left) and the new input  $\bar{u}_i = (u_{x,i}, u_{y,i})^T$  (right).

at the front end of the robot.

For each robot  $i$  its input  $u_i = (u_{f,i}, u_{\phi,i})^T$  is transformed into a new input  $\bar{u}_i = (u_{x,i}, u_{y,i})^T$  where  $u_{x,i}$  ( $u_{y,i}$ ) denotes a force along the  $x$  direction ( $y$  direction) acting on the point  $(x_{B,i}, y_{B,i})$  (see Figure 4.2). The output  $y_i$  is transformed accordingly into the new output  $\bar{y}_i = (\bar{y}_{x,i}, \bar{y}_{y,i})^T$ . The transformation  $(u_i, y_i) \mapsto (\bar{u}_i, \bar{y}_i)$  is given by [10]

$$\begin{aligned} u_i &= \bar{G}_i(\phi_i)\bar{u}_i, \\ \bar{y}_i &= \bar{G}_i^T(\phi_i)y_i, \end{aligned} \tag{4.8}$$

where

$$\bar{G}_i(\phi_i) = \begin{pmatrix} \cos \phi_i & \sin \phi_i \\ -d_{AB,i} \sin \phi_i & d_{AB,i} \cos \phi_i \end{pmatrix}.$$

Note that controlling the point  $(x_{B,i}, y_{B,i})$  is also quite natural from a practical point of view, since on-board sensors and end-effectors are usually positioned at the front end of the robot. The dynamics of robot  $i$  with new input  $\bar{u}_i$  and new output  $\bar{y}_i$  follow from (4.5) and (4.8) and are given by

$$\begin{aligned} \begin{pmatrix} \dot{q}_i \\ \dot{p}_i \end{pmatrix} &= \begin{pmatrix} 0 & S_i(q_i) \\ -S_i^T(q_i) & -D_i^r \end{pmatrix} \begin{pmatrix} \frac{\partial H_i^r}{\partial q_i} \\ \frac{\partial H_i^r}{\partial p_i} \end{pmatrix} + \begin{pmatrix} 0 \\ \bar{G}_i(q_i) \end{pmatrix} \bar{u}_i, \\ \bar{y}_i &= \bar{G}_i^T(q_i) \frac{\partial H_i^r}{\partial p_i}, \end{aligned} \tag{4.9}$$

where the Hamiltonian  $H_i^r$  remains the same as in (4.5). To compactly write the dynamics of  $N$  robots with new input  $\bar{u} = (\bar{u}_1, \dots, \bar{u}_N)^T$  and output  $\bar{y} = (\bar{y}_1, \dots, \bar{y}_N)^T$  define  $\bar{G}(q) = \text{block.diag}(\bar{G}_1(q_1), \dots, \bar{G}_N(q_N))$ , such that the dynamics for  $N$

robots are given by

$$\begin{pmatrix} \dot{q} \\ \dot{p} \end{pmatrix} = \begin{pmatrix} 0 & S(q) \\ -S^T(q) & -D^r \end{pmatrix} \begin{pmatrix} \frac{\partial H^r}{\partial q} \\ \frac{\partial H^r}{\partial p} \end{pmatrix} + \begin{pmatrix} 0 \\ \bar{G}(q) \end{pmatrix} \bar{u}, \quad (4.10)$$

$$\bar{y} = \bar{G}^T(q) \frac{\partial H^r}{\partial p},$$

with Hamiltonian  $H^r = \sum_{i=1}^N H_i^r(p_i) = \frac{1}{2} p^T M^r^{-1} p$ .

### 4.3 Formation control

The formation control problem was already introduced in Section 3.4. The main objective in formation control is to achieve a prescribed geometrical shape for the network of agents. The main difference with the results presented in Section 3.4 is the type of systems considered. In Section 3.4 formation control for a network of fully actuated agents in the presence of ideal Coulomb friction is considered, while here the focus is on a network of nonholonomic wheeled robots.

Formation control for a network of wheeled robots is achieved by assigning virtual couplings between the *front ends* (point  $(x_{B,i}, y_{B,i})$  in Figure 4.1) of the robots. This differs from the previous chapter, where the springs were assigned between the *centers of mass* of the agents. This change of point of action is required to deal with the nonholonomic constraint on the wheel axle of the robot.

To formally define the formation control objective, let  $z_j, z_j^* \in \mathbb{R}^2$  denote respectively the relative displacement between two wheeled robots and the desired relative displacement. For  $E$  virtual couplings define the collocated vectors  $z = (z_1, \dots, z_E)^T$  and  $z^* = (z_1^*, \dots, z_E^*)^T$ . Then, the *formation control objective* is formally defined as

$$\begin{cases} p \rightarrow 0, \\ z \rightarrow z^*, \end{cases} \quad \text{as } t \rightarrow \infty. \quad (4.11)$$

Note that the formation control objective (4.11) is exactly the same as the objective stated in (3.3). The results presented here to achieve (4.11) are published in [113, 115].

#### 4.3.1 Formation control of wheeled robots using virtual springs

To achieve formation control the  $N$  wheeled robots are interconnected using  $E$  virtual couplings (i.e., virtual springs and dampers) in a similar way as Chapter 3. A tree graph (i.e., an undirected connected acyclic graph) is used to describe the interconnection topology: the nodes of the graph correspond to robots, while the edges of the graph correspond to virtual couplings.

Let  $z_j \in \mathbb{R}^2$  denote the length along the  $x$  and  $y$  direction of the virtual coupling. The input to the control system  $w_j \in \mathbb{R}^2$  is a velocity, while the corresponding output  $\tau_j \in \mathbb{R}^2$  is a force. Furthermore let  $D_j^c \in \mathbb{R}^{2 \times 2}$  denote the corresponding virtual dissipation matrix, defined as  $D_j^c = \text{diag}(d_{x,j}^c, d_{y,j}^c)$ . The dynamics of virtual coupling  $j$  are given by [39, 99]

$$\begin{aligned}\dot{z}_j &= w_j, \\ \tau_j &= \frac{\partial H_j^c}{\partial z_j}(z_j) + D_j^c w_j,\end{aligned}\tag{4.12}$$

with Hamiltonian  $H_j^c(z_j)$ . The Hamiltonian  $H_j^c(z_j)$  equals the potential energy stored in virtual spring  $j$  which is given by

$$H_j^c(z_j) = \frac{1}{2} (z_j - z_j^*)^T K_j^c (z_j - z_j^*),$$

where  $z_j^* = (z_{x,j}^*, z_{y,j}^*)^T$  denotes the nominal spring length and  $K_j^c = \text{diag}(k_{x,j}^c, k_{y,j}^c)$  denotes the virtual spring constant matrix. Defining the springs in this way corresponds to position-based control in terms of [3], which implies that not only the inter-robot displacement, but also the inter-robot heading is controlled.

To compactly write the dynamics of  $E$  virtual couplings of the form (4.12), define the collocated vectors  $z = (z_1, \dots, z_E)^T$ ,  $z^* = (z_1^*, \dots, z_E^*)^T$ ,  $w = (w_1, \dots, w_E)^T$ ,  $\tau = (\tau_1, \dots, \tau_E)^T$ , and system matrices  $K^c = \text{block.diag}(K_1^c, \dots, K_E^c)$ ,  $D^c = \text{block.diag}(D_1^c, \dots, D_E^c)$ . Then the dynamics of  $E$  virtual couplings of the form (4.12) are given by

$$\begin{aligned}\dot{z} &= w, \\ \tau &= \frac{\partial H^c}{\partial z}(z) + D^c w,\end{aligned}\tag{4.13}$$

with Hamiltonian  $H^c(z) = \sum_{j=1}^E H_j^c(z_j) = \frac{1}{2} (z - z^*)^T K^c (z - z^*)$ .

The incidence matrix  $B$  of the graph describes which robots (nodes) are interconnected by virtual couplings (edges). The corresponding coupling of robots on the nodes and virtual couplings at the edges is given by [3, 100]

$$\begin{cases} \bar{u} = -(B \otimes I_2) \tau, \\ w = (B^T \otimes I_2)^T \bar{y}. \end{cases}\tag{4.14}$$

The closed-loop network dynamics are obtained by eliminating the interconnection constraint (4.14) using the wheeled robot dynamics (4.10) and virtual

coupling dynamics (4.13) and are given by

$$\begin{pmatrix} \dot{q} \\ \dot{p} \\ \dot{z} \end{pmatrix} = \begin{pmatrix} 0 & S(q) & 0 \\ -S^T(q) & -(D^r + BD^cB^T) & -\bar{G}(q)(B \otimes I_2) \\ 0 & (B \otimes I_2)^T \bar{G}^T(q) & 0 \end{pmatrix} \begin{pmatrix} \frac{\partial H}{\partial q} \\ \frac{\partial H}{\partial p} \\ \frac{\partial H}{\partial z} \end{pmatrix}, \quad (4.15)$$

with closed-loop Hamiltonian  $H(p, z) = H^r(p) + H^c(z)$ .

The main result of this section is now presented in the following theorem.

**Theorem 4.2.** *Interconnect wheeled robots (4.10) with virtual couplings (4.13) via coupling (4.14) using a tree graph topology. Then, the solutions to the closed-loop system (4.15) converge to  $p = 0, z = z^*$ , thereby achieving the control objectives (4.11).*

*Proof.* Take the closed-loop Hamiltonian  $H(p, z)$  as a candidate Lyapunov function. Since  $H(p, z)$  is quadratic in  $p$  and  $z$ , it follows that  $H(x) \geq 0$  for all  $p \in \mathbb{R}^{2N}, z \in \mathbb{R}^{2E}$ . The time derivative  $\dot{H}(p, z)$  follows from (4.15) and is given by

$$\dot{H}(p, z) = -\frac{\partial^T H}{\partial p}(p)(D^r + BD^cB^T)\frac{\partial H}{\partial p}(p) \leq 0.$$

Invoking LaSalle's invariance principle (Theorem 2.4) gives that (4.15) converges to the largest invariant set where  $\dot{H}(p, z) = 0$ . On this set  $\frac{\partial H}{\partial p}(p) = 0$ , which implies that  $p = 0$  and  $\dot{p} = 0$ . Substituting  $p = 0, \dot{p} = 0$  into (4.15) and rearranging the terms along the  $x$  and  $y$  direction gives

$$-\cos \phi B K_x^c (z_x - z_x^*) - \sin \phi B K_y^c (z_y - z_y^*) = 0, \quad (4.16)$$

$$D_{AB} \sin \phi B K_x^c (z_x - z_x^*) - D_{AB} \cos \phi B K_y^c (z_y - z_y^*) = 0, \quad (4.17)$$

where  $z_x = (z_{x,1}, \dots, z_{x,E})^T$ ,  $z_y = (z_{y,1}, \dots, z_{y,E})^T$  and  $K_x^c = \text{diag}(k_{x,1}^c, \dots, k_{x,E}^c)$ ,  $K_y^c = \text{diag}(k_{y,1}^c, \dots, k_{y,E}^c)$ ,  $\sin \phi = \text{diag}(\sin \phi_1, \dots, \sin \phi_N)$ ,  $\cos \phi = \text{diag}(\cos \phi_1, \dots, \cos \phi_N)$ ,  $D_{AB} = \text{diag}(d_{AB,1}, \dots, d_{AB,N})$ .

Multiplying (4.16) from the left by  $D_{AB} \cos \phi$ , (4.17) by  $\sin \phi$ , and summing the result gives

$$D_{AB} B K_x^c (z_x - z_x^*) = 0. \quad (4.18)$$

Since  $D_{AB}$  is a positive definite diagonal matrix, it follows from (4.18) that  $K_x^c(z_x - z_x^*) \in \ker B$ . Noting that  $K_x^c$  is also a positive definite diagonal matrix and  $\ker B = 0$  it follows from (4.18)  $z_x = z_x^*$ . In a similar fashion, by multiplying (4.16) from the left by  $D_{AB} \sin \phi$ , (4.17) by  $\cos \phi$ , and summing the result it follows that  $z_y = z_y^*$ . This completes the proof.  $\square$

*Remark 4.3 (Cyclic graphs).* In this approach, the graph topology is a design freedom and considering only acyclic graphs is not restrictive. Cyclic graphs might

give rise to undesired equilibria [9]. A work around to prevent undesired equilibria is to have an a priori condition of the form  $K_x(z_x - z_x^*) \in \ker B, K_y(z_y - z_y^*) \in \ker B$  (see [3, 100]). In contrast, Chapter 5 does consider cyclic graphs for modeling the interaction amongst satellites on a (circular) orbit.

*Remark 4.4* (Control law formation control). The resulting control law  $u$  for the robots is easily obtained from (4.8) and (4.14) as

$$\begin{aligned} u &= \bar{G}(q)\bar{u} \\ &= -\bar{G}(q)(B \otimes I_2) \frac{\partial H}{\partial z}(z) - \bar{G}(q)(B \otimes I_2) D^c (B \otimes I_2)^T \frac{\partial H}{\partial p}(p) \\ &= -\bar{G}(q) \underbrace{(B \otimes I_2) K^c(z - z^*)}_{\text{virtual spring force}} - \bar{G}(q) \underbrace{(B \otimes I_2) D^c (B \otimes I_2)^T M^{r-1} p}_{\text{virtual damping force}}. \end{aligned} \quad (4.19)$$

Due the use of virtual couplings in the port-Hamiltonian framework, the two terms in (4.19) have a clear physical interpretation. The virtual spring forces along the  $x$  and  $y$  direction ensures that the formation control objectives (4.11) are achieved, while the virtual damping forces can be used to shape the transient response. The input matrix  $\bar{G}(q)$  transforms these virtual spring and damper forces into inputs for the wheeled robots (see equation (4.8)).

Furthermore, note that (4.19) is a distributed control law. Each agent only requires measurements on the relative displacement  $z$  and relative velocity  $\dot{z}$  with respect to its two neighbors, which are characterized by the graph topology.

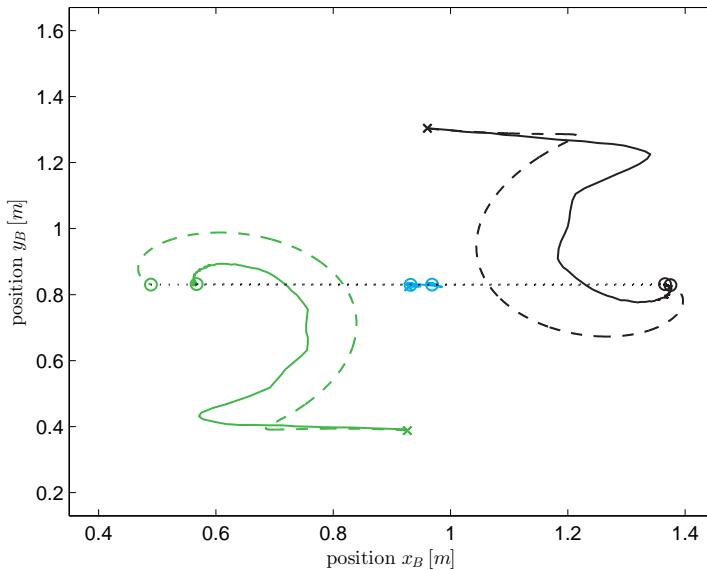
Simulation and experimental results are presented in the next section to illustrate the effectiveness of Theorem 4.2.

### 4.3.2 Simulation and experimental results

Consider a network of  $N = 3$  wheeled robots of the form (4.5), with model parameters  $m_i = 0.167 \text{ kg}$ ,  $I_{cm,i} = 9.69 \cdot 10^{-5} \text{ kg m}^2$ ,  $d_{f,i} = 2 \text{ kg/s}$ ,  $d_{\phi,i} = 0.2 \text{ kg m}^2/\text{s}$ ,  $d_{AB,i} = 0.06 \text{ m}$  for  $i = 1, 2, 3$ . The model parameters are chosen in accordance with the e-puck wheeled robot (Figure C.1) which are given in Table C.1. The three robots are interconnected using the incidence matrix

$$B = \begin{pmatrix} -1 & 0 \\ +1 & -1 \\ 0 & +1 \end{pmatrix},$$

which is associated to a so-called *path graph*. Since each robot is strictly passive (i.e.,  $d_{f,i} > 0, d_{\phi,i} > 0$ ) no virtual dampers are required for asymptotic stability ( $d_{x,j}^c = d_{y,j}^c = 0 \text{ kg/s}$  for  $j = 1, 2$ ). Hence, each virtual coupling only corresponds to a virtual spring with spring parameters  $z_{x,j}^* = 0.4 \text{ m}$ ,  $z_{y,j}^* = 0 \text{ m}$ ,  $k_{x,j}^c = k_{y,j}^c =$



**Figure 4.3:** Trajectories of the e-puck wheeled robots for the simulation (dashed) and the experiment (solid). The crosses ( $\times$ ) and circles ( $\circ$ ) denote the initial and final positions respectively.

$2 \text{ kg}/\text{s}^2$ , for  $j = 1, 2$ . This choice for the graph topology and  $z_x^*, z_y^*$  yields a *line formation*. Note that this formation might look similar to the experimental results in Chapter 3 for formation control and deployment. However, other formation shapes (e.g. star and zig-zag formation) are also possible within the current setup and here the line formation was chosen because of its simplicity.

The simulations are performed using MATLAB and Simulink, while the experiments make use of the experimental setup discussed in Appendix A. The experimental setup consists of  $2.6 \times 2.0 \text{ m}$  table with an overhead camera for localization. Each robot is identified and localized using a data-matrix, attached on top of the e-puck (see Figure C.1 (right)). A vision algorithm runs in parallel to MATLAB and provides the ID, position, and heading of each e-puck. MATLAB calculates the corresponding control inputs (4.19) and sends them to the e-pucks via a Bluetooth connection.

Both the simulation and the experiment were run for  $t = 50 \text{ s}$ , starting from the same initial conditions:  $x_B(0) = (0.87, 0.99, 1.02) \text{ m}$ ,  $y_B(0) = (0.39, 0.82, 1.30) \text{ m}$ ,  $\phi(0) = (6.23, 3.05, 3.02) \text{ rad}$ ,  $p_f(0) = (0, 0, 0) \text{ kg m/s}$ ,  $h(0) = (0, 0, 0) \text{ kg m}^2/\text{s}$ . The results are shown in Figures 4.3, 4.4, and 4.5.

Figure 4.3 shows the trajectories of the e-pucks for the simulation (dashed) and the experiment (solid). The dotted black lines represent the virtual springs at the final position. Even though there is a difference between the trajectories, the

simulation and the experiment show a similar trend. Consider for example the left e-puck (blue): initially driving straight, the e-puck takes a sharp turn to the right followed by a circular path towards its final position. The differences between the trajectories are due to localization errors of the vision algorithm, model parameter uncertainty, and a difference in the actuators of the model and the e-puck (the robot dynamics (4.5) do not consider the differential drive of the e-puck).

To clarify that objectives (4.11) are achieved, Figures 4.4 and 4.5 show the time evolution of the forward velocity  $v_f$  and the relative displacement  $z_x$  for the simulation and experiment respectively. The angular velocity  $v_\phi$  and relative displacement  $z_y$  follow a similar trend and are given in Appendix D. Both figures clearly show that  $v_f \rightarrow 0, z_x \rightarrow z^*$  as  $t \rightarrow \infty$ , thereby achieving (4.11).

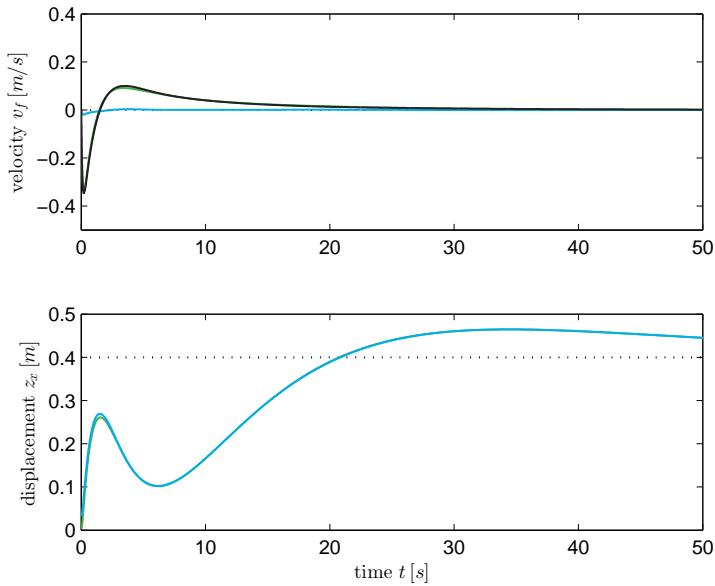
## 4.4 Formation control with velocity tracking

The goal of this approach is to make the wheeled robot (4.5) track a reference velocity along a prescribed heading angle, while converging to a desired formation shape. In addition to the formation control design in Section 4.3, this section consider an additional *velocity tracking* and *heading controller*. The velocity tracking controller is a local controller which makes each wheeled robot track a predefined forward velocity  $v_i^*$  and builds upon generalized canonical transformations [45]. To ensure that the movement of the robots is along the desired heading  $\phi_i^*$ , an additional heading controller is proposed. In this work the desired heading is assumed to be along the  $y$ -axis (see Figure 4.1), such that  $\phi_i^* = \frac{\pi}{2}$  (see Figure 4.6).

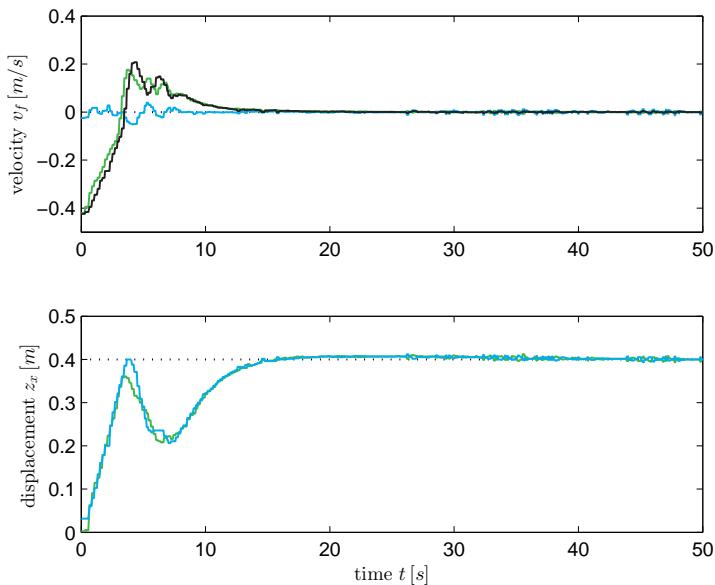
To achieve a desired formation shape virtual couplings are assigned between the agents. Each virtual coupling consists of a virtual spring, which steers robots to the desired formation, and a virtual damper, which may be used to shape the transient response. The interconnection topology (i.e., which robots are interconnected to each other by a virtual coupling) is modeled by a tree graph (i.e., an undirected connected acyclic graph). Each edge is labeled with a positive and a negative end arbitrarily. Consider two robots  $i$  and  $j$  which are interconnected using virtual coupling  $k$ , where robot  $j$  is at the positive end of the corresponding edge. The relative displacement  $z_k$  of robot  $j$  with respect to robot  $i$  is defined as  $z_k = (x_{B,j}, y_{B,j})^T - (x_{B,i}, y_{B,i})^T$ . Let  $z = (z_1, \dots, z_E)^T$  and let  $z^* = (z_1^*, \dots, z_E^*)^T$  denote the desired relative displacement.

The objectives of formation control with velocity tracking are now formally stated in terms of (4.7) as

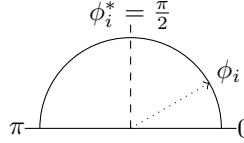
$$\begin{cases} \phi \rightarrow \phi^*, \\ p \rightarrow M^r v^*, & \text{as } t \rightarrow \infty, \\ z \rightarrow z^*, \end{cases} \quad (4.20)$$



**Figure 4.4:** Time evolution of the forward velocity  $v_f$  and relative position  $z_x$  for formation controller (4.19) (simulation). The dotted lines show the reference values.



**Figure 4.5:** Time evolution of the forward velocity  $v_f$  and relative position  $z_x$  for formation controller (4.19) (experiment). The dotted lines show the reference values.



**Figure 4.6:** The robot heading  $\phi_i$  is defined with respect to the  $x$ -axis, while the desired heading  $\phi_i^* = \frac{\pi}{2}$  is along the  $y$ -axis (see also Figure 4.1).

where  $v^*$  denotes the reference velocity to be defined later on. There are two differences with respect to the formation control objectives (4.11). In (4.11) the momentum  $p$  should converge zero, while in (4.20) the momentum  $p$  should converge to a nonzero constant  $M^r v^*$ . Furthermore, (4.11) imposes no requirement on the robots' heading, while in (4.20) the heading  $\phi$  of each robot should converge to a desired value  $\phi^*$ .

The results presented here to achieve (4.20) are published in [113].

#### 4.4.1 Velocity tracking control and formation control using virtual couplings

The heading and velocity tracking controller are local controllers. Therefore these two controllers are developed for a single robot  $i$ . The formation controller is a distributed controller, where (some) robots exchange local information. Section 4.4.1 presents the design and analysis of the whole network of  $N$  robots.

The control input  $u$  designed in this section consists of three parts (i.e.,  $u = u_1 + u_2 + u_3$ ). The three control input signals  $u_1, u_2, u_3$  correspond to respectively the heading controller, velocity tracking controller, and formation controller which are designed below. Since the heading and velocity tracking controllers are local controllers, they provide control inputs  $u_{1i}$  and  $u_{2i}$  respectively.

##### Heading control

Each robot is required to track a forward reference velocity along the  $y$ -axis (see Figure 4.1). The desired heading for each robot is therefore set at  $\phi_i^* = \frac{\pi}{2}$  (see Figure 4.6). The heading controller assigns a nonlinear spring to the heading of each robot, which guarantees convergence to the desired heading under the following natural assumption on the initial heading:

**Assumption 4.5.** The initial heading  $\phi_i(0)$  of robot  $i$  is contained in the interval  $\phi_i(0) \in (0, \pi)$ .

Let  $\hat{\phi}_i = \phi_i - \phi_i^*$  denote the error state of the heading controller, then the

heading control dynamics are given by

$$\begin{aligned}\dot{\hat{\phi}}_i &= u_i^h, \\ y_i^h &= \frac{\partial H_i^h}{\partial \hat{\phi}_i},\end{aligned}\tag{4.21}$$

with  $u_i^h$  the input and  $y_i^h$  the output of the control system. Hamiltonian  $H_i^h(\hat{\phi}_i)$  is defined as

$$H_i^h(\hat{\phi}_i) := -k_i^\phi \ln |\cos \hat{\phi}_i|,$$

with  $k_i^\phi > 0$  the controller gain. This nonlinear Hamiltonian guarantees that under Assumption 4.5 it follows that  $\phi_i(t) \in (0, \pi)$  for all  $t$  (see further on for the proof). On the interval  $\hat{\phi}_i \in (-\pi/2, \pi/2)$  it is easy to verify that  $H_i^h(\hat{\phi}_i) \geq 0$  since  $|\cos \hat{\phi}_i| \leq 1$  and hence  $\ln |\cos \hat{\phi}_i| < 0$ . Note that the partial derivative of  $H_i^h$  with respect to  $\hat{\phi}_i$  is simply  $\frac{\partial H_i^h}{\partial \hat{\phi}_i} = k_i^\phi \tan \hat{\phi}_i$ .

The coupling of robot (4.5) to the heading controller (4.21) is given by

$$\begin{cases} u_{1i} = -L_i y_i^h, \\ u_i^h = L_i^T y_i, \end{cases}\tag{4.22}$$

where  $L_i^T = \begin{pmatrix} 0 & 1 \end{pmatrix}$ . Before stating the proposition on the heading controller, first consider the following natural assumption on the initial heading  $\phi_i(0)$ .

*Remark 4.6* (Relaxation of Assumption 4.5). The interval  $(0, \pi)$  in Assumption 4.5 is used for simplicity of notation. This interval may be ‘stretched’ to e.g. the interval  $(-\pi/2, 3\pi/2)$  resulting however in a more complicated expression.

The following proposition provides the result for the heading controller.

**Proposition 4.7.** *Consider the system (4.5). Using  $u_i = u_{1i}$ , with  $u_{1i}$  defined in (4.21)-(4.22), the system (4.5) converges to  $\phi_i = \phi_i^*, p_i = 0$ .*

*Proof.* Let  $\hat{q}_i = (x_{A,i}, y_{A,i}, \hat{\phi}_i)^T$ , then setting  $u_i = u_{1i}$  the closed-loop dynamics follow from (4.5), (4.21), (4.22)

$$\begin{pmatrix} \dot{\hat{q}}_i \\ \dot{p}_i \end{pmatrix} = \begin{pmatrix} 0 & \hat{S}_i(\hat{q}_i) \\ -\hat{S}_i^T(\hat{q}_i) & -D_i^r \end{pmatrix} \begin{pmatrix} \frac{\partial \hat{H}_i^r}{\partial \hat{q}_i} \\ \frac{\partial \hat{H}_i^r}{\partial p_i} \end{pmatrix},\tag{4.23}$$

with

$$\hat{S}_i(\hat{q}_i) = \begin{pmatrix} \cos(\hat{\phi}_i + \phi_i^*) & 0 \\ \sin(\hat{\phi}_i + \phi_i^*) & 0 \\ 0 & 1 \end{pmatrix},$$

and Hamiltonian

$$\begin{aligned}\hat{H}_i^r(\hat{\phi}_i, p_i) &= H_i^h(\hat{\phi}_i) + H_i^r(p_i) \\ &= -k_i^\phi \ln |\cos \hat{\phi}_i| + \frac{1}{2} p_i^T (M_i^r)^{-1} p_i.\end{aligned}$$

Take the Hamiltonian  $\hat{H}_i^r(\hat{\phi}_i, p_i)$  as a Lyapunov candidate function and calculate its time derivative as

$$\dot{\hat{H}}_i^r = -\frac{\partial^T \hat{H}_i^r}{\partial p_i} D_i^r \frac{\partial \hat{H}_i^r}{\partial p_i} \leq 0.$$

Invoking LaSalle's invariance principle one obtains from  $\dot{\hat{H}}_i^r = 0$  that  $\frac{\partial \hat{H}_i^r}{\partial p_i} = 0$  and thus  $p_i = 0$  and  $\dot{p}_i = 0$ . Substituting into (4.23) yields

$$-\hat{S}_i^T(\hat{q}_i) \frac{\partial \hat{H}_i^r}{\partial \hat{q}_i} = \begin{pmatrix} \cos(\hat{\phi}_i + \phi_i^*) & \sin(\hat{\phi}_i + \phi_i^*) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \frac{\partial \hat{H}_i^r}{\partial x_{A,i}} \\ \frac{\partial \hat{H}_i^r}{\partial y_{A,i}} \\ \frac{\partial \hat{H}_i^r}{\partial \hat{\phi}_i} \end{pmatrix} = 0,$$

which implies that  $\frac{\partial \hat{H}_i^r}{\partial \hat{\phi}_i} = 0$ , implying that  $k_i^\phi \tan \hat{\phi}_i = 0$ .

From Assumption 4.5 and  $\phi_i^* = \frac{\pi}{2}$  it follows that  $\hat{\phi}_i(0) \in (-\frac{\pi}{2}, \frac{\pi}{2})$ . Since  $\hat{H}_i^r(t) \leq \hat{H}_i^r(0) \leq M < \infty$  for all  $t$  and  $\hat{H}_i^r \rightarrow \infty$  as  $\hat{\phi}_i \rightarrow \pm \frac{\pi}{2}$ , it follows that  $\hat{\phi}_i(t) \in (-\frac{\pi}{2}, \frac{\pi}{2})$  for all  $t$ . Then, it immediately follows that  $\hat{\phi}_i = 0$  since  $\hat{\phi}_i = 0$  is the only value on the interval  $\hat{\phi}_i(t) \in (-\frac{\pi}{2}, \frac{\pi}{2})$  for which  $k_i^\phi \tan \hat{\phi}_i = 0$ . Since  $\hat{\phi}_i = 0$  implies  $\phi_i = \phi_i^*$ , this completes the proof.  $\square$

The next subsection continues with the velocity tracking control.

### Velocity tracking control

The design of the velocity tracking controller is based on the use of generalized canonical coordinate transformations [45]. The main idea is to first derive the error dynamics w.r.t. the reference velocity and then to stabilize these dynamics. In this section, a constant forward velocity is considered, where the wheeled robots move along a straight line. Let  $v_i^* \in \mathbb{R}^2$  denote the desired velocity and define the error variables

$$\begin{pmatrix} \bar{q}_i \\ \bar{p}_i \end{pmatrix} = \begin{pmatrix} \hat{q}_i - \hat{S}_i(\hat{q}_i)v_i^*t \\ p_i - M_i^r v_i^* \end{pmatrix} =: \Phi_i(q_i, p_i, t), \quad (4.24)$$

where  $v_i^* = (v_{f,i}^*, 0)^T$  with forward reference velocity  $v_{f,i}^* \in \mathbb{R}$ . Note that the heading  $\hat{\phi}_i$  and angular momentum  $h_i$  are not affected by (4.24) since  $h_i^* = 0$  (i.e.,

$\bar{\phi}_i = \hat{\phi}_i$ ,  $\bar{h}_i = h_i$ ). The corresponding error Hamiltonian is defined as

$$\begin{aligned}\bar{H}_i^r(\bar{q}_i, \bar{p}_i) &:= -k_i^\phi \ln |\cos \bar{\phi}_i| + \frac{1}{2} \bar{p}_i^T (M_i^r)^{-1} \bar{p}_i \\ &= \underbrace{-k_i^\phi \ln |\cos \hat{\phi}_i| + \frac{1}{2} p_i^T (M_i^r)^{-1} p_i}_{\bar{H}_i^r(\hat{\phi}_i, p_i)} \underbrace{-p_i^T v_i^* + \frac{1}{2} v_i^{*T} M_i^r v_i^*}_{U_i(p_i)},\end{aligned}\quad (4.25)$$

where  $U_i(p_i)$  is a fictitious potential. Now consider the following proposition.

**Proposition 4.8.** Define  $u_{2i} = -\beta - D_i^t(y_i + \alpha_i)$  with  $\beta_i(p_i) = D_i^r v_i^*$ ,  $\alpha_i = \frac{\partial U_i}{\partial p_i} = -(v_{f,i}^*, 0)^T$ , and  $D_i^t \geq I_2 > 0 \in \mathbb{R}^{2 \times 2}$ . Using  $u_i = u_{1i} + u_{2i}$ , the system (4.5) converges to  $\phi_i = \phi_i^*$ ,  $p_i = M_i^r v_i^*$ .

*Proof.* To verify whether  $\Phi_i$  yields a generalized canonical transformation, substitute  $\Phi_i$ ,  $U_i(p_i)$  and  $\beta_i = D_i v_i^*$  into PDE (2.13). It is easily checked that the PDE holds and the error dynamics of (4.23) w.r.t. the reference velocity are given by

$$\begin{pmatrix} \dot{\bar{q}}_i \\ \dot{\bar{p}}_i \end{pmatrix} = \begin{pmatrix} 0 & \bar{S}_i(\bar{q}_i) \\ -\bar{S}_i^T(\bar{q}_i) & -(D_i^r + D_i^t) \end{pmatrix} \begin{pmatrix} \frac{\partial \bar{H}_i^r}{\partial \bar{q}_i} \\ \frac{\partial \bar{H}_i^r}{\partial \bar{p}_i} \end{pmatrix}, \quad (4.26)$$

$$\bar{y}_i = \frac{\partial \bar{H}_i^r}{\partial \bar{p}_i},$$

where

$$\bar{S}_i(\bar{q}_i) = \begin{pmatrix} \cos(\bar{\phi}_i + \phi_i^*) & 0 \\ \sin(\bar{\phi}_i + \phi_i^*) & 0 \\ 0 & 1 \end{pmatrix}.$$

Note that matrix  $\bar{S}_i(\bar{q}_i)$  follows directly from (2.14). Take the error Hamiltonian  $\bar{H}_i^r(\bar{q}_i, \bar{p}_i)$  as a candidate Lyapunov function. It is easily verified that  $\bar{H}_i^r(\bar{q}_i, \bar{p}_i) \geq 0$  and that the time derivative is given by

$$\dot{\bar{H}}_i^r = -\frac{\partial^T \bar{H}_i^r}{\partial \bar{p}_i} (D_i^r + D_i^t) \frac{\partial^T \bar{H}_i^r}{\partial \bar{p}_i}.$$

Invoking LaSalle's invariance principle (Theorem 2.4) gives that (4.26) converges to the largest invariant set where  $\dot{\bar{H}}_i^r = 0$ . On this set  $\frac{\partial^T \bar{H}_i^r}{\partial \bar{p}_i} = 0$ , which implies that  $\bar{p}_i = 0$  and  $\dot{\bar{p}}_i = 0$ . Substituting  $\bar{p}_i = 0$ ,  $\dot{\bar{p}}_i = 0$  into (4.26) gives

$$-\bar{S}_i^T(\bar{q}_i) \frac{\partial \bar{H}_i^r}{\partial \bar{q}_i} = 0.$$

Using similar arguments as in the proof of Proposition 4.7 it immediately follows that  $\bar{\phi} = 0$ , thereby completing the proof.  $\square$

*Remark 4.9* (Time-varying velocities). Instead of a constant forward velocity  $v_i^*$  also time-varying forward velocities  $v_i^*(t)$  can be incorporated in the approach. Then transformation (4.24) needs to be redefined accordingly and the state feedback term  $\beta_i(p_i)$  requires an additional term  $M_i^r v_i^*(t)$ , corresponding to the time derivative of the reference velocity  $v_i^*(t)$ . The details are left to the reader.

Finally, the next section deals with the formation keeping controller.

### Formation control using virtual couplings

The third controller is a distributed formation controller, where robots exchange local information. This controller assigns virtual couplings between the front ends of the wheeled robots (point  $(x_{B,i}, y_{B,i})$  in Figure 4.1) [115]. Each virtual coupling consists of a virtual spring and a virtual damper. The virtual coupling dynamics are given in (4.13).

The way in which robots are interconnected by a virtual coupling is modeled by a tree graph  $\mathcal{G}(\mathcal{V}, \mathcal{E})$ . The node-set  $\mathcal{V}$  corresponds to  $N$  wheeled robots, while the edge-set  $\mathcal{E}$  corresponds to  $E$  virtual couplings. Let  $B$  denote the incidence matrix associated to  $\mathcal{G}(\mathcal{V}, \mathcal{E})$  then the coupling to the formation controller is given by [3, 100]

$$\begin{cases} u_3 = -\bar{G}(\phi)(B \otimes I_2)\tau, \\ w = (B^T \otimes I_2)\bar{G}^T(\phi)y, \end{cases} \quad (4.27)$$

with  $u_3 = (u_{31}, \dots, u_{3N})^T$ ,  $y = (y_1, \dots, y_N)^T$ ,  $w = (w_1, \dots, w_E)^T$ ,  $\tau = (\tau_1, \dots, \tau_E)^T$ . The only difference with respect to (4.14) is matrix  $G(\phi)$ , which assigns the virtual couplings to the front ends of robots in a similar way as (4.8) transforms the inputs and outputs from (4.5) to (4.9). Recall that  $\bar{G}(\phi) = \text{block.diag}(\bar{G}_1(\phi_1), \dots, \bar{G}_N(\phi_N))$ , where

$$\bar{G}_i(\phi_i) = \begin{pmatrix} \cos \phi_i & \sin \phi_i \\ -d_{AB,i} \sin \phi_i & d_{AB,i} \cos \phi_i \end{pmatrix}.$$

The main result of this approach is now stated as follows.

**Theorem 4.10.** *Using  $u = u_1 + u_2 + u_3$ , where coupling (4.27) uses a tree graph topology, the system (4.5) converges to  $\phi = \phi^*, p = M^r v^*, z = z^*$ , thereby achieving the control goals (4.20).*

*Proof.* For simplicity of notation let  $\bar{z}_j = z_j - z_j^*$ . To analyze the closed-loop dynamics of the network, let  $\bar{q} = (\bar{q}_1, \dots, \bar{q}_N)^T$ ,  $\bar{p} = (\bar{p}_1, \dots, \bar{p}_N)^T$ ,  $\bar{z} = (\bar{z}_1, \dots, \bar{z}_E)^T$ , and define matrices  $D^r = \text{diag}(D_1^r, \dots, D_N^r)$ ,  $D^t = \text{diag}(D_1^t, \dots, D_N^t)$ ,  $D^c = \text{diag}(D_1^c, \dots, D_E^c)$ ,  $K^c = \text{diag}(K_1^c, \dots, K_E^c)$ ,  $\bar{S}(\bar{q}) = \text{block.diag}(\bar{S}_1(\bar{q}_1), \dots, \bar{S}_N(\bar{q}_N))$ .

From (4.13), (4.26), (4.27) the closed-loop dynamics are obtained as

$$\begin{pmatrix} \dot{\bar{q}} \\ \dot{\bar{p}} \\ \dot{\bar{z}} \end{pmatrix} = \begin{pmatrix} 0 & \bar{S}(\bar{q}) & 0 \\ -\bar{S}^T(\bar{q}) & -\bar{D}(\bar{q}) & -\bar{G}(\bar{q})(B \otimes I_2) \\ 0 & (B^T \otimes I_2)\bar{G}^T(\bar{q}) & 0 \end{pmatrix} \begin{pmatrix} \frac{\partial \bar{H}}{\partial \bar{q}} \\ \frac{\partial \bar{H}}{\partial \bar{p}} \\ \frac{\partial \bar{H}}{\partial \bar{z}} \end{pmatrix}, \quad (4.28)$$

$$\bar{y} = \frac{\partial \bar{H}}{\partial \bar{p}},$$

with  $\bar{D}(\bar{q}) = D^r + D^t + \bar{G}(\bar{q})(B \otimes I_2)D^c(B^T \otimes I_2)\bar{G}^T(\bar{q})$ . Note that in (4.28) the controller state  $\bar{\phi}$  is contained in  $\bar{q}$ . The closed-loop Hamiltonian  $\bar{H}$  is given by

$$\begin{aligned} \bar{H}(\bar{\phi}, \bar{p}, \bar{z}) &= \bar{H}^r(\bar{\phi}, \bar{p}) + H^s(\bar{z}) \\ &= -\sum_{i=1}^N k_i^\phi \ln |\cos \bar{\phi}_i| + \frac{1}{2} \bar{p}^T M^{r-1} \bar{p} + \frac{1}{2} \bar{z}^T K^c \bar{z}. \end{aligned}$$

Take the closed-loop Hamiltonian  $\bar{H}(\bar{q}, \bar{p}, \bar{z})$  as a candidate Lyapunov function. It is easily verified that  $\bar{H}(\bar{q}, \bar{p}, \bar{z}) \geq 0$  and that the time derivative is given by

$$\dot{\bar{H}}(\bar{q}, \bar{p}, \bar{z}) = -\frac{\partial^T \bar{H}}{\partial \bar{p}} \bar{D}(\bar{q}) \frac{\partial^T \bar{H}^r}{\partial \bar{p}}.$$

Invoking LaSalle's invariance principle (Theorem 2.4) gives that (4.26) converges to the largest invariant set where  $\dot{\bar{H}} = 0$ . On this set  $\frac{\partial^T \bar{H}}{\partial \bar{p}} = 0$ , which implies that  $\bar{p} = 0$  and  $\dot{\bar{p}} = 0$ . Substituting  $\bar{p} = 0, \dot{\bar{p}} = 0$  into (4.28) yields

$$-\bar{S}^T(\bar{q}) \frac{\partial \bar{H}}{\partial \bar{q}} - \bar{G}(\bar{q})(B \otimes I_2) \frac{\partial \bar{H}}{\partial \bar{z}} = 0. \quad (4.29)$$

Rearranging the terms, (4.29) can be rewritten as

$$\cos(\bar{\phi} + \phi^*) B K_x \bar{z}_x + \sin(\bar{\phi} + \phi^*) B K_y \bar{z}_y = 0, \quad (4.30)$$

$$K^\phi \tan \bar{\phi} - D_{AB} \sin(\bar{\phi} + \phi^*) B K_x \bar{z}_x + D_{AB} \cos(\bar{\phi} + \phi^*) B K_y \bar{z}_y = 0, \quad (4.31)$$

with  $K^\phi = \text{diag}(k_1^\phi, \dots, k_N^\phi)$ ,  $\tan \bar{\phi} = (\tan \bar{\phi}_1, \dots, \tan \bar{\phi}_N)^T$ , and  $D_{AB} = \text{diag}(d_{AB,1}, \dots, d_{AB,N})$ . Now multiply (4.30) by  $\sin(\bar{\phi} + \phi^*)$ , multiply (4.31) by  $\cos(\bar{\phi} + \phi^*) D_{AB}^{-1}$  and sum the result to obtain

$$\cos(\bar{\phi} + \phi^*) D_{AB}^{-1} K^\phi \tan \bar{\phi} + B K_y \bar{z}_y = 0. \quad (4.32)$$

Since the graph is connected, it follows that  $\ker B^T = \alpha \mathbf{1}_N$  with  $\alpha$  some arbitrary

constant. Multiplying (4.32) by  $\mathbb{1}_N^T$  from the left results in

$$\mathbb{1}_N^T \cos(\bar{\phi} + \phi^*) D_{AB}^{-1} K^\phi \tan \bar{\phi} = 0. \quad (4.33)$$

Then, since  $\cos(\bar{\phi} + \phi^*)$ ,  $D_{AB}$ , and  $K_\phi$  are diagonal matrices, (4.33) can be rewritten element wise as

$$\sum_{i=1}^N \frac{k_i^\phi}{d_{AB,i}} \cos(\bar{\phi}_i + \phi_i^*) \tan \bar{\phi}_i = 0. \quad (4.34)$$

To prove that (4.34) implies that  $\bar{\phi}_i = 0$ , first show that under Assumption 4.5 it follows that  $\bar{\phi}_i(t) \in (-\frac{\pi}{2}, \frac{\pi}{2})$  for all  $t$ . From Assumption 4.5 and  $\phi_i^* = \frac{\pi}{2}$  it follows that  $\bar{\phi}_i(0) = \phi_i(0) - \phi_i^* \in (-\frac{\pi}{2}, \frac{\pi}{2})$ . Then, since  $\bar{H}(t)$  is bounded from above ( $\bar{H}(t) \leq \bar{H}(0) \leq M < \infty$  for all  $t$ ) and  $\bar{H}_i \rightarrow \infty$  as  $\bar{\phi}_i \rightarrow \pm \frac{\pi}{2}$  it follows that  $\bar{\phi}_i(t) \in (-\frac{\pi}{2}, \frac{\pi}{2})$  for all  $t$ .

Now it is proven that (4.34) implies that  $\bar{\phi}_i = 0$  by contradiction. Assume that  $\bar{\phi}_i \neq 0$ , then either  $\bar{\phi}_i \in (-\frac{\pi}{2}, 0)$  or  $\bar{\phi}_i \in (0, \frac{\pi}{2})$  since  $\bar{\phi}_i(t) \in (-\frac{\pi}{2}, \frac{\pi}{2})$  for all  $t$ . For the case that  $\bar{\phi}_i \in (-\frac{\pi}{2}, 0)$  it immediately follows that  $\cos(\bar{\phi}_i + \phi_i^*) > 0$  while  $\tan \bar{\phi}_i < 0$ . On the other hand for  $\bar{\phi}_i \in (0, \frac{\pi}{2})$  it follows that  $\cos(\bar{\phi}_i + \phi_i^*) < 0$  and  $\tan \bar{\phi}_i > 0$ . Hence, if  $\bar{\phi}_i \neq 0$  always  $\cos(\bar{\phi}_i + \phi_i^*) \tan \bar{\phi}_i < 0$  for all  $i = 1, \dots, N$ . Substituting into (4.34) gives

$$\sum_{i=1}^N \frac{k_{\phi,i}}{d_{AB,i}} \cos(\bar{\phi}_i + \phi_i^*) \tan \bar{\phi}_i < 0,$$

which contradicts (4.34) and hence one may conclude that  $\bar{\phi} = 0$ . Substituting  $\bar{\phi} = 0$ ,  $\phi^* = \frac{\pi}{2}$  into (4.30)-(4.31) yields

$$\begin{aligned} B K_y \bar{z}_y &= 0, \\ D_{AB} B K_x \bar{z}_x &= 0. \end{aligned}$$

The graph is assumed to be acyclic, implying that  $\ker B = 0$ . Since  $D_{AB}$  is a positive definite diagonal matrix, it follows that  $\bar{z}_x = 0$ ,  $\bar{z}_y = 0$ , thus completing the proof.  $\square$

*Remark 4.11 (Cyclic graphs).* This section considered acyclic graphs to model the interaction topology. Since the interaction topology is a design freedom, considering only acyclic graphs is not restrictive. To deal with cyclic graphs an additional a priori condition is required on the desired relative displacements (see Remark 4.3).

*Remark 4.12 (Velocity tracking without local feedback control).* Experimental and simulation results seem to indicate that it is possible to achieve formation control with velocity tracking using two virtual robots which drag the whole network, without requiring a local velocity tracking controller for each robot [52, 92]. Due

to the lack of friction compensation the geometrical shape of the formation is disturbed to a greater or lesser extend depending on the virtual spring constant. Section 6.2 provides more details and a direction for future research along this avenue.

*Remark 4.13* (Control law formation control with velocity tracking). The control law  $u$  for the wheeled robots is easily derived from Proposition 4.7, Proposition 4.8 and Theorem 4.10 and is given by

$$\begin{aligned} u = & u_1 + u_2 + u_3 \\ = & \underbrace{-L K^\phi \tan(\phi - \phi^*)}_{\text{heading control}} + \underbrace{D^r v^* - D^t(y - \alpha)}_{\text{velocity tracking control}} \\ & \underbrace{-G(\phi) (B \otimes I_2) K^c(z - z^*)}_{\text{virtual spring force}} - \underbrace{G(\phi) (B \otimes I_2) D^c(B^T \otimes I_2) G^T(\phi)y}_{\text{virtual damper force}}, \end{aligned} \quad (4.35)$$

where  $L = \text{block.diag}(L_1, \dots, L_N)$  with  $L_i$  defined below (4.22). From (4.35) it follows that the control input  $u_i$  for robot  $i$  requires knowledge of its heading, forward and angular velocity (if  $D^t \neq 0$ ), while the formation controller requires the relative displacement and velocity with respect to its neighbors.

Note that (4.35) has a clear physical interpretation. The first term corresponds to a nonlinear spring force, which ensures each robot's heading  $\phi$  is converging to the desired heading  $\phi^*$ . The second term inserts a damping corresponding to the reference velocity  $v^*$ , while  $D^t(y - \alpha)$  inserts damping with respect to the error velocity. Finally, the fourth and fifth term correspond to the spring and damper forces along  $x$  and  $y$ , which are assigned between the robots. Due to the incidence matrix  $B$  each agent only requires local information from its neighbors, thereby making (4.35) a distributed control law.

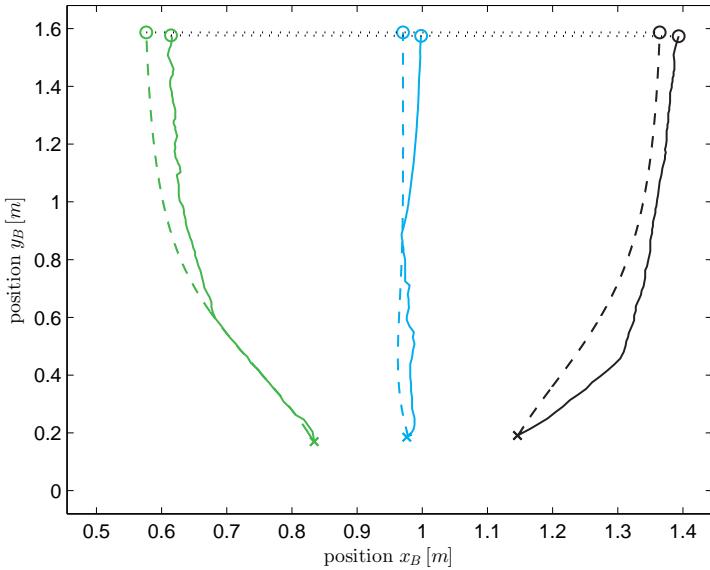
To illustrate the effectiveness of Theorem 4.10, simulation and experimental results are provided in the next section.

#### 4.4.2 Simulation and experimental results

Consider a network of  $N = 3$  wheeled robots of the form (4.5), with model parameters  $m_i = 0.167 \text{ kg}$ ,  $I_{cm,i} = 9.69 \cdot 10^{-5} \text{ kg m}^2$ ,  $d_{f,i} = 2 \text{ kg/s}$ ,  $d_{\phi,i} = 0.2 \text{ kg m}^2/\text{s}$ ,  $d_{AB,i} = 0.06 \text{ m}$  for  $i = 1, 2, 3$ . The model parameters are chosen in accordance with the e-puck wheeled robot (Figure C.1) which are given in Table C.1.

The three robots are interconnected using virtual couplings according to the incidence matrix

$$B = \begin{pmatrix} -1 & 0 \\ +1 & -1 \\ 0 & +1 \end{pmatrix},$$



**Figure 4.7:** Trajectories of the wheeled robots for the simulation (dashed) and the experiment (solid). The crosses ( $\times$ ) and circles ( $\circ$ ) denote the initial and final positions respectively.

which is associated to a so-called *path graph*. Since each robot is strictly passive (i.e.,  $d_{f,i} > 0, d_{\phi,i} > 0$ ) no virtual dampers are required for asymptotic stability ( $d_{x,j}^c = d_{y,j}^c = 0 \text{ kg/s}$  for  $j = 1, 2$ ). For the heading control and velocity tracking control the control gains are set at respectively  $\phi_i^* = \frac{\pi}{2} \text{ rad}, k_i^\phi = 0.1 \text{ kg/s}^2$  and  $v_{f,i}^* = 0.05 \text{ m/s}, D_i^t = 0$  for  $i = 1, 2, 3$ . Finally, for the formation control set  $z_{x,j}^* = 0.4 \text{ m}, z_{y,i}^* = 0 \text{ m}, k_j^x = 2 \text{ kg/s}^2, k_j^y = 2 \text{ kg/s}^2, D_{xj}^c = D_{yj}^c = 0 \text{ kg/s}$  for  $j = 1, 2$ . These settings correspond to a line formation.

The simulations are performed using MATLAB and Simulink, while the experiments make use of the experimental setup discussed in Appendix A. The experimental setup consists of  $2.6 \times 2.0 \text{ m}$  table with an overhead camera for localization. Each robot is identified and localized using a data-matrix, attached on top of the e-puck (see Figure C.1 (right)). A vision algorithm runs in parallel to MATLAB and provides the ID, position, and heading of each e-puck. MATLAB calculates the corresponding control inputs (4.35) and sends them to the e-pucks via a Bluetooth connection.

Both the simulation and the experiment are run for  $t = 21 \text{ s}$ , starting from the same initial conditions:  $x_B(0) = (0.83, 1.00, 1.15) \text{ m}, y_B(0) = (0.17, 0.19, 0.19) \text{ m}, \phi(0) = (1.68, 1.89, 1.48) \text{ rad}, p_f(0) = (0, 0, 0) \frac{\text{kg m}}{\text{s}}, h(0) = (0, 0, 0) \frac{\text{kg m}^2}{\text{s}}$ . Note that the run time is significantly shorter than for the simulation and experiments in

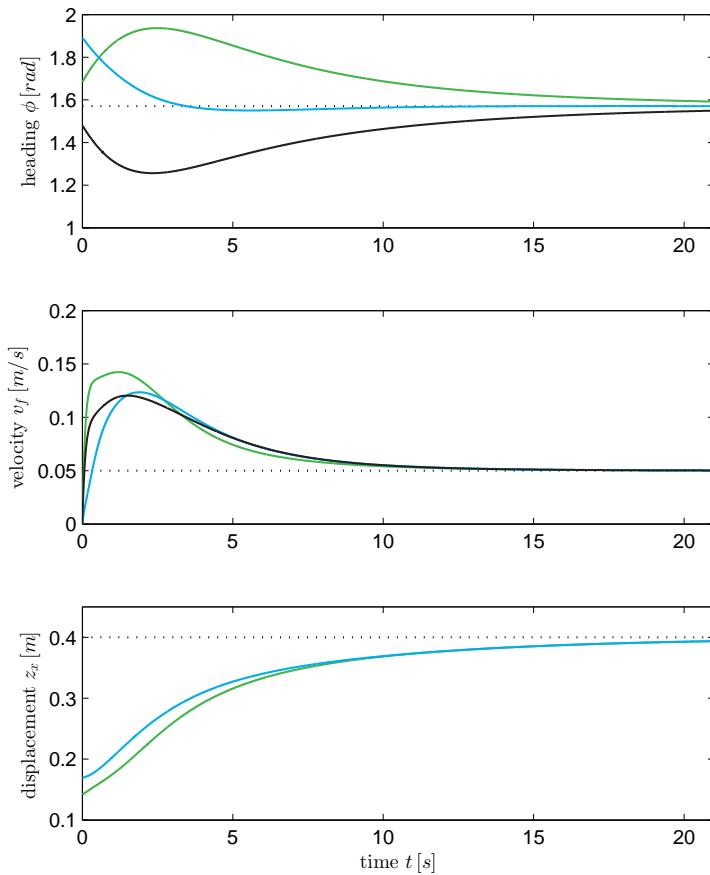
Section 4.3 due to the limited size of the table in the experimental setup (i.e., the e-pucks would run into the boundaries of the setup for a longer run time). The results are shown in Figure 4.3-4.5.

The trajectories of the e-pucks are shown in Figure 4.7. The (small) differences are due to localization errors of the vision algorithm, model parameter uncertainty and a difference in the actuators of the model and the e-puck (the robot dynamics (4.5) do not consider the differential drive of the e-puck).

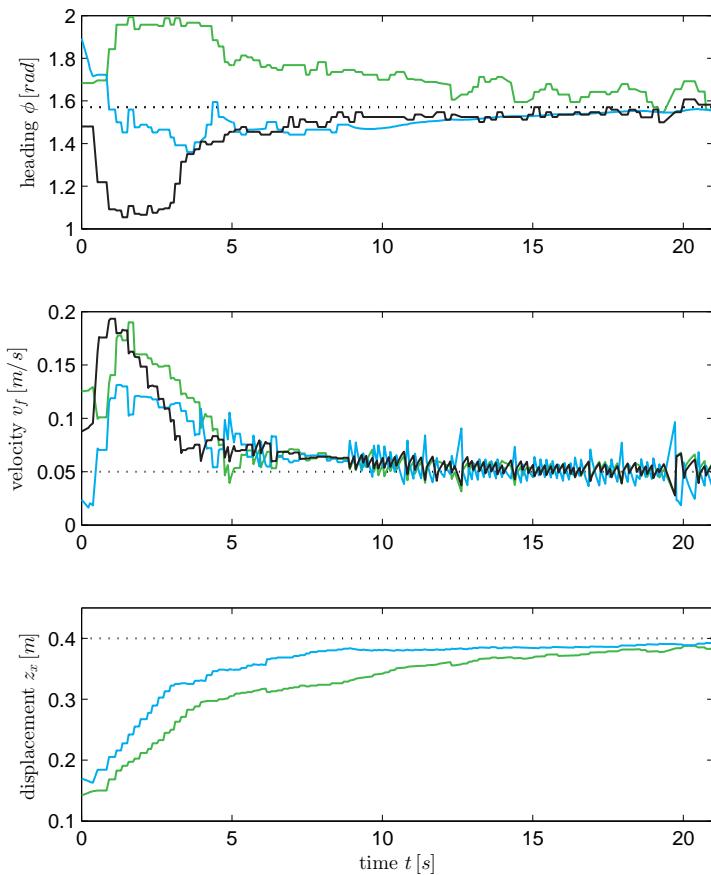
To illustrate that the control objectives (4.20) are achieved, Figure 4.8 and 4.9 show the time evolution of the heading  $\phi$ , forward velocity  $v_f$ , and relative displacement  $z_x$ . The dotted lines correspond the reference values  $\phi^*, v_f^*, z_x^*$  respectively. The angular velocity  $v_\phi$  and relative displacement  $z_y$  show a similar trend and are given in Appendix D. Figure 4.8 and 4.9 show that for both the simulation and the experiment all variables converge to their reference value in accordance with Theorem 4.10.

Preliminary results on tuning of the controller gains in an extensive simulation and experimental setting may be found in respectively [52] and [20, 92]. For the velocity tracking controller considered in [52, 92] only some robots in the network require knowledge on the reference velocity. While the simulation and experimental results are promising, no formal proofs are available (see page 121 for a recommendation for future research).

In addition to formation control and velocity tracking, [20, 52] consider obstacle avoidance using artificial potential fields. The performance indicators considered in tuning the controller gains include energy consumption [20, 52, 92], formation error [92], number of collisions [20, 52], settling time [20] and coverage [52].



**Figure 4.8:** Time evolution of the heading  $\phi$ , forward velocity  $v_f$  and relative displacement  $z_x$  (simulation). The dotted lines show the reference values.



**Figure 4.9:** Time evolution of the heading  $\phi$ , forward velocity  $v_f$  and relative displacement  $z_x$  (experiment). The dotted lines show the reference values.

## 4.5 Formation control in the presence of matched input disturbances

In practice there are often disturbances influencing the robots' behavior. In Sections 4.3 and 4.4 it was assumed that no disturbances were present. In this section *matched input disturbances* are considered, which act on the same channel as the control input. The disturbances are assumed to be generated by an external *exosystem* with internal state  $w$ . An *internal model controller* with state  $\theta$  is designed, which asymptotically compensates the unknown disturbance.

The objectives of formation control in the presence of matched input disturbances can now be formulated in terms of (4.15) as

$$\begin{cases} p \rightarrow 0, \\ z \rightarrow z^*, \\ \theta \rightarrow w, \end{cases} \quad \text{as } t \rightarrow \infty. \quad (4.36)$$

Note that the first two objectives are the same as the formation control objectives (4.11). Different is the addition of an objective for the internal model controller state, which guarantees that the disturbances considered here are rejected. In contrast with (4.20) velocity tracking and heading control are not considered here.

The results presented here to achieve (4.36) are published in [60, 116] and are based on a collaboration with Matin Jafarian and Claudio De Persis.

### 4.5.1 Matched input disturbance rejection using an internal-model-based approach

Before continuing with the controller design, first the matched input disturbance is introduced. A matched input disturbance acts on the same channel as the control input  $u_i$  in (4.9) such that the dynamics of robot  $i$  subject to a matched input disturbance  $d_i$  is given by

$$\begin{aligned} \begin{pmatrix} \dot{q}_i \\ \dot{p}_i \end{pmatrix} &= \begin{pmatrix} 0 & S_i(q_i) \\ -S_i^T(q_i) & -D_i^r \end{pmatrix} \begin{pmatrix} \frac{\partial H_i^r}{\partial q_i} \\ \frac{\partial H_i^r}{\partial p_i} \end{pmatrix} + \begin{pmatrix} 0 \\ \bar{G}_i(q_i) \end{pmatrix} (\bar{u}_i + d_i), \\ \bar{y}_i &= \bar{G}_i^T(q_i) \frac{\partial H_i^r}{\partial p_i}. \end{aligned} \quad (4.37)$$

with  $q_i \in \mathbb{R}^3$ ,  $p_i, \bar{u}_i, d_i, \bar{y}_i \in \mathbb{R}^2$ . In a similar fashion as for (4.10) the compact dynamics for  $N$  robots of the form (4.37) are given by

$$\begin{pmatrix} \dot{q} \\ \dot{p} \end{pmatrix} = \begin{pmatrix} 0 & S(q) \\ -S^T(q) & -D^r \end{pmatrix} \begin{pmatrix} \frac{\partial H^r}{\partial q} \\ \frac{\partial H^r}{\partial p} \end{pmatrix} + \begin{pmatrix} 0 \\ \bar{G}(q) \end{pmatrix} (\bar{u} + d),$$

$$\bar{y} = \bar{G}^T(q) \frac{\partial H^r}{\partial p}, \quad (4.38)$$

with  $d = (d_1, \dots, d_N)^T$ .

The controller to achieve the control objectives (4.36) for (4.37) consists of two parts. The first part is a formation controller of the form (4.12), which provides the control input  $\bar{u}_i$  given in (4.14). The second part is an internal-model-based disturbance rejection controller. The corresponding control input  $\check{d}_i$  is designed below.

The disturbance signal  $d_i$  is assumed to be generated by an autonomous exosystem. Given two matrices  $\Phi_i^d \in \mathbb{R}^{4 \times 4}$  and  $\Gamma_i^d \in \mathbb{R}^{2 \times 4}$ , whose properties will be made precise later on. Let  $w_i^d \in \mathbb{R}^4$  denote the ecosystem state, then the exosystem of robot  $i$  obeys the following dynamics

$$\begin{aligned} \dot{w}_i^d &= \Phi_i^d w_i^d, \\ d_i &= \Gamma_i^d w_i^d, \end{aligned} \quad (4.39)$$

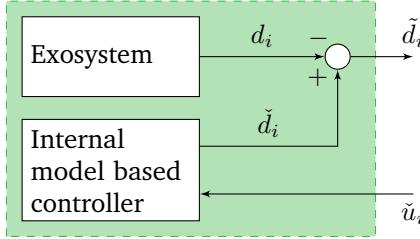
for  $i = 1, 2, \dots, N$ . Here it is assumed that  $\Phi_i^d$  is a skew-symmetric matrix, implying that (4.39) is able to generate harmonic and constant disturbance signals. Inspired by the theory of output regulation (see e.g. [54]), an internal-model-based controller is adopted to counteract the effect of the disturbance  $d_i$  generated by (4.39). Let  $\theta_i \in \mathbb{R}^4$ ,  $\check{u}_i, \check{d}_i \in \mathbb{R}^2$  denote respectively the state, input and output of the internal-model-based controller. Then the internal model dynamics are given by

$$\begin{aligned} \dot{\theta}_i &= \Phi_i^d \theta_i + \Gamma_i^{d^T} \check{u}_i, \\ \check{d}_i &= \Gamma_i^d \theta_i, \end{aligned} \quad (4.40)$$

for  $i = 1, \dots, N$ . When  $\check{u}_i = 0$  and the system is appropriately initialized, the latter system is able to generate any  $w_i^d$  solution to (4.39). Now, define the error variables  $\tilde{d}_i = \check{d}_i - d_i$  and  $\tilde{\theta}_i = \theta_i - w_i^d$ , then the error dynamics of the internal model controller including the exosystem are given by

$$\begin{aligned} \dot{\tilde{\theta}}_i &= \Phi_i^d \tilde{\theta}_i + \Gamma_i^{d^T} \check{u}_i, \\ \tilde{d}_i &= \Gamma_i^d \tilde{\theta}_i. \end{aligned} \quad (4.41)$$

To compactly denote the dynamics of  $N$  internal model controllers of the form



**Figure 4.10:** The exosystem (4.39) together with the internal-model-based disturbance rejecting controller (4.40) form the lossless subsystem (4.41) with port variables  $\check{u}_i, \check{d}_i$ .

(4.41) denote vectors  $\tilde{\theta} = (\tilde{\theta}_1, \dots, \tilde{\theta}_N)^T$ ,  $\check{u} = (\check{u}_1, \dots, \check{u}_N)^T$ ,  $\tilde{d} = (\tilde{d}_1, \dots, \tilde{d}_N)^T$ , and matrices  $\Phi^d = \text{block.diag}(\Phi_1^d, \dots, \Phi_N^d)$ ,  $\Gamma^d = \text{block.diag}(\Gamma_1^d, \dots, \Gamma_N^d)$ . Then, the total internal model controller dynamics are given by

$$\begin{aligned}\dot{\tilde{\theta}} &= \Phi^d \tilde{\theta} + \Gamma^{dT} \check{u}, \\ \tilde{d} &= \Gamma^d \tilde{\theta}.\end{aligned}\tag{4.42}$$

Note that the exosystem (4.39) itself is not a passive system, since it has no input. However, interconnecting exosystem (4.39) with the internal model controller (4.40) the resulting system (4.41) is lossless with respect to the port variables  $\check{u}_i, \check{d}_i$  (see Figure 4.10). Furthermore (4.42) can easily be represented in the port-Hamiltonian framework by defining the Hamiltonian  $H^d(\tilde{\theta}) = \frac{1}{2} \tilde{\theta}^T \tilde{\theta}$ , such that (4.42) can be rewritten in the form (2.5) as

$$\begin{aligned}\dot{\tilde{\theta}} &= \Phi^d \frac{\partial H^d}{\partial \tilde{\theta}} + \Gamma^{dT} \check{u}, \\ \tilde{d} &= \Gamma^d \frac{\partial H^d}{\partial \tilde{\theta}}.\end{aligned}\tag{4.43}$$

Note that in (4.43) the port-Hamiltonian structure is preserved, since  $\Phi^d$  is skew-symmetric.

### Closed-loop analysis

Now, continue with the closed-loop analysis of the two control systems: one for reaching the desired formation ( $\bar{u}_i$  from (4.14)) and one for counteracting matched input disturbances ( $\check{d}_i$  from (4.40)). By interconnecting the systems appropriately, the closed-loop system preserves passivity properties and the port-Hamiltonian structure. The coupling of the two controllers with the robots is power-preserving

and given by [3, 100]

$$\begin{cases} u = -(B \otimes I_2)\tau - \check{d}, \\ v = (B^T \otimes I_2)y, \\ \check{u} = y, \end{cases} \quad (4.44)$$

with  $\check{d} = (\check{d}_1, \dots, \check{d}_N)^T$ . Note that  $(u, y)$ ,  $(v, \tau)$  and  $(\check{u}, \check{d})$  are the port-variables for respectively the robots (4.38), the virtual couplings (4.13) and the internal-model-based controller (4.40). Coupling (4.44) is a combination of formation control coupling (4.14) and standard negative feedback.

The overall closed-loop dynamics follow from (4.38), (4.13), (4.43), (4.44) and are given by

$$\begin{pmatrix} \dot{q} \\ \dot{p} \\ \dot{z} \\ \dot{\tilde{\theta}} \end{pmatrix} = \begin{pmatrix} 0 & S(q) & 0 & 0 \\ -S^T(q) & -\check{D}(q) & -\check{G}(q) & -\bar{G}(q)\Gamma^d \\ 0 & \check{G}^T(q) & 0 & 0 \\ 0 & \Gamma^d \bar{G}^T(q) & 0 & \Phi^d \end{pmatrix} \begin{pmatrix} \frac{\partial H}{\partial q} \\ \frac{\partial H}{\partial p} \\ \frac{\partial H}{\partial z} \\ \frac{\partial H}{\partial \tilde{\theta}} \end{pmatrix}, \quad (4.45)$$

with  $\check{G}(q) = \bar{G}(q)(B \otimes I_2)$  and  $\check{D}(q) = D^r + \check{G}(q)D^c\check{G}^T(q)$ . Finally, the closed-loop Hamiltonian is given by

$$H(p, z, \tilde{\theta}) = H^r(p) + H^c(z) + H^d(\tilde{\theta}).$$

Now the results of the stability and convergence analysis of the closed loop system are presented, starting with the following proposition.

**Proposition 4.14.** *Interconnect wheeled robots (4.38) with virtual couplings (4.13) and internal model controller (4.43) via coupling (4.44) using a tree graph topology. Then, the closed-loop system (4.45) asymptotically converges to  $p = 0$  provided that  $D^c > 0$  and there is at least one robot  $i$  satisfying  $D_i^r > 0$ .*

*Proof.* Take the closed-loop Hamiltonian  $H$  as the candidate Lyapunov function. Calculating the time derivative of  $H$ , one obtains

$$\begin{aligned} \dot{H} = & -\frac{\partial^T H}{\partial p} \check{D}(q) \frac{\partial H}{\partial p} - \frac{\partial^T H}{\partial p} \check{G}(q) \frac{\partial H}{\partial z} + \frac{\partial^T H}{\partial z} \check{G}^T(q) \frac{\partial H}{\partial p} \\ & + \frac{\partial^T H}{\partial \tilde{\theta}} \Phi^d \frac{\partial H}{\partial \tilde{\theta}} - \frac{\partial^T H}{\partial p} \bar{G}(q)\Gamma^d \frac{\partial H}{\partial \tilde{\theta}} + \frac{\partial^T H}{\partial \tilde{\theta}} \Gamma^d \bar{G}^T(q) \frac{\partial H}{\partial p}. \end{aligned} \quad (4.46)$$

Due to skew-symmetry of  $\Phi^d$  it immediately follows that

$$\dot{H}(x) = -\frac{\partial^T H}{\partial p} (D^r + \check{G}(q)D^c\check{G}^T(q)) \frac{\partial H}{\partial p}. \quad (4.47)$$

Note that since  $D^c > 0$  and  $D_i^r \geq 0$  it follows that  $\dot{H}(x) \leq 0$  and therefore the system is stable. Invoking LaSalle's invariance principle (Theorem 2.4) provides that the system (4.45) converges to the largest invariant set where

$$\check{G}^T(q) \frac{\partial H}{\partial p} = (B^T \otimes I_2) \bar{G}^T(q) (M^r)^{-1} p = 0, \quad (4.48)$$

$$\frac{\partial H}{\partial p_i} = (M_i^r)^{-1} p_i = 0, \quad (4.49)$$

where  $p_i$  is the momentum of strictly passive robot  $i$  (i.e.,  $D_i^r > 0$ ).

Since the graph topology is assumed to be undirected and connected, it follows that  $\ker B^T = \alpha_1 \mathbb{1}_N$ , for some constant  $\alpha_1 \in \mathbb{R}$ . Noting that  $G^T(q)$  has no singularity for any  $q$  it follows from (4.48) that  $p = \alpha_2 \otimes \mathbb{1}_N$  for some constant vector  $\alpha_2 \in \mathbb{R}^2$ . Moreover, from (4.49) it follows that  $p_i = 0$  (since  $m_i > 0$  and  $I_{CM,i} > 0$ ). This implies that  $\alpha_2 = 0$  and therefore  $p = 0$ , thereby completing the proof.  $\square$

**Remark 4.15** (Strictly passive robots). The results of the Proposition 4.14 still hold if  $D^c = 0$  provided that for all robots  $D_i^r > 0$  (i.e., a network of strictly passive robots).

Proposition 4.14 guarantees stability of the system. However, it guarantees neither reaching the desired formation (i.e.,  $z = z^*$ ) nor rejecting the matched input disturbances (i.e.,  $\tilde{\theta} = 0$ ). To verify more, two special types of the disturbances generated by the exosystem (4.39) are considered next, namely harmonic and constant disturbances.

**Corollary 4.16** (Harmonic disturbances). *Assume that the exosystems' matrices  $(\Gamma_i^d, \Phi_i^d)$  are of the form*

$$\Phi_i^d = \text{block.diag}\left(\begin{pmatrix} 0 & \omega_{i1} \\ -\omega_{i1} & 0 \end{pmatrix}, \begin{pmatrix} 0 & \omega_{i2} \\ -\omega_{i2} & 0 \end{pmatrix}\right), \quad (4.50)$$

with  $\omega_{i\ell} \neq 0$  for  $\ell = 1, 2$ , and  $\Gamma_i^d = \text{block.diag}(\Gamma_{i1}^d, \Gamma_{i2}^d)$ , with  $\Gamma_{i\ell}^d \neq 0$ , for all  $\ell = 1, 2$ , and the pair  $(\Gamma_i^d, \Phi_i^d)$  is observable. Then the closed loop system (4.45) converges to  $p = 0$ ,  $\tilde{\theta} = 0$ , and  $z = z^*$ .

*Proof.* Corollary 4.16 is a special case of Proposition 4.14. Therefore, one can start by substituting  $p = 0$  into (4.45) to obtain

$$\begin{aligned} \dot{q} &= 0, \\ 0 &= -\bar{G}(q)(B \otimes I_2)K^c(z - z^*) - \Gamma^d \tilde{\theta}, \\ \dot{z} &= 0, \\ \dot{\tilde{\theta}} &= \Phi^d \tilde{\theta}, \end{aligned} \quad (4.51)$$

First calculate the time derivative of the second equality. Since,  $\dot{q} = 0$  and  $\dot{z} = 0$ , it follows that  $\Gamma^d \dot{\tilde{\theta}} = 0$ . Now, replacing  $\dot{\tilde{\theta}}$  from the last equality in (4.51) gives  $\Gamma^d \Phi^d \tilde{\theta} = 0$ . Calculate the time derivative of the latter gives  $\Gamma^d \Phi^d \Gamma^d \tilde{\theta} = 0$ . Since  $\Phi^d$  is a skew-symmetric matrix with the structure given in (4.50),  $\Phi^d \Phi^d$  is equal to  $\gamma I_4$ , where  $\gamma$  is a constant. Hence,  $\Gamma^d \Phi^d \Phi^d \tilde{\theta} = 0$  implies that  $\Gamma^d \tilde{\theta} = 0$ . Hence,

$$\begin{aligned}\dot{\tilde{\theta}} &= \Phi^d \tilde{\theta}, \\ \Gamma^d \tilde{\theta} &= 0.\end{aligned}$$

Since the pair  $(\Gamma^d, \Phi^d)$  is observable, it immediately follows that  $\tilde{\theta} = 0$ . Now, consider the second equality in (4.51). Substituting  $\tilde{\theta} = 0$  results in

$$\bar{G}(q)(B \otimes I_2)K^c(z - z^*) = 0.$$

For an undirected acyclic graph the kernel of the incidence matrix is given by  $\ker(B \otimes I_2) = 0$ . Since  $\bar{G}(q)$  has no singularities for any  $q$  and  $K^c$  is a diagonal matrix with positive constants on the diagonal it immediately follows that  $z = z^*$ , thereby completing the proof.  $\square$

Corollary 4.16 shows that the internal-model-based controller (4.42) can counteract harmonic disturbances. The following corollary presents the result for constant disturbances. Note that constant disturbances are generated by (4.43) by setting  $\Phi^d = 0$ .

**Corollary 4.17** (Constant disturbances). *Assume that  $\Phi^d = 0$ , and  $\Gamma^d$  is nonsingular. Then the closed loop system (4.45) converges to  $p = 0$ ,  $\tilde{\theta} = c_{\tilde{\theta}}$ , and  $\tilde{z} = c_{\tilde{z}}$ , with  $c_{\tilde{\theta}} \in \mathbb{R}^N$ ,  $c_{\tilde{z}} \in \mathbb{R}^E$  arbitrary constants.*

*Proof.* Corollary 4.17 is again a special case of Proposition 4.14. Substituting  $p = 0$  into (4.45) gives (4.51). For constant disturbances  $\Phi^d = 0$ , substituting into (4.51) gives

$$\begin{aligned}\dot{q} &= 0, \\ 0 &= -\bar{G}(q)(B \otimes I_2)K^c(z - z^*) - \Gamma^d \tilde{\theta}, \\ \dot{\tilde{z}} &= 0, \\ \dot{\tilde{\theta}} &= 0.\end{aligned}\tag{4.52}$$

Since  $\dot{\tilde{z}} = 0$  and  $\dot{\tilde{\theta}} = 0$ ,  $\tilde{z}$  and  $\tilde{\theta}$  converge to arbitrary constants, thereby completing the proof.  $\square$

**Remark 4.18** (Constant disturbances). For the constant disturbance case, one can only conclude that the error position vector  $z - z^*$  and the error disturbance vector  $\tilde{\theta}$  are constant. Therefore the internal-model-based controller (4.43) guarantees

neither rejecting a constant disturbance nor achieving the desired formation. However, the stability of the network is maintained and the robots' velocities converge to zero.

The difference between Corollary 4.16 and 4.17 is that for constant disturbances observability of the pair  $(\Gamma^d, \Phi^d)$  is lost. This observability property plays a crucial role in the proof of Corollary 4.16 to prove that  $\tilde{\theta} = 0$ .

*Remark 4.19* (Cyclic graphs). This section considered acyclic graphs to model the interaction topology. Since the interaction topology is a design freedom, considering only acyclic graphs is not restrictive. To deal with cyclic graphs an additional a priori condition is required on the desired relative displacements (see Remark 4.3).

*Remark 4.20* (Control input formation control and disturbance rejection). The input to each robot follows directly from (4.13), (4.40) and (4.44) and is given by

$$\begin{aligned} u &= -(B \otimes I_2)\tau - \check{d} \\ &= -\underbrace{(B \otimes I_2)K^c(z - z^*)}_{\text{virtual spring force}} - \underbrace{(B \otimes I_2)D^c(B^T \otimes I_2)M^{-1}p}_{\text{virtual damper force}} - \underbrace{\Gamma^d\theta}_{\text{internal model control}}. \end{aligned} \quad (4.53)$$

The first two parts in (4.53) corresponds to the virtual springs and virtual dampers assigned between the robots. The presence of the incidence matrix  $B$  shows that (4.53) is in fact a distributed control law. The last term in (4.53) corresponds to the local internal model controller, which requires only the controller state  $\theta$ .

## 4.5.2 Simulation results

Consider a network of  $N = 5$  wheeled robots of the form (4.5), with model parameters  $m_i = 0.167 \text{ kg}$ ,  $I_{cm,i} = 9.69 \cdot 10^{-5} \text{ kg m}^2$ ,  $d_{AB,i} = 0.06 \text{ m}$  for  $i = 1, \dots, 5$ . In accordance with Proposition 4.14 set  $d_{f,1} = 2 \text{ kg/s}$ ,  $d_{\phi,1} = 0.2 \text{ kg m}^2/\text{s}$  and  $d_{f,i} = 0 \text{ kg/s}$ ,  $d_{\phi,i} = 0 \text{ kg m}^2/\text{s}$  for  $i = 2, \dots, 5$  such that only robot 1 is strictly passive, while the other robots are lossless. The five robots are interconnected using the incidence matrix

$$B = \begin{pmatrix} -1 & 0 & 0 & 0 \\ +1 & -1 & 0 & 0 \\ 0 & +1 & -1 & 0 \\ 0 & 0 & +1 & -1 \\ 0 & 0 & 0 & +1 \end{pmatrix},$$

which is associated to a *path graph*. The virtual coupling parameters are set at  $z_{x,k}^* = 1 \text{ m}$ ,  $z_{y,k}^* = 0 \text{ m}$ ,  $\kappa_{x,k} = \kappa_{y,k} = 2 \text{ kg/s}^2$ ,  $d_{x,k} = d_{y,k} = 1$  for  $k = 1, \dots, 4$ . Note that this choice for  $z_{x,k}^*, z_{y,k}^*$  corresponds to a *line formation*.

To illustrate the effectiveness of Corollary 4.16 the exosystem parameters generating the harmonic disturbance signal are set at

$$\Phi_i^d = I_2 \otimes \begin{pmatrix} 0 & i \\ -i & 0 \end{pmatrix}, \Gamma_i^d = I_2 \otimes \begin{pmatrix} 0 & 2 \end{pmatrix},$$

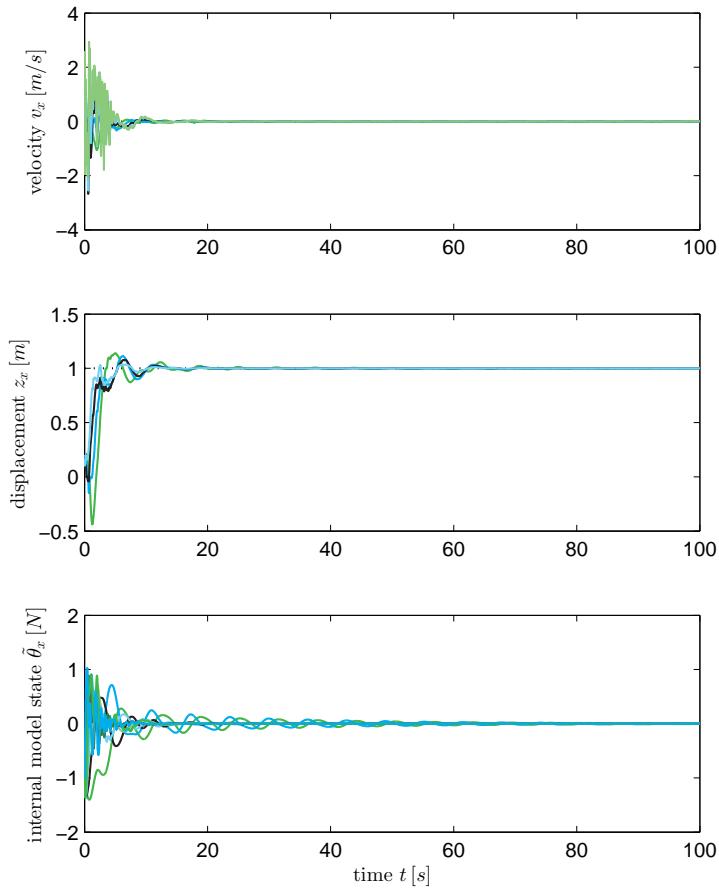
$$w_i^d(0) = (1, 1, 1, 1)^T, \theta_i^d(0) = (0, 0, 0, 0)^T,$$

for  $i = 1, \dots, 5$ . For the illustration of constant disturbance case (Corollary 4.17) set  $\Phi_i^d = 0$  and keep  $\Gamma_i^d, w_i^d(0), \theta_i^d(0)$  the same.

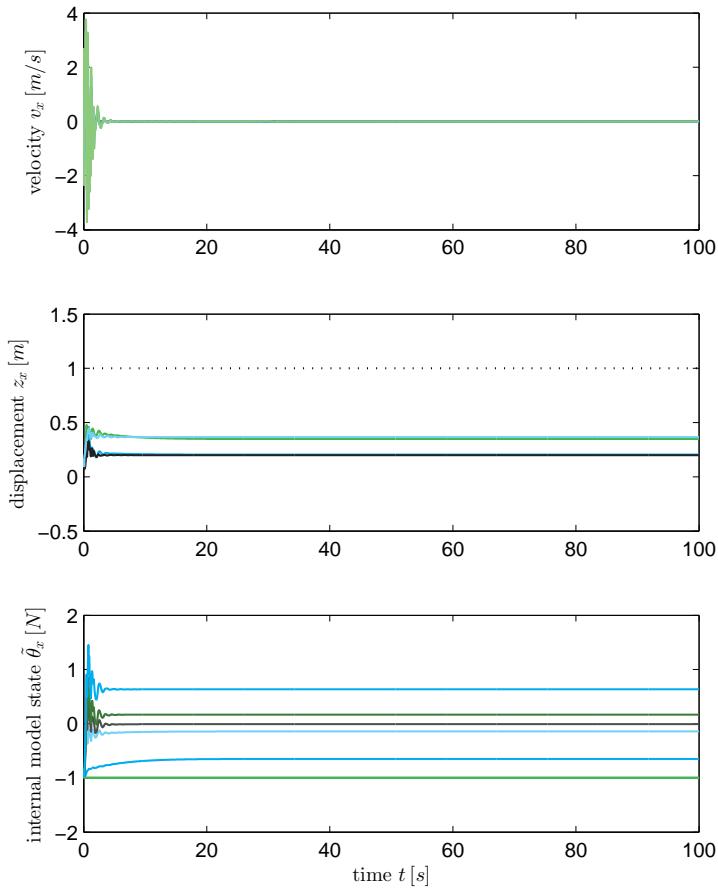
The simulations are performed using MATLAB and Simulink and were run for  $t = 100\text{ s}$ , starting from the initial conditions  $x_B(0) = (0.0035, 0.0981, 0.2354, 0.3467, 0.4797)\text{ m}$ ,  $y_B(0) = (0.0599, -0.0537, 0.0582, -0.0529, -0.0565)\text{ m}$ ,  $\phi(0) = (1.5124, 4.2482, 1.8162, 4.2211, 4.3677)\text{ rad}$ ,  $p_f(0) = 0\text{ kg m/s}$ ,  $h(0) = 0\text{ kg m}^2/\text{s}$ . The results are shown in Figures 4.11, 4.12, and 4.13. Only the results along the  $x$  direction are shown here, the plots along the  $y$  direction show a similar trend and are given in Appendix D.

Figure 4.11 shows the time evolution of the velocity  $v_x$ , relative displacement  $z_x$ , and internal model controller state  $\tilde{\theta}_x$  in the presence of harmonic disturbances, while Figure 4.12 shows the same variables for constant disturbances. For the harmonic disturbances, all three variables converge to zero in accordance with Corollary 4.16 (see Figure 4.11). For the constant disturbance case  $v_x$  does converge to zero, while  $z_x$  and  $\tilde{\theta}_x$  converge to a constant different from the desired value (see Figure 4.12).

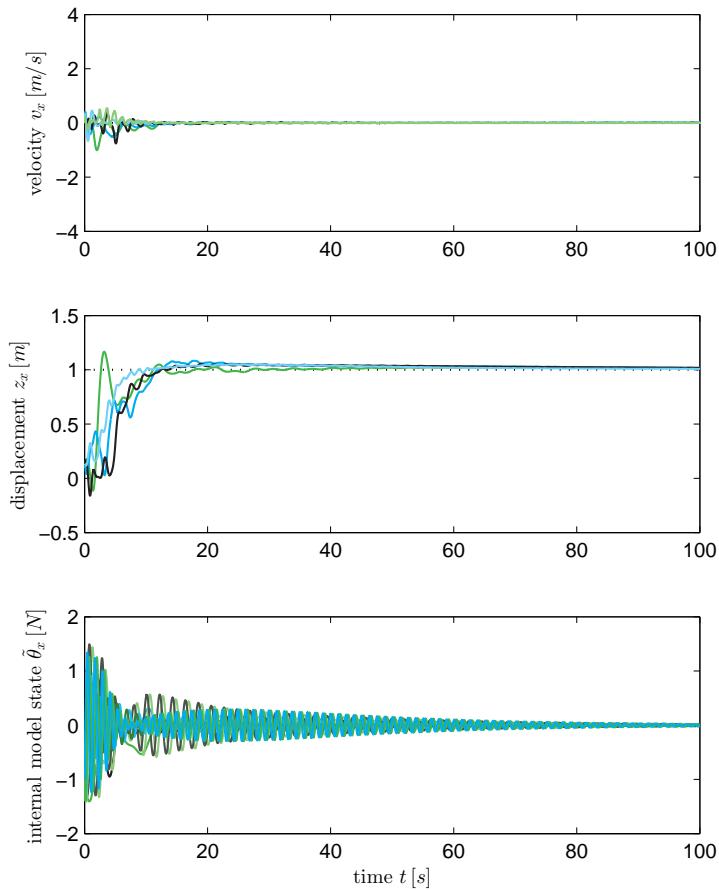
Finally, to illustrate Remark 4.15 a network where all five robots are strictly passive is simulated (i.e.,  $d_{f,i} = 2\text{ kg/s}$ ,  $d_{\phi,i} = 0.2\text{ kg m}^2/\text{s}$  for  $i = 1, \dots, 5$ ). The virtual damping coefficients are set at  $d_{x,k} = d_{y,k} = 0$  for  $k = 1, \dots, 4$ . Here, the robots are subject to the same harmonic disturbances as in Figure 4.11. For this setting all variables converge to zero as pointed out by Remark 4.15 (see Figure 4.13).



**Figure 4.11:** Time evolution of the velocity  $v_x$ , relative displacement  $z_x$  and internal model controller state  $\tilde{\theta}_x$  in the presence of harmonic disturbances. The dotted lines show the reference values.



**Figure 4.12:** Time evolution of the velocity  $v_x$ , relative displacement  $z_x$  and internal model controller state  $\tilde{\theta}_x$  in the presence of constant disturbances. The dotted lines show the reference values.



**Figure 4.13:** Time evolution of the velocity  $v_x$ , relative displacement  $z_x$  and internal model controller state  $\tilde{\theta}_x$  in the presence of harmonic disturbances when all robots are strictly passive and  $D^c = 0$ . The dotted lines show the reference values.

## 4.6 Concluding remarks

This chapter considers three formation control problems for a network of nonholonomic wheeled robots. The wheeled robots are modeled as rigid bodies with a nonholonomic constraint on the wheel axle. Solving for the constraint provides the wheeled robot dynamics on the constrained state space.

The formation control problem is tackled by assigning virtual couplings between the front ends of the robots, where the interconnection topology is modeled as a tree graph. Assigning the couplings to the front ends differs from the coupling assignment in Chapter 3, where the springs are assigned to the center of mass.

The same approach is then used to achieve formation control while tracking a reference velocity along a desired heading. The velocity controller builds upon generalized canonical transformations to derive and stabilize the error dynamics with respect to the reference velocity. The nonlinear heading controller guarantees that under a natural assumption on the initial heading, all robots move along the desired heading.

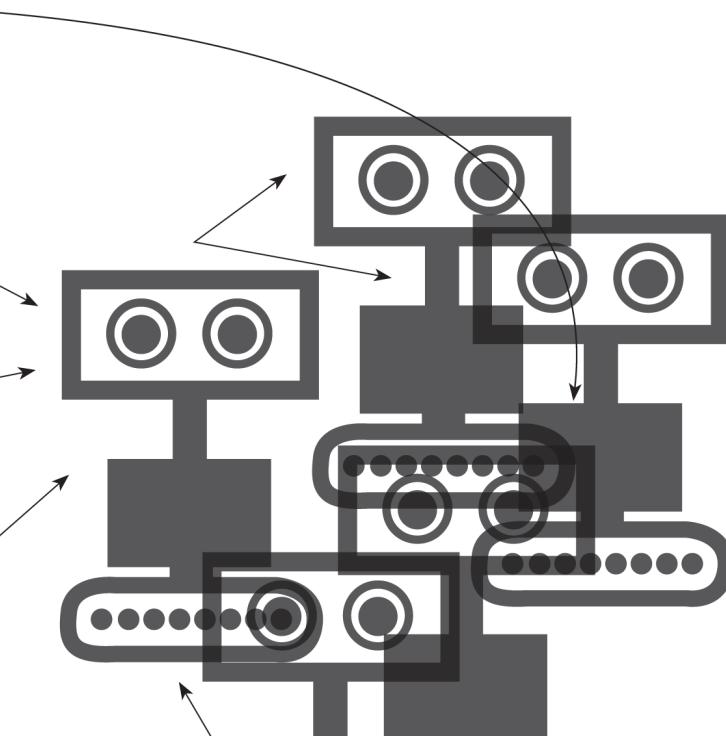
Finally an internal-model-based controller is designed to reject matched input disturbances. The internal model controller is able to compensate for harmonic disturbances if at least one robot is strictly passive and the other robots are connected using (a chain of) virtual dampers in parallel with the virtual springs. For constant disturbances the controller does prevent the robots from drifting away (i.e., stability is guaranteed), but it does not completely reject the disturbance. Ongoing research investigates alternative controller designs to deal with constant disturbances.

The results in this chapter are illustrated by simulation and experimental results using the e-puck wheeled robot.



# Chapter 5

Orbital phasing of satellites on circular orbits





## Chapter 5

# Orbital phasing of satellites on circular orbits

This chapter considers a port-Hamiltonian approach to the problem of orbital phasing of satellite constellations on circular orbits. The problem setting and a theoretical background for the problem are discussed in Section 5.1. Section 5.2 continues with the dynamical model for each satellite and using generalized canonical transformations the error system with respect to a circular orbit is derived. A formal problem definition of the orbital phasing problem is given in Section 5.3, followed by the controller design and closed-loop analysis. The highlights and concluding remarks of the chapter are given in Section 5.4.

The results in this chapter are published in [111, 112, 114].

## 5.1 Introduction

Formation flying of satellite constellations has received quite some attention in recent years. Different definitions of the terms satellite constellation and formation flying have been used in literature, for example based on the fact whether the states of the satellites are coupled (formation flying [101]) or not (constellation [102]). In this chapter the word *satellite constellation* refers to a group of satellites which collaborate in order to achieve a higher level goal. The term *formation flying* refers to this higher level goal, which in this chapter corresponds to achieving an equal distribution of the satellites in the constellation on a circular orbit.

Using satellite constellations opens up possibilities for new types of missions, which are not possible with the traditional one satellite setup [26]. For example, the OLFAR mission aims at exploring the below 30MHz frequency bandwidth radio signals. To achieve sufficient spatial resolution, such a low frequency telescope in space needs an aperture diameter of 10 to 100 kilometers.<sup>1</sup> Clearly, these types of applications are not feasible with a single satellite.

The dynamic environment where constellation operates can be divided into two regimes [101], namely deep space and planetary orbits. In deep space the relative dynamics of a constellation are usually approximated, often by a double

<sup>1</sup> [www.lr.tudelft.nl/en/organisation/departments/space-engineering/space-systems-engineering/projects/olfar-orbiting-low-frequency](http://www.lr.tudelft.nl/en/organisation/departments/space-engineering/space-systems-engineering/projects/olfar-orbiting-low-frequency)

integrator. For planetary orbits on the other hand the constellation dynamics are considered explicitly, including gravitational forces and disturbances such as drag. In this chapter the focus is on one type of planetary orbits, namely *circular orbits*.

Much formation flying research has focused on the case where only one point in the formation (e.g. the center of mass or the leader satellite) is on the planetary orbit [23, 65], while the individual satellites themselves do not need to be on the orbit. In contrast, [76, 108] and this chapter address the problem where each satellite is on the orbit, while the satellites equally distribute spatially on the orbit. This problem is also known as *orbital phasing*, since the control goal is to keep spacecrafts phased on the orbit [102].

Orbital phasing on circular orbits is of special interest to Global Navigation Satellite Systems (GNSS) like the Global Positioning System (GPS) and more recently Galileo. GPS requires 24 satellites to phase on six circular orbits, while Galileo requires 30 satellites to phase on three orbits. Other applications may be found in meteorological, environmental and military applications.

In Section 5.3 a controller is presented, based on energy considerations to solve the orbital phasing problem. Recently there has been an increasing interest in these so-called energy-based models [86], which allow for analysis and control design of nonlinear, multi-domain systems such as satellite constellations. The energy function of a system determines not only the static, but also the transient behavior [86] thereby enabling stabilization and performance studies. Furthermore, practitioners are familiar with energy concepts and therefore energy-based models may serve as a *lingua franca* amongst (control) engineers [86]. Two common energy-based modeling frameworks are the Lagrangian [23] and port-Hamiltonian framework [41, 45].

This chapter provides a theoretical framework for the orbital phasing of satellites on circular orbits. The control systems consists of virtual couplings and is divided into two parts. The internal control system is a local controller which assigns virtual couplings between each satellite and the target orbit. The external control system assigns virtual couplings between the satellites in the constellation in a similar way as Chapters 3 and 4. Stability is proven using the energy function of the closed-loop system as a Lyapunov function candidate. The main differences with respect to Chapters 3 and 4 is that satellites are lossless systems subject to a nonlinear gravitational potential and that the graph. modeling the interaction topology is cyclic.

For satellite constellations the limited availability of propellant asks for energy-efficient control schemes. Insight into the controller's power requirements and energy consumption is inherent to the port-Hamiltonian framework making use of the energy functions (see Remark 5.5).

### Utilization within the ROSE project

The algorithms in this chapter achieve orbital phasing for satellite constellations on circular target orbits, the main application area being Global Navigation Satellite Systems. The algorithms presented in this chapter provide a first step for the application of virtual couplings in space applications on planetary orbits. The next step is to generalize these results to the formation flying of satellites in deep space. This is of particular interest for the Far-InfraRed Interferometry (FIRI) mission of ESA and SRON, where the goal is to achieve a high angular resolutions (in the order of a few arcseconds) to investigate wavelengths between 25 and 300 micron. Achieving such high resolutions requires huge baselines, which can only be achieved using tight formation control of the satellites in the constellation.

The next section provides the dynamical modeling of the satellite constellation in the port-Hamiltonian framework.

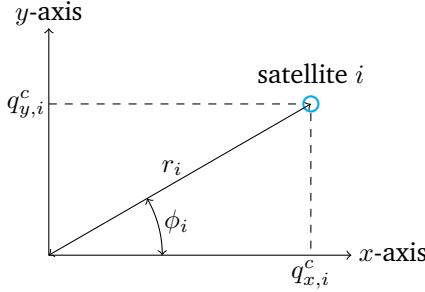
## 5.2 Dynamical model

Consider an earth-centered inertial frame of reference in Cartesian coordinates, where the origin of the inertial frame is at the center of planet earth. The  $z$ -axis of the reference frame is assumed to be normal to the orbital plane of interest. The interest here is merely on the satellite dynamics in the orbital plane, hence the dynamics along the  $z$ -axis are omitted. Each satellite is modeled as a point mass which is subject to the gravitational field of planet earth. Let  $q_i^c = (q_{x,i}^c, q_{y,i}^c)^T$  denote the position and  $p_i^c = (p_{x,i}^c, p_{y,i}^c)^T$  the momentum in Cartesian coordinates of satellite  $i$  in the earth-centered inertial frame of reference (see Figure 5.1). Each satellite is modeled as a point  $m_i$  moving in the orbital plane. Setting  $n = 2$  and  $D_i^a = 0$  the dynamics of satellite  $i$  are equal to the fully actuated agent dynamics (3.1) and are given by

$$\begin{pmatrix} \dot{q}_i^c \\ \dot{p}_i^c \end{pmatrix} = \begin{pmatrix} 0 & I_2 \\ -I_2 & 0 \end{pmatrix} \begin{pmatrix} \frac{\partial H_i^c}{\partial q_i^c} \\ \frac{\partial H_i^c}{\partial p_i^c} \end{pmatrix} + \begin{pmatrix} 0 \\ I_2 \end{pmatrix} f_i^c, \quad (5.1)$$

$$v_i^c = \frac{\partial H_i^c}{\partial p_i^c},$$

with input force  $f_i^c = (f_{x,i}^c, f_{y,i}^c)^T$ , and output velocity  $v_i^c = (v_{x,i}^c, v_{y,i}^c)^T$ . The Hamiltonian  $H_i^c(q_i^c, p_i^c)$  is the sum of the kinetic energy stored in satellite mass  $m_i$  and the gravitational potential energy. Here, planet earth is assumed to be a perfect sphere (i.e., deviations like the  $J_2$ -perturbation are neglected) [2] and its kinetic



**Figure 5.1:** Position of satellite  $i$  in Cartesian coordinates  $(q_{x,i}^c, q_{y,i}^c)$  and polar coordinates  $(r_i, \phi_i)$  in an earth-centered inertial frame.

and gravitational potential energy is modeled as

$$H_i^c(q_i^c, p_i^c) = \underbrace{\frac{(p_{x,i}^c)^2}{2m_i} + \frac{(p_{y,i}^c)^2}{2m_i}}_{\text{kinetic energy}} - \underbrace{\frac{\mu_e m_i}{\|q_i^c\|}}_{\text{gravitational potential energy}}, \quad (5.2)$$

with  $\|q_i^c\|$  the distance with respect to the center of planet earth, which is defined as  $\|q_i^c\| = \sqrt{(q_{x,i}^c)^2 + (q_{y,i}^c)^2}$ .

In contrast with most energy functions, Hamiltonian (5.2) is not bounded from below and therefore  $H_i^c(q_i^c, p_i^c)$  can not be directly used as a Lyapunov function candidate. However, from the physics it follows that  $\|q_i^c\| > R_E > 0$ , with  $R_E$  the radius of planet earth, (see Assumption 5.1). Thus  $H_i^c(q_i^c, p_i^c)$  is in fact bounded from below. Moreover,  $H_i^c(q_i^c, p_i^c)$  has multiple critical points which might give rise to undesired equilibria. The critical points for  $H_i^c(q_i^c, p_i^c)$  are defined as those  $q_i^c, p_i^c$  for which  $\frac{\partial H_i^c}{\partial q_i^c} = \frac{\partial H_i^c}{\partial p_i^c} = 0$ .

In order to facilitate the control design of equal distribution on a circular orbit polar coordinates are employed. Let  $r_i \in \mathbb{R}$  denote the radial distance and  $\phi_i \in [0, 2\pi]$  the azimuthal angle (see Figure 5.1), and let  $p_i \in \mathbb{R}$ ,  $h_i \in \mathbb{R}$  denote the corresponding (angular) momenta. Transforming (5.1) from Cartesian to polar coordinates is achieved by a canonical coordinate transformation  $(q_{x,i}^c, q_{y,i}^c, p_{x,i}^c, p_{y,i}^c) \mapsto (r_i, \phi_i, p_i, h_i)$  which is given by [98]

$$\begin{pmatrix} q_{x,i}^c \\ q_{y,i}^c \end{pmatrix} = L_i(r_i, \phi_i) = \begin{pmatrix} r_i \cos \phi_i \\ r_i \sin \phi_i \end{pmatrix},$$

$$\begin{pmatrix} p_{x,i}^c \\ p_{y,i}^c \end{pmatrix} = \left( \frac{\partial L_i(r_i, \phi_i)}{\partial (r_i, \phi_i)} \right)^{-T} \begin{pmatrix} p_i \\ h_i \end{pmatrix},$$

where  $\frac{\partial L_i(r_i, \phi_i)}{\partial(r_i, \phi_i)}$  denotes the Jacobian matrix, which is given by

$$\frac{\partial L_i(r_i, \phi_i)}{\partial(r_i, \phi_i)} = \begin{pmatrix} \cos \phi_i & -r_i \sin \phi_i \\ \sin \phi_i & r_i \cos \phi_i \end{pmatrix}.$$

The dynamics (5.1) of satellite  $i$  in polar coordinates<sup>2</sup> are now given by

$$\begin{aligned} \begin{pmatrix} \dot{r}_i \\ \dot{\phi}_i \\ \dot{p}_i \\ \dot{h}_i \end{pmatrix} &= \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{pmatrix} \begin{pmatrix} \frac{\partial H_i}{\partial r_i} \\ \frac{\partial H_i}{\partial \phi_i} \\ \frac{\partial H_i}{\partial p_i} \\ \frac{\partial H_i}{\partial h_i} \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} f_i, \\ v_i &= \begin{pmatrix} \frac{\partial H_i}{\partial p_i} \\ \frac{\partial H_i}{\partial h_i} \end{pmatrix}, \end{aligned} \quad (5.3)$$

with input force  $f_i = (f_{r,i}, f_{\phi,i})^T$ , along the radial distance  $r_i$  and azimuthal angle  $\phi_i$ , and corresponding output velocity  $v_i = (v_{r,i}, v_{\phi,i})^T$ . The Hamiltonian (5.2) in polar coordinates is given by

$$H_i(r_i, p_i, h_i) = \underbrace{\frac{p_i^2}{2m_i} + \frac{h_i^2}{2m_i r_i^2}}_{\text{kinetic energy}} - \underbrace{\frac{\mu_e m_i}{r_i}}_{\text{gravitational potential energy}}. \quad (5.4)$$

Physics impose the following natural assumption on the radial distance  $r_i$ .

**Assumption 5.1.** The radial distance  $r_i$  is bounded from below by  $r_i > R_E > 0$ , with  $R_E \approx 6.371 \cdot 10^6 \text{ m}$  the radius of planet earth, for all  $i = 1, \dots, N$  satellites.

Note that Assumption 5.1 corresponds to the physical fact that satellites always fly above the surface of planet earth. It ensures that the Hamiltonian  $H_i(p_i, h_i)$  is lower bounded and that it may be used as a Lyapunov function candidate. Furthermore, it guarantees that the Jacobian  $\frac{\partial L_i(r_i, \phi_i)}{\partial(r_i, \phi_i)}$  is nonsingular.

Next, the error system of (5.3) with respect to a circular target orbit is presented.

### 5.2.1 Derivation of the error dynamics with respect to a circular orbit using generalized canonical transformations

To facilitate the controller design the error dynamics of (5.3) with respect to a circular orbit are derived. A circular orbit is characterized by its radius  $R_0$  and the corresponding angular velocity  $\omega_0 = \sqrt{\mu_e/R_0^3}$ , with  $\mu_e = 398.6004418 \cdot 10^{12} \text{ m}^3/\text{s}^2$  the gravitational constant [23]. Next a generalized canonical transformation of

---

<sup>2</sup>The remainder of this chapter uses polar coordinates. For easy of notation a  ${}^p$  superscript is omitted in the remainder.

the form (2.11) is applied to derive the error dynamics with respect to the circular orbit.

Let  $\bar{r}_i, \bar{\phi}_i, \bar{p}_i, \bar{h}_i$  denote the error state variables, which are defined by the following time-dependent coordinate transformation function  $\Phi(r_i, \phi_i, p_i, h_i)$

$$\begin{pmatrix} \bar{r}_i \\ \bar{\phi}_i \\ \bar{p}_i \\ \bar{h}_i \end{pmatrix} = \begin{pmatrix} r_i - R_0 \\ \phi_i - \omega_0 t \\ p_i \\ h_i - h_0(r_i) \end{pmatrix} =: \Phi(r_i, \phi_i, p_i, h_i), \quad (5.5)$$

with desired angular momentum  $h_{0i}(r_i) = m_i \omega_0 r_i^2$ . Furthermore define the fictitious potential  $U(h_i) = -\omega_0 h_i + c_i$ , with  $c_i$  some arbitrary constant, and set  $\beta = 0, K = S = 0$ . Substituting  $\Phi, U, \beta$  into the partial differential equation (2.13) results in

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & -\omega_0 \\ 0 & 0 & 1 & 0 & 0 \\ m_i \omega_0 r_i & 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 0 \\ -\omega_0 \end{pmatrix} = 0.$$

Hence by Theorem 2.7 this choice of  $\Phi(r_i, \phi_i, p_i, h_i, t), U_i(h_i), \beta_i$  yields a canonical transformation such that the error dynamics with respect to a circular orbit follow from (2.12) and are given by

$$\begin{pmatrix} \dot{\bar{r}}_i \\ \dot{\bar{\phi}}_i \\ \dot{\bar{p}}_i \\ \dot{\bar{h}}_i \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & \bar{c}_i(\bar{r}_i) \\ 0 & -1 & -\bar{c}_i(\bar{r}_i) & 0 \end{pmatrix} \begin{pmatrix} \frac{\partial \bar{H}_i}{\partial \bar{r}_i} \\ \frac{\partial \bar{H}_i}{\partial \bar{\phi}_i} \\ \frac{\partial \bar{H}_i}{\partial \bar{p}_i} \\ \frac{\partial \bar{H}_i}{\partial \bar{h}_i} \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} f_i, \quad (5.6)$$

$$\bar{v}_i = \begin{pmatrix} \frac{\partial \bar{H}_i}{\partial \bar{p}_i} \\ \frac{\partial \bar{H}_i}{\partial \bar{h}_i} \end{pmatrix},$$

with  $\bar{c}_i(\bar{r}_i) = 2m_i \omega_0 (\bar{r}_i + R_0)$ . The new Hamiltonian  $\bar{H}_i(\bar{r}_i, \bar{\phi}_i, \bar{p}_i, \bar{h}_i)$  is calculated as

$$\begin{aligned} \bar{H}_i(\bar{r}_i, \bar{\phi}_i, \bar{p}_i, \bar{h}_i) &= H_i(\Phi^{-1}(\bar{r}_i, \bar{\phi}_i, \bar{p}_i, \bar{h}_i, t)) + U_i(\Phi^{-1}(\bar{h}_i)) \\ &= \frac{\bar{p}_i^2}{2m_i} + \frac{(\bar{h}_i + h_{0i}(r_i))^2}{2m_i(\bar{r}_i + R_0)^2} - \frac{\mu_e m_i}{\bar{r}_i + R_0} - \omega_0 (\bar{h}_i + h_{0i}(r_i)) + c_i, \end{aligned}$$

which, using  $h_{0i}(\bar{r}_i) = m_i \omega_0 (\bar{r}_i + R_0)^2$  and  $\mu_e = \omega_0^2 R_0^3$ , can be simplified to

$$\begin{aligned}\bar{H}_i(\bar{r}_i, \bar{\phi}_i, \bar{p}_i, \bar{h}_i) &= \frac{\bar{p}_i^2}{2m_i} + \frac{\bar{h}_i^2}{2m_i(\bar{r}_i + R_0)^2} + \frac{h_{0i}(r_i)\bar{h}_i}{m_i(\bar{r}_i + R_0)^2} + \frac{h_{0i}(r_i)^2}{2m_i(\bar{r}_i + R_0)^2} \\ &\quad - \frac{\omega_0^2 R_0^3 m_i}{\bar{r}_i + R_0} - \omega_0 \bar{h}_i - \omega_0 h_{0i}(r_i) + c_i \\ &= \frac{\bar{p}_i^2}{2m_i} + \frac{\bar{h}_i^2}{2m_i(\bar{r}_i + R_0)^2} + \frac{m_i \omega_0 (\bar{r}_i + R_0)^2 \bar{h}_i}{m_i(\bar{r}_i + R_0)^2} + \frac{m_i^2 \omega_0^2 (\bar{r}_i + R_0)^4}{2m_i(\bar{r}_i + R_0)^2} \\ &\quad - \frac{\omega_0^2 R_0^3 m_i}{\bar{r}_i + R_0} - \omega_0 \bar{h}_i - m_i \omega_0^2 (\bar{r}_i + R_0)^2 + c_i \\ &= \frac{\bar{p}_i^2}{2m_i} + \frac{\bar{h}_i^2}{2m_i(\bar{r}_i + R_0)^2} - \frac{m_i \omega_0^2 (\bar{r}_i + R_0)^2}{2} - \frac{\omega_0^2 R_0^3 m_i}{\bar{r}_i + R_0} + c_i \\ &= \frac{\bar{p}_i^2}{2m_i} + \frac{\bar{h}_i^2}{2m_i(\bar{r}_i + R_0)^2} - \frac{m_i \omega_0^2}{2} \left( \frac{(\bar{r}_i + R_0)^3 + 2R_0^3}{\bar{r}_i + R_0} \right) + c_i.\end{aligned}\tag{5.7}$$

To compactly rewrite (5.6) for all  $N$  satellites denote the collocated vectors  $\bar{r} = (\bar{r}_1, \dots, \bar{r}_N)^T$ ,  $\bar{\phi} = (\bar{\phi}_1, \dots, \bar{\phi}_N)^T$ ,  $\bar{p} = (\bar{p}_1, \dots, \bar{p}_N)^T$ ,  $\bar{h} = (\bar{h}_1, \dots, \bar{h}_N)^T$ ,  $f_r = (\bar{f}_{r,1}, \dots, \bar{f}_{r,N})^T$ ,  $f_\phi = (\bar{f}_{\phi,1}, \dots, \bar{f}_{\phi,N})^T$ ,  $\bar{v}_r = (\bar{v}_{r,1}, \dots, \bar{v}_{r,N})^T$ ,  $\bar{v}_\phi = (\bar{v}_{\phi,1}, \dots, \bar{v}_{\phi,N})^T$ ,  $f = (f_r, f_\phi)^T$ ,  $\bar{v} = (\bar{v}_r, \bar{v}_\phi)^T$ ,  $\bar{\omega}_0 = \mathbb{1}_N \omega_0$  and system matrices  $M = \text{diag}(m_1, \dots, m_N)$ ,  $\bar{R}(\bar{r}) = \text{diag}(\bar{r}_1 + R_0, \dots, \bar{r}_N + R_0)$ ,  $\bar{C}(\bar{r}) = \text{diag}(\bar{c}_1(\bar{r}_1), \dots, \bar{c}_N(\bar{r}_N))$ . The error dynamics for the  $N$  satellites (5.3) in the constellation are given by

$$\begin{aligned}\begin{pmatrix} \dot{\bar{r}} \\ \dot{\bar{\phi}} \\ \dot{\bar{p}} \\ \dot{\bar{h}} \end{pmatrix} &= \begin{pmatrix} 0 & 0 & I_N & 0 \\ 0 & 0 & 0 & I_N \\ -I_N & 0 & 0 & \bar{C}(\bar{r}) \\ 0 & -I_N & -\bar{C}^T(\bar{r}) & 0 \end{pmatrix} \begin{pmatrix} \frac{\partial \bar{H}}{\partial \bar{r}} \\ \frac{\partial \bar{H}}{\partial \bar{\phi}} \\ \frac{\partial \bar{H}}{\partial \bar{p}} \\ \frac{\partial \bar{H}}{\partial \bar{h}} \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ I_N & 0 \\ 0 & I_N \end{pmatrix} f, \\ \bar{v} &= \begin{pmatrix} \frac{\partial \bar{H}}{\partial \bar{p}} \\ \frac{\partial \bar{H}}{\partial \bar{h}} \end{pmatrix}.\end{aligned}\tag{5.8}$$

The Hamiltonian  $\bar{H}(\bar{r}, \bar{\phi}, \bar{p}, \bar{h})$  for the whole constellation follows from directly from (5.7) and is given by

$$\begin{aligned}\bar{H}(\bar{r}, \bar{\phi}, \bar{p}, \bar{h}) &= \sum_{i=1}^N \bar{H}_i(\bar{r}_i, \bar{\phi}_i, \bar{p}_i, \bar{h}_i) \\ &= \frac{1}{2} \bar{p}^T M^{-1} \bar{p} + \frac{1}{2} \bar{h}^T M^{-1} \bar{R}(\bar{r})^{-2} \bar{h} \\ &\quad - \frac{1}{2} \bar{\omega}_0^T M [\bar{R}(\bar{r})^3 + 2\bar{R}_0^3] \bar{R}(\bar{r})^{-1} \bar{\omega}_0 + c,\end{aligned}\tag{5.9}$$

with  $c = \sum_{i=1}^N c_i$ .

Having derived the error dynamics (5.8) of the satellite constellation, the next section continues with the design of the control system to achieve the orbital phasing objectives on a circular orbit.

## 5.3 Orbital phasing on circular orbits

The control objective is to make all satellites in the constellation converge to a circular orbit, while phasing them on the orbit. The main idea behind this approach is to use virtual springs and dampers to control the satellites in the constellation: An *internal control system* stabilizes each satellite with respect to the circular orbit by assigning a virtual spring to  $\bar{r}$  and a virtual damper to  $\dot{\bar{r}}$  and  $\dot{\phi}$ , while an *external control system* distributes the constellation on the orbit by assigning virtual springs and dampers to the relative azimuthal angle  $\varphi$  and relative angular velocity  $\dot{\varphi}$ .

The interconnection topology captures which satellites are interconnected using a virtual coupling of the external control system. In the present approach the interaction topology is modeled as a ring graph  $\mathcal{G}(\mathcal{V}, \mathcal{E})$ , where the  $N$  nodes in  $\mathcal{V}$  refer to the internally controlled satellites in the constellation and the  $E = N$  edges in  $\mathcal{E}$  refers to the the external control systems.

In terms of (5.8) and the *relative azimuthal angle*  $\varphi = (\varphi_1, \dots, \varphi_N)^T$  the orbital phasing objectives are given by to

$$\begin{cases} \bar{r} \rightarrow 0, \\ \bar{p} \rightarrow 0, \\ \bar{h} \rightarrow 0, \\ \varphi \rightarrow \varphi^*, \end{cases} \quad \text{as } t \rightarrow \infty, \quad (5.10)$$

with  $\varphi^* = (\varphi_1^*, \dots, \varphi_N^*)^T$  the desired relative azimuthal angle for the external control systems. Note that there is no requirement on the *absolute* azimuthal angle  $\phi$ .

### 5.3.1 Internal control system

The internal control system consists of a virtual spring on  $\bar{r}$  and virtual dampers on  $\dot{\bar{r}}$  and  $\dot{\phi}$ . The dynamics of such spring-damper systems are well known (see Chapters 3 and 4) and given by

$$\begin{aligned} \dot{\bar{r}} &= \begin{pmatrix} I_N & 0 \end{pmatrix} v_{r\phi}, \\ f_{r\phi} &= \begin{pmatrix} I_N \\ 0 \end{pmatrix} \frac{\partial S^r}{\partial \bar{r}}(\bar{r}) + \begin{pmatrix} D_r & 0 \\ 0 & D_\phi \end{pmatrix} v_{r\phi}, \end{aligned} \quad (5.11)$$

with dissipation matrices  $D_r, D_\phi \in \mathbb{R}^{N \times N}$ , input  $v_{r\phi} = (v_r, v_\phi)^T \in \mathbb{R}^{2N}$  and output  $f_{r\phi} = (f_r, f_\phi)^T \in \mathbb{R}^{2N}$ . Here  $v_r$  is the radial velocity and  $v_\phi$  is the angular velocity around the center of planet earth, while  $f_r, f_\phi$  are the corresponding forces. The Hamiltonian  $S^r(\bar{r})$  of the internal control system equals the potential energy stored in the virtual springs, which is given by

$$S^r(\bar{r}) = \frac{1}{2}\bar{r}^T K_r \bar{r},$$

with virtual spring constants matrix  $K_r \in \mathbb{R}^{N \times N}$ . Each internal control system (5.11) is interconnected with (5.6) via a negative feedback interconnection

$$\begin{cases} f = -f_{r\phi} + \bar{f}, \\ v_{r\phi} = \bar{v}. \end{cases} \quad (5.12)$$

### 5.3.2 External control system

Let  $\varphi \in \mathbb{R}^N$  denote the *relative* azimuthal angle (not to be confused with the absolute azimuthal angle  $\phi$ ). The external control system assigns a virtual spring on  $\bar{\varphi} = \varphi - \varphi^*$ , with  $\varphi^*$  the desired relative azimuthal angle<sup>3</sup> and a virtual damper on  $\dot{\varphi}$ . The dynamics of the external control system are similar to (5.11) and are given by

$$\begin{aligned} \dot{\varphi} &= v_\varphi, \\ f_\varphi &= \frac{\partial S^\varphi}{\partial \bar{\varphi}}(\bar{\varphi}) + D_\varphi v_\varphi. \end{aligned} \quad (5.13)$$

with dissipation matrix  $D_\varphi \in \mathbb{R}^{N \times N}$ , input  $v_\varphi \in \mathbb{R}^N$ , and output  $f_\varphi \in \mathbb{R}^N$ .

The Hamiltonian  $S^\varphi(\bar{\varphi})$  of the internal control system equals the potential energy stored in the virtual springs, which is given by

$$S^\varphi(\bar{\varphi}) = \frac{1}{2}\bar{\varphi}^T K_\varphi \bar{\varphi},$$

with diagonal spring constants matrix  $K_\varphi \in \mathbb{R}^{N \times N}$ .

The interconnection of the external control system with the internally controlled constellation is described by the incidence matrix  $B$  of the graph, such that the coupling of satellites on the nodes and virtual couplings at the edges is given by

---

<sup>3</sup>In order for the desired relative azimuthal angles to be feasible, they need to satisfy  $\sum_{j=1}^N \varphi_j^* = 0$ . By choosing the azimuthal angle domain as  $\bar{\varphi} \in [0, 2\pi]$ , it follows that  $\varphi_j^* = \frac{2\pi}{N}$  for  $j = 1, \dots, N-1$  and  $\varphi_e^* = -\frac{2\pi(N-1)}{N}$ .

[3, 100]

$$\begin{cases} \bar{f} = \begin{pmatrix} 0 \\ -B \end{pmatrix} f_\varphi, \\ v_\varphi = \begin{pmatrix} 0 & B^T \end{pmatrix} \bar{v}. \end{cases} \quad (5.14)$$

Note that (5.14) results in an autonomous system. If required, an external control port may be added (see [100]). The next section presents the closed-loop analysis and a sufficient condition for the origin  $(\bar{r}, \bar{p}, \bar{h}, \bar{\varphi}) = 0$  to be locally asymptotically stable.

### 5.3.3 Closed-loop analysis

Combining the error system (5.8) with the internal control system dynamics (5.11) and the external control system dynamics (5.13) using couplings (5.12) and (5.14) gives the closed-loop satellite constellation dynamics

$$\begin{pmatrix} \bar{r} \\ \bar{p} \\ \bar{h} \\ \bar{\varphi} \end{pmatrix} = \begin{pmatrix} 0 & I_N & 0 & 0 \\ -I_N & -D_r & \bar{C}(\bar{r}) & 0 \\ 0 & -\bar{C}(\bar{r})^T & -(D_\phi + BD_\varphi B^T) & -B \\ 0 & 0 & B^T & 0 \end{pmatrix} \begin{pmatrix} \frac{\partial \hat{H}}{\partial \bar{r}} \\ \frac{\partial \hat{H}}{\partial \bar{p}} \\ \frac{\partial \hat{H}}{\partial \bar{h}} \\ \frac{\partial \hat{H}}{\partial \bar{\varphi}} \end{pmatrix}. \quad (5.15)$$

Note that the azimuthal angle  $\bar{\varphi}$  is eliminated, since the distribution objective (5.10) merely considers the relative azimuthal angle  $\bar{\varphi}$ . Eliminating  $\bar{\varphi}$  is straightforward, since it is a *cyclic coordinate* (i.e.,  $\frac{\partial \hat{H}}{\partial \bar{\varphi}} = 0$ ).

The Hamiltonian of the closed-loop satellite constellation is simply the sum of the subsystems Hamiltonians and is given by

$$\hat{H}(\bar{r}, \bar{p}, \bar{h}, \bar{\varphi}) = \bar{H}(\bar{r}, \bar{p}, \bar{h}) + S^r(\bar{r}) + S^\varphi(\bar{\varphi}). \quad (5.16)$$

Note that the origin  $(\bar{r}, \bar{p}, \bar{h}, \bar{\varphi}) = 0$  of (5.15) corresponds to objectives (5.10) being achieved.

Now consider an open set  $\mathcal{D}$  in which the origin  $(\bar{r}, \bar{p}, \bar{h}, \bar{\varphi}) = 0$  is the unique critical point for  $\hat{H}$  (i.e.,  $(\bar{r}, \bar{p}, \bar{h}, \bar{\varphi}) = 0$  is the only point in  $\mathcal{D}$  such that  $\frac{\partial \hat{H}}{\partial (\bar{r}, \bar{p}, \bar{h}, \bar{\varphi})} = 0$ ). In the remainder of the stability analysis this set  $\mathcal{D}$  is used to prove local asymptotic stability of the origin. First, calculate the column vector of partial

derivatives  $\frac{\partial \hat{H}}{\partial(\bar{r}, \bar{p}, \bar{h}, \bar{\varphi})}$  as

$$\begin{aligned}\frac{\partial \hat{H}}{\partial \bar{r}} &= -\bar{h}^T M^{-1} \bar{R}(\bar{r})^{-3} \bar{h} + K_r \bar{r} \\ &\quad - \bar{\omega}_0^T [M (\bar{R}(\bar{r})^3 - \bar{R}_0^3) \bar{R}(\bar{r})^{-2}] \bar{\omega}_0, \\ \frac{\partial \hat{H}}{\partial \bar{p}} &= M^{-1} \bar{p}, \\ \frac{\partial \hat{H}}{\partial \bar{h}} &= M^{-1} \bar{R}(\bar{r})^{-2} \bar{h}, \\ \frac{\partial \hat{H}}{\partial \bar{\varphi}} &= K_\varphi \bar{\varphi}.\end{aligned}\tag{5.17}$$

Now continue with the Hessians  $\frac{\partial^2 \hat{H}}{\partial(\bar{r}, \bar{p}, \bar{h}, \bar{\varphi})^2}$ , where the nonzero terms are given by

$$\begin{aligned}\frac{\partial^2 \hat{H}}{\partial \bar{r}^2} &= 3\bar{h}^T M^{-1} \bar{R}(\bar{r})^{-4} \bar{h} \\ &\quad - \bar{\omega}_0^2 [M (\bar{R}(\bar{r})^3 + 2\bar{R}_0^3) \bar{R}(\bar{r})^{-3}] + K_r, \\ \frac{\partial^2 \hat{H}}{\partial \bar{p}^2} &= M^{-1}, \\ \frac{\partial^2 \hat{H}}{\partial \bar{h}^2} &= M^{-1} \bar{R}(\bar{r})^{-2}, \\ \frac{\partial^2 \hat{H}}{\partial \bar{\varphi}^2} &= K_\varphi, \\ \frac{\partial^2 \hat{H}}{\partial \bar{r} \partial \bar{h}} &= -2M^{-1} \bar{R}(\bar{r})^{-3} \bar{W}(\bar{h}),\end{aligned}\tag{5.18}$$

with  $\bar{W}(\bar{h}) = \text{diag}(\bar{h}_1, \dots, \bar{h}_N)$ .

From Assumption 5.1 and substituting  $(\bar{r}, \bar{p}, \bar{h}, \bar{\varphi}) = 0$  into (5.18) it follows that  $\frac{\partial^2 \hat{H}}{\partial \bar{r}^2}(0) = -3\bar{\omega}_0^2 M + K_r$ ,  $\frac{\partial^2 \hat{H}}{\partial \bar{p}^2}(0) = M^{-1} > 0$ ,  $\frac{\partial^2 \hat{H}}{\partial \bar{h}^2}(0) = R_0^{-2} M^{-1} > 0$ ,  $\frac{\partial^2 \hat{H}}{\partial \bar{\varphi}^2}(0) = K_\varphi > 0$ ,  $\frac{\partial^2 \hat{H}}{\partial \bar{r} \partial \bar{h}}(0) = 0$ , which implies that the origin  $(\bar{r}, \bar{p}, \bar{h}, \bar{\varphi}) = 0$  is a local minimum for  $\hat{H}(\bar{r}, \bar{p}, \bar{h}, \bar{\varphi})$  if  $K_r > 3\bar{\omega}_0^2 M$ .

To find the critical point nearest to the origin, one should solve for  $\frac{\partial \hat{H}}{\partial \bar{r}} = 0$ . After some rewriting and noting that all matrices in (5.17) are diagonal matrices, the critical point nearest to the origin is given by

$$r_i^* = \min \left| \frac{-R_0(2k_{r,i} - 3m_i \bar{\omega}_0^2) \pm \sqrt{m_i \bar{\omega}_0^2 R_0^2 (4k_{r,i} - 3m_i \bar{\omega}_0^2)}}{2(k_{r,i} - m_i \bar{\omega}_0^2)} \right|,\tag{5.19}$$

for  $i = 1, \dots, N$ . The open set  $\mathcal{D}$ , where the origin is a unique minimum is therefore

given by

$$\mathcal{D} = \left\{ (\bar{r}, \bar{p}, \bar{h}, \bar{\varphi}) \in \mathbb{R}^{5n} \mid |\bar{r}_i| < r_i^* \text{ for all } i = 1, \dots, N \right\}. \quad (5.20)$$

Now the main result of this chapter is stated in the following theorem.

**Theorem 5.2.** *For  $K_r > 3\omega_0^2 M$ ,  $D_r > 0$ ,  $D_\phi > 0$ ,  $K_\varphi > 0$ ,  $D_\varphi \geq 0$  the solutions of the closed-loop system (5.15) converge to  $\bar{r} = 0, \bar{p} = 0, \bar{h} = 0, \bar{\varphi} = 0$ . Hence objectives (5.10) are achieved for (5.3) on the set  $\Omega = \left\{ (\bar{r}, \bar{p}, \bar{h}, \bar{\varphi}) \in \mathcal{D} \mid \hat{H}(\bar{r}, \bar{p}, \bar{h}, \bar{\varphi}) \leq H^* \right\}$ , for some  $H^* > 0$ .*

*Proof.* Note that achieving objectives (5.10) corresponds to asymptotic stability of the origin  $(\bar{r}, \bar{p}, \bar{h}, \bar{\varphi}) = 0$  of (5.15). The proof of Theorem 5.2 is obtained by invoking LaSalle's invariance principle (Theorem 2.4). Substituting  $(\bar{r}, \bar{p}, \bar{h}, \bar{\varphi}) = 0$  into (5.17) shows directly that  $(\dot{\bar{r}}, \dot{\bar{p}}, \dot{\bar{h}}, \dot{\bar{\varphi}}) = 0$  is an equilibrium for (5.15). Calculating the time derivative  $\dot{\hat{H}}(\bar{r}, \bar{p}, \bar{h}, \bar{\varphi})$  it is easily seen that

$$\dot{\hat{H}}(\bar{r}, \bar{p}, \bar{h}, \bar{\varphi}) = -\frac{\partial^T \hat{H}}{\partial \bar{p}} D_r \frac{\partial \hat{H}}{\partial \bar{p}} - \frac{\partial^T \hat{H}}{\partial \bar{h}} (D_\phi + BD_\varphi B^T) \frac{\partial \hat{H}}{\partial \bar{h}}.$$

Let  $\mathcal{S} = \left\{ (\bar{r}, \bar{p}, \bar{h}, \bar{\varphi}) \in \mathcal{D} \mid \dot{\hat{H}}(\bar{r}, \bar{p}, \bar{h}, \bar{\varphi}) = 0 \right\}$ . To find  $\mathcal{S}$ , note that from (5.17) it follows that  $\bar{p} = 0, \bar{h} = 0$ . Hence  $\mathcal{S} = \left\{ (\bar{r}, \bar{p}, \bar{h}, \bar{\varphi}) \in \mathcal{D} \mid \bar{p} = 0, \bar{h} = 0 \right\}$ . Let  $(\bar{r}(t), \bar{p}(t), \bar{h}(t), \bar{\varphi}(t))$  be a solution that belongs to  $\mathcal{S}$ :

$$\bar{p}(t) \equiv 0 \Rightarrow \dot{\bar{p}}(t) \equiv 0 \Rightarrow \frac{\partial \hat{H}}{\partial \bar{r}} \equiv 0 \Rightarrow \bar{r}(t) \equiv 0 \text{ in } \Omega,$$

where the last part follows from the fact that  $(\bar{r}, \bar{p}, \bar{h}, \bar{\varphi}) = 0$  is the only critical point for  $\hat{H}(\bar{r}, \bar{p}, \bar{h}, \bar{\varphi})$  in  $\Omega$  by definition. Furthermore

$$\bar{h}(t) \equiv 0 \Rightarrow \dot{\bar{h}}(t) \equiv 0 \Rightarrow B \frac{\partial \hat{H}}{\partial \bar{\varphi}} \equiv 0 \Rightarrow BK_\varphi \bar{\varphi} \equiv 0 \text{ in } \Omega. \quad (5.21)$$

From (5.21) it follows that  $\bar{\varphi} \in \ker B$ , which implies for a undirected ring graph that  $\bar{\varphi} \in \text{im } \mathbf{1}_N$ . Furthermore, since  $\sum_{j=1}^N \bar{\varphi}_j = 0$  it follows that within the set  $\mathcal{S}$  to should hold that  $\bar{\varphi}(t) \equiv 0$ . Hence the smallest invariant set is given by  $(\bar{r}, \bar{p}, \bar{h}, \bar{\varphi}) = 0$  and by LaSalle's invariance principle (Theorem 2.4) the origin of (5.15) is asymptotically stable and hence objectives (5.10) are achieved, thereby completing the proof.  $\square$

*Remark 5.3* (Tuning controller gains). The ratio between the spring constants  $K_r$  and  $K_\phi$  determines whether internal stabilization w.r.t. the orbit or spatial distribution gets a higher priority. Careful tuning of  $K_r$  and  $K_\phi$  is required to prevent large energy consumption due to satellites moving far away from their orbit (see Remark 5.5).

*Remark 5.4* (Control input). The satellites' control inputs follow directly from (5.8), (5.11)-(5.14).

$$\begin{aligned} f_r &= -\underbrace{K_r(r - R_0)}_{\text{spring force}} - \underbrace{D_r M^{-1} p}_{\text{damping force}}, \\ f_\phi &= -\underbrace{B K_\varphi (\varphi - \varphi^*)}_{\text{spring force}} - \underbrace{(D_\phi + BD_\varphi B^T) H(r, h)}_{\text{damping forces}}, \end{aligned} \quad (5.22)$$

where  $H(r, h) = M^{-1} R(r)^{-2} (h - \omega_0 M R(r) r)$ , with  $R(r) = \text{diag}(r_1, \dots, r_N)$ . Note that (5.22) has a clear physical interpretation in terms of spring and damping forces.

*Remark 5.5* (Energy consumption). The energy the controller consumes follows directly from the power supplied to each satellite, which is simply the product of the inputs  $f_{r,i}, f_{\phi,i}$  and output  $v_{r,i}, v_{\phi,i}$ . Hence the total power supplied at time  $t$  is given by

$$P(t) = \sum_{i=1}^N |f_{r,i}^T(t)v_{r,i}(t)| + \sum_{i=1}^N |f_{\phi,i}^T(t)v_{\phi,i}(t)|. \quad (5.23)$$

The total energy consumption equals the time-integral of the supplied power (5.23) and is given by  $E(t) = \int_0^t |P(\tau)| d\tau$ . Using the power (5.23) and the related energy consumption allows for an easy comparison between different controller settings (see the simulation results in the next section for a worked example).

### 5.3.4 Simulation results

To illustrate the effectiveness of Theorem 5.2, simulation results are presented here. For the simulation, a circular orbit with an altitude of 20,000 km is considered (i.e.,  $R_0 = 20 \cdot 10^6 + R_E = 20 \cdot 10^6 + 6.371 \cdot 10^6 = 26.371 \cdot 10^6$  m). An orbit with this altitude is called a Medium Earth orbit (MEO) and is most commonly used for the navigation satellite systems mentioned in Section 5.1.

The simulation is run for  $t = 500$  hr, in which the satellites encircle planet Earth approximately 42 times on the chosen MEO orbit (the period of the orbit is  $T_0 = \frac{2\pi}{\omega_0} = 11.83$  hr). The constellation consists of  $N = 8$  satellites with mass  $m_i = 2$  kg. The controller gains are set at  $k_{r,i} = 10$ ,  $d_{r,i} = 15$ ,  $d_{\phi,i} = 10$  for the  $i = 1, \dots, 8$  internal control systems and  $k_{\varphi,j} = 5$ ,  $d_{\varphi,j} = 10$  for the  $j = 1, \dots, 8$  external control systems. These values were found after some trial-and-error.

The initial conditions are given in Table 5.1. Note that these initial conditions were chosen to clearly illustrate the simulation results. From a practical point of view they are considered to be quite far away from the desired position on the orbit. However, in this simulation  $K_r \gg 3M\omega_0$  ( $k_{r,i} = 10$ ,  $3m_i\omega_0 = 3m_i\sqrt{\mu_e/R_0^3} =$

satellite	initial state [unit]				$h [10^6 \text{ kg km}^2/\text{hr}]$
	$r [10^3 \text{ km}]$	$\phi [\text{rad}]$	$p [\text{kg km/hr}]$	$\dot{r} [\text{km/hr}]$	
1	27.00	0.29	0	-0.0001	-26.93
2	27.18	1.08	0	-0.0001	-34.79
3	25.62	1.36	0	-0.0001	30.82
4	27.20	2.65	0	-0.0001	-35.67
5	26.63	3.43	0	-0.0001	-11.00
6	25.57	3.92	0	-0.0001	32.83
7	25.93	4.90	0	-0.0001	18.24
8	26.46	5.27	0	-0.0001	-3.79

**Table 5.1:** Initial conditions for the simulation.

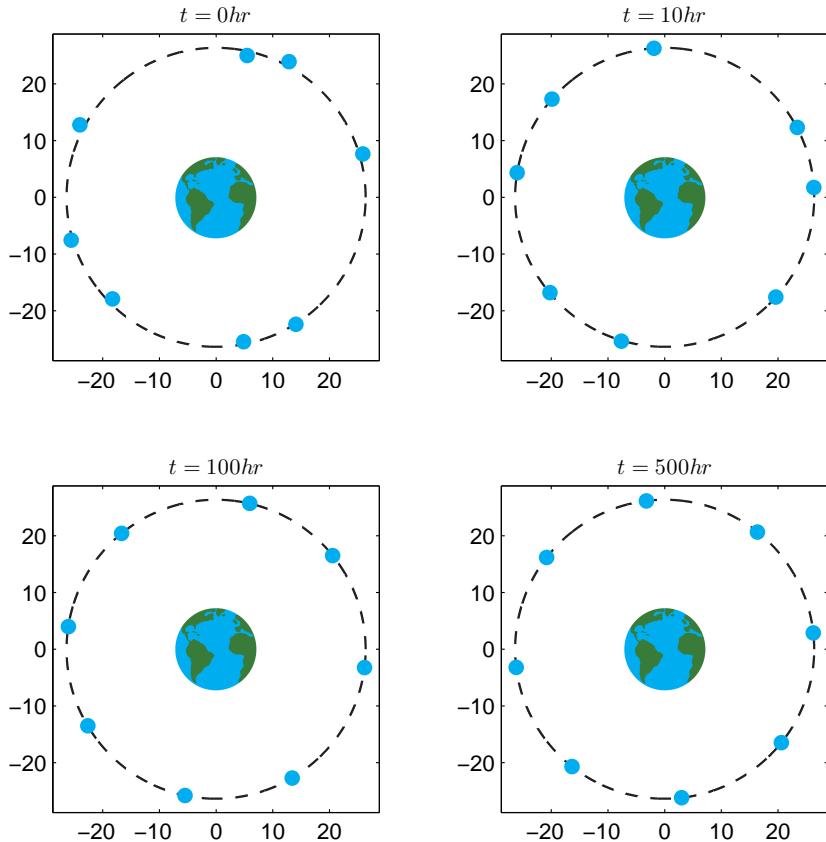
$8.8457 \cdot 10^{-4}$  for all  $i = 1, \dots, 8$ ), such that the convergence is still quite fast. The simulation results are shown in Figures 5.2, 5.3, 5.4, and 5.5.

Figure 5.2 shows four snapshots of the satellite positions at times  $t = \{0, 10, 100, 500\} \text{ hr}$ . Note that in all four figures, the satellites have a counterclockwise movement on the orbit. Initially (top left) the satellites are neither on the orbit, nor are they equally distributed on the orbit. After  $t = 500 \text{ hr}$  the satellites have converged to the circular orbit and are phased on the orbit.

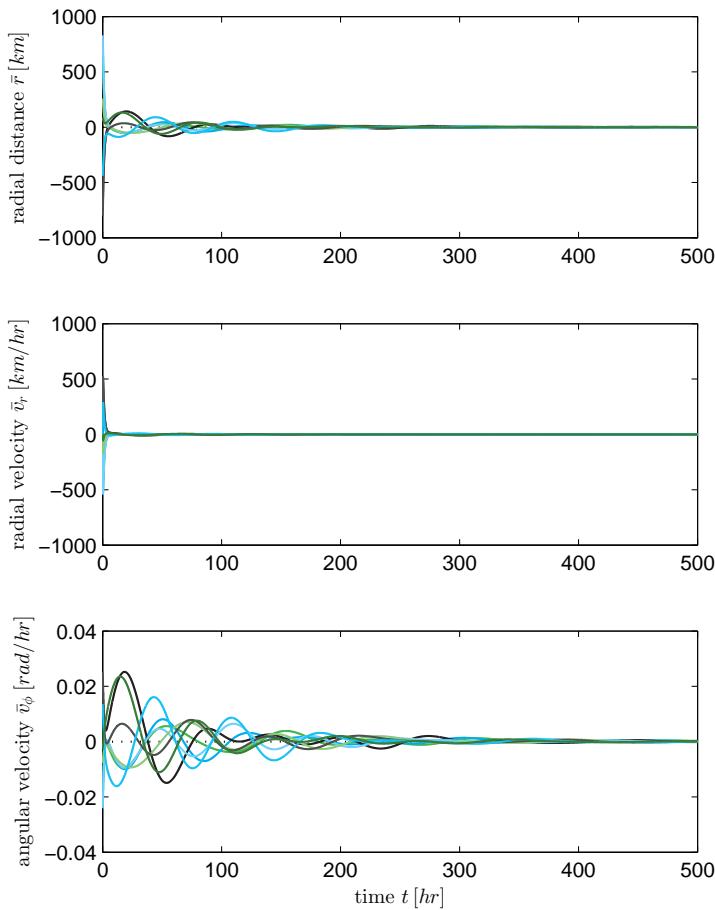
Due to the scaling of Figure 5.2 it is not immediately seen that the orbital phasing objectives (5.10) are achieved. For clarification Figure 5.3 shows the time evolution of the radial distance  $\bar{r}$  (top), radial velocity  $\bar{v}_r$  (middle) and angular velocity  $\bar{v}_\phi$  (bottom), while Figure 5.4 shows the time evolution of the azimuthal angle  $\bar{\phi}$ . Here, Figures 5.3 and 5.4 correspond to the first three objectives and the final objective in (5.10) respectively. Note that Figure 5.3 shows the *error* variables defined in (5.5), such that convergence to the origin corresponds to achievement of first three objectives in (5.10). For the azimuthal angle depicted in Figure 5.4 it is easily seen that the angle distributes evenly on the orbit (i.e.,  $\phi_{i+1} - \phi_i \rightarrow \frac{\pi}{4}$  for  $i = 1, \dots, 7$  and  $\phi_1 - \phi_8 \rightarrow -\frac{14\pi}{8}$  as  $t \rightarrow \infty$ ).

From Figures 5.3 and 5.4 and (5.22) it follows that the control inputs converge to zero and no control effort is needed once the satellites are phased on the orbit. Furthermore, Figure 5.5 shows the total power (5.23) supplied to each satellite by the internal and external control system. The  $y$ -axis is cut off at  $1.6W$  for clarity. The maximal power usage ranges from  $1.30 \text{ W}$  for satellite 8 to  $24.80 \text{ W}$  for satellite 4. Looking at micro propulsion systems like the *T3 Cold-Gas Micropulsion system*<sup>4</sup>, these control inputs are well within the achievable thrust levels. On the other hand, considering electrical propulsion systems the peak power (see Figure 5.5) is well within the range of modern power supplies (e.g. solar power).

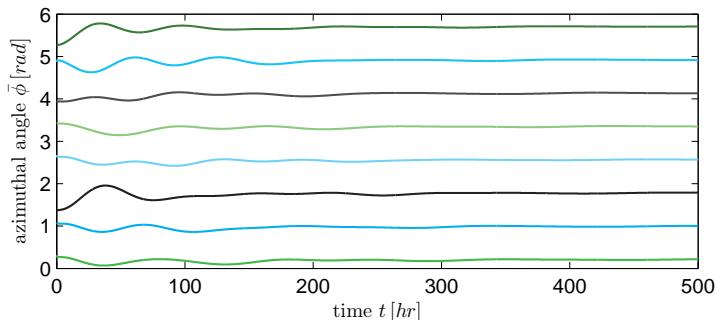
<sup>4</sup>[www.lr.tudelft.nl/organisatie/afdelingen/space-engineering/space-systems-engineering/expertise-areas/space-propulsion/propulsion-research/t3-mps/](http://www.lr.tudelft.nl/organisatie/afdelingen/space-engineering/space-systems-engineering/expertise-areas/space-propulsion/propulsion-research/t3-mps/)



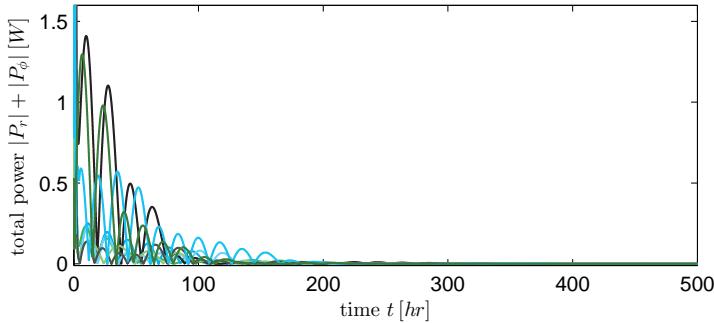
**Figure 5.2:** Snapshots of the satellites' position in the orbital plane at  $t = \{0, 10, 100, 500\} \text{ hr}$ . The blue circles represent the satellites, while the dashed line indicates the circular orbit with radius  $R_0$ . The orbital motion is counterclockwise. All axis are in  $1000\text{km}$ .



**Figure 5.3:** Time evolution of the radial distance  $\bar{r}$ , radial velocity  $\bar{v}_r$ , and angular velocity  $\bar{v}_\phi$ . The dotted lines show the reference values.



**Figure 5.4:** Time evolution of the azimuthal angle  $\bar{\phi}$ .



**Figure 5.5:** Time evolution of the total power  $|P_r| + |P_\phi|$ .

## 5.4 Concluding remarks

This chapter presents a port-Hamiltonian approach to the orbital phasing problem of satellite constellations on circular orbits. A dynamical model is obtained and using generalized canonical transformations the error system with respect to the target orbit is derived. The control objective of orbital phasing is achieved by assigning virtual couplings in between each satellite and the orbit as well as virtual couplings in between the satellites. Under a mild condition on one of the virtual spring constants the orbital phasing objectives are achieved and a complete stability proof is given. Simulation results show that the control systems fits well within the specification of modern satellite systems.

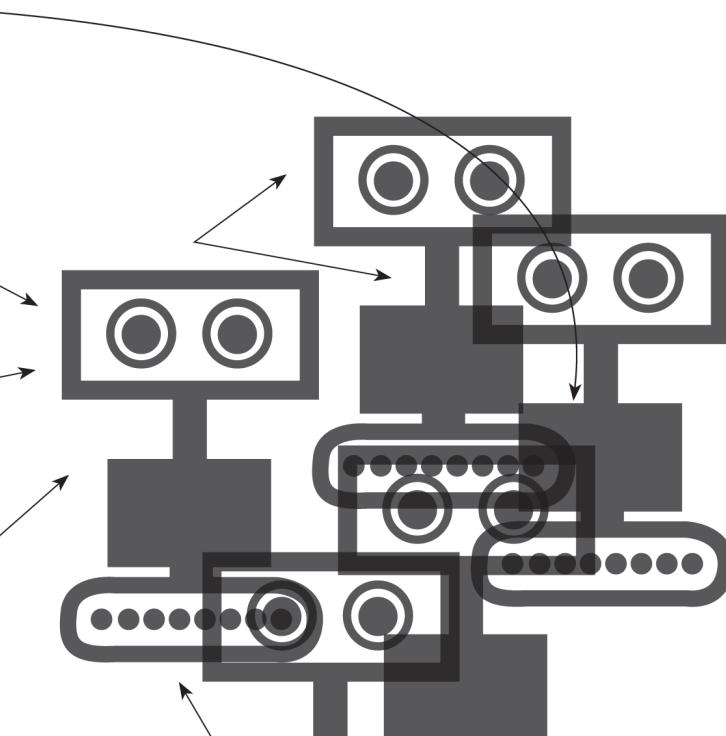
The algorithms for orbital phasing in this chapter use a similar approach as the formation control algorithms in Chapters 3 and 4. All algorithms use virtual couplings to shape the total energy of the network and inject damping for stability, while the physical structure is exploited in the analysis and design.





# Chapter 6

Conclusions and future research





# Chapter 6

---

## Conclusions and future research

Chapter 1 presents the (theoretical) background for this thesis and the corresponding application areas. The main results of this thesis are presented in Chapter 3 (formation control of fully actuated systems), Chapter 4 (formation control of wheeled robots), and Chapter 5 (orbital phasing of satellites). The present chapter reflects on these results and points out some directions for future research.

### 6.1 Conclusions

In 2010 the ROSE project started with the ambition to develop a robotic sensor network for the inspection and monitoring of dikes. The project encompasses the development of a mobile sensor-equipped robot by the University of Twente and the design of coordination algorithms for a network of such robotic sensors by the University of Groningen. During several user committee meetings the academic and industrial partners (see Appendix B) discussed the progress of the project and set directions for the research. This section reflects on the contributions of this thesis on the design of coordination algorithms.

The innovative paper “Passivity as a design tool for group coordination” [3] served as a starting point for the results of this thesis. The passivity-based design approach in [3] is extended to networks of port-Hamiltonian systems. Using generalized canonical transformations [45] the internal feedback for velocity tracking is designed explicitly for networks of wheeled robots (Section 4.4) and satellites (Chapter 5). Furthermore, the concept of virtual springs for the external feedback provides a clear physical intuition of the control action. The addition of virtual dampers provides more design freedom for the external feedback.

Port-Hamiltonian systems theory has proven its value from a theoretical and a communication point of view. Exploiting the (physical) structure of the models provides more insight, while robustness is inherited from the intrinsic physical robustness of the components. The physical interpretation of the models also facilitates clear presentations at international conferences and serves as a common ground amongst collaborators.

The virtual coupling concept has proven to be versatile, by applying it to formation control of networks of different types of systems. The three types of

systems considered in this thesis are in different ways all aligned with utilization in the ROSE project. Fully actuated systems represent the legged robot, which is developed by the University of Twente. Nonholonomic wheeled robots are considered to test the algorithms on an existing testbed of e-puck robots. Finally, the satellites considered in Chapter 5 provide a first step towards formation flying algorithms for satellites in the future FIRI mission (Section 1.2.3).

The interconnection topology of the robots and the virtual couplings is assumed to be undirected, connected and acyclic (i.e., the topology is a tree graph). These types of graphs enable a broad class of formation shapes relevant for the dike inspection application, including the line formation for coverage [52, 92] and the star formation for borehole inspection [20].

Although non-smooth Hamiltonians have been considered before [46], the non-smooth analysis tools presented in Chapter 3 (Krasovskii notion of solution, generalized Clarke gradient, non-smooth version of LaSalle's invariance principle)) have not been considered in port-Hamiltonian systems theory before. Furthermore, the internal-model-based controller design for matched input disturbance rejection in the context of formation control (Section 4.5) provides an extension of previous work for a single system [47, 48].

Finally, experimental results (and simulation results to a lesser extent) provide a big step towards the implementation of the algorithms in practice. The physical interpretations behind the algorithms help engineers to interpret the results and provide directions for tuning the controller gains. On the other hand, the experiments give rise to new directions for research, which would not have been developed otherwise (see some of the recommendations in the next section). It is this interplay between theory, practice and engineering which is the foundation of this work.

## 6.2 Recommendations for future research

Four years of PhD research provide answers to many questions, while at the same time even more questions pop up and/or remain unanswered. This section provides some directions for future research, which are divided into four topics: overall, Chapter 3, Chapter 4, and Chapter 5.

### Overall

Previous research has shown that port-Hamiltonian system theory is instrumental to the design of energy-efficient and robust control systems. Robustness of the proposed solutions is inherited from the intrinsic robustness of physical systems like springs and dampers. Chapter 5 also shows that the power-consumption of the

orbital phasing controller is well within nowadays power supplies like solar panels. However, providing analytical solutions to the energy-efficiency problem of the algorithms was not considered and sets a challenge for future research (see (1.1)).

Another direction is relaxing the assumption that the interconnection and communication topology are the same (Section 1.1.2). In practice, the communication topology is often directed which does not align with the undirected nature of the physical interconnection structure.

All algorithms in this thesis are continuous-time laws, which in practice may be hard to implement because of hardware constraints. Recent work linking formation control to theoretical computer science provides a starting point for sampled-data control laws for applications in real-robot systems [117].

### **Formation control of fully actuated systems**

Section 3.4.3 presents a discontinuous approach to counteract the discontinuous Coulomb friction in order to achieve exact formations. While the intuition of the discontinuous virtual springs is clear, the discontinuous springs do have a (major) drawback. As the agents in the network approach the desired formation, the control action shows undesired fast switching (see Figure 3.8 (bottom)). This chattering behavior can be mitigated by the adoption of e.g. hysteretic quantizers [18] or self-triggered control algorithms [27].

Instead of using discontinuous controllers, another direction to circumvent the problem of the continuous virtual springs in Section 3.4.2 is to add an integral controller [32]. Integral control is a classical tool to eliminate steady state errors [35] and should therefore be able to counteract the formation error caused by the Coulomb friction (see Figure 3.7 (middle)). Using integral control also prevents the chattering behavior caused by the discontinuous springs, but it lacks a clear physical interpretation.

### **Formation control of nonholonomic wheeled robots**

Preliminary research has included obstacle avoidance using virtual potential fields in the formation control setting of Chapter 4 [20, 52]. The physical interpretation of the potential field method fits easily into the port-Hamiltonian framework and simulation [52] and experimental results [20] have been obtained. There is still the need for a rigorous closed-loop analysis of the potential field method presented in [20, 52] for example using similar arguments as in [96]. Another way to achieve obstacle avoidance is to introduce a time-varying heading controller in Section 4.4, which adjusts the heading of the robots as they approach obstacles.

Section 4.4 presents a local velocity tracking controller for each robot, which assumes that each robot knows the reference velocity. Simulation and experimental

results have shown that velocity tracking for the network can also be achieved by introducing two (virtual) leader robots, which “drag” the other robots in the network [52, 92]. Apart from the two leaders, the other robots only require knowledge on the relative distance with respect to their neighbors. A drawback of this approach is the slight deformation of the formation shape due to friction. Efforts have been made in analyzing the closed-loop stability of such a system using the theory of *forced Hamiltonian systems* [74]. However, no Casimir functions could be found (yet) to complete the proofs.

The velocity tracking controller (4.35) requires velocity measurements of each robot. However, experiments have shown that velocity measurements are either poor or not available [20, 92]. Using a dynamic extension to estimate the velocity [33] could overcome this problem in a straightforward manner.

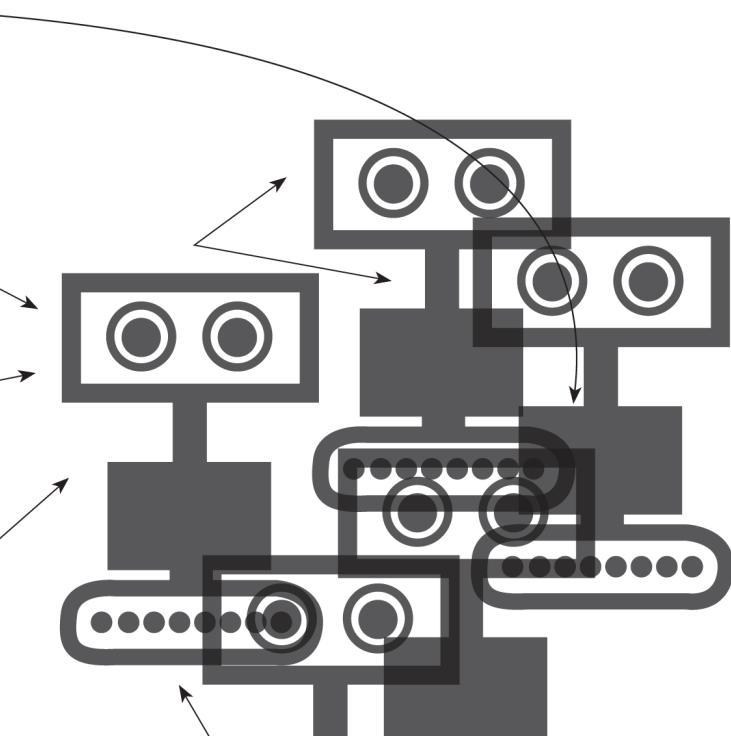
### Formation control of satellite constellations

Chapter 5 considers circular target orbits (i.e., orbits with an eccentricity equal to one). From a practical point of view, many orbits have a different eccentricities and extending the results to these elliptical orbits is therefore of interest. Another strong assumption to be relaxed is that each satellite is fully actuated. In practice each satellite has only one single thruster, which is tangential to the satellite’s orbit. *Hohmann maneuvers* have potential to achieve the desired phase shift on the orbit, but it is not yet clear how to apply this maneuver in a control setting.

Finally, for the FIRI project the satellites are no longer on planetary orbits but are positioned in deep space. Hence, the (complex) orbital dynamics may be omitted, but several types of disturbances (e.g.  $J_2$  perturbation) have to be included. One direction for future research is to investigate the robustness of the external control system of Section 5.3.2 when applied in a deep space setting.



## Appendix





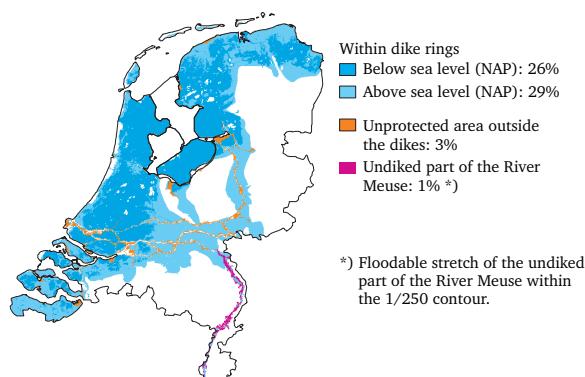
## Appendix A

---

# Background on dike inspection

A quick look at the map in Figure A.1 shows why The Netherlands are known as the “lower countries”. About 26% of the Dutch land surface is below sea level (Dutch: Normaal Amsterdams Peil (NAP)), while an additional 33% above NAP is susceptible to flooding. Out of this total of 59% of the Dutch land surface 55% is located within dike rings, while 4% is located outside dike rings and is therefore unprotected from flooding ([www.pbl.nl](http://www.pbl.nl)).

Dike rings are contiguous rings of water barriers like dunes, dikes, dams, and other artificial structures. The area within a dike ring is therefore protected from flooding. This thesis uses the word dike, when referring to any of the structures in a dike ring. Formally a dike (also known as levee, dyke or embankment) is an elongated naturally occurring ridge or artificially constructed fill or wall, which regulates water levels [89]. With most of the Dutch land surface being flood-prone and protected by dike rings, ensuring the quality and safety of dikes is of the utmost importance. The Netherlands has a long tradition in the construction and inspection of dikes and has a global reputation for its expertise in advanced water defense systems (Figure A.2). The next section provides an overview of current dike inspection methods in The Netherlands.



**Figure A.1:** Flood-prone areas within the Netherlands in 2005 (Source: PBL Netherlands Environmental Assessment Agency ([www.pbl.nl](http://www.pbl.nl))).



**Figure A.2:** Maeslantkering near Rotterdam, part of the Delta works (Source: Rijkswaterstaat).

## A.1 Dike inspection in The Netherlands

All dikes in The Netherlands can be divided into two types: primary dikes, which protect The Netherlands from water from outside (North Sea, Wadden Sea, big rivers, IJssel- and Markerlake), and regional dikes, which protect The Netherlands from water from inside (lakes, small rivers, canals). In total there are about 17,500km of dikes, out of which 3,558km are primary and approximately 14,000km is regional [52]. Needless to say, this creates an enormous task for the water boards, who are responsible for inspecting and evaluating the dikes.

The Ministry of Infrastructure and the Environment (Rijkswaterstaat) created the Act on the Water Defenses (Dutch: Wet op de Waterkering) which provides standards for a maximum exceedance probability for different types of dikes. The Act on the Water Defences obliges water boards to inspect dikes every five years. Although some (sensor) technology has entered dike inspection activities (see also Section A.2), the vast majority is done via visual inspection by dike wardens (Dutch: dijkgraven) and inspectors.

Visual inspection is carried out by well-trained inspectors. Training used to be organized in a master apprentice setup, where new inspectors learn by experience under the supervision of experienced inspectors. To ensure that inspectors are qualified, in 2014 an official training program has been established to teach inspectors basic knowledge on dikes, damage, failure mechanisms, and how to perform a dike inspection unambiguously (see [www.wateropleidingen.nl](http://www.wateropleidingen.nl) for more information on the training programme). In short, visual inspection is about checking the dike surface for irregularities like holes, cracks, and shearing of the dike surface.



**Figure A.3:** The IJkdijk test facility near Bellingwolde, The Netherlands. The two basins in the front were used during the piping experiment in 2009 (Source: Stichting IJkdijk).

Looking back, it shows that visual inspections actually perform quite well. However, recent dike breaches such as in Wilnis in 2003, Stein in 2004, and Terbregge in 2003; near-disasters such as in the Betuwe in 1994, 1995 and 1996; and the evacuation in Woltersum in 2012 point out that visual inspections alone are insufficient to guarantee safety. In words of the newspaper Trouw: “*After the dike breach in Wilnis, the utility of visual inspection is open for discussion*”<sup>1</sup>. A new line of research applying sensor technology to dikes started up with the foundation of the Stichting IJkdijk with the goal to develop the *dike of the future*.

## A.2 Dike inspection using sensor technology

After Wilnis a thorough investigation demonstrated that the dike breach was caused by a combination of the dike material (peat, Dutch: veen) and severe dryness during the summer of 2003. The peat dried out and started floating on the ground water causing a shift in the dike body. Wilnis showed that there was a lack of data structure in combination with a lack of dike inspection and observation. This observation gave rise to the establishment of the *Stichting IJkdijk* in 2004.

Stichting IJkdijk was initialized by the Investment and Development Agency for the Northern Netherland (N.V. NOM), Foundation for Applied Water Research (STOWA), Foundation IDL Sensor Solutions, Deltares, and the Netherlands Organization for Applied Scientific Research (TNO). The goal of Stichting IJkdijk was the development of a “*smart dike*” using a combination of dike technology, sensor

<sup>1</sup> 2003, September 1. Gevaar voor de dijken nog niet geweken. *Trouw*. Retrieved from [www.trouw.nl](http://www.trouw.nl).

networks and monitoring systems. In 2014 Stichting IJkdijk merged with Stichting FloodControl into *Stichting FloodControl IJkdijk*. The goal of FloodControl IJkdijk is to develop international marketable monitoring systems for dikes to contribute to the improvement and renovation of Dutch and international dike management.

To enable the development and experimental validation of such sensor systems a test facility was constructed near Bellingwolde, The Netherlands (see Figure A.3). The IJkdijk facility enabled three groundbreaking experiments during the period 2007–2012. The macro stability test in 2008, piping experiment in 2009 (see Figure A.3) and all-in-one/sensor validation experiment in 2012 (see Figure 1.3) provided many new insights into dike failure mechanisms and the important role of sensor technology in the monitoring and inspection of dikes.

Current dike monitoring systems can be divided into three types of sensor systems: *in-situ* sensors, *ex-situ* sensors, and remote sensing. *In-situ* sensors are positioned inside the dike. The main advantages of *in-situ* sensors are the ability to measure at large depths and the extensive experimental validation during the IJkdijk experiments. Drawbacks include the high failure rates [20], the high investment costs and the risk for collateral damage when installing the sensors [52].

Contrary to *in-situ* sensors, *ex-situ* sensors are positioned on the dike surface, rather than inside the dike. The main advantage over *in-situ* sensors is their mobility, thereby extending their action radius significantly. Major drawbacks of *ex-situ* sensors are the limited penetration depth and the sensitivity to external disturbances [52].

*Remote* sensors are another non-invasive sensor technology. The difference with *ex-situ* sensors is that remote sensor systems are located (far) away from the dike, while *ex-situ* are positioned on the dike surface. The major advantage of remote sensing is the fast coverage of large surfaces (e.g. using satellites for deformation measurements). The main drawback is that remote sensors can only provide superficial measurements (i.e., zero penetration depth). Therefore, these systems provide a similar kind of information as visual inspection.

While for the construction of new dikes installation of static *in-situ* sensor networks might be feasible, for existing dikes it is not, due to the high costs of installation (more than 1 million euro per km of dike) and the huge number of sensors needed (17,500 km dike). An alternative inspection and monitoring strategy is to employ a network of mobile robotic sensors. Sensor-equipped autonomous mobile robots have great potential to be applied to infrastructural security like dikes [21]. It is exactly this (dike inspection using robotic sensor networks) that motivated the ROSE project (see Section 1.2.1 and Appendix B).

## Appendix B

# ROSE project partners and utilization

ROSE is a collaboration between academia and industry under the auspices of technology foundation STW. A short description of the ROSE partners and their interest in the utilization of the results is given below.

**Controllab Products B.V.** is an engineering company with a broad experience in model based design. Tailor made solutions using model based design enable companies faster and more accurate development of their machine controllers. Controllab offers a range of products, of which 20-sim is the most well-known. Simulation package 20-sim enables the modeling of multi-domain systems using equations, block diagrams, physical components and bond graphs. These models may be used for simulation, analysis, control system design, and even rapid prototyping and hardware-in-the-loop-simulations.



For the ROSE project Controllab provided their 20-sim simulation package, which is very suitable for the modeling and design of a robotic sensor network. Modeling systems using (a combination of) equations, physical components and bond graphs, nicely align with the features of the port-Hamiltonian modeling framework presented in this thesis. In this project, 20-sim was used for the modeling of a network of robotic sensors, the analysis of deployment (Section 3.5) and obstacle avoidance, and three-dimensional visualization [52].

More information: [www.controllab.nl](http://www.controllab.nl).

**DEMCON** is a high-end technology supplier with a focus on high tech systems and medical devices. Mechatronic systems engineering is the multi-disciplinary specialism of DEMCON. In accordance with customers needs, DEMCON provides support starting from proof-of-principle and pre-production up till series production. DEMCON features specialist knowledge, an international network of suppliers and advanced facilities, like assembly lines and clean rooms.



Within the ROSE project DEMCON provides staff hours and laboratory facilities for the realization of a proof-of-principle of the new locomotion system developed

in collaboration with the University of Twente (Section 1.2.1). The design and realization of the prototype is developed in close collaboration with the Robotics and Mechatronics research group at the University of Twente. DEMCON's particular interest is in developing new businesses in robotics applications.

More information: [www.demcon.nl](http://www.demcon.nl).

**ESA** (European Space Agency) is the European institute engaged with projects in the field of spaceflight, Earth investigation, space research, and development of satellite system related technology. ESA outsources the design and development of individual satellite systems to industrial partners located in the ESA member states. During the realization and test phase, satellite systems are tested at the Space Research and Technology Center (ESTEC) in Noordwijk, The Netherlands. ESA is only engaged in civilian applications and not in any military applications.



The design of an autonomous robotic sensor for dike inspection is closely related to the design of planet exploration rovers like the ExoMars rover. ESA is particularly interested in the energy-efficiency of the new robotic locomotion design, since there is only a limited amount of energy available during planetary exploration.

More information: [www.esa.int](http://www.esa.int).

**Stichting FloodControl IJkdijk** came into existence with the merge of Stichting FloodControl and Stichting IJkdijk in 2014. FloodControl IJkdijk is dedicated to the monitoring of dikes using sensor technology for inspection and review purposes. In the period 2007–2014 several groundbreaking experiments were conducted in the field of dike monitoring using sensor technology (see also Section A.2). The goal of FloodControl IJkdijk is to develop international marketable monitoring systems for dikes (so-called “smart levees”) to contribute to the improvement and renovation of Dutch and international dike management.



Robotic sensor networks fit perfectly within the scope of FloodControl IJkdijk as an innovative sensor technology for the inspection of dikes. During the course of the ROSE project, FloodControl IJkdijk provided access to the test facility at Bellingwolde (Figure A.3), as well as access to information and support about the IJkdijk. Chapters 3 and 4 present formation control algorithms for two types robotic sensors, which are enabling the coordination of robotic sensors for dike monitoring.

More information: [www.ijkdijk.nl](http://www.ijkdijk.nl).

**SRON Netherlands Institute for Space Research** develops pioneering technology and advanced space instruments for fundamental astrophysical research, Earth science and exoplanetary research. The acronym originates from its Dutch name “Stichting RuimteOnderzoek Nederland”. The institute follows four research lines: low-energy astrophysics, high-energy astrophysics, atmospheric composition and chemistry, and planetary research.



Regarding the ROSE project SRON's main interest lies in energy efficient formation control algorithms for satellites in the light of their long term project Far-InfraRed Interferometry (FIRI) (see Section 1.2.3). Although the FIRI project is currently on hold, there is other start up research with small satellite systems for atmospheric investigation which is aligned with SRON's scientific work. Chapter 5 presents a distributed control algorithm for the orbital phasing of satellites on circular orbits.

More information: [home.sron.nl](http://home.sron.nl).

**Technology Foundation STW** is the funding agency for the ROSE project. The acronym STW stands for foundation for applied sciences (Dutch: “Stichting voor de Technische Wetenschappen”). STW's goal is to transfer knowledge between technical sciences and users from industry by funding projects which bring the two together. The vast majority of STW research fits within the top sectors and the strategic agendas of the Dutch government, NWO and universities. STW's strategy is to call for proposals from the field, concerning innovative research with a high potential for utilization. Related projects are combined into programs around a central subject (e.g. the ROSE project is part of the Autonomous Sensor SYStems (ASSYS) program). Representatives of the users form a user committee, which is supervising the research in the project.



More information: [www.stw.nl](http://www.stw.nl).

**TNO** stands for Netherlands Organisation for Applied Scientific Research (Dutch: “Nederlandse Organisatie voor Toegepast Natuurwetenschappelijk Onderzoek”). The institute creates innovations by developing and applying knowledge for practical applications. In addition, TNO tests, certifies, evaluates quality and sets up new companies for innovations. The research of TNO is categorized according to five themes: industry, healthy living, defense, safety and security, and urbanization and energy.



As a founding father of the Stichting IJkdijk (which merged into FloodControl

IJkdijk in 2014), TNO has been a sparring partner in setting up the contents of the ROSE project. TNO has provided consultation about current dike inspection and sensor technologies, support at the IJkdijk test facility, and scientific support. Current interest is on remote sensing technology for dikes, where no invasive sensors are needed. An exploratory study using boreholes for dike monitoring has been carried out in [20].

More information: [www.tno.nl](http://www.tno.nl).

**University of Groningen** is represented by the research groups Discrete Technology and Production Automation (DTPA) and Systems, Control and Applied Analysis (SCAA) which are embedded in respectively the ENgineering and TEchnology institute Groningen (ENTEG) and the Johann Bernoulli Institute for Mathematics and Computer Science (JBI). Both research groups are part of the Jan C. Willems Center for Systems and Control.



ENTEG is an engineering science and technology institute that focuses on a number of processing and production sectors such as chemical processing, high tech and discrete production, and interface industries. The DTPA group develops quantitative and analytical theories and methodologies, based on mathematical models for design and control of complex industrial processes and systems. Application areas for the group are found in robotics, sensor networks, micro-assembly systems, energy systems, mechatronic systems, semi-conductor devices, and space systems.

The main goal of JBI is to perform performing research at a high international level, leading to publications in international scientific journals and a steady stream of highly qualified researchers (at PhD level) in mathematics and computer science. The SCAA group focuses on the analysis and design of complex and heterogeneous systems and optimization. Application areas are found in physical engineering systems, networked systems, and systems biology.

DTPA and SCAA have a shared interest in the modeling, analysis, and design of robotic sensor networks. The aim is to develop generic algorithms which are not only applicable to dike inspection, but to a much broader class of applications. Furthermore the groups' expertise on modeling, analysis, and design of complex multi-domain systems within the port-Hamiltonian framework has a direct connection to the content of this thesis.

More information:

[www.rug.nl/research/enteg](http://www.rug.nl/research/enteg) (ENTEG),

[www.rug.nl/research/jbi](http://www.rug.nl/research/jbi) (JBI),

[www.math.rug.nl/gcsc](http://www.math.rug.nl/gcsc) (Jan C. Willems Center for Systems and Control).

**University of Twente** is represented by the research group Robotics and Mechatronics (RaM), which is embedded in the Institute for ICT research in context (CTIT) and the Research Institute for Biomedical Technology and Technical Medicine (MIRA). RaM is application oriented and has a focus on modeling and simulation of physical systems, intelligent control, robotic actuators, computer vision and medical imaging, and embedded control systems. Application area's are in inspection robotics, medical robotics and service robotics.

The main interest for RaM in the ROSE project is the design and realization of a energy-efficient legged locomotion system for the mobile sensor-equipped robot (Section 1.2.1). In close collaboration with DEMCON the aim is to realize a prototype of the new controlled passive actuation locomotion system using the concept of Continuously Variable Transmissions (CVT).

More information: [www.ce.utwente.nl](http://www.ce.utwente.nl).



## Appendix C

---

# Experimental setup e-pucks

This appendix describes the e-puck wheeled robot and the experimental setup which is used for the execution of the experiments in Sections 3.3, 3.5, 4.3 and 4.4.

### C.1 The e-puck robot

The e-puck wheeled robot (Figure C.1) is wheeled robot developed for engineering education at university level [79]. Due to its particular design, the e-puck can also be used for experimental validation as in Sections 3.3, 3.5, 4.3 and 4.4.

The robot has a diameter of 75 mm and a height which varies with the (possibly) connected extensions. The robot structure is made from only four plastic parts: the main body, light ring and two wheels. Two stepper motors for movement are screwed onto the main body, with the wheels directly attached to the motor axes. The model parameters for the dynamical model (4.5) are determined by [20, 92] and given in Table C.1.

The e-puck on-board microcontroller embeds a 16 bit processor, 8 kB of RAM



**Figure C.1:** The e-puck robot with (right) and without (left) data matrix for localization. The blue dot in front refers to point  $(x_B, y_B)$  (Figure 4.1), while the blue dot in the center refers to point  $(x_A, y_A)$ . These blue dots are for illustration and are not part of the data matrix.

parameter	SI unit	value
mass $m$	$kg$	0.167
moment of inertia $I_{cm}$	$kg\ m^2$	$9.69 \cdot 10^{-5}$
damping coefficient $d_f$	$kg/s$	2
damping coefficient $d_\phi$	$kg\ m^2/s$	0.2
distance $d_{AB}$	$m$	0.06

**Table C.1:** Model parameters for the e-puck robot [20, 92].

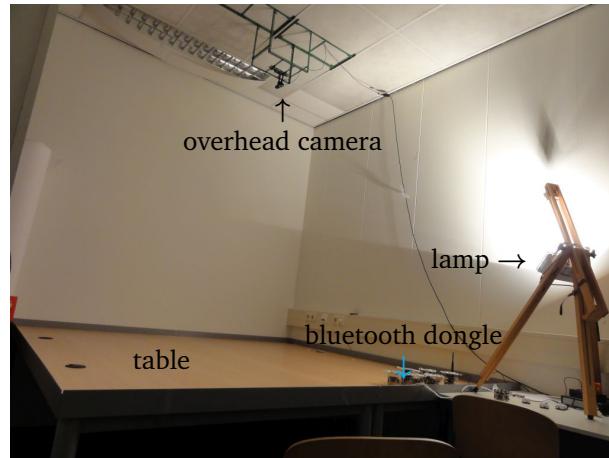
and 144 kB of flash memory. Each robot is equipped with several sensors, including infrared proximity sensors, a 3D accelerometer, microphones, and a color CMOS camera. There are several extensions available, which can equip the robot with even more sensing capabilities (e.g. the range and bearing turret depicted in Figure C.1 (left)). Actuation is provided through two stepper motors, a speaker, and light emitting diodes (LEDs). Interaction between the user and the robots is achieved by status LEDs, a connector to interface to the in-circuit debugger, infrared remote control receiver, RS232 serial port, Bluetooth radio link, and a rotary switch.

During the experiments the Bluetooth radio link was used to send control inputs to the e-puck. The two stepper motors were used to move the robot around. For more (technical) details on the e-puck robot see [79] or [www.e-puck.org](http://www.e-puck.org).

## C.2 Experimental setup

The experimental setup (Figure C.2) consists of  $2.6 \times 2.0m$  table. An overhead camera captures the robots on the table at a frame rate of 5 Hz with a resolution of  $1200 \times 1600$  pixels. A lamp with a power of 3000 Watt is used to provide a uniform non-flashing lighting of the setup (see Figure C.2). Each e-puck has a data matrix on top (see Figure C.1 (right)) which is used for the localization and identification of the robots. A vision algorithm processes the images captures by the camera and distills the ID,  $x_B$ ,  $y_B$ -position, and orientation of the robots. The newest version of the algorithm provides both the position of the front end as well as the center of the robots (see blue dots in Figure C.1).

In parallel with the vision algorithm MATLAB is running to compute the control inputs based on the localization data from the vision algorithm (see Figure C.3). The control inputs designed Chapters 3 and 4 are than transformed into a common (linear velocity) and differential (angular velocity) input, which are send to the e-puck robot via the Bluetooth radio link.



**Figure C.2:** Experimental setup for the e-puck robots.



**Figure C.3:** Computer setup running the localization algorithm, MATLAB and Bluetooth radio link to e-pucks.



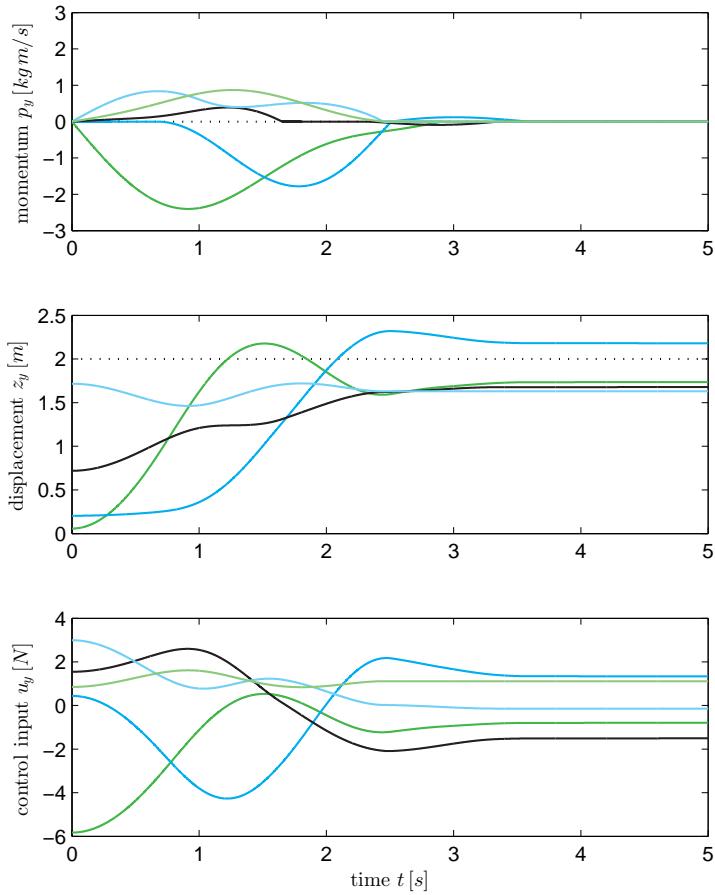
## **Appendix D**

---

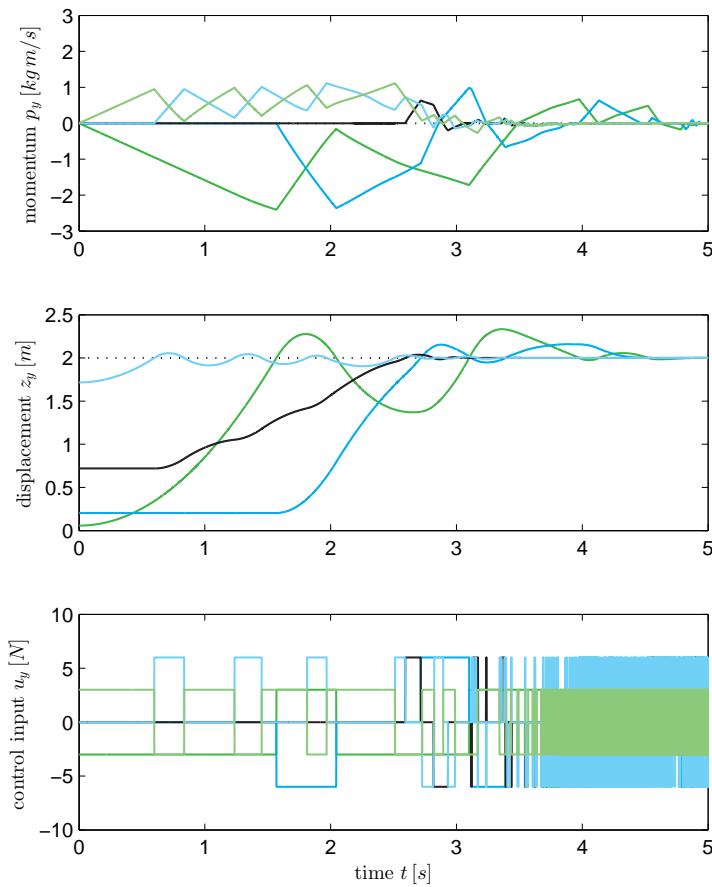
# **Complementary simulation and experimental data**

This appendix provides complementary simulation and experimental results to Sections 3.4, 4.3, 4.4, and 4.5. The figures on the following pages are organized as follows:

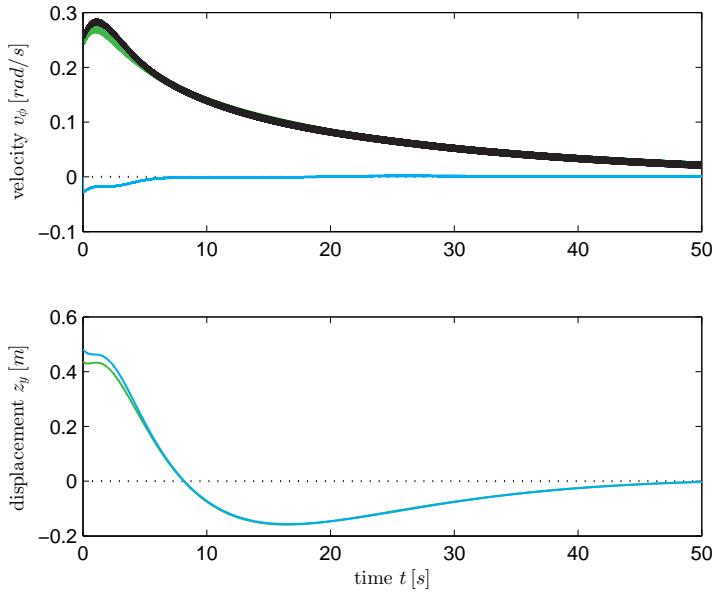
- Figures D.1 and D.2 on pages 138 and 139 complement the simulation results of Section 3.4.
- Figures D.3 and D.4 on page 140 complement the simulation and experimental results of Section 4.3.
- Figures D.5 and D.6 on page 141 complement the simulation and experimental results of Section 4.4.
- Figures D.7, D.8, and D.9 on pages 142-144 complement the simulation results of Section 4.5.



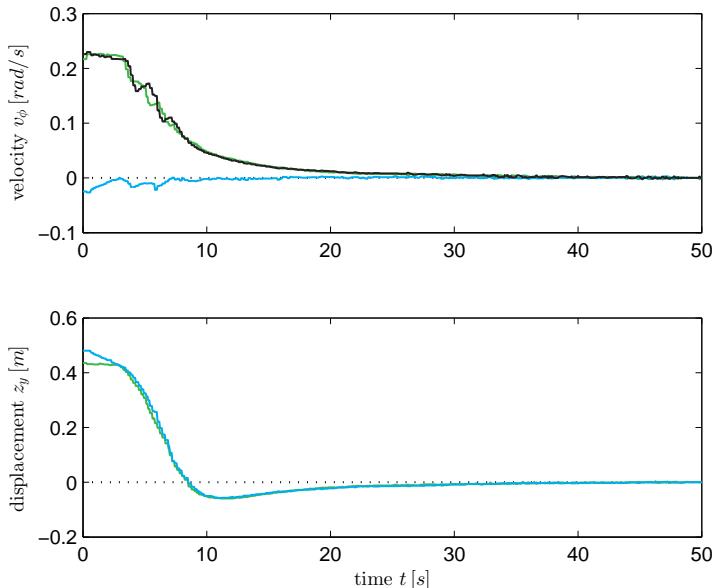
**Figure D.1:** Time evolution of the momentum  $p_y$ , relative displacement  $z_y$ , and control input  $u_y$  using continuous springs (complementary to Figure 3.7). The dotted lines show the reference values.



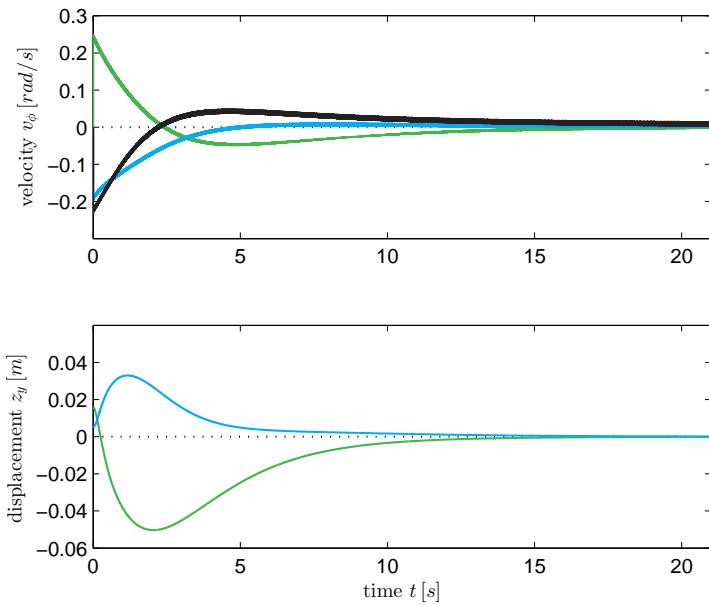
**Figure D.2:** Time evolution of the momentum  $p_y$ , relative displacement  $z_y$ , and control input  $u_y$  using discontinuous springs (complementary to Figure 3.8). The dotted lines show the reference values.



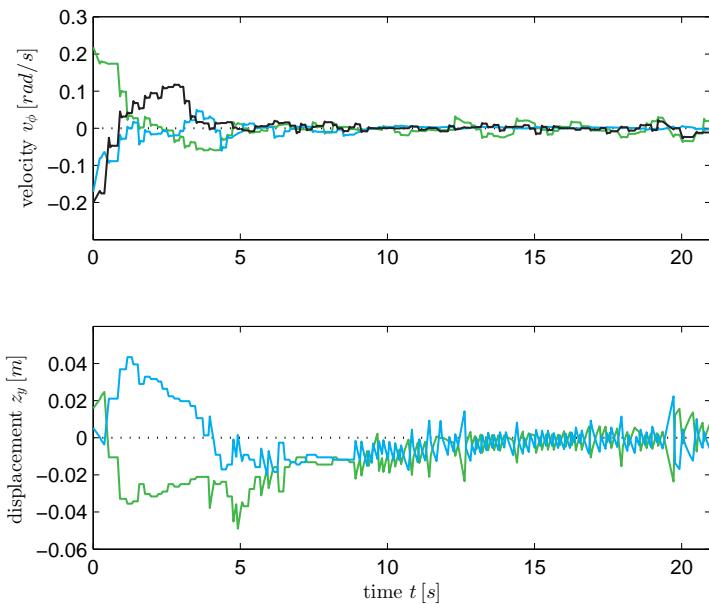
**Figure D.3:** Time evolution of the angular velocity  $v_\phi$  and relative displacement  $z_y$  for formation controller (4.19) (simulation complementary to Figure 4.4). The dotted lines show the reference values.



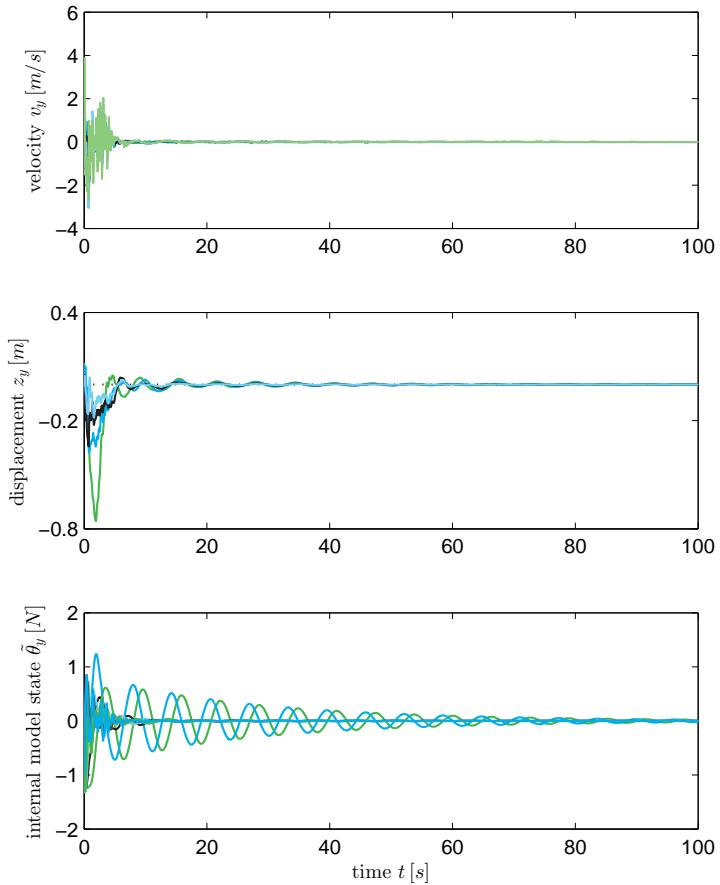
**Figure D.4:** Time evolution of the angular velocity  $v_\phi$  and relative displacement  $z_y$  for formation controller (4.19) (experiment complementary to Figure 4.5). The dotted lines show the reference values.



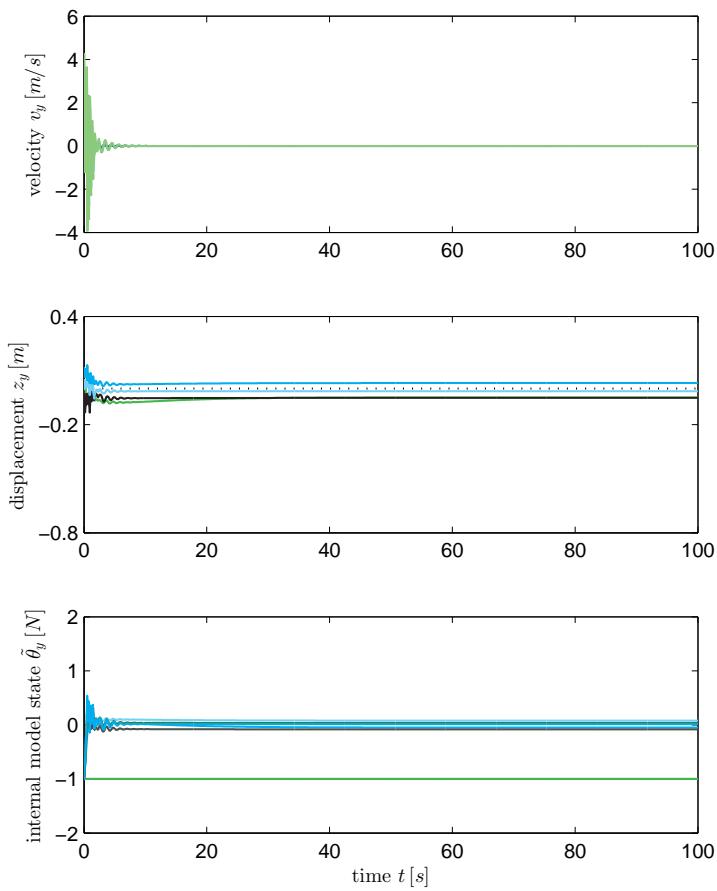
**Figure D.5:** Time evolution of the angular velocity  $v_f$  and relative distance  $z_y$  (simulation complementary to Figure 4.8). The dotted lines show the reference values.



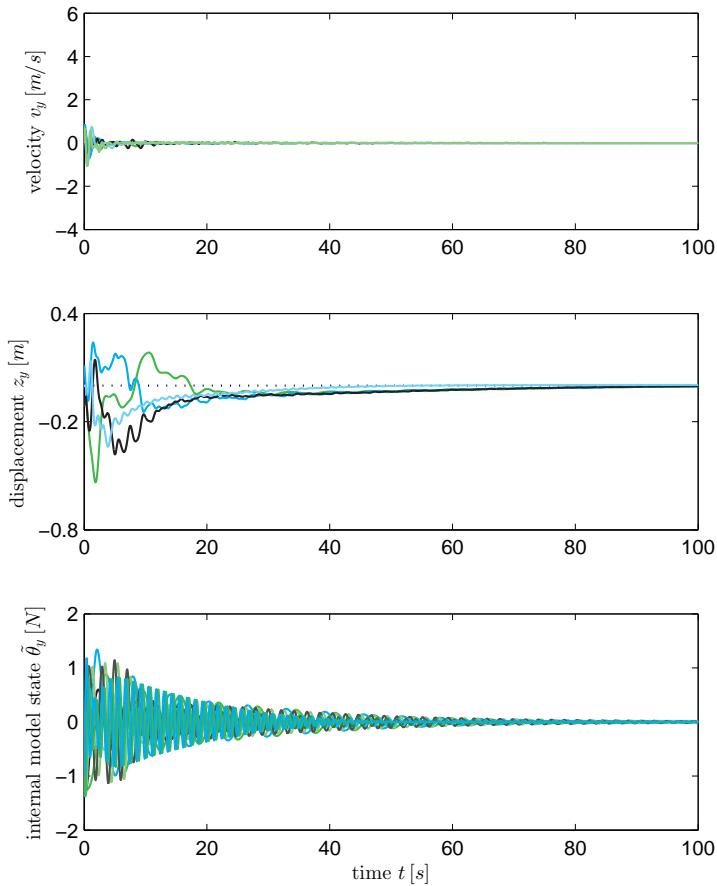
**Figure D.6:** Time evolution of the angular velocity  $v_\phi$  and relative distance  $z_y$  (experiment complementary to Figure 4.9). The dotted lines show the reference values.



**Figure D.7:** Time evolution of the velocity  $v_y$ , relative distance  $z_y$  and internal model controller state  $\tilde{\theta}_y$  in the presence of harmonic disturbances (complementary to Figure 4.11). The dotted lines show the reference values.



**Figure D.8:** Time evolution of the velocity  $v_y$ , relative distance  $z_y$  and internal model controller state  $\tilde{\theta}_y$  in the presence of constant disturbances (complementary to Figure 4.12). The dotted lines show the reference values.



**Figure D.9:** Time evolution of the velocity  $v_y$ , relative distance  $z_y$  and internal model controller state  $\tilde{\theta}_y$  in the presence of harmonic disturbances when all robots are strictly passive and  $D^v = 0$  (complementary to Figure 4.13). The dotted lines show the reference values.

---

## Bibliography

- [1] R.A. Abraham and J.E. Marsden. *Foundations of Mechanics*. Addison-Wesley, Reading, MA, 2nd edition, 1978.
- [2] K.T. Alfriend, R.V. Srinivas, P. Gurfil, J.P. How, and L.S. Breger. *Spacecraft formation flying: dynamics, control, and navigation*, volume 2. Butterworth-Heinemann, 2010.
- [3] M. Arcak. Passivity as a design tool for group coordination. *IEEE Transactions on Automatic Control*, 52(8):1380–1390, 2007.
- [4] Ö. Arslan. Design of a collaboration strategy for autonomous multi-robot systems. an application in plastic marine debris cleanup. Bachelor’s thesis, University of Groningen, 2012.
- [5] A. Astolfi. Exponential stabilization of a wheeled mobile robot via discontinuous control. *Journal of dynamic systems, measurement, and control*, 121(1): 121–126, 1999.
- [6] A. Bacciotti and F. Ceragioli. Stability and stabilization of discontinuous systems and nonsmooth Lyapunov functions. *Control Optimisation and Calculus of Variations*, 4:361–376, 1999.
- [7] R. Baheti and H. Gill. Cyber-physical systems. *The impact of control technology*, pages 161–166, 2011.
- [8] H. Bai. The undesired equilibria of formation control with ring graphs. In *American Control Conference*, Baltimore, MD, USA, 2010.
- [9] H. Bai, M. Arcak, and J. Wen. *Cooperative control design: A systematic, passivity-based approach*, volume 89. Springer, 2011.

- [10] A. Bara and S. Dale. Dynamic modeling and stabilization of wheeled mobile robot. In *WSEAS International Conference on Dynamical Systems and Control*, 2009.
- [11] B. Bollobás. *Modern graph theory*, volume 184. Springer Verlag, 1998.
- [12] P.C. Breedveld. *Modeling and control of complex physical systems: The port-Hamiltonian approach*, chapter Port-based modeling of dynamic systems. Springer, 2009.
- [13] R.W. Brockett. *Asymptotic stability and feedback stabilization*. Birkhäuser, Boston, 1983.
- [14] F. Bullo and A.D. Lewis. Geometric control of mechanical systems. In *Texts in Applied Mathematics*, volume 49. Springer-Verlag, New York-Heidelberg-Berlin, 2004.
- [15] F. Bullo, J. Cortés, and S. Martínez. *Distributed Control of Robotic Networks*. Applied Mathematics Series. Princeton University Press, 2009.
- [16] M. Bürger and C. De Persis. Dynamic coupling design for nonlinear output agreement and time-varying flow control. *Automatica*, 51:210–222, 2015.
- [17] C. Canudas-de-Wit and O.J. Sordalen. Exponential stabilization of mobile robots with nonholonomic constraints. *IEEE Transactions on Automatic Control*, 37(11):1791–1797, 1992.
- [18] F. Ceragioli, C. De Persis, and P. Frasca. Discontinuities and hysteresis in quantized average consensus. *Automatica*, 47(9):1916–1928, 2011.
- [19] J. Cervera, A.J. van der Schaft, and A. Banos. Interconnection of port-Hamiltonian systems and composition of Dirac structures. *Automatica*, 43(2):212–225, 2007.
- [20] M. Chahbari. Port-Hamiltonian modeling and control of a mobile multi robot system for borehole radar dike inspections. Master’s thesis, University of Groningen, 2014.
- [21] C.Y. Chong and S.P. Kumar. Sensor networks: Evolution, opportunities, and challenges. *Proceedings of the IEEE*, 91(8):1247–1256, 2003.
- [22] H. Choset. Coverage for robotics – A survey of recent results. *Annals of Mathematics and Artificial Intelligence*, 31(1):113–126, 2001.
- [23] S.J. Chung, U. Ahsun, and J.J.E. Slotine. Application of synchronization to formation flying spacecraft: Lagrangian approach. *Journal of Guidance Control and Dynamics*, 32(2):512–526, 2009.

- [24] J. Cortés. Finite-time convergent gradient flows with applications to network consensus. *Automatica*, 42(11):1993–2000, 2006.
- [25] J. Cortés, S. Martínez, T. Karatas, and F. Bullo. Coverage control for mobile sensing networks. *IEEE Transactions on Robotics and Automation*, 20(2):243–255, 2004.
- [26] A. Das and R. Cobb. Techsat 21-space missions using collaborating constellations of satellites. In *Conference on Small Satellites*, pages 125–129, Logan, UT, USA, 1998.
- [27] C. De Persis and P. Frasca. Robust self-triggered coordination with ternary controllers. *IEEE Transactions on Automatic Control*, 58(12):3024–3038, 2013.
- [28] C. De Persis and B. Jayawardhana. Coordination of passive systems under quantized measurements. *SIAM Journal on Control and Optimization*, 50(6):3155–3177, 2012.
- [29] C. De Persis and B. Jayawardhana. On the internal model principle in the coordination of nonlinear systems. *IEEE Transactions on Control of Network Systems*, 1(3):272–282, 2014.
- [30] J.P. Desai, J.P. Ostrowski, and V. Kumar. Modeling and control of formations of nonholonomic mobile robots. *IEEE Transactions on Robotics and Automation*, 17(6):905–908, 2001.
- [31] D.A. Dirksz. *Robust energy- and power-based control design*. PhD thesis, University of Groningen, 2010.
- [32] D.A. Dirksz and J.M.A. Scherpen. Power-based control: Canonical coordinate transformations, integral and adaptive control. *Automatica*, 48(6):1045–1056, 2012.
- [33] D.A. Dirksz and J.M.A. Scherpen. On tracking control of rigid-joint robots with only position measurements. *IEEE Transactions on Control Systems Technology*, 21(4):1510–1513, 2013.
- [34] K.D. Do and J. Pan. Nonlinear formation control of unicycle-type mobile robots. *Robotics and Autonomous Systems*, 55(3):191–204, 2007.
- [35] R.C. Dorf and R.H. Bishop. *Modern control systems*. Prentice Hall, 12th edition, 2010.
- [36] D. Dresscher and S. Stramigioli. Applying energy autonomous robots for dike inspection.

- [37] D. Dresscher, T. de Vries, and S. Stramigioli. Controlled Passive Actuation for legged locomotion.
- [38] V. Duindam, S. Stramigioli, and J.M.A. Scherpen. Passive compensation of nonlinear robot dynamics. *IEEE Transactions on Robotics and Automation*, 20(3):480–488, 2004.
- [39] V. Duindam, A. Macchelli, and S. Stramigioli. *Modeling and control of complex physical systems: The port-Hamiltonian approach*. Springer Verlag, 2009.
- [40] M.I. El-Hawwary and M. Maggiore. Distributed circular formation stabilization for dynamic unicycles. *IEEE Transactions on Automatic Control*, 58(1):149–162, 2013.
- [41] S. Fiaz, D. Zonetti, R. Ortega, J.M.A. Scherpen, and A.J. van der Schaft. A port-Hamiltonian approach to power network modeling and analysis. *European Journal of Control*, 19(6):477 – 485, 2013.
- [42] P. Frasca. Continuous-time quantized consensus: Convergence of Krasovskii solutions. *Systems & Control Letters*, 61(2):273 – 278, 2012.
- [43] K. Fujibayashi, S. Murata, K. Sugawara, and M. Yamamura. Self-organizing formation algorithm for active elements. In *IEEE Symposium on Reliable Distributed Systems*, pages 416–421, 2002.
- [44] K. Fujimoto and T. Sugie. Canonical transformation and stabilization of generalized Hamiltonian systems. *Systems & Control Letters*, 42(3):217–227, 2001.
- [45] K. Fujimoto, K. Sakurama, and T. Sugie. Trajectory tracking control of port-controlled Hamiltonian systems via generalized canonical transformations. *Automatica*, 39(12):2059–2069, 2003.
- [46] K. Fujimoto, S. Sakai, and T. Sugie. Passivity based control of a class of Hamiltonian systems with nonholonomic constraints. *Automatica*, 48(12):3054 – 3063, 2012.
- [47] L. Gentili and A.J. van der Schaft. Regulation and input disturbance suppression for port-controlled Hamiltonian systems. In *IFAC workshop on Lagrangian and Hamiltonian methods for nonlinear control*, Seville, Spain, 2003.
- [48] L. Gentili, A. Paoli, and C. Bonivento. Input disturbance suppression for port-Hamiltonian systems: An internal model approach. In *Advances in Control Theory and Applications*, pages 85–98. Springer, 2007.

- [49] R. Ghabcheloo, A. Pascoal, C. Silvestre, and I. Kaminer. Coordinated path following control of multiple wheeled robots using linearization techniques. *International Journal of Systems Science*, 37(6):399–414, 2006.
- [50] C.D. Godsil and G. Royle. *Algebraic graph theory*, volume 207. Springer New York, 2001.
- [51] F. Gómez-Estern and A.J. van der Schaft. Physical damping in IDA-PBC controlled underactuated mechanical systems. *European Journal of Control*, 10(5):451–468, 2004.
- [52] E. Halbertsma. Potential field based obstacle avoidance with multiple robotic agents in the port-Hamiltonian framework. Master’s thesis, University of Groningen, 2013.
- [53] T. Hoogerwerf. Dijken verkennen door groepen autonome robots. Bachelor’s thesis, University of Groningen, 2011.
- [54] A. Isidori, L. Marconi, and A. Serrani. *Robust autonomous guidance: an internal model approach*. Springer, 2003.
- [55] A. Jadbabaie, J. Lin, and A.S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control*, 48(6):988–1001, 2003.
- [56] M. Jafarian. A note on formation keeping control with coarse information. In *International Symposium on Mathematical Theory of Networks and Systems*, pages 1724–1727, Groningen, The Netherlands, 2014.
- [57] M. Jafarian and C. De Persis. Formation control with binary information. *Automatica*, 53:125–135, 2015.
- [58] M. Jafarian, E. Vos, C. De Persis, A.J. van der Schaft, and J.M.A. Scherpen. Formation control of a multi-agent system subject to ideal Coulomb friction, 2014. *Submitted*.
- [59] M. Jafarian, E. Vos, C. De Persis, A.J. van der Schaft, and J.M.A. Scherpen. On formation control of agents subject to ideal Coulomb friction. In *International Symposium on Mathematical Theory of Networks and Systems*, pages 1736–1739, Groningen, The Netherlands, 2014.
- [60] M. Jafarian, E. Vos, C. De Persis, J.M.A. Scherpen, and A.J. van der Schaft. Disturbance rejection in formation keeping control of nonholonomic wheeled robots, 2015. *In preparation*.

- [61] D. Jeltsema and J.M.A. Scherpen. Multidomain modeling of nonlinear networks and systems. *IEEE Control Systems Magazine*, 29(4):28–59, 2009.
- [62] D. Jeltsema, R. Ortega, and J.M.A. Scherpen. An energy-balancing perspective of interconnection and damping assignment control of nonlinear systems. *Automatica*, 40(9):1643–1646, 2004.
- [63] R. Kamps. Reinforcement learning: As a strategy in a robotic fish waterpolo competition, as a proof of concept for the enhancement of order picking performance. Bachelor’s thesis, University of Groningen, 2012.
- [64] H.K. Khalil. *Nonlinear systems*, volume 3. Prentice hall Englewood Cliffs, NJ, 2002.
- [65] R. Kristiansen and P.J. Nicklasson. Spacecraft formation flying: A review and new results on state feedback control. *Acta Astronautica*, 65(11–12):1537–1552, 2009.
- [66] D. Kurabayashi, J. Ota, T. Arai, and E. Yoshida. Cooperative sweeping by multiple mobile robots. In *IEEE International Conference on Robotics and Automation*, volume 2, pages 1744–1749, 1996.
- [67] J.R.T. Lawton, R.W. Beard, and B.J. Young. A decentralized approach to formation maneuvers. *IEEE Transactions on Robotics and Automation*, 19(6):933–941, 2003.
- [68] D. Lee. Passivity-based switching control for stabilization of wheeled mobile robots. In *Proceedings of Robotics: Science and Systems*, pages 70–77, 2007.
- [69] E.A. Lee. Cyber physical systems: Design challenges. In *IEEE International Symposium on Object Oriented Real-Time Distributed Computing*, pages 363–369, 2008.
- [70] E.Z. MacArthur and C.D. Crane. Compliant formation control of a multi-vehicle system. In *International Symposium on Computational Intelligence in Robotics and Automation*, pages 479–484, 2007.
- [71] J.E. Marsden and T.S. Ratiu. Introduction to mechanics and symmetry: a basic exposition to classical mechanics. In *London Mathematical Society Lecture Notes Series*. Springer, New York, 2nd edition, 1999.
- [72] J.A. Marshall, M.E. Broucke, and B.A. Francis. Formations of vehicles in cyclic pursuit. *IEEE Transactions on Automatic Control*, 49(11):1963–1974, 2004.

- [73] B.M. Maschke and A.J. van der Schaft. Port-controlled Hamiltonian systems: Modeling origins and system-theoretic properties. In *IFAC symposium on nonlinear control systems*, pages 282–288, Bordeaux, France, 1992.
- [74] B.M. Maschke, R. Ortega, and A.J. van der Schaft. Energy-based Lyapunov functions for forced Hamiltonian systems with dissipation. *IEEE Transactions on Automatic Control*, 45(8):1498–1502, 2000.
- [75] J. Mathews and R.L. Walker. *Mathematical Methods of Physics*. W.A. Benjamin, 2nd edition, 1970.
- [76] C.R. McInnes. Autonomous ring formation for a planar constellation of satellites. *Journal of Guidance, Control, and Dynamics*, 18(5):1215–1217, 1995.
- [77] Y. Mei, Y.H. Lu, Y.C. Hu, and C.S.G. Lee. Deployment of mobile robots with energy and timing constraints. *IEEE Transactions on Robotics*, 22(3):507–522, 2006.
- [78] M. Mesbahi and M. Egerstedt. *Graph theoretic methods in multi-agent networks*. Princeton University Press, 2010.
- [79] F. Mondada, M. Bonani, X. Raemy, J. Pugh, C. Cianci, A. Klaptocz, S. Magnenat, J.C. Zufferey, D. Floreano, and A. Martinoli. The e-puck, a robot designed for education in engineering. In *Conference on autonomous robot systems and competitions*, volume 1, pages 59–65, 2009.
- [80] S. Nair and N.E. Leonard. Stable synchronization of mechanical system networks. *SIAM Journal on Control and Optimization*, 47(2):661–683, 2008.
- [81] E. Nuño, R. Ortega, L. Basañez, and D. Hill. Synchronization of networks of nonidentical Euler-Lagrange systems with uncertain parameters and communication delays. *IEEE Transactions on Automatic Control*, 56(4):935–941, 2011.
- [82] K.J. Obermeyer, A. Ganguli, and F. Bullo. Multi-agent deployment for visibility coverage in polygonal environments with holes. *International Journal of Robust and Nonlinear Control*, 21(12):1467–1492, 2011.
- [83] P. Ögren, E. Fiorelli, and N.E. Leonard. Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment. *IEEE Transactions on Automatic Control*, 49(8):1292–1302, Aug 2004.
- [84] R. Olfati-Saber, J.A. Fax, and R.M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007.

- [85] R. Ortega, A. Loria, P.J. Nicklasson, and H. Sira-Ramirez. *Passivity-based control of Euler-Lagrange systems*, volume 388. Springer Berlin, 1998.
- [86] R. Ortega, A.J. van der Schaft, I. Mareels, and B. Maschke. Putting energy back in control. *IEEE Control Systems Magazine*, 21(2):18–33, 2002.
- [87] R. Ortega, A.J. van der Schaft, B. Maschke, and G. Escobar. Interconnection and damping assignment passivity-based control of port-controlled Hamiltonian systems. *Automatica*, 38(4):585–596, 2002.
- [88] H.M. Paynter. *Analysis and design of engineering systems*. M.I.T. Press, MA,, 1960.
- [89] H. Petroski. Levees and other raised ground. *American Scientist*, 94(1):7–11, 2006.
- [90] S.A. Pigg. *Adaptive algorithms for the rejection of sinusoidal disturbances acting on unknown plants*. PhD thesis, University of Utah, 2011.
- [91] J.B. Pomet, B. Thuiilot, G. Bastin, and G. Campion. A hybrid strategy for the feedback stabilization of nonholonomic mobile robots. In *IEEE International Conference on Robotics and Automation*, pages 129–134, 1992.
- [92] A. Postma. Port-Hamiltonian based deployment of dike robots. Master’s thesis, University of Groningen, 2013.
- [93] W. Ren. Distributed leaderless consensus algorithms for networked Euler-Lagrange systems. *International Journal of Control*, 82(11):2137–2149, 2009.
- [94] W. Ren and R.W. Beard. Decentralized scheme for spacecraft formation flying via the virtual structure approach. *Journal of Guidance, Control, and Dynamics*, 27(1):73–82, 2004.
- [95] W. Ren, R. W. Beard, and E. M. Atkins. Information consensus in multivehicle cooperative control. *IEEE Control Systems Magazine*, 27(2):71–82, 2007.
- [96] E. Rimon and D.E. Koditschek. Exact robot navigation using artificial potential functions. *IEEE Transactions on Robotics and Automation*, 8(5):501–518, 1992.
- [97] C. Samson. Time-varying feedback stabilization of car-like wheeled mobile robots. *The International journal of robotics research*, 12(1):55–64, 1993.
- [98] A.J. van der Schaft.  *$\mathcal{L}_2$ -gain and passivity techniques in nonlinear control*. Springer Verlag, 2000.

- [99] A.J. van der Schaft and D. Jeltsema. Port-Hamiltonian systems theory: An introductory overview. *Foundations and Trends in Systems and Control*, 1(2-3):173–378, 2014.
- [100] A.J. van der Schaft and B.M. Maschke. Port-Hamiltonian systems on graphs. *SIAM Journal on Control and Optimization*, 51(2):906–937, 2013.
- [101] D.P. Scharf, S.R. Ploen, and F.Y. Hadaegh. A survey of spacecraft formation flying guidance and control (Part I): Guidance. In *American Control Conference*, pages 1733–1739, Denver, CO, USA, 2003.
- [102] D.P. Scharf, F.Y. Hadaegh, and S.R. Ploen. A survey of spacecraft formation flying guidance and control (Part II): Control. In *American Control Conference*, pages 2976–2985, Boston, MA, USA, 2004.
- [103] L. Sha, S. Gopalakrishnan, X. Liu, and Q. Wang. Cyber-physical systems: A new frontier. In *Machine Learning in Cyber Trust*, pages 3–13. Springer, 2009.
- [104] G. Stacey and R. Mahony. A port-Hamiltonian approach to formation control using bearing measurements and range observers. In *IEEE Conference on Decision and Control*, pages 7641–7646, Florence, Italy, 2013.
- [105] G. Stacey, R. Mahony, and P. Corke. A bondgraph approach to formation control using relative state measurements. In *European Control Conference*, pages 1262–1267, Zürich, Switzerland, 2013.
- [106] P. Tabuada and G.J. Pappas. From nonlinear to Hamiltonian via feedback. In *IEEE Conference on Decision and Control*, volume 2, pages 1515–1520, 2002.
- [107] P. Tabuada and G.J. Pappas. Abstractions of Hamiltonian control systems. *Automatica*, 39(12):2025–2033, 2003.
- [108] Y. Ulybyshev. Long-term formation keeping of satellite constellation using linear-quadratic controller. *Journal of Guidance, Control, and Dynamics*, 21(1):109–115, 1998.
- [109] G.J.O. Vermeer. 3-D symmetric sampling. *Geophysics*, 63(5):1629–1647, 1998.
- [110] E. Vos, J.M.A. Scherpen, and A.J. van der Schaft. Port-Hamiltonian approach to deployment. In *International Symposium on Mathematical Theory of Networks and Systems*, Melbourne, Australia, 2012.

- [111] E. Vos, S. Fiaz, J.M.A. Scherpen, and A.J. van der Schaft. Control for formation flying of satellites: port-Hamiltonian approach. In *International Workshop on Satellite Constellations and Formation Flying*, Lisbon, Portugal, 2013. Paper ID: IWSCFF-2013-02-02.
- [112] E. Vos, J.M.A. Scherpen, and A.J. van der Schaft. Spatial distribution of satellite constellations on circular orbits. In *IEEE Conference on Decision and Control*, pages 829–834, Florence, Italy, 2013.
- [113] E. Vos, A.J. van der Schaft, and J.M.A. Scherpen. Formation control and velocity tracking of wheeled robots, 2014. *Submitted*.
- [114] E. Vos, J.M.A. Scherpen, and A.J. van der Schaft. Equal distribution of satellite constellations on circular target orbits. *Automatica*, 50(10):2641–2647, 2014.
- [115] E. Vos, J.M.A. Scherpen, A.J. van der Schaft, and A. Postma. Formation control of wheeled robots in the port-Hamiltonian framework. In *World Congress of the International Federation of Automatic Control*, pages 6662–6667, Cape Town, South Africa, 2014.
- [116] E. Vos, M. Jafarian, C. De Persis, J.M.A. Scherpen, and A.J. van der Schaft. Formation control of nonholonomic wheeled robots in the presence of matched input disturbances, 2015. *Submitted*.
- [117] C. Wang, G. Xie, and M. Cao. Controlling anonymous mobile agents with unidirectional locomotion to form formations on a circle. *Automatica*, 50(4):1100–1108, 2014.
- [118] N. van de Wouw and R.I. Leine. Attractivity of equilibrium sets of systems with dry friction. *Nonlinear Dynamics*, 35(1):19–39, 2004.
- [119] N. van de Wouw and R.I. Leine. Robust impulsive control of motion systems with uncertain friction. *International Journal of Robust and Nonlinear Control*, 22(4):369–397, 2012.
- [120] E. Xargay, R. Choe, N. Hovakimyan, and I. Kaminer. Convergence of a PI coordination protocol in networks with switching topology and quantized measurements. In *IEEE Conference on Decision and Control*, pages 6107–6112, Maui, HI, USA, 2012.

---

## Summary

This thesis proposes new formation control algorithms for networks of port-Hamiltonian systems. Formation control is a specific type of coordination control, where the goal is to achieve a prescribed geometrical shape for the network. Port-Hamiltonian systems theory provides an energy-based framework for the modeling, analysis, and control design for nonlinear, multi-domain systems. The two application areas considered in this thesis are dike inspection using robotic sensor networks and formation flying of satellites. Both application areas require exact formation control to obtain high resolution measurements. Formation control algorithms are developed for three different types of agents: point masses, wheeled robots, and satellites. In addition to the standard formation control problem, the related problems of formation control in the presence of Coulomb friction, deployment, velocity tracking , input disturbance rejection, and orbital phasing are studied.

Formation control of networks of port-Hamiltonian systems requires distributed controllers which can deal with large networks of complex dynamical agents. Distributed controllers provide modular, scalable controllers for networks with many agents. For the three types of dynamical agents considered in this thesis, the complex dynamics pose additional challenges for the controller design. For point masses, Coulomb friction renders the agent dynamics non-smooth, which disables standard continuous controllers to achieve exact formation control. The wheeled robots are subject to a nonholonomic constraint, which reduces the degrees of freedom of the robot. For satellites the gravitational field of planet earth renders the satellite dynamics nonlinear.

The results in this thesis build upon the theory of port-Hamiltonian systems on graphs and a passivity-based approach for the coordination of networks. For the three types of agents considered, formation control is achieved by assigning virtual couplings between the agents in the network. Each virtual coupling consists of a spring and damper in parallel, while the interconnection topology is modeled by an undirected graph. For a network of point masses subject to discontinuous Coulomb

friction it is shown that a discontinuous counterpart to the standard continuous spring is required to achieve exact formations. Deployment is achieved by assigning virtual couplings between the agents and two virtual walls, which are positioned at prescribed reference points. Formation control for a network of wheeled robots is achieved by assigning virtual couplings between the front end of the robots, rather than the center of mass. For velocity tracking, error dynamics with respect to the reference velocity are derived using generalized canonical transformations. Stabilizing these error dynamics achieves convergence to the reference velocity. An internal model controller counteracts harmonic matched input disturbances with unknown frequency and amplitude. Finally, the concept of virtual couplings has shown its versatility, by a successful application to the orbital phasing problem for a network of satellites on circular orbits.

The approach presented in this thesis provides an extension of a passivity-based approach for group coordination to port-Hamiltonian systems. Moreover, the use of non-smooth analysis tools is completely new in port-Hamiltonian systems theory, which enables the generalization of existing non-smooth theory to port-Hamiltonian systems. Virtual couplings are applicable to various types of systems and their physical interpretation facilitates their use by practitioners. Furthermore, the physical structure of port-Hamiltonian models provides intrinsic robustness to the results. Results are illustrated and validated with experimental and extensive simulation results, which provide an important first step towards future implementation in practice.

---

## Samenvatting

Dit proefschrift komt met nieuwe regelalgoritmes voor het regelen van formaties van netwerken met poort-Hamiltonse systemen. Bij het regelen van formaties draait het om het vormen van een vooraf bepaalde geometrische vorm van een netwerk. De theorie van poort-Hamiltonse systemen biedt een op energie gebaseerd kader voor het modelleren, analyseren, en ontwerpen van regelaars voor niet-lineaire systemen in verschillende fysische domeinen. De twee toepassingsgebieden in dit proefschrift zijn dijkinspectie met groepen robots en formatie vliegen met satellieten. Beide toepassingen verlangen metingen met hoge resoluties, wat vraagt om nauwkeurige regeling van de formatie. De formatie regelalgoritmes zijn ontwikkeld voor drie verschillende soorten agenten: puntmassa's, robots met wielen, en satellieten. Naast het regelen van formaties, wordt in dit proefschrift ook aandacht besteed aan het effect van Coulomb wrijving, ontplooiing, het volgen van een referentiesnelheid, het tegengaan van ingangsverstoringen, en baan fasering.

Het regelen van formaties van netwerken vraagt om gedistribueerde regelaars die kunnen omgaan met grote netwerken met complexe dynamische agenten. Gedistribueerde regelaars bieden modulaire, schaalbare regelaars, die kunnen omgaan met netwerken met (zeer) veel agenten. Dit proefschrift bestudeert drie soorten agenten: puntmassa's, robots met wielen, en satellieten. De complexe dynamica van deze agenten zijn een extra uitdaging voor het ontwerp van de regelaars. Voor puntmassa's maakt Coulomb wrijving de dynamica niet-glad, wat standaard continue regelaars belet om exacte formaties te vormen. Voor de robot met wielen geldt een niet-holome randvoorwaarde, welke het aantal vrijheidsgraden van de robot beperkt. Voor satellieten maakt het zwaartekrachtveld van de Aarde de dynamica van satellieten niet-lineair.

De resultaten in dit proefschrift bouwen voort op de theorie van poort-Hamiltonse systemen op grafen en een op passiviteit gebaseerde benadering voor de coördinatie van netwerken. Voor de drie soorten agenten in dit proefschrift worden formaties gevormd door virtuele koppelingen tussen de agenten in het netwerk te plaatsen.

Elke virtuele koppeling bestaat uit een veer die parallel staat aan een demper. Daarbij is de verbindingstopologie gemodelleerd met een niet-gerichte graaf. Voor een netwerk van puntmassa's die onderhevig zijn aan discontinue Coulomb wrijving is een discontinue tegenhanger van de virtuele veer vereist om exacte formaties te vormen. Wat betreft het ontplooiingsvraagstuk, worden virtuele koppelingen geplaatst tussen de agenten en twee virtuele muren, die gepositioneerd zijn op voorgeschreven referentiepunten. Voor netwerken van robots met wielen worden formaties gevormd door virtuele koppeling te plaatsen tussen de voorkanten van de robots, in plaats van de massamiddelpunten. Voor het volgen van een referentiesnelheid zijn de foutdynamica afgeleid met behulp van gegeneraliseerde canonieke transformaties. Door deze foutdynamica te stabiliseren wordt convergentie naar de referentiesnelheid gegarandeerd. Een interne model regelaar gaat het effect van harmonische gepaarde ingangsverstoringen met onbekende frequentie en amplitude tegen. Door een succesvolle toepassing op het vraagstuk van baan fasering voor satellieten op cirkelvormige banen, tonen virtuele koppelingen hun veelzijdigheid als regeling voor formaties.

Dit proefschrift biedt een uitbreiding van een op passiviteit gebaseerde benadering voor groepscoördinatie naar poort-Hamiltonse systemen. Daarnaast is het gebruik van niet-gladde analyse gereedschappen volledig nieuw binnen de theorie van poort-Hamiltonse systemen en maakt deze generalisatie van bestaande niet-gladde theorie mogelijk. Virtuele koppelingen zijn toepasbaar voor verschillende soorten agenten en hun fysische interpretatie vergemakkelijkt hun gebruik. De fysische structuur van poort-Hamiltonse modellen zorgt voor robuuste resultaten. Experimenten en uitgebreide simulaties illustreren en valideren de resultaten. Dit is een belangrijke eerste stap richting toekomstige implementaties in de praktijk.