Distributed IDA-PBC for Non-holonomic Mechanical Systems

Anastasios Tsolakis

Master of Science Thesis

TU Delft Delft University of Technology

Delft Center for Systems and Control

Distributed IDA-PBC for Nonholonomic Mechanical Systems

Master of Science Thesis

For the degree of Master of Science in Systems and Control at Delft University of Technology

Anastasios Tsolakis

January 4, 2021

Faculty of Mechanical, Maritime and Materials Engineering (3mE) · Delft University of Technology

Delft University of Technology
Department of
Delft Center for Systems and Control (dcsc)

The undersigned hereby certify that they have read and recommend to the Faculty of Mechanical, Maritime and Materials Engineering (3mE) for acceptance a thesis entitled

Distributed IDA-PBC for Nonholonomic Mechanical Systems

by

Anastasios Tsolakis

in partial fulfillment of the requirements for the degree of

Master of Science Systems and Control

Dated: January 4, 2021

Supervisor(s):

Prof. dr. ir. T. Keviczky

Reader(s):

Dr. L. Ferranti

Dr. V. Reppa

Ir. O.M. de Groot

Abstract

Nonholonomic mechanical systems encompass a large class of practically interesting robotic structures, such as wheeled mobile robots, space manipulators, and multi-fingered robot hands. However, few results exist on the cooperative control of such systems in a generic, distributed approach. In this work we extend a recently developed distributed Interconnection and Damping Assignment Passivity-Based Control (IDA-PBC) method to such systems. More specifically, relying on port-Hamiltonian system modelling for networks of mechanical systems, we propose a full-state stabilization control law for a class of nonholonomic systems within the framework of distributed IDA-PBC. This enables the cooperative control of heterogeneous, underactuated and nonholonomic systems with a unified control law. This control law primarily relies on the notion of Passive Configuration Decomposition and a novel, non-smooth control law proposed here. A low-level collision avoidance protocol based on the Artificial Potential Fields (APF) method is also implemented in order to achieve dynamic inter-agent collision avoidance, enhancing the practical relevance of this work. Theoretical results are tested in different simulation scenarios in order to highlight the applicability of the derived method.

Table of Contents

List of Figures

List of Tables

Acknowledgements

This thesis is a continuation of the work of Laurens Valk [1, 2] and Oscar de Groot [3, 4]. First of all, I would like to thank my supervisor Tamás Keviczky for his guidance and support throughout this project. I would also like to thank Oscar de Groot for his willingness to assist me, offering his time for discussions and valuable advice.

Last but not least, I would like to express my deepest gratitude to my family and friends for their love and support. Special thanks to my friend Alexandros for assisting me with the explanatory drawings.

Delft, University of Technology
January 4, 2021

Anastasios Tsolakis

Chapter 1

Introduction

An increasing demand in multi-agent systems has been spurred by the benefits obtained when a single complex system is transformed to an equivalent set of multiple yet simpler systems. Controlling this set of systems can rely on either a centralized or a distributed approach. In the distributed approach, lower level components operate on local information in an appropriate manner to accomplish global goals. This decomposition of a complex system into simpler units and their distributed control entails great advantages among which, decreased operational cost, robustness to failure, strong adaptivity and system scalability [5]. Distributed control of mechanical systems is used in numerous applications such as collaborative transportation, exploration of unknown or dangerous terrains, large scale sensing and area monitoring, collaborative construction and autonomously operating vehicle platoons or spacecraft constellations.

The dynamics of mechanical systems are highly nonlinear. Since control system design relied primarily on linear control theory, nonlinear systems were often linearized and controlled around an operating point. Another common approach was to render the systems linear through an internal feedback loop that would cancel any nonlinearities. Both approaches however have well-known limitations. In the presence of strong nonlinearities, the operation of the linearized system is restricted in a small region around the operating point while cancelling nonlinearities usually leads to excessive amount of feedback gains and in general suffers from robustness issues. For these reasons, a large number of nonlinear control methods has been developed. For the control of mechanical systems, a class of nonlinear control methods known as energy-based control has been proven to be especially suitable. The key idea in modelling such systems stems from analytical dynamics where the energy of the system (namely the sum of potential, kinetic and dissipation energy) fully determines its behaviour while another fundamental property, passivity, is instrumental for deriving stabilizing control laws. The nonlinear system can be controlled by shaping its closed loop energy while respecting the original dynamics, an intrinsically less conservative method which provides higher performance, cost-effective controllers.

1-1 Motivation and Problem Statement

A generic distributed control method that enables heterogeneous groups of underactuated and fully-actuated, highly nonlinear mechanical systems to cooperatively assume desired task-space formations has been proposed in [2]. This work extended the nonlinear, energy-based control method of Interconnection and Damping Assignment Passivity-Based Control (IDA-PBC) [6] to a distributed setting. This allowed the distributed control of a group of (possibly underactuated) mechanical systems with non-identical dynamics so that they can achieve low-level cooperative group objectives known as consensus, synchronization, or formation problems. These problems are mathematically similar across the wide range of the previously mentioned applications highlighting the practical importance of the results in [2].

The generality and potential applicability of the work in [2] motivates researchers to seek ways of relaxing several simplifying assumptions. Initially motivated by this work, in [4] a novel r-Passivity Based Control (r-PBC) approach has been proposed which can successfully handle network effects such as communication delays and packet dropouts. Other topics such as collision avoidance and discretization were also considered and the efficacy of the proposed novel control law was validated with experimental results.

Motivated to broaden the scope of [2], in this work we aim to extend these results to nonholonomic mechanical systems and implement a simple low-level collision avoidance protocol. Nonholonomic mechanical systems encompass a great variety of mechanical systems including wheeled mobile robots, space robots and other mobile mechanical systems which are often found in various applications. Moreover, these applications often lead to a cluttered environment where the agents are likely to collide while performing their collaborative tasks. This motivates the attempt to implement a low-level, inter-agent collision avoidance protocol that has inherent similarities with the distributed IDA-PBC approach and can work with the nonholonomic nature of such systems successfully. The problem we are addressing in this work can be simply stated as follows:

"Develop a distributed control method which allows a group of non-identical, possibly nonholonomic mechanical systems achieve cooperative objectives while locally avoiding collision with each other."

1-2 Novelty of This Work

Many powerful techniques have been developed for the feedback control of nonholonomic mechanical systems, yet, the majority of them rely on specific control forms (e.g. normal form, power form, chained form) which can be attained typically only by using feedback linearization [7]. In this work we aim to develop a feedback control method for nonholonomic systems relying on the benefits of passivity based control. More specifically, we address the problem of full-state stabilization for nonholonomic mechanical systems applying the method of Passive Configuration Decomposition developed in [7] to port-Hamiltonian systems. This is instrumental to expand full-state stabilization in the IDA-PBC framework allowing for control of both underactuated and nonholonomic mechanical systems. Moreover, we propose a new potential energy for the shaped, closed-loop dynamics which yields improved transient response for the stabilization problem of nonholonomic systems. Last but not least

we implement a simple protocol that can locally ensure inter-agent collision avoidance. The efficacy of the aforementioned is illustrated in simulations.

1-3 Notation

In order to distinguish among scalars, vectors and matrices we introduce typographical emphasis. Scalars are denoted by non-bold lower-case characters, vectors by bold lower-case characters while matrices are denoted by bold, upper-case characters.

$$q \in \mathbb{R}, \quad \boldsymbol{q} \in \mathbb{R}^n, \quad \boldsymbol{Q} \in \mathbb{R}^{m \times n} \tag{1-1}$$

The notation $\mathbf{0_n}$ denotes an $n \times n$ matrix with zeros, whereas $\boldsymbol{I_n}$ denotes the $n \times n$ identity matrix. The notation $\mathbf{0}$ denotes a zero vector of the appropriate dimension (often expressed explicitly with subscripts). The subscript ij of Q_{ij} denotes the element at the i^{th} row and the j^{th} column. The transpose of a partial derivative is written as $\frac{\partial^\top f}{\partial \boldsymbol{x}}$. The Jacobian of scalar function f to vector $\boldsymbol{x} \in \mathbb{R}^n$, is denoted by the column vector:

$$\frac{\partial^\top f}{\partial \boldsymbol{x}} = \begin{bmatrix} \frac{\partial f}{\partial x_1} & \cdots & \frac{\partial f}{\partial x_n} \end{bmatrix}^\top \tag{1-2}$$

In the following we denote the real numbers with $\mathbb{R}$ and the positive real numbers including zero with $\mathbb{R}^+$. The set of natural numbers excluding zero is denoted N and the set of integers is written as Z.

For matrix $\boldsymbol{A}$, we denote the transpose as $\boldsymbol{A}^\top$, the inverse as $\boldsymbol{A}^{-1}$ and the inverse of the transpose as $\boldsymbol{A}^{-\top}$. The Moore-Penrose pseudo inverse is given by $\boldsymbol{A}^\dagger = \left(\boldsymbol{A}^\top \boldsymbol{A}\right)^{-1} \boldsymbol{A}^\top$. An annihilator for $\boldsymbol{A}$ is $\boldsymbol{A}^\perp$ such that $\boldsymbol{A}^\perp \boldsymbol{A} = \mathbf{0}$ variable dependencies are often left out if their dependency is clear from context. For example $\boldsymbol{z}(\boldsymbol{q}(t))$ is written as $\boldsymbol{z}(\boldsymbol{q})$ where the dependency on time is implicit. In general the notation $\|\boldsymbol{q}\|$ denotes the 2-norm unless specifically specified otherwise.

1-4 Report Overview

This report is organized as follows. In Chapter 2 we provide the fundamentals of mechanical system modelling, nonholonomic constraints and port-Hamiltonian systems which constitute the stepping stone of this work. In Chapter 3 we present a complete approach of IDA-PBC for nonholonomic systems starting with system matching and concluding with a novel stabilization approach. Chapter 4 extends the single-agent results of the previous chapter to a distributed setting following the steps in [2] thus completing the problem of distributed IDA-PBC for nonholonomic agents. We then attempt to make the whole design more practically relevant in Chapter 5 by introducing a simple collision avoidance protocol that relies on the Artificial Potential Fields (APF) method. We show that the chosen collision avoidance scheme can work in accordance with the control design derived in the previous chapters. In Chapter 6 we provide simulation results for two different scenarios in order to illustrate the working principles of our control design. We conclude this work in Chapter 7 with a discussion and some conclusions on the derived results, the advantages and the shortcomings of our approach, followed by recommendations for future work and improvements.

Chapter 2

Nonholonomic Mechanical Systems

The aim of this chapter is to present a formal mathematical definition of the systems we analyze in this report and discuss their fundamental properties. In order to keep the report concise, this is by no means an exhaustive review on mechanical system modelling. Nevertheless, it provides the necessary background for the subsequent chapters. For a more detailed description of mechanical systems modelling the reader is referred to [8, 9, 10].

2-1 Modelling Mechanical Systems

The study of dynamics of particles often relies on the principles of Newtonian Mechanics which is vectorial in nature. These principles were first introduced by Newton for single point masses and later expanded by Euler for systems of point masses. According to Newtonian Mechanics the central notions are expressed with vectors (e.g. acceleration, forces) which are in general difficult to calculate. Moreover for complex systems the equations of motion are derived by individually studying each component of the system which results in the necessity of calculating forces that result from kinematic constraints increasing the number of unknowns in the problem. For such complex systems the application of Newtonian Mechanics can present difficulties. An alternate approach is that of Analytical Dynamics which was founded by Leibniz and Lagrange. In this approach central notions are work and energy which are scalar quantities. Moreover, the foundation of these principles is realized in such a way that the calculation of reaction forces from kinematic constraints is not needed. This is the result of introducing the generalized coordinates which are more abstract mathematical quantities than physical coordinates but lead to the aforementioned advantages. Last but not least, the equations of motion derived in Analytical Dynamics have a form that does not depend on the choice of the generalized coordinates. This approach of Analytical Dynamics is represented by the Euler-Lagrange equations and Hamilton's equations. These methods enable one to obtain a complete set of equations of motion by differentiation of a single scalar function, namely the Lagrangian function or the Hamiltonian function [9].

2-1-1 Generalized Coordinates and Momenta

Every set of variables that can define the exact location and orientation of the components of a system is called the set of *generalized coordinates*. In mechanical systems these generalized coordinates usually represent length or angle but some times can even represent other more abstract quantities like energy or momentum. They are often expressed as $\boldsymbol{q} \in \mathbb{R}^n$ where n is the number of generalized coordinates. The set of generalized coordinates to define a dynamical system is not unique. The time derivatives of the generalized coordinates expressed as $\dot{\boldsymbol{q}} \in \mathbb{R}^n$ are the *generalized velocities*.

A set of generalized coordinates defines the *configuration* of the system. The *configuration space* $\mathcal{Q} \in \mathbb{R}^n$ (or *configuration manifold*) is the vector space of all possible configurations of the system. The tangent space of $\mathcal{Q}$ at $\boldsymbol{q}$ is denoted as $\dot{\boldsymbol{q}} \in \mathcal{T}_q\mathcal{Q}$. The tangent space contains all possible directions on $\mathcal{Q}$ that can be followed when the system is at configuration $\boldsymbol{q}$. In this work, modelling relies on Hamilton's equations where the system's unknowns are the generalized coordinates and the *generalized momenta* $\boldsymbol{p} \in \mathbb{R}^n$ defined as:

$$\boldsymbol{p} = \boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}} \tag{2-1}$$

where $\boldsymbol{M}(\boldsymbol{q})$ denotes the symmetric, positive-definite, *mass matrix* of the system. The generalized coordinates and momenta belong to the *cotangent bundle* $(\boldsymbol{q}, \boldsymbol{p}) \in \mathcal{T}_q^*\mathcal{Q}$, which is the set of possible configurations and momenta that the system can reach. We also refer to the pair $(\boldsymbol{q}, \boldsymbol{p})$ as the *state* of the system and denote the *state space* as $\mathcal{X} \in \mathbb{R}^{2n}$.

2-1-2 Nonholonomic Constraints

In many cases, the motion of the components of a mechanical system is *constrained* by kinematic relations called *kinematic constraints*. These kinematic constraints are in general a set of differential and/or algebraic equations that further dictate the evolution of the system additionally to the equations of motion. This additional set of equations is satisfied due to the generation of corresponding generalized constraint forces that are applied on the system.

Constraints can be classified as *holonomic* when they are algebraic equations of the form $\boldsymbol{\alpha}(\boldsymbol{q}, t) = \boldsymbol{0}$ or *non-holonomic* when they are equality or inequality equations of the forms $\boldsymbol{\alpha}(\boldsymbol{q}, \dot{\boldsymbol{q}}, t) = \boldsymbol{0}$ (bilateral) or $\boldsymbol{\alpha}(\boldsymbol{q}, \dot{\boldsymbol{q}}, t) \leq \boldsymbol{0}$ (unilateral) correspondingly. When these expressions are explicit functions of time then we refer to them as *rheonomic* otherwise they are known as *scelronomic*. The aforementioned classification is important as to how these constraints affect a system. Holonomic constraints are algebraic expressions of the generalized coordinates and can most of the times vanish if a proper, minimal set of generalized coordinates is chosen to describe the system. On the other hand, nonholonomic constraints are differential equations of both the generalized coordinates and velocities. These expressions are neither total differentials nor they can be converted into one via an integrating factor and thus they are not integrable. As a result, these nonholonomic constraints are additional differential equations that further dictate the evolution of the system and they need to be solved along with the system's equations of motion complicating the analysis. One difference of critical importance is that nonholonomic constraints do not reduce the configuration space $\mathcal{Q}$ meaning that any configuration is still accessible by the system. Nonetheless, they reduce the tangent space $\mathcal{T}_q\mathcal{Q}$ meaning that a nonholonomic system can go anywhere but not howsoever.

A particularly interesting form of nonholonomic constraints that arise in mechanical systems is that of *Pfaffian Constraints*. These are sclerononomic, bilateral nonholonomic constraints which are linear expressions with respect to the generalized velocities:

$$\boldsymbol{A}^{\top}(\boldsymbol{q})\dot{\boldsymbol{q}} = \boldsymbol{0} \tag{2-2}$$

where $\boldsymbol{A}(\boldsymbol{q}) \in \mathbb{R}^{n\times k}$ is the *constraint matrix* for a system with n generalized coordinates and k distinct nonholonomic constraints. These constraints describe various practical phenomena that arise in mechanical systems including but not limited to:

- Rolling without slipping constraints: wheeled mobile robots (such as differential and car-like robots) and also in manipulation with multi-fingered robotic hands
- Conservation of angular momentum: arises in space robots, satellites as well as gymnastic robots (e.g.: hopping robots)
- Knife edge motion (sledges, sailboats)

These types of systems are very often met in a vast area of robotic applications including logistics, transportation, area monitoring, space exploration etc.

2-1-3 Hamilton's Equations

The dynamics of a simple mechanical system with generalized coordinates $\boldsymbol{q} \in \mathbb{R}^n$, generalized momenta $\boldsymbol{p} \in \mathbb{R}^n$, generalized forces $\boldsymbol{f_q} \in \mathbb{R}^n$ and which is further imposed to k kinematic constraints of Pfaffian form can be described by the following set of $2n+k$ nonlinear Ordinary Differential Equations (ODEs):

$$\begin{bmatrix} \dot{\boldsymbol{q}} \\ \dot{\boldsymbol{p}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{0_n} & \boldsymbol{I_n} \\ -\boldsymbol{I_n} & \boldsymbol{0_n} \end{bmatrix} \begin{bmatrix} \frac{\partial H}{\partial \boldsymbol{q}}(\boldsymbol{q},\boldsymbol{p}) \\ \frac{\partial H}{\partial \boldsymbol{p}}(\boldsymbol{q},\boldsymbol{p}) \end{bmatrix} + \begin{bmatrix} \boldsymbol{0_{n\times k}} \\ \boldsymbol{A}(\boldsymbol{q}) \end{bmatrix} \boldsymbol{\lambda} + \boldsymbol{f_q} \tag{2-3a}$$

$$\boldsymbol{0} = \boldsymbol{A}^{\top}(\boldsymbol{q}) \frac{\partial H}{\partial \boldsymbol{p}}(\boldsymbol{q},\boldsymbol{p}) \tag{2-3b}$$

These are the standard Hamilton's Equations written in matrix notation. Note that the first term of the right-hand side in (2-3a) represents all inertial and potential forces, the second term represents forces arising from constraints described in (2-3b) and the third term represents exogenous forces that act on the system. The pair $(\boldsymbol{q},\boldsymbol{p})^{\top} \in \mathbb{R}^{2n}$ is the *state* of the system and $H(\boldsymbol{q},\boldsymbol{p}) \in \mathbb{R}$ is the *Hamiltonian* function of the system. For simple mechanical systems the Hamiltonian function is the *mechanical energy* of the system defined as the sum of *kinetic energy* and *potential energy*:

$$H = \frac{1}{2}\boldsymbol{p}^{\top}\boldsymbol{M}^{-1}\boldsymbol{p} + V(\boldsymbol{q}) \tag{2-4}$$

where $\boldsymbol{M}(\boldsymbol{q}) = \boldsymbol{M}^{\top}(\boldsymbol{q}) > \boldsymbol{0}_{n\times n}$ denotes the symmetric, positive-definite, *mass matrix* as stated before and $V(\boldsymbol{q}) \in \mathbb{R}$ denotes the potential energy of the system, a scalar quantity that we assume is a function of the generalized coordinates only. Note also that Equation (2-3) represent the *frictionless* dynamics. The Lagrange multipliers $\boldsymbol{\lambda} \in \mathbb{R}^k$ denote the magnitude of the constraint forces which ensure that the constraints are satisfied and $\boldsymbol{A}(\boldsymbol{q}) \in \mathbb{R}^{n\times k}$ is the *constraint matrix*. In the next section we present how the standard Hamilton's Equations can be expressed as a port-Hamiltonian system which is more favourable for the control purposes in subsequent chapters.

2-2 Port-Hamiltonian Systems

In the previous section we briefly presented Hamilton's equations which originate from the filed of analytical dynamics. Another approach that stems from electrical engineering, is the *network* approach which constitutes a cornerstone of mathematical systems theory. While most of the analysis of physical systems has been performed within the Lagrangian and Hamiltonian framework, the network point of view is prevailing in modelling and simulation of (complex) physical engineering systems. A framework that combines both points of view is that of port-Hamiltonian systems. The port-Hamiltonian description offers a systematic framework for analysis, control and simulation of complex physical systems [11].

2-2-1 System Input and Output

A port-Hamiltonian system consists of two internal ports, an energy-storing element and an energy-dissipating element as well as an external port which models the interaction of the system with its environment [11]. A natural candidate for an energy-storing element of (2-3) is the Hamiltonian while an energy-dissipating element does not exist having assumed that the dynamics are frictionless. The external port consists of a type of port variables accessible for controller action (input) and another type of external port variables corresponds to an interaction port (output). First, we assume that the generalized forces in Equation (2-3) comprise the known, input forces which have the input-affine form:

$$\boldsymbol{f_q} = \boldsymbol{F}(\boldsymbol{q})\boldsymbol{\tau} \tag{2-5}$$

where $\boldsymbol{F}(\boldsymbol{q}) \in \mathbb{R}^{n\times m}$ is the *input matrix* and $\boldsymbol{\tau} \in \mathbb{R}^m$ the system's input. We continue by driving the following energy balance from (2-3) and (2-5):

$$\frac{dH(\boldsymbol{q},\boldsymbol{p})}{dt} = \frac{\partial^\top H}{\partial \boldsymbol{q}}(\boldsymbol{q},\boldsymbol{p})\dot{\boldsymbol{q}} + \frac{\partial^\top H}{\partial \boldsymbol{p}}(\boldsymbol{q},\boldsymbol{p})\dot{\boldsymbol{p}} = (-\dot{\boldsymbol{p}} + \boldsymbol{A}\boldsymbol{\lambda} + \boldsymbol{F}\boldsymbol{\tau})^\top \dot{\boldsymbol{q}} + \dot{\boldsymbol{q}}^\top \dot{\boldsymbol{p}} = \boldsymbol{\tau}^\top(\boldsymbol{F}^\top \dot{\boldsymbol{q}}) \tag{2-6}$$

expressing that the increase in the mechanical energy of the system is equal to the supplied work (conservation of energy). This motivates to define the output of the system as:

$$\boldsymbol{y} = \boldsymbol{F}^\top \dot{\boldsymbol{q}} = \boldsymbol{F}^\top(\boldsymbol{q})\frac{\partial H}{\partial \boldsymbol{p}}(\boldsymbol{q},\boldsymbol{p}) \tag{2-7}$$

Throughout this work, each agent is assumed to be a simple mechanical system with generalized coordinates $\boldsymbol{q} \in \mathbb{R}^n$, generalized momenta $\boldsymbol{p} \in \mathbb{R}^n$, input $\boldsymbol{\tau} \in \mathbb{R}^m$ and output $\boldsymbol{y} \in \mathbb{R}^m$ and its open-loop, constrained, frictionless dynamics described as:

$$\begin{cases} \begin{bmatrix} \dot{\boldsymbol{q}} \\ \dot{\boldsymbol{p}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{0_n} & \boldsymbol{I_n} \\ -\boldsymbol{I_n} & \boldsymbol{0_n} \end{bmatrix} \begin{bmatrix} \frac{\partial H}{\partial \boldsymbol{q}}(\boldsymbol{q},\boldsymbol{p}) \\ \frac{\partial H}{\partial \boldsymbol{p}}(\boldsymbol{q},\boldsymbol{p}) \end{bmatrix} + \begin{bmatrix} \boldsymbol{0_{n\times k}} \\ \boldsymbol{A}(\boldsymbol{q}) \end{bmatrix} \boldsymbol{\lambda} + \begin{bmatrix} \boldsymbol{0_{n\times m}} \\ \boldsymbol{F}(\boldsymbol{q}) \end{bmatrix} \boldsymbol{\tau} \\ \boldsymbol{y} = \boldsymbol{F}^\top(\boldsymbol{q})\frac{\partial H}{\partial \boldsymbol{p}}(\boldsymbol{q},\boldsymbol{p}) \\ \boldsymbol{0} = \boldsymbol{A}^\top(\boldsymbol{q})\frac{\partial H}{\partial \boldsymbol{p}}(\boldsymbol{q},\boldsymbol{p}) \\ H = \frac{1}{2}\boldsymbol{p}^\top \boldsymbol{M}^{-1}\boldsymbol{p} + V(\boldsymbol{q}) \end{cases} \tag{2-8}$$

2-2-2 Constraint Transformation

This system described by (2-8) is now closer to the port-Hamiltonian structure that we need to proceed with the control tasks later on. Notice however that the constraint equation still appears explicitly in the system description. A classical approach to solve such a problem, is differentiating expression (2-3b) with respect to time and then after proper substitutions solve for the Lagrange multipliers $\boldsymbol{\lambda}$ and get an expression for the constraint forces. However, the constraint forces, albeit dictating the evolution of the system, are not useful to compute explicitly. A much more efficient way for handling systems subjected to this class of constraints is described in [12] which yields the desired port-Hamiltonian form. In the following we briefly present and explain this approach.

The constraint equation (2-3b) forces the system to evolve on the *constrained state space* described as:

$$\mathcal{X}_c = \left\{ (\boldsymbol{q}, \boldsymbol{p}) \in \mathcal{X} \middle| \boldsymbol{A}^\top(\boldsymbol{q}) \frac{\partial H}{\partial \boldsymbol{p}}(\boldsymbol{q}, \boldsymbol{p}) = \boldsymbol{0} \right\} \tag{2-9}$$

The idea proposed in [12] is to perform a coordinate transformation of the generalized momenta since it comprises the part of the state that is affected by the constraints. This coordinate transformation is essentially a projection of the system's state from $\mathcal{X}$ onto the constrained space $\mathcal{X}_c$ described by (2-9). The new coordinates are not canonical in general but nonetheless they provide new capabilities to the analysis of systems under constraints.

We begin with defining the transformation:

$$\tilde{\boldsymbol{p}} = \boldsymbol{T}(\boldsymbol{q})\boldsymbol{p} \tag{2-10}$$

with $\boldsymbol{T}(\boldsymbol{q}) \in \mathbb{R}^{n \times n}$ denoting the square, invertible transformation matrix. A natural selection for this matrix is [13]:

$$\boldsymbol{T}(\boldsymbol{q}) = \begin{bmatrix} \boldsymbol{S}^\top(\boldsymbol{q}) \\ \boldsymbol{A}^\top(\boldsymbol{q})\boldsymbol{M}^{-1}(\boldsymbol{q}) \end{bmatrix} \tag{2-11}$$

where matrix $\boldsymbol{S}^\top(\boldsymbol{q}) \in \mathbb{R}^{(n-k)\times n}$ is the full-rank *left annihilator* of the constraint matrix $\boldsymbol{A}(\boldsymbol{q}) \in \mathbb{R}^{n \times k}$ such that:

$$\boldsymbol{S}^\top(\boldsymbol{q})\boldsymbol{A}(\boldsymbol{q}) = \boldsymbol{0}_{(\boldsymbol{n-k})\times \boldsymbol{k}} \tag{2-12}$$

When matrix $\boldsymbol{S}^\top(\boldsymbol{q})$ is multiplied with the original generalized momenta (or velocities), results to the transformed momenta (velocities) which lie on the vector space of "allowable" momenta (velocities). This is then the *orthogonal complement* of the constraint velocities given from the expression $\boldsymbol{A}^\top(\boldsymbol{q})\dot{\boldsymbol{q}}$. The transformation matrix as defined in (2-11), projects the generalized momenta onto the constraint state space.

Next, partition the generalized momenta as $\tilde{\boldsymbol{p}} = \begin{bmatrix} \tilde{\boldsymbol{p}}^{\boldsymbol{1}} \\ \tilde{\boldsymbol{p}}^{\boldsymbol{2}} \end{bmatrix}$ with $\tilde{\boldsymbol{p}}^{\boldsymbol{1}} \in \mathbb{R}^{n-k}$ and $\tilde{\boldsymbol{p}}^{\boldsymbol{2}} \in \mathbb{R}^k$ which results to:

$$\begin{aligned} \tilde{\boldsymbol{p}}^{\boldsymbol{1}} &= \boldsymbol{S}^\top(\boldsymbol{q})\boldsymbol{p} \in \mathbb{R}^{n-k} \\ \tilde{\boldsymbol{p}}^{\boldsymbol{2}} &= \boldsymbol{A}^\top(\boldsymbol{q})\boldsymbol{M}^{-1}(\boldsymbol{q})\boldsymbol{p} \in \mathbb{R}^k \end{aligned} \tag{2-13}$$

Notice first that $\tilde{\boldsymbol{p}}^{\boldsymbol{2}} = \boldsymbol{0}$ over the constraint manifold. Hence, the last k equations give the constraints instead of the dynamics. Moreover, the introduction of the annihilator removes the constraint forces from the dynamics of $\tilde{\boldsymbol{p}}^{\boldsymbol{1}}$ in (2-8). For the sake of notation we denote here $\tilde{\boldsymbol{p}}^{\boldsymbol{1}} \triangleq \tilde{\boldsymbol{p}}$.

2-2-3 System Dynamics

Equations (2-8) can then be expressed in the reduced form to which we conclude as the final form of the open-loop dynamics. The frictionless, implicitly constrained, open-loop dynamics of each agent with generalized coordinates $\boldsymbol{q} \in \mathbb{R}^n$, generalized momenta $\tilde{\boldsymbol{p}} \in \mathbb{R}^{n-k}$, input $\boldsymbol{\tau} \in \mathbb{R}^m$ and output $\boldsymbol{y} \in \mathbb{R}^m$ are expressed as:

$$\begin{cases} \begin{bmatrix} \dot{\boldsymbol{q}} \\ \dot{\tilde{\boldsymbol{p}}} \end{bmatrix} = \underbrace{\begin{bmatrix} \boldsymbol{0_n} & \boldsymbol{S}(\boldsymbol{q}) \\ -\boldsymbol{S}^\top(\boldsymbol{q}) & \boldsymbol{Y}(\boldsymbol{q},\tilde{\boldsymbol{p}}) \end{bmatrix}}_{\tilde{\mathcal{J}}} \begin{bmatrix} \frac{\partial \tilde{H}}{\partial \boldsymbol{q}}(\boldsymbol{q},\tilde{\boldsymbol{p}}) \\ \frac{\partial \tilde{H}}{\partial \tilde{\boldsymbol{p}}}(\boldsymbol{q},\tilde{\boldsymbol{p}}) \end{bmatrix} + \begin{bmatrix} \boldsymbol{0_{n\times m}} \\ \tilde{\boldsymbol{F}}(\boldsymbol{q}) \end{bmatrix} \boldsymbol{\tau} \\ \tilde{\boldsymbol{y}} = \tilde{\boldsymbol{F}}^\top(\boldsymbol{q}) \dfrac{\partial \tilde{H}}{\partial \tilde{\boldsymbol{p}}}(\boldsymbol{q},\tilde{\boldsymbol{p}}) \\ \tilde{H}(\boldsymbol{q},\tilde{\boldsymbol{p}}) = \dfrac{1}{2}\tilde{\boldsymbol{p}}^\top \tilde{\boldsymbol{M}}^{-1}(\boldsymbol{q})\tilde{\boldsymbol{p}} + V(\boldsymbol{q}) \end{cases} \tag{2-14}$$

where $\tilde{\boldsymbol{y}}(\boldsymbol{q},\tilde{\boldsymbol{p}}) \in \mathbb{R}^m$ is the transformed output, $\tilde{H}(\boldsymbol{q},\tilde{\boldsymbol{p}}) \in \mathbb{R}$ the transformed Hamiltonian, $\tilde{\boldsymbol{M}}(\boldsymbol{q}) \in \mathbb{R}^{(n-k)\times(n-k)}$ the transformed generalized mass matrix, $\tilde{\boldsymbol{F}}(\boldsymbol{q}) \in \mathbb{R}^{(n-k)\times m}$, the transformed control matrix and $\boldsymbol{Y}(\boldsymbol{q},\tilde{\boldsymbol{p}}) \in \mathbb{R}^{(n-k)\times(n-k)}$ is a skew-symmetric matrix that arises from the existence of constraints and is defined as $\boldsymbol{Y} = \left(-\boldsymbol{p}^T[S_i,S_j](\boldsymbol{q})\right)_{i,j=1,\dots,n-k}$ with $[S_i,S_j]$ denoting the Lie bracket (thus satisfying skew-symmetry). More elaborated expressions of these components are the following:

$$\boldsymbol{Y} = -\boldsymbol{Y}^\top = \sum_{i=1}^{n}\left(\frac{\partial \boldsymbol{S}^\top}{\partial q_i}\boldsymbol{T}^{-1}\tilde{\boldsymbol{p}}\boldsymbol{e}_i^T\boldsymbol{S}\right) - \sum_{i=1}^{n}\left(\boldsymbol{S}^\top\boldsymbol{e}_i\tilde{\boldsymbol{p}}^\top\boldsymbol{T}^{-\top}\frac{\partial \boldsymbol{S}}{\partial q_i}\right) \tag{2-15}$$

$$\tilde{\boldsymbol{M}} = \tilde{\boldsymbol{M}}^\top = \boldsymbol{S}^\top\boldsymbol{M}\boldsymbol{S} > \boldsymbol{0_k} \tag{2-16}$$

$$\tilde{\boldsymbol{F}} = \boldsymbol{S}^\top\boldsymbol{F} \tag{2-17}$$

with $\boldsymbol{e}_i$ the standard basis vector in $\mathbb{R}^n$. The full derivation of the resulting terms is not presented here in order to keep the report concise. The interconnection matrix $\tilde{\boldsymbol{\mathcal{J}}}$ remains skew symmetric which ensures that the resulting system admits to the same Hamiltonian structure as the original system [12].

If we denote $\boldsymbol{x} = (\boldsymbol{q}^\top, \tilde{\boldsymbol{p}}^\top)^\top \in \mathbb{R}^{2n-k}$ the state of the system described by (2-14) then we can write it in the more general form:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})\boldsymbol{\tau} \\ \tilde{\boldsymbol{y}} &= \boldsymbol{h}(\boldsymbol{x}) \end{aligned} \tag{2-18}$$

where the vector fields $\boldsymbol{f}(\cdot)$ and $\boldsymbol{g}(\cdot)$ are contained in the distribution $\mathcal{D}$ given in (2-20). The system is described by a set of $2n-k$ nonlinear, input-affine ODEs with an m-dimensional input $\boldsymbol{\tau}$ and a set of $2n-k$ initial conditions $\boldsymbol{x_0} = (\boldsymbol{q_0}^\top, \tilde{\boldsymbol{p}}_{\boldsymbol{0}}^\top)^\top \in \mathbb{R}^{n-k}$ which can be derived using the transformation matrix defined in (2-11) as:

$$\tilde{\boldsymbol{p}}_{\boldsymbol{0}} = \boldsymbol{T}(\boldsymbol{q_0})\boldsymbol{p_0} \tag{2-19}$$

The physical meaning of the transformed generalized momenta $\tilde{\boldsymbol{p}}$ (and velocity) can be better interpreted in the example with which we conclude this chapter.

2-3 System Properties

In this section we discuss a few properties of the systems described by Equation (2-14) so as to provide some further insights and hopefully a more elaborate understanding.

2-3-1 Constraints

A system that is subjected to nonholonomic constraints is known as a *Nonholonomic System* otherwise, as a *Holonomic System*. The terms *Constrained System* or *Unconstrained System* are also used interchangeably. Constraints are described by Equation (2-3b) with $rank\left(\boldsymbol{A}(\boldsymbol{q})\right) = k < n$ meaning that the columns of $\boldsymbol{A}(\boldsymbol{q})$ describe k linearly independent constraints. If there are no constraints ($k = 0$) then the constraint matrix $\boldsymbol{A(q)}$ is an empty matrix and thus, matrix $\boldsymbol{S}^{\top}(\boldsymbol{q})$ can be any $n \times n$ full-rank matrix as it will satisfy (2-12). We can then choose $\boldsymbol{S}^{\top}(\boldsymbol{q}) = \boldsymbol{I_n}$ which makes the transformation matrix also become $\boldsymbol{T}(\boldsymbol{q}) = \boldsymbol{I_n}$. Thus, system (2-14) will then take the same form as in (2-8) without the constraints: $\boldsymbol{A}(\boldsymbol{q})$ and the second term of the sum in (2-8) vanish, reducing the system's description to a simpler, holonomic one consisting of $2n$ nonlinear ODEs. Note also that the annihilator $\boldsymbol{S}^{\top}(\boldsymbol{q})$ is not unique. The constraint equation (2-3b) determines a k-dimensional distribution $\mathcal{D}$ on $\mathcal{Q}$ given at every point $\boldsymbol{q} \in \mathcal{Q}$ as:

$$\mathcal{D}(\boldsymbol{q}) = \ker \boldsymbol{A}^{\top}(\boldsymbol{q}) \tag{2-20}$$

Since $\boldsymbol{S(q)}$ is the orthogonal complement of $\boldsymbol{A}^{\top}(\boldsymbol{q})$, Equation (2-20) can be expressed equivalently as:

$$\mathcal{D}(\boldsymbol{q}) = \text{col}\boldsymbol{S}(\boldsymbol{q}) \tag{2-21}$$

Meaning that any matrix $\boldsymbol{S}(\boldsymbol{q})$ that satisfies (2-12) is an appropriate choice. Notice that since this matrix is a function of $\boldsymbol{q}$ some entries may be ill-defined (division by zero). In order to circumvent this problem we can divide each column by the product of its denominators.

2-3-2 Actuation

The input matrix $\boldsymbol{F}(\boldsymbol{q}) \in \mathbb{R}^{n \times m}$ is assumed to be of full rank with $rank\left(\boldsymbol{F}(\boldsymbol{q})\right) = m \leq n$ that is, its columns describe m linearly independent actuators. If there are no input forces ($m = 0$),the system is referred to as *Unforced*. If the system is not unforced then we make the following distinctions: In case that $m = n$ the system is *Fully-actuated* and matrix $\boldsymbol{F}(\boldsymbol{q})$ is a square, full-rank, invertible matrix. In case that $m < n$ then the system is called *Underactuated* meaning that there are fewer distinct actuators than degrees of freedom. Notice that constrained systems are underactuated by definition. Despite the fact that there are fewer actuators, all degrees of freedom can be controlled if a proper coupling among them exists. Such systems are particularly common. A characteristic example is that of a quad-rotor with $n = 6$ degrees of freedom (full spatial motion) and $m = 4$ inputs (the four rotors). The quad-rotor cannot translate sideways directly but this is possible implicitly if a proper tilt is first realised. We will see in Chapter 3 however that under-actuation complicates that analysis and makes control of such systems more challenging. Lastly, in the case where $m > n$ the system is referred to as *Over-actuated* meaning that there is redundancy in actuators. Such systems are outside of the scope of this work.

2-3-3 Heterogeneity

Notice that the system equations (2-14) comprise a minimal set of components with which we can essentially describe *any* simple mechanical system. For an unconstrained and unforced system, if we define the set of generalized coordinates $\boldsymbol{q}$ and the system's energy $H(\boldsymbol{q},\boldsymbol{p})$, for which we need the mass matrix $\boldsymbol{M}(\boldsymbol{q})$ and the potential energy $V(\boldsymbol{q})$, then we can plug this expressions in (2-14) and derive the equations of motion. If further the system is subjected to nonholonomic constraints or to input forces we can simply define the matrices $\boldsymbol{A}(\boldsymbol{q})$ and $\boldsymbol{F}(\boldsymbol{q})$ correspondingly. Thus, by a minimal set of components ($\boldsymbol{q}$, $\boldsymbol{M}(\boldsymbol{q})$, $V(\boldsymbol{q})$, $\boldsymbol{A}(\boldsymbol{q})$, $\boldsymbol{F}(\boldsymbol{q})$) we can describe any mechanical system that fulfills the assumptions made so far. This idea of modelling in the framework of analytical dynamics allows as to use the same set of equations, namely (2-14), with just different components and still describe any system. This fact is extremely useful for two main reasons. First of all, it gives us the capability to automatically derive the equations of motion algorithmically even for very complex systems with a large number of degrees of freedom which would be impossible to do analytically. Furthermore, from a control point of view, it is now possible to control a multi-agent system that consists of *Heterogeneous* agents, meaning agents that have completely different dynamics (e.g. a manipulator cooperating with a quad-rotor). This is a great advantage compared to methods that require the systems controlled to be *Homogeneous* (i.e. have identical dynamics).

2-3-4 Passivity

We conclude this discussion by introducing the fundamental property of passivity for port-Hamiltonian systems.

Definition 2.1: *Suppose that the system* (2-14) *with a supply rate* $\boldsymbol{w} = \boldsymbol{\tau}^\top \boldsymbol{y}$ *and a storage function* $V(\boldsymbol{x}) : \mathbb{R}^n \to \mathbb{R}$ *with* $V(\boldsymbol{0}) = 0$ *for which the following relation holds:*

$$V(\boldsymbol{x}(t)) - V(\boldsymbol{x}(0)) \leq \int_0^t \boldsymbol{y}(s)^\top \boldsymbol{\tau}(s) ds \tag{2-22}$$

Then the system is passive [14].

Differentiating this expression with respect to time we can get the equivalent one:

$$\dot{V} \leq \boldsymbol{y}^\top \boldsymbol{\tau} \tag{2-23}$$

For the port-Hamiltonian system (2-14), with the Hamiltonian function as storage function, the control input $\boldsymbol{\tau}$ and the conjugate output $\boldsymbol{y}$ as defined in (2-7) we have already shown in (2-6) that the system is already passive. Passivity is an important property which comes naturally in mechanical systems stating that the power supply in the system is less or equal than the rate of energy stored into the system. Rearranging equation (2-6) we have:

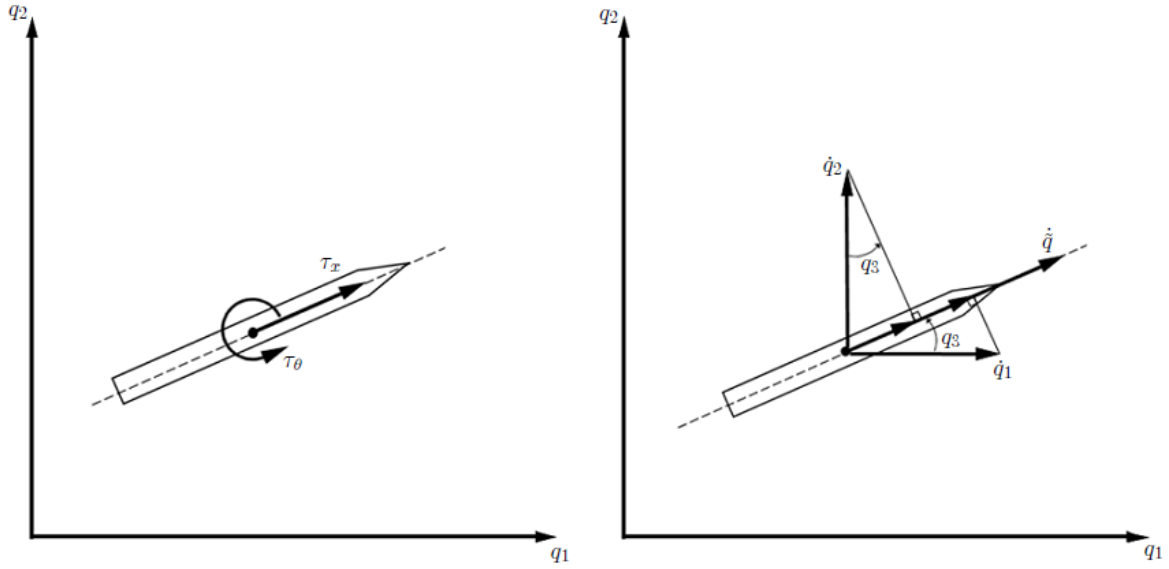
$$\dot{H}(\boldsymbol{q},\boldsymbol{p}) = \dot{\boldsymbol{q}}^\top (\boldsymbol{F}\boldsymbol{\tau}) \tag{2-24}$$

which states from a physical point of view that the rate of change of the mechanical energy is equal (due to the assumption that it is frictionless) to the power supply to the system where power is given as the multiplication of the generalized velocities and the conjugate generalized

forces. This property has been a crucial tool for robot control as it allows for exploitation of the nonlinear dynamics of robotic systems instead of attempting to completely cancel out and replace it by linear dynamics (e.g. feedback linearization). Furthermore, with its storage function often serving as a basis for a Lyapunov function construction, the property of passivity has been instrumental to enforce stability of the closed-loop system in a robust manner.

Example 1: Constrained Dynamics

In this example we present a simple nonholonomic system in order to elucidate the resulted open-loop dynamics (2-14). The "knife edge" in Figure (2-1) is a simple nonholonomic system that moves on a plane with $n = 3$ degrees of freedom, $k = 1$ constraint and $m = 2$ inputs. The degrees of freedom denoted here as (q_1, q_2) define its position in this plane and q_3 its orientation. Since the motion is planar, the potential energy is zero $V = 0$ and the mass matrix is given as $\boldsymbol{M} = diag\,(m, m, I)$. Assuming that we can a apply a force on the longitudinal



(a) The control inputs on the knife edge system **(b)** The transformed generalized velocities

Figure 2-1: The knife edge example: inputs and transformed velocity

direction of the knife edge let it be τ_x, and a torque, τ_θ, that can rotate the body with respect to axis z .The control input is then $\boldsymbol{\tau} = (\tau_x \quad \tau_\theta)^\top$ and the input matrix:

$$\boldsymbol{F} = \begin{bmatrix} \cos q_3 & 0 \\ \sin q_3 & 0 \\ 0 & 1 \end{bmatrix}$$

The nonholonomic constraint arises from the assumption that the "knife edge" does not slide sideways. This is expressed as:

$$\sin q_3 \dot{q}_1 - \cos q_3 \dot{q}_2 = 0 \Rightarrow \begin{pmatrix} \sin q_3 & -\cos q_3 & 0 \end{pmatrix} \dot{\boldsymbol{q}} = 0 \Rightarrow$$
$$\boldsymbol{A}^\top(\boldsymbol{q}) = \begin{pmatrix} \sin q_3 & -\cos q_3 & 0 \end{pmatrix}$$

An annihilator is then:

$$\boldsymbol{S}^\top \boldsymbol{A} = \boldsymbol{0} \Rightarrow \boldsymbol{S} = \begin{bmatrix} \frac{\cos q_3}{\sin q_3} & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \xrightarrow{\text{see (2-3-1)}} \boldsymbol{S} = \begin{bmatrix} \cos q_3 & 0 \\ \sin q_3 & 0 \\ 0 & 1 \end{bmatrix}$$

Now transform the generalized velocities with the derived annihilator:

$$\dot{\tilde{\boldsymbol{q}}} = \boldsymbol{S}^\top \dot{\boldsymbol{q}} = \begin{bmatrix} \cos q_3 & \sin q_3 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{pmatrix} = \begin{pmatrix} \cos q_3 \dot{q}_1 + \sin q_3 \dot{q}_2 \\ \dot{q}_3 \end{pmatrix}$$

The first component of the transformed velocity $\dot{\tilde{\boldsymbol{q}}}$ is the vector sum of the projections of the generalized velocities $\dot{q}_1$ and $\dot{q}_2$ on the longitudinal axis of the knife edge which is the vector space of permissible velocities as shown in Figure (2-1). Notice also that the generalized velocity $\dot{q}_3$ remains intact as it does not appear in the constraint equation. This means that the generalized variable q_3 is free from the constraints, a critical observation for the stabilization results of Chapter 3. The rest of the components are:

$$\tilde{\boldsymbol{M}} = \boldsymbol{S}^\top \boldsymbol{M} \boldsymbol{S} = \begin{bmatrix} m & 0 \\ 0 & I \end{bmatrix} \qquad \tilde{\boldsymbol{F}} = \boldsymbol{S}^\top \boldsymbol{F} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad \boldsymbol{Y} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

The first entry of the transformed mass matrix is the linear mass component m which is the inertia term in the longitudinal direction of the knife edge body. Notice also how in this example the original input matrix is the same with the annihilator $\boldsymbol{S}$ leading to the transformed input matrix being the identity matrix.This implies that the control input $\boldsymbol{\tau}$ is directly applied on the acceleration components $\dot{p}_1$ and $\dot{p}_2$.

Chapter 3

IDA-PBC for Nonholonomic Systems

In this chapter we present an extension of the method of Interconnection and Damping Assignment Passivity-Based Control (IDA-PBC), first proposed in [6], for the system dynamics that we concluded with in the previous chapter. An extension of the method for nonholonomic systems was first proposed in [15] and later in [13]. However, the stabilization problem of the whole configuration is not solved in any existing work as the nature of nonholonomic constraints prohibits the stabilization with a smooth control law. This chapter aims to present a complete proposal tackling first the "Matching Problem" and then the "Stabilization Problem".

3-1 System Matching

3-1-1 Desired System Dynamics

IDA-PBC is a form of nonlinear control that relies on passivity properties to systematically derive asymptotically stabilizing control laws for systems with a known model or model structure. The control goal is achieved by energy shaping instead of assigning a fully-prescribed set of dynamics in order to exploit rather than destroy the internal dynamics of the original system. This approach has the advantage of not needing high gains and improves robustness. Specifically, we define the dynamics of the *desired system* in the Hamiltonian framework by designing favourable energy terms (i.e. kinetic, potential and dissipation energy). Then we seek a feedback control law $\boldsymbol{\tau} \in \mathbb{R}^m$ which can transform the original open-loop system to the desired, closed-loop system. First, we rewrite the open-loop dynamics of the agents of interest here for completeness.

$$\begin{cases} \begin{bmatrix} \dot{\boldsymbol{q}} \\ \dot{\tilde{\boldsymbol{p}}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{0}_n & \boldsymbol{S}(\boldsymbol{q}) \\ -\boldsymbol{S}^\top(\boldsymbol{q}) & \boldsymbol{Y}(\boldsymbol{q},\tilde{\boldsymbol{p}}) \end{bmatrix} \begin{bmatrix} \frac{\partial \tilde{H}}{\partial \boldsymbol{q}}(\boldsymbol{q},\tilde{\boldsymbol{p}}) \\ \frac{\partial \tilde{H}}{\partial \tilde{\boldsymbol{p}}}(\boldsymbol{q},\tilde{\boldsymbol{p}}) \end{bmatrix} + \begin{bmatrix} \boldsymbol{0}_{n\times m} \\ \tilde{\boldsymbol{F}}(\boldsymbol{q}) \end{bmatrix} \boldsymbol{\tau} \\ \tilde{\boldsymbol{y}} = \tilde{\boldsymbol{F}}^\top(\boldsymbol{q}) \dfrac{\partial \tilde{H}}{\partial \tilde{\boldsymbol{p}}}(\boldsymbol{q},\tilde{\boldsymbol{p}}) \\ \tilde{H}(\boldsymbol{q},\tilde{\boldsymbol{p}}) = \dfrac{1}{2}\tilde{\boldsymbol{p}}^\top \tilde{\boldsymbol{M}}^{-1}(\boldsymbol{q})\tilde{\boldsymbol{p}} + V(\boldsymbol{q}) \end{cases} \tag{3-1}$$

Consider the partitioning of the control law $\boldsymbol{\tau}$ as:

$$\boldsymbol{\tau} = \boldsymbol{\beta}(\boldsymbol{q}, \boldsymbol{p}) + \boldsymbol{\tau_d} \tag{3-2}$$

where $\boldsymbol{\beta}(\boldsymbol{q}, \boldsymbol{p})$ implements the feedback law while $\boldsymbol{\tau}_d$ denotes the new, closed-loop input (a feed-forward term). Since we aim to shape the potential and kinetic energy of the system, following the classical approach in [6] and the extension in [13], we define the *desired dynamics* in Hamiltonian form as:

$$\begin{cases} \begin{bmatrix} \dot{\boldsymbol{q}} \\ \dot{\tilde{\boldsymbol{p}}} \end{bmatrix} = \begin{bmatrix} \mathbf{0_n} & \boldsymbol{S}\tilde{\boldsymbol{M}}^{-1}\boldsymbol{M_d} \\ -\boldsymbol{M_d}\tilde{\boldsymbol{M}}^{-1}\boldsymbol{S}^\top & \boldsymbol{J} - \tilde{\boldsymbol{F}}\boldsymbol{K_v}\tilde{\boldsymbol{F}}^\top \end{bmatrix} \begin{bmatrix} \frac{\partial H_d}{\partial \boldsymbol{q}}(\boldsymbol{q}, \tilde{\boldsymbol{p}}) \\ \frac{\partial H_d}{\partial \tilde{\boldsymbol{p}}}(\boldsymbol{q}, \tilde{\boldsymbol{p}}) \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{n\times m} \\ \tilde{\boldsymbol{F}}(\boldsymbol{q}) \end{bmatrix} \boldsymbol{\tau_d} \\ \boldsymbol{y_d} = \tilde{\boldsymbol{F}}^\top(\boldsymbol{q}) \dfrac{\partial H_d}{\partial \tilde{\boldsymbol{p}}}(\boldsymbol{q}, \tilde{\boldsymbol{p}}) \\ H_d(\boldsymbol{q}, \tilde{\boldsymbol{p}}) = \dfrac{1}{2}\tilde{\boldsymbol{p}}^\top \boldsymbol{M_d}^{-1}(\boldsymbol{q})\tilde{\boldsymbol{p}} + V_d(\boldsymbol{q}) \end{cases} \tag{3-3}$$

where $V_d \in \mathbb{R}$ is the *desired potential energy* (which is the shaped potential energy of the closed loop system) and $\boldsymbol{M_d} \in \mathbb{R}^{(n-k)\times(n-k)}$ is the *desired mass matrix* (which shapes the kinetic energy of the closed loop system). The desired potential energy V_d aims to make the system evolve towards a new goal configuration denoted as $\boldsymbol{q}^*$ thus having the property:

$$\boldsymbol{q}^* = \arg\min_{\boldsymbol{q}} V_d(\boldsymbol{q}) \tag{3-4}$$

Kinetic energy shaping aims to solve the matching problem and in addition shapes the transient response. Matrix $\boldsymbol{J} = -\boldsymbol{J}^\top \in \mathbb{R}^{(n-k)\times(n-k)}$ is the skew-symmetric, *gyroscopic force matrix* which aids in the solution of the matching problem as well, by creating one extra degree of freedom in the matching conditions explained later. The *damping matrix* denoted as $\boldsymbol{K_v} = \boldsymbol{K_v}^\top > \mathbf{0_m}$ induces dissipation to the closed-loop system for asymptotic convergence. This matrix is free to choose as it does not appear in the so-called matching conditions.

3-1-2 Control Law and Matching Conditions

IDA-PBC aims to find a control input $\boldsymbol{\tau}$ of the form (3-2) that transforms the open-loop plant (3-1) to the desired, closed-loop dynamics (3-3). This is known as the *matching problem* since we need to match the controlled system with the desired dynamics. In order to solve the problem we begin with equating the open-loop dynamics (3-1) with control input (3-2) to the closed-loop dynamics (3-3) which yields the following equations:

$$\boldsymbol{S}\frac{\partial \tilde{H}}{\partial \tilde{\boldsymbol{p}}} = \boldsymbol{S}\tilde{\boldsymbol{M}}^{-1}\boldsymbol{M_d}\frac{\partial H_d}{\partial \tilde{\boldsymbol{p}}} \tag{3-5}$$

$$-\boldsymbol{S}^\top\frac{\partial \tilde{H}}{\partial \boldsymbol{q}} + \boldsymbol{Y}\frac{\partial \tilde{H}}{\partial \tilde{\boldsymbol{p}}} + \tilde{\boldsymbol{F}}\boldsymbol{\beta} = -\boldsymbol{M_d}\tilde{\boldsymbol{M}}^{-1}\boldsymbol{S}^\top\frac{\partial H_d}{\partial \boldsymbol{q}} + \boldsymbol{J}\frac{\partial H_d}{\partial \tilde{\boldsymbol{p}}} - \tilde{\boldsymbol{F}}\boldsymbol{K_v}\tilde{\boldsymbol{F}}^\top\frac{\partial H_d}{\partial \tilde{\boldsymbol{p}}} \tag{3-6}$$

Equation (3-5) is trivially satisfied. In order to solve (3-6) for the feedback law $\boldsymbol{\beta}$, define the left annihilator $\tilde{\boldsymbol{F}}^\perp \in \mathbb{R}^{(n-k-m)\times(n-k)}$ of $\tilde{\boldsymbol{F}}$ such that:

$$\tilde{\boldsymbol{F}}^\perp\tilde{\boldsymbol{F}} = \mathbf{0}_{(\boldsymbol{n-k-m})\times\boldsymbol{m}} \tag{3-7}$$

and the full rank matrix $[(\tilde{\boldsymbol{F}}^{\perp})^{\top}, \tilde{\boldsymbol{F}}]^{\top} \in \mathbb{R}^{(n-k)\times(n-k)}$. Multiplying (3-6) with the latter matrix from the left gives:

$$\begin{bmatrix} \tilde{\boldsymbol{F}}^{\perp} \\ \tilde{\boldsymbol{F}}^{\top} \end{bmatrix} \tilde{\boldsymbol{F}}\boldsymbol{\beta} = \begin{bmatrix} \tilde{\boldsymbol{F}}^{\perp} \\ \tilde{\boldsymbol{F}}^{\top} \end{bmatrix} \left(\boldsymbol{S}^{\top}\frac{\partial \tilde{H}}{\partial \boldsymbol{q}} - \boldsymbol{M_d}\tilde{\boldsymbol{M}}^{-1}\boldsymbol{S}^{\top}\frac{\partial H_d}{\partial \boldsymbol{q}} - \boldsymbol{Y}\frac{\partial \tilde{H}}{\partial \tilde{\boldsymbol{p}}} + \boldsymbol{J}\frac{\partial H_d}{\partial \tilde{\boldsymbol{p}}} - \tilde{\boldsymbol{F}}\boldsymbol{K_v}\tilde{\boldsymbol{F}}^{\top}\frac{\partial H_d}{\partial \tilde{\boldsymbol{p}}} \right) \tag{3-8}$$

Since $\tilde{\boldsymbol{F}}^{\perp}\tilde{\boldsymbol{F}} = \boldsymbol{0_{(n-k-m)\times m}}$ by construction the first $n-k-m$ equations of (3-8) give the necessary conditions and are termed as *matching conditions*. These conditions are a set of nonlinear, Partial Differential Equations (PDEs) which can be expressed as:

$$\tilde{\boldsymbol{F}}^{\perp}\left(\boldsymbol{S}^{\top}\frac{\partial \tilde{H}}{\partial \boldsymbol{q}} - \boldsymbol{M_d}\tilde{\boldsymbol{M}}^{-1}\boldsymbol{S}^{\top}\frac{\partial H_d}{\partial \boldsymbol{q}} - \boldsymbol{Y}\frac{\partial \tilde{H}}{\partial \tilde{\boldsymbol{p}}} + \boldsymbol{J}\frac{\partial H_d}{\partial \tilde{\boldsymbol{p}}} - \cancel{\tilde{\boldsymbol{F}}\boldsymbol{K_v}\tilde{\boldsymbol{F}}^{\top}\frac{\partial H_d}{\partial \tilde{\boldsymbol{p}}}} \right) = \boldsymbol{0_{n-k-m}} \tag{3-9}$$

Notice that these conditions are not invariant of the control law $\boldsymbol{\beta}$ and can only be satisfied by properly selecting $\boldsymbol{M_d}$, $\boldsymbol{J}$ and V_d. Notice also that the last term in this equation vanishes because of (3-7) rendering the damping matrix free. The last m equations can be solved for the feedback law since $[(\tilde{\boldsymbol{F}}^{\perp})^{\top}, \tilde{\boldsymbol{F}}]^{\top}$ is full rank and thus, invertible:

$$\boldsymbol{\beta} = \left(\tilde{\boldsymbol{F}}^{\top}\tilde{\boldsymbol{F}}\right)^{-1}\tilde{\boldsymbol{F}}^{\top}\left(\boldsymbol{S}^{\top}\frac{\partial \tilde{H}}{\partial \boldsymbol{q}} - \boldsymbol{M_d}\tilde{\boldsymbol{M}}^{-1}\boldsymbol{S}^{\top}\frac{\partial H_d}{\partial \boldsymbol{q}} - \boldsymbol{Y}\frac{\partial \tilde{H}}{\partial \tilde{\boldsymbol{p}}} + \boldsymbol{J}\frac{\partial H_d}{\partial \tilde{\boldsymbol{p}}} \right) - \boldsymbol{K_v}\underbrace{\tilde{\boldsymbol{F}}^{\top}\frac{\partial H_d}{\partial \tilde{\boldsymbol{p}}}}_{\boldsymbol{y_d}} \tag{3-10}$$

The selection of components that satisfy the matching condition (3-9) is not a trivial problem. Nevertheless, this condition can be reformulated by expanding the Hamiltonian and by recalling that $\frac{\partial H_d}{\partial \tilde{\boldsymbol{p}}} = \boldsymbol{M_d}^{-1}\tilde{\boldsymbol{p}}$. After these considerations, (3-9) can be grouped in its kinetic and potential energy contributions as:

$$\begin{aligned} &\tilde{\boldsymbol{F}}^{\perp}\left(\boldsymbol{S}^{\top}\frac{\partial \tilde{\boldsymbol{p}}^{\top}\tilde{\boldsymbol{M}}^{-1}\tilde{\boldsymbol{p}}}{\partial \boldsymbol{q}} - \boldsymbol{M_d}\tilde{\boldsymbol{M}}^{-1}\boldsymbol{S}^{\top}\frac{\partial \tilde{\boldsymbol{p}}^{\top}\boldsymbol{M_d}^{-1}\tilde{\boldsymbol{p}}}{\partial \boldsymbol{q}} - 2\boldsymbol{Y}\tilde{\boldsymbol{M}}^{-1}\tilde{\boldsymbol{p}} + 2\boldsymbol{J}\boldsymbol{M_d}^{-1}\tilde{\boldsymbol{p}} \right) + \\ &\tilde{\boldsymbol{F}}^{\perp}\left(\frac{\partial V}{\partial \boldsymbol{q}} - \boldsymbol{M_d}\tilde{\boldsymbol{M}}^{-1}\boldsymbol{S}^{\top}\frac{\partial V_d}{\partial \boldsymbol{q}} \right) = \boldsymbol{0_{n-k-m}} \end{aligned} \tag{3-11}$$

To simplify the selection of components for the desired dynamics, (3-11) is split into matching conditions for the kinetic and potential energy respectively given by:

$$\tilde{\boldsymbol{F}}^{\perp}\left(\boldsymbol{S}^{\top}\frac{\partial \tilde{\boldsymbol{p}}^{\top}\tilde{\boldsymbol{M}}^{-1}\tilde{\boldsymbol{p}}}{\partial \boldsymbol{q}} - \boldsymbol{M_d}\tilde{\boldsymbol{M}}^{-1}\boldsymbol{S}^{\top}\frac{\partial \tilde{\boldsymbol{p}}^{\top}\boldsymbol{M_d}^{-1}\tilde{\boldsymbol{p}}}{\partial \boldsymbol{q}} - 2\boldsymbol{Y}\tilde{\boldsymbol{M}}^{-1}\tilde{\boldsymbol{p}} + 2\boldsymbol{J}\boldsymbol{M_d}^{-1}\tilde{\boldsymbol{p}} \right) = \boldsymbol{0_{n-k-m}} \tag{3-12}$$

$$\tilde{\boldsymbol{F}}^{\perp}\left(\frac{\partial V}{\partial \boldsymbol{q}} - \boldsymbol{M_d}\tilde{\boldsymbol{M}}^{-1}\boldsymbol{S}^{\top}\frac{\partial V_d}{\partial \boldsymbol{q}} \right) = \boldsymbol{0_{n-k-m}} \tag{3-13}$$

Notice that satisfying (3-12) and (3-13) is more conservative that satisfying (3-11). The feedback law given by (3-10) can be reformulated after considering that $\boldsymbol{y_d} = \tilde{\boldsymbol{F}}^{\top}\frac{\partial H_d}{\partial \tilde{\boldsymbol{p}}}$. Substitution of (3-10) into (3-2) gives:

$$\boldsymbol{\tau} = \left(\tilde{\boldsymbol{F}}^{\top}\tilde{\boldsymbol{F}}\right)^{-1}\tilde{\boldsymbol{F}}^{\top}\left(\boldsymbol{S}^{\top}\frac{\partial \tilde{H}}{\partial \boldsymbol{q}} - \boldsymbol{M_d}\tilde{\boldsymbol{M}}^{-1}\boldsymbol{S}^{\top}\frac{\partial H_d}{\partial \boldsymbol{q}} - \boldsymbol{Y}\frac{\partial \tilde{H}}{\partial \tilde{\boldsymbol{p}}} + \boldsymbol{J}\frac{\partial H_d}{\partial \tilde{\boldsymbol{p}}} \right) - \boldsymbol{K_v}\boldsymbol{y_d} + \boldsymbol{\tau_d} \tag{3-14}$$

3-1-3 Energy Shaping Components

In the method described earlier our goal is to shape the energy of the closed-loop system so that we can achieve a desired behaviour. Notice that, the desired mass matrix $\boldsymbol{M_d}$ and the gyroscopic force matrix $\boldsymbol{J}$, have reduced dimensions due to the system reduction explained in Section 2-2-2. Since these are user-defined parameters we can choose them arbitrarily if their respected properties are satisfied (symmetry, positive definiteness and skew-symmetry). However, as seen in the example of Chapter 2 the reduced generalized momenta do not correspond directly to the physical properties of the system. Due to this fact, we propose choosing the desired mass matrix and gyroscopic force matrix in the original configuration space $\mathcal{X}$ denoted as $\boldsymbol{M_D}$ and $\boldsymbol{J_D}$ since their entries will then have a straightforward physical meaning. Subsequently we can use the same reduction method that we used for the generalized momenta and transform them in the constraint configuration space $\mathcal{X}_c$:

$$\boldsymbol{M_d} = \boldsymbol{S}^\top \boldsymbol{M_D} \boldsymbol{S} \tag{3-15}$$

$$\boldsymbol{J} = \boldsymbol{S}^\top \boldsymbol{J_D} \boldsymbol{S} \tag{3-16}$$

The damping matrix $\boldsymbol{K_v}$is an important tuning parameter that guarantees we achieve asymptotic stability and also affects the speed of the transient response. Since as seen in the previous section it does not appear in the matching conditions, it's a free design parameter that has the minimal requirements of symmetry and positive definiteness. Inspired by classical control theory and the idea of critical damping, in Section 7-3-4 we propose a possible tuning approach of $\boldsymbol{K_v}$ based on the desired mass matrix $\boldsymbol{M_D}$ and the desired potential V_d that can potentially simplify the tuning process.

3-1-4 The Annihilators

We briefly explain the physical meaning of the annihilators $\boldsymbol{S}$ and $\tilde{\boldsymbol{F}}^\perp$. As already discussed, the column space of the constraint matrix, col$\boldsymbol{A}(\boldsymbol{q})$, is the space of non-permissible velocities due to constraints. Its orthogonal complement is the column space of the annihilator, col$\boldsymbol{S}^\top(\boldsymbol{q})$, which is the space of permissible velocities. Similarly, as the column space of the input matrix col$\boldsymbol{F}(\boldsymbol{q})$ denotes the space of allowable forces, its orthogonal complement, the annihilator $\tilde{\boldsymbol{F}}^\perp$ is a map to the non-permissible forces. The terms in the parenthesis of Equation (3-11) denote generalized forces that when multiplied with $\tilde{\boldsymbol{F}}^\perp$ need to vanish meaning that they lie in the space of permissible forces. This is why under-actuation can equivalently be seen as a fully-actuated system imposed to acceleration constraints [16].

Notice that in the case of nonholonomic systems the input matrix is transformed as $\tilde{\boldsymbol{F}} = \boldsymbol{S}^\top \boldsymbol{F}$. In this example of Chapter 2 these two matrices coincide leading to an invertible transformed input matrix $\tilde{\boldsymbol{F}}$. More specifically, if these matrices coincide while they are semi-orthogonal, then $\tilde{\boldsymbol{F}} = \boldsymbol{I_{n-k}}$. Systems with an invertible matrix $\tilde{\boldsymbol{F}}$ are *fully-actuated constrained systems* meaning that they are fully actuated in the constrained space. A formal condition to characterize an n-degree of freedom constraint system as fully-actuated is the following:

$$\text{col}\boldsymbol{A}(\boldsymbol{q}) \oplus \text{col}\boldsymbol{F}(\boldsymbol{q}) = \mathbb{R}^n \tag{3-17}$$

Such systems are fully-actuated in the constrained space.

3-2 System Stabilization

3-2-1 The Largest Invariant Set

The problem of feedback stabilization of nonlinear systems has occupied a central role in the literature of nonlinear systems. One of the most challenging topics in this area is the design of local (or global) stabilizing control laws for nonholonomic systems with more degrees of freedom than controls [17]. A well-known negative result regarding the stabilization of such systems is **Brockett's necessary condition** [18]:

Control systems of the form (2-18) *with a smooth distribution* $\mathcal{D}$ *that contains* $\boldsymbol{f}(\cdot)$ *and* $\boldsymbol{g}(\cdot)$ *cannot be stabilized by a smooth, time-invariant feedback law.*

This result spurred a considerable effort to find either smooth time-invariant control laws or non-smooth control laws for the stabilization of this class of systems. Many powerful techniques have been proposed so far, yet, the majority of them relies on specific control forms (e.g. normal form, power form, chained form etc.) that can be attained typically only by using feedback linearization or state transformation as in [17]. In the passivity based control framework, results for nonholonomic systems can be seen in [19] where they rely on coordinate transformation and the design of non-smooth desired potentials. However, this coordinate transformation is system specific and the transformed generalized coordinates loose their physical meaning which makes analysis and control considerably difficult. In [13] they use IDA-PBC but are concerned with the stabilization of variables that are not affected by the constraints. The interested reader is motivated to study [20] for a more in-depth analysis on nonholonomic system stabilization. Before proposing a solution to the stabilization problem we first elucidate the problem that nonholonomic constraints pose with an additional illustrative example.

Consider the closed-loop system (3-3) which has resulted from the matching process described before. We proceed to find the equilibrium of this dynamical system as follows:

$$\dot{\boldsymbol{q}} = \boldsymbol{0} \Rightarrow \boldsymbol{S}\tilde{\boldsymbol{M}}^{-1}\boldsymbol{M_d}\frac{\partial H_d}{\partial \tilde{\boldsymbol{p}}} = \boldsymbol{0} \Rightarrow \boldsymbol{S}^\top \boldsymbol{M}\boldsymbol{S}\left(\boldsymbol{S}^\top \boldsymbol{M}\boldsymbol{S}\right)^{-1}\boldsymbol{M_d}\boldsymbol{M_d}^{-1}\tilde{\boldsymbol{p}} = \boldsymbol{0} \Rightarrow \tilde{\boldsymbol{p}} = \boldsymbol{0} \tag{3-18}$$

and using the result from (3-18)

$$\dot{\tilde{\boldsymbol{p}}} = \boldsymbol{0} \Rightarrow -\boldsymbol{M_d}\tilde{\boldsymbol{M}}^{-1}\boldsymbol{S}^\top\frac{\partial H_d}{\partial \boldsymbol{q}} + \left(\boldsymbol{J} - \tilde{\boldsymbol{F}}\boldsymbol{K_v}\tilde{\boldsymbol{F}}^\top\right)\frac{\partial H_d}{\partial \tilde{\boldsymbol{p}}} = \boldsymbol{0} \xrightarrow{\tilde{\boldsymbol{p}}=\boldsymbol{0}} \boldsymbol{S}^\top\frac{\partial V_d}{\partial \boldsymbol{q}} = \boldsymbol{0} \tag{3-19}$$

Which shows that the system will converge to the largest invariant set given as:

$$\boldsymbol{\Omega}_{inv} = \left\{(\boldsymbol{q}, \boldsymbol{0}) \in \mathcal{X} \middle| \boldsymbol{S}^\top(\boldsymbol{q})\frac{\partial V_d}{\partial \boldsymbol{q}}(\boldsymbol{q}) = \boldsymbol{0}\right\} \tag{3-20}$$

Thus, due to the existence of nonholonomic constraints, the system does not asymptotically converge to the desired equilibrium expressed in (3-4) but instead at the invariant set defined in (3-20) validating Brockett's condition.

Notice that since $\boldsymbol{S}^\top(\boldsymbol{q}) \in \mathbb{R}^{n-k}$, the largest invariant set is given by $n-k$ equations with n unknowns leading to an under-determined system of equations. The solution of this system is a subset of the configuration space that does not necessarily contain the desired equilibrium $\boldsymbol{q}^*$. We elaborate on this result in the following example.

Example 2 (Continued): The Largest Invariant Set

In this example we show the result of using a smooth feedback law to stabilize the knife edge, the nonholonomic systems used in an earlier example, in order to provide some insight on the complications that arise in constrained systems.

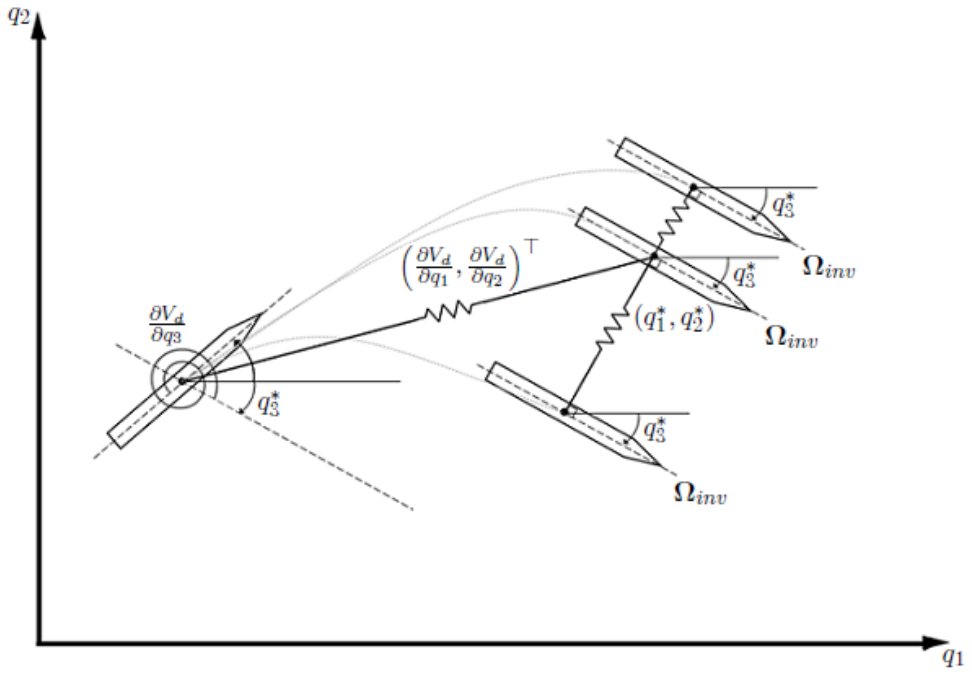


Figure 3-1: The largest invariant set in the knife edge example

The knife edge system with $n = 3$ degrees of freedom ((q_1, q_2) define its position in this plane and q_3 its orientation), is described by the following components in the constraint velocity space:

$$V = 0, \quad \tilde{\boldsymbol{M}} = \begin{bmatrix} m & 0 \\ 0 & I \end{bmatrix}, \quad \boldsymbol{S} = \begin{bmatrix} \cos q_3 & 0 \\ \sin q_3 & 0 \\ 0 & 1 \end{bmatrix}, \quad \tilde{\boldsymbol{F}} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \boldsymbol{Y} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

Since matrix $\tilde{\boldsymbol{F}}$ is full-rank the matching conditions seize to exist. For simplicity we also assume that $\boldsymbol{\tau_d} = \boldsymbol{0}$, $\boldsymbol{J} = \boldsymbol{0}$ and $\boldsymbol{M_d} = \tilde{\boldsymbol{M}}$ with the mass matrix being constant. We also assign an arbitrary symmetric, positive definite matrix $\boldsymbol{K_v}$. Hence, the control law (3-14) takes the simpler form:

$$\boldsymbol{\tau} = -\boldsymbol{S}^\top \frac{\partial V_d}{\partial \boldsymbol{q}} - \boldsymbol{K_v} \dot{\tilde{\boldsymbol{q}}}$$

The first term represents the potential force that is a function of the generalized coordinates while the second term denotes the damping forces that are proportional to the transformed generalised velocities. A common choice for the desired potential function V_d is a smooth, quadratic function:

$$V_d = \frac{1}{2} \left(\boldsymbol{q} - \boldsymbol{q}^*\right)^\top \boldsymbol{Q} \left(\boldsymbol{q} - \boldsymbol{q}^*\right)$$

The control law becomes:

$$\boldsymbol{\tau} = -\boldsymbol{S}^{\top}\boldsymbol{Q}\left(\boldsymbol{q}-\boldsymbol{q}^{*}\right) - \boldsymbol{K}_{v}\dot{\tilde{\boldsymbol{q}}}$$

Assuming for simplicity that $\boldsymbol{Q} = diag\left(Q_1, Q_2, Q_3\right)$ the invariant set (3-20) becomes:

$$\boldsymbol{\Omega}_{inv} = \begin{cases} \cos q_3 Q_1(q_1 - q_1^*) + \sin q_3 Q_2(q_2 - q_2^*) + Q_3(q_3 - q_3^*) = 0 \\ Q_3(q_3 - q_3^*) = 0 \end{cases}$$

The second equation yields the desirable result $q_3 \to q_3^*$. However, notice that the first equation of the invariant set becomes:

$$\boldsymbol{\varepsilon}: \quad \cos q_3^* Q_1(q_1 - q_1^*) + \sin q_3^* Q_2(q_2 - q_2^*) = 0$$

which is under-defined and hence has infinite solutions (q_1, q_2) other than the desired one (q_1^*, q_2^*) . This equation represents the straight line on which the system will be stabilized. Notice how irrespective of the selection of the weighting matrix $\boldsymbol{Q}$, our selection of q_3^* interferes with the stabilization of (q_1, q_2). This result illustrates the fact that there is no smooth control law that can stabilize the nonholonomic knife edge system at a desired equilibrium $\boldsymbol{q}^*$.

The aforementioned are also depicted in Figure 3-1 where the invariant set is depicted as the straight line $\boldsymbol{\varepsilon} \in \mathbb{R}^2$. Each one of the three straight lines depicted in Figure 3-1 results from different tuning of $\boldsymbol{Q}$. We can see the components of the potential control force depicted as one linear spring for (q_1, q_2) to converge and one torsional spring for the convergence of q_3. As q_3 is not tampered with the constraint the system can successfully reach q_3^* but on the other hand, (q_1^*, q_2^*) will not in general be reached as the system will be stabilized at a position on the largest invariant set that has the minimum distance from the desired location (q_1^*, q_2^*) at an angle q_3^*.

3-2-2 Proposed Solution Overview

The proposed solution consists of two parts. The first part is the adaptation of Passive Configuration Decomposition to port-Hamiltonian systems. The Passive Configuration Decomposition was introduced in [7] and it was applied on the open-loop Lagrangian Dynamics of nonholonomic systems. In our case, we apply this method to the Hamiltonian Dynamics and also to the closed-loop system (3-3) so that we can use the control law (3-14) already derived. There are two main reasons to extend this result in the framework of IDA-PBC. First of all, since IDA-PBC has been proven a favourable approach for underactuated systems, an extension of Passive Configuration Decomposition to port-Hamiltonian systems may allow the development of stabilizing control laws for systems that are both nonholonomic and underactuated like the Mobile Inverted Pendulum studied in [13]. Moreover, this result is fundamental so that in the distributed method developed originally in [2] a combination of nonholonomic and underactuated systems can participate in a multi-agent cooperative task (for example a set of quad-rotors reaching consensus with a team of differential robots).

With Passive Configuration Decomposition achieved under some assumptions, the second part of the proposed solution is a proper choice of the desired potential V_d that relies on the aforementioned decomposition. More specifically, based on the insight that some of the configuration variables are free from the nonholonomic constraints, we can use the latter

to drive the system to the desired equilibrium $\boldsymbol{q}^*$ from the arbitrary equilibrium, let it be $\boldsymbol{q}^{\boldsymbol{\omega}} \in \boldsymbol{\Omega}_{inv}$, that the system is stabilized to with a smooth feedback law due to the constraints. Of course, in order to stabilize the system we conclude to a non-smooth feedback law for the variables that are free from constraints thus not contradicting with Brockett's necessary condition.

3-2-3 Passive Configuration Decomposition for Hamiltonian Systems

The original idea presented in [7] aims to decompose a system's Lagrange-D'Alembert Dynamics into two systems, each evolving on their respective configuration spaces and also individually inheriting Lagrangian structure and passivity from the original dynamics. Based on that decomposition they later propose Passivity Based Time-Varying Control (PBVC) and Passivity Based Switching Control (PBSC) schemes, that by utilizing some control actions defined on each of the configuration spaces of the two systems, can achieve stabilization.

In this work, we attempt to use the same idea for decomposition but for the port-Hamiltonian system we have been working on so far. Moreover, in our case this decomposition is applied on the closed-loop system (3-3) so that we can use the same control law derived in (3-14) and thus the only remaining task would be to "shape" the energy of the decomposed, closed-loop system in a suitable manner. That is, after we show that decomposition is possible under some conditions, we shape the desired potential energy of each subsystem individually in order to achieve stabilization of the full state space.

First, recall the closed-loop system derived in the previous section:

$$\begin{cases} \begin{bmatrix} \dot{\boldsymbol{q}} \\ \dot{\tilde{\boldsymbol{p}}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{0_n} & \boldsymbol{S}\tilde{\boldsymbol{M}}^{-1}\boldsymbol{M_d} \\ -\boldsymbol{M_d}\tilde{\boldsymbol{M}}^{-1}\boldsymbol{S}^\top & \boldsymbol{J} - \tilde{\boldsymbol{F}}\boldsymbol{K_v}\tilde{\boldsymbol{F}}^\top \end{bmatrix} \begin{bmatrix} \frac{\partial H_d}{\partial \boldsymbol{q}}(\boldsymbol{q},\tilde{\boldsymbol{p}}) \\ \frac{\partial H_d}{\partial \boldsymbol{p}}(\boldsymbol{q},\tilde{\boldsymbol{p}}) \end{bmatrix} + \begin{bmatrix} \boldsymbol{0}_{n\times m} \\ \tilde{\boldsymbol{F}}(\boldsymbol{q}) \end{bmatrix} \boldsymbol{\tau_d} \\ \boldsymbol{y_d} = \tilde{\boldsymbol{F}}^\top(\boldsymbol{q}) \dfrac{\partial H_d}{\partial \tilde{\boldsymbol{p}}}(\boldsymbol{q},\tilde{\boldsymbol{p}}) \\ H_d(\boldsymbol{q},\tilde{\boldsymbol{p}}) = \dfrac{1}{2}\tilde{\boldsymbol{p}}^\top \boldsymbol{M_d}^{-1}(\boldsymbol{q})\tilde{\boldsymbol{p}} + V_d(\boldsymbol{q}) \end{cases} \tag{3-21}$$

We are interested in the class of mechanical systems described by (3-21) for which the following assumptions are made [7]:

1. The system's configuration space $\mathcal{Q}$ can be endowed with the product structure such that:
$$\mathcal{Q} = \mathcal{S} \times \mathcal{R} \tag{3-22}$$
with $\boldsymbol{q} = \begin{pmatrix} \boldsymbol{s}^\top & \boldsymbol{r}^\top \end{pmatrix}^\top$, $\boldsymbol{s} \in \mathbb{R}^{n-p}$, $\boldsymbol{r} \in \mathbb{R}^p$.
2. The constraint matrix of the nonholonomic Pfaffian constraint (2-3b) is also a function of only $\boldsymbol{r} \in \mathcal{R}$ and the constraint acts only on $\boldsymbol{s} \in \mathcal{S}$:
$$\boldsymbol{A}^\top(\boldsymbol{q})\dot{\boldsymbol{q}} = \begin{bmatrix} \boldsymbol{A_s}^\top(\boldsymbol{r}) & \boldsymbol{0_{k\times p}} \end{bmatrix} \dot{\boldsymbol{q}} = \boldsymbol{A_s}^\top(\boldsymbol{r})\dot{\boldsymbol{s}} = \boldsymbol{0_k} \tag{3-23}$$
with $\boldsymbol{A_s}(\boldsymbol{r}) \in \mathbb{R}$ being full row rank.
3. Its inertia matrix is a function of only $\boldsymbol{r} \in \mathcal{R}$, that is, $\boldsymbol{M}(\boldsymbol{q}) = \boldsymbol{M}(\boldsymbol{r})$

The aforementioned properties may seem restrictive but in fact encompass many practically important and interesting systems (e.g. vertical coin [20], wheeled mobile robots [21], mobile manipulator [22], beanie [23], hopping robot [10]).

With this structure, we can write the unconstrained distribution (2-21) as:

$$\mathcal{D}(\boldsymbol{r}) = span \begin{bmatrix} \mathcal{D}_s(\boldsymbol{r}) & \mathbf{0} \\ \mathbf{0} & \boldsymbol{I_p} \end{bmatrix} \tag{3-24}$$

where $\mathcal{D}_s(\boldsymbol{r}) \in \mathbb{R}^{(n-p)\times(n-p-k)}$ defines the unconstrained distribution on $\mathcal{S}$ such that:

$$\mathcal{D}_s(\boldsymbol{r})^\top \boldsymbol{A_s}(\boldsymbol{r}) = \mathbf{0_{(n-p-k)\times k}} \tag{3-25}$$

Since $\boldsymbol{A}(\boldsymbol{r})$ is regular and smooth so are $\mathcal{D}$ and $\mathcal{D}_s$ with $\text{rank}(\mathcal{D}) = n-p-k,\ \forall \boldsymbol{r} \in \mathcal{R}$. We can then write the generalized velocities as:

$$\dot{\boldsymbol{q}} = \begin{pmatrix} \dot{\boldsymbol{s}} \\ \dot{\boldsymbol{r}} \end{pmatrix} = \underbrace{\begin{bmatrix} \mathcal{D}_s & \mathbf{0} \\ \mathbf{0} & \boldsymbol{I_p} \end{bmatrix}}_{\boldsymbol{S}(\boldsymbol{q})} \begin{pmatrix} \dot{\tilde{\boldsymbol{s}}} \\ \dot{\boldsymbol{r}} \end{pmatrix} \tag{3-26}$$

where $\dot{\tilde{\boldsymbol{s}}} \in \mathbb{R}^{(n-p-k)}$ encodes the permissible velocity on $\mathcal{S}$ respecting the nonholonomic constraint (3-23). We can then partition the mass matrix $\boldsymbol{M}(\boldsymbol{r})$ as:

$$\boldsymbol{M}(\boldsymbol{r}) = \begin{bmatrix} \boldsymbol{M_s}(\boldsymbol{r}) & \boldsymbol{M_{sr}}(\boldsymbol{r}) \\ \boldsymbol{M_{sr}}^\top(\boldsymbol{r}) & \boldsymbol{M_r}(\boldsymbol{r}) \end{bmatrix} \tag{3-27}$$

such that the transformed mass matrix becomes:

$$\tilde{\boldsymbol{M}}(\boldsymbol{r}) = \boldsymbol{S}(\boldsymbol{r})^\top \boldsymbol{M}(\boldsymbol{r}) \boldsymbol{S}(\boldsymbol{r}) = \begin{bmatrix} \mathcal{D}_s^\top \boldsymbol{M_s} \mathcal{D}_s & \mathcal{D}_s^\top \boldsymbol{M_{sr}} \\ \boldsymbol{M_{sr}}^\top \mathcal{D}_s & \boldsymbol{M_r} \end{bmatrix} \tag{3-28}$$

In order to avoid acceleration couplings via the inertia matrix between the s-dynamics and the r-dynamics which is usually not cancellable we follow another assumption from [7]:

$$\boldsymbol{M_{sr}^\top}(\boldsymbol{r}) \mathcal{D}_s(\boldsymbol{r}) = \mathbf{0_{p\times(n-p-k)}},\ \forall \boldsymbol{r} \in \mathbb{R}^p \tag{3-29}$$

Thus $\tilde{\boldsymbol{M}}(\boldsymbol{r})$ becomes block-diagonal leading to decoupling of $\boldsymbol{s}$ and $\boldsymbol{r}$ via the inertia matrix. We can further decompose the transformed generalized momenta as $\tilde{\boldsymbol{p}} = \begin{pmatrix} \tilde{\boldsymbol{p}}_s \\ \boldsymbol{p_r} \end{pmatrix}$ and choose a block diagonal desired mass matrix $\boldsymbol{M_d}(\boldsymbol{r})$ as:

$$\boldsymbol{M_d}(\boldsymbol{r}) = \begin{bmatrix} \boldsymbol{M_{ds}} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{M_{dr}}(\boldsymbol{r}) \end{bmatrix} \tag{3-30}$$

with with $\boldsymbol{M_{ds}}$ independent from $\boldsymbol{r}$. The Hamiltonian in (3-21) can be decomposed as:

$$\begin{aligned} H_d &= \frac{1}{2} \begin{pmatrix} \tilde{\boldsymbol{p}}_s^\top & \boldsymbol{p_r}^\top \end{pmatrix} \begin{bmatrix} \boldsymbol{M_{ds}} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{M_{dr}}(\boldsymbol{r}) \end{bmatrix} \begin{pmatrix} \tilde{\boldsymbol{p}}_s \\ \boldsymbol{p_r} \end{pmatrix} + V_{ds}(\boldsymbol{s}) + V_{dr}(\boldsymbol{r}) \\ &= \underbrace{\frac{1}{2} \tilde{\boldsymbol{p}}_s^\top \boldsymbol{M_{ds}}(\boldsymbol{r}) \tilde{\boldsymbol{p}}_s + V_{ds}(\boldsymbol{s})}_{H_{ds}} + \underbrace{\frac{1}{2} \boldsymbol{p_r}^\top \boldsymbol{M_{dr}} \boldsymbol{p_r} + V_{dr}(\boldsymbol{r})}_{H_{dr}} \end{aligned} \tag{3-31}$$

Note that the generalized momenta $\boldsymbol{p_r}$ is denoted without a tilde symbol since it is invariant to the constraints. With zero feed-forward control $\boldsymbol{\tau_d} = \mathbf{0}$ and with proper block-diagonal choices for matrices $\boldsymbol{J}$ and $\boldsymbol{K_v}$, system (3-21) can be decomposed in two Hamiltonian systems (each evolving on its respective configuration space) as:

$$\begin{cases} \begin{bmatrix} \dot{\boldsymbol{s}} \\ \dot{\tilde{\boldsymbol{p}}}_{\boldsymbol{s}} \end{bmatrix} = \begin{bmatrix} \mathbf{0_{n-p}} & \mathcal{D}_s(\mathcal{D}_s^\top \boldsymbol{M_s}\mathcal{D}_s)^{-1}\boldsymbol{M_{ds}} \\ -\boldsymbol{M_{ds}}(\mathcal{D}_s^\top \boldsymbol{M_s}\mathcal{D}_s)^{-1}\mathcal{D}_s^\top & \boldsymbol{J_s} - \tilde{\boldsymbol{F}}_{\boldsymbol{s}}\boldsymbol{K_{vs}}\tilde{\boldsymbol{F}}_{\boldsymbol{s}}{}^\top \end{bmatrix} \begin{bmatrix} \frac{\partial H_{ds}}{\partial \boldsymbol{s}}(\boldsymbol{s}) \\ \frac{\partial H_{ds}}{\partial \tilde{\boldsymbol{p}}_{\boldsymbol{s}}}(\tilde{\boldsymbol{p}}_{\boldsymbol{s}}) \end{bmatrix} \\ \boldsymbol{y_{ds}} = \tilde{\boldsymbol{F}}_{\boldsymbol{s}}{}^\top \dfrac{\partial H_{ds}}{\partial \tilde{\boldsymbol{p}}_{\boldsymbol{s}}} \\ H_{ds} = \dfrac{1}{2}\tilde{\boldsymbol{p}}_{\boldsymbol{s}}{}^\top \boldsymbol{M_{ds}}{}^{-1}\tilde{\boldsymbol{p}}_{\boldsymbol{s}} + V_{ds}(\boldsymbol{s}) \end{cases} \tag{3-32}$$

$$\begin{cases} \begin{bmatrix} \dot{\boldsymbol{r}} \\ \dot{\boldsymbol{p}}_{\boldsymbol{r}} \end{bmatrix} = \begin{bmatrix} \mathbf{0_p} & \boldsymbol{M_r}{}^{-1}\boldsymbol{M_{dr}} \\ -\boldsymbol{M_{dr}}\boldsymbol{M_r}{}^{-1} & \boldsymbol{J_r} - \tilde{\boldsymbol{F}}_{\boldsymbol{r}}\boldsymbol{K_{vr}}\tilde{\boldsymbol{F}}_{\boldsymbol{r}}{}^\top \end{bmatrix} \begin{bmatrix} \frac{\partial H_{dr}}{\partial \boldsymbol{r}}(\boldsymbol{r},\boldsymbol{p_r}) \\ \frac{\partial H_{dr}}{\partial \boldsymbol{p_r}}(\boldsymbol{r},\boldsymbol{p_r}) \end{bmatrix} \\ \boldsymbol{y_{dr}} = \tilde{\boldsymbol{F}}_{\boldsymbol{r}}{}^\top \dfrac{\partial H_{ds}}{\partial \boldsymbol{p_r}} \\ H_{dr} = \dfrac{1}{2}\boldsymbol{p_r}{}^\top \boldsymbol{M_{dr}}{}^{-1}(\boldsymbol{r})\boldsymbol{p_r} + V_{dr}(\boldsymbol{r}) \end{cases} \tag{3-33}$$

The two systems (3-32) and (3-33) are decoupled and each one evolves on its own configuration manifold $\mathcal{S}$ and $\mathcal{R}$ respectively thought with a coupling due to the nonholonomic constraint. Note also that system (3-33) is of the original unconstrained Hamiltonian form. Thus, the unconstrained variables $\boldsymbol{r}$ are easy to stabilize with a smooth control law whereas, for $\boldsymbol{s}$, the stabilization is not straightforward. Based on that fact that $\dot{\boldsymbol{s}} = \mathcal{D}_s(\boldsymbol{r})\dot{\tilde{\boldsymbol{s}}}$ we can see that a promising attempt is the following: Drive the $\boldsymbol{s}$-dynamics to the invariant set $\boldsymbol{\Omega}_{inv}$ at a stabilization point, let it be $\boldsymbol{s^\omega}$, while recruiting $\boldsymbol{r}$ to "guide" the direction of $\mathcal{D}_s(\boldsymbol{r})$ so as to steer the system from $\boldsymbol{s^\omega}$ towards $\boldsymbol{s^*}$. Using this passive configuration decomposition as in [7] we can proceed in designing a passivity-based switching control law that can asymptotically stabilize the system in any configuration $\boldsymbol{q^*} = \begin{pmatrix} \boldsymbol{s^*}^\top & \boldsymbol{r^*}^\top \end{pmatrix}^\top$.

3-2-4 Passivity-Based Switching Control

With the control law (3-14) and the aforementioned assumptions we can control independently the two decomposed systems derived in the previous sections. We begin with analytically deriving the energy evolution of (3-21) baring in mind the decomposition in the previous section yielding:

$$\dot{H}_d = \frac{\partial^\top H_d}{\partial \boldsymbol{q}}\dot{\boldsymbol{q}} + \frac{\partial^\top H_d}{\partial \tilde{\boldsymbol{p}}}\dot{\tilde{\boldsymbol{p}}} = \underbrace{\frac{\partial^\top H_d}{\partial \boldsymbol{s}}\dot{\boldsymbol{s}} + \frac{\partial^\top H_d}{\partial \tilde{\boldsymbol{p}}_{\boldsymbol{s}}}\dot{\tilde{\boldsymbol{p}}}_{\boldsymbol{s}}}_{\dot{H}_{ds}} + \underbrace{\frac{\partial^\top H_d}{\partial \boldsymbol{r}}\dot{\boldsymbol{r}} + \frac{\partial^\top H_d}{\partial \boldsymbol{p_r}}\dot{\boldsymbol{p}}_{\boldsymbol{r}}}_{\dot{H}_{dr}} \tag{3-34}$$

So for the system evolving on $\mathcal{S}$ we have:

$$\dot{H}_{ds} = \cancel{\frac{\partial^\top H_d}{\partial \boldsymbol{s}} \mathcal{D}_s (\mathcal{D}_s^\top \boldsymbol{M_1} \mathcal{D}_s)^{-1} \boldsymbol{M_{ds}} \frac{\partial H_d}{\partial \tilde{\boldsymbol{p}}_s}} - \cancel{\frac{\partial^\top H_d}{\partial \tilde{\boldsymbol{p}}_s} \boldsymbol{M_{ds}} (\mathcal{D}_s^\top \boldsymbol{M_1} \mathcal{D}_s)^{-1} \mathcal{D}_s^\top \frac{\partial H_d}{\partial \boldsymbol{s}}} + \frac{\partial^\top H_d}{\partial \tilde{\boldsymbol{p}}_s} \underbrace{\left(\boldsymbol{J}_s - \tilde{\boldsymbol{F}}_s \boldsymbol{K}_{vs} \tilde{\boldsymbol{F}}_s{}^\top \right)}_{<0} \frac{\partial H_d}{\partial \tilde{\boldsymbol{p}}_s} \leq \boldsymbol{0} \tag{3-35}$$

with:

$$\dot{H}_{ds} = 0 \Rightarrow \boldsymbol{M_{ds}}^{-1} \tilde{\boldsymbol{p}}_s = \boldsymbol{0} \Rightarrow \tilde{\boldsymbol{p}}_s = \boldsymbol{0} \tag{3-36}$$

$$\dot{\tilde{\boldsymbol{p}}}_s = \boldsymbol{0} \Rightarrow -\boldsymbol{M_{ds}} (\mathcal{D}_s^\top \boldsymbol{M_1} \mathcal{D}_s)^{-1} \mathcal{D}_s^\top \frac{\partial^\top H_d}{\partial \boldsymbol{s}} + \left(\boldsymbol{J}_s - \tilde{\boldsymbol{F}}_s \boldsymbol{K}_{vs} \tilde{\boldsymbol{F}}_s{}^\top \right) \boldsymbol{M_d} \boldsymbol{s}^{-1} \cancel{\tilde{\boldsymbol{p}}_s}^{\boldsymbol{0}} = \boldsymbol{0}$$
$$\Rightarrow \boxed{\mathcal{D}_s^\top (\boldsymbol{r}) \frac{\partial V_{ds}(\boldsymbol{s})}{\partial \boldsymbol{s}} = \boldsymbol{0}_{\boldsymbol{n-p-k}}} \tag{3-37}$$

and for the system evolving on $\mathcal{R}$ we have in a similar manner:

$$\dot{H}_{dr} = \cancel{\frac{\partial^\top H_d}{\partial \boldsymbol{r}} \boldsymbol{M_r}^{-1} \boldsymbol{M_{dr}} \frac{\partial H_d}{\partial \boldsymbol{p_r}}} - \cancel{\frac{\partial^\top H_d}{\partial \boldsymbol{p_r}} \boldsymbol{M_{dr}} \boldsymbol{M_r}^{-1} \frac{\partial H_d}{\partial \boldsymbol{r}}} + \frac{\partial^\top H_d}{\partial \boldsymbol{p_r}} \underbrace{\left(\boldsymbol{J}_r - \tilde{\boldsymbol{F}}_r \boldsymbol{K}_{vr} \tilde{\boldsymbol{F}}_r{}^\top \right)}_{<0} \frac{\partial H_d}{\partial \boldsymbol{p_r}} \leq \boldsymbol{0} \tag{3-38}$$

with:

$$\dot{H}_{dr} = 0 \Rightarrow \boldsymbol{M_{dr}}^{-1} \boldsymbol{p_r} = \boldsymbol{0} \Rightarrow \boldsymbol{p_r} = \boldsymbol{0} \tag{3-39}$$

$$\dot{\boldsymbol{p}}_{\boldsymbol{r}} = \boldsymbol{0} \Rightarrow -\boldsymbol{M_{dr}} \boldsymbol{M_r}^{-1} \frac{1}{2} \cancel{\boldsymbol{p_r}}^{\boldsymbol{0}} \frac{\partial \boldsymbol{M_{dr}}}{\partial \boldsymbol{r}} \cancel{\boldsymbol{p_r}}^{\boldsymbol{0}} + \frac{\partial V_{dr}(\boldsymbol{r})}{\partial \boldsymbol{r}} + \left(\boldsymbol{J}_r - \tilde{\boldsymbol{F}}_r \boldsymbol{K}_{vr} \tilde{\boldsymbol{F}}_r{}^\top \right) \boldsymbol{M_d} \boldsymbol{s}^{-1} \cancel{\boldsymbol{p_r}}^{\boldsymbol{0}} = \boldsymbol{0}$$
$$\Rightarrow \boxed{\frac{\partial V_{dr}(\boldsymbol{r})}{\partial \boldsymbol{r}} = \boldsymbol{0}_{\boldsymbol{p}}} \tag{3-40}$$

The goal to stabilize the system at a desired configuration $\boldsymbol{q} \rightarrow \boldsymbol{q}^*$ can be achieved by stabilizing the decomposed variables separately by driving $\boldsymbol{s} \rightarrow \boldsymbol{s}^*$ by utilizing $\boldsymbol{r}$ to assist us and then $\boldsymbol{r} \rightarrow \boldsymbol{r}^*$. Thus, according to equations (3-37) and (3-40) we need to design the desired potential functions V_{ds} and V_{dr} respectively in a suitable manner. Note also that this design needs to lead in a time-variant or switching control so as not to be in contradiction with Brockett's necessary condition. Based on [7], function $V_{ds} : \mathcal{S} \rightarrow \mathbb{R}$ is subjected to the following:

1. $V_{ds} \geq 0$ with the equality holding when $\boldsymbol{s} = \boldsymbol{s}^*$
2. $\frac{\partial V_{ds}}{\partial \boldsymbol{s}} = 0$, iff $\boldsymbol{s} = \boldsymbol{s}^*$
3. V_{ds} is radially unbounded

We begin with the s-dynamics for which we choose a quadratic function that satisfies the aforementioned requirements defined as:

$$V_{ds} = \frac{1}{2} (\boldsymbol{s} - \boldsymbol{s}^*)^\top \boldsymbol{Q}_s (\boldsymbol{s} - \boldsymbol{s}^*) \tag{3-41}$$

with $\boldsymbol{Q_s} \in \mathbb{R}^{(n-p)\times(n-p)}$ a constant symmetric matrix serving tuning purposes. With this choice equation (3-37) gives:

$$\mathcal{D}_s^\top(\boldsymbol{r})\boldsymbol{Q_s}(\boldsymbol{s}-\boldsymbol{s^*}) = \boldsymbol{0_{n-p-k}} \tag{3-42}$$

which describes a k-dimensional affine hyperplane in $\mathcal{S} \in \mathbb{R}^{n-p}$ that is defined by a set of $n-p-k$ linear equations. Thus the system will not be stabilized at $\boldsymbol{s^*}$ but rather at another point denoted as $\boldsymbol{s^\omega} \in \boldsymbol{\Omega}_{inv}$. Let $\boldsymbol{v_s} = \boldsymbol{s} - \boldsymbol{s^*}$ be the vector that we want to drive to zero. Then, matrix $\mathcal{D}_s(\boldsymbol{r})\boldsymbol{Q_s}$ maps this vector to the constrained space as a new vector:

$$\boldsymbol{v_\alpha} = \mathcal{D}_s^\top(\boldsymbol{r})\boldsymbol{Q_s}\boldsymbol{v_s} \in \mathbb{R}^{n-p-k} \tag{3-43}$$

which is the vector on the constraint space that we are able to drive to zero $\boldsymbol{v_\alpha} \to \boldsymbol{0}$ with the potential function V_{ds} chosen as in (3-41), and acting only on the constrained variables $\boldsymbol{s}$ thus driving $\boldsymbol{s} \to \boldsymbol{s^\omega}$.

We continue with the following critical observation: Since we have assumed that the nonholonomic constraints are a function of only $\boldsymbol{r} \in \mathcal{R}$ and act only on $\boldsymbol{s} \in \mathcal{S}$ the constraint equation (3-23) is now integrable in $\mathcal{S}$ and can get the form:

$$\boldsymbol{A_s}^\top(\boldsymbol{r})\frac{d\boldsymbol{s}}{dt} = \boldsymbol{0} \Rightarrow \int_{\boldsymbol{s}}^{\boldsymbol{s^\omega}} \boldsymbol{A_s}^\top(\boldsymbol{r})\, d\boldsymbol{s} = \int_0^{t^\omega} \boldsymbol{0}\, dt \Rightarrow \boldsymbol{A_s}^\top(\boldsymbol{r})\,(\boldsymbol{s^\omega} - \boldsymbol{s}) = \boldsymbol{0} \tag{3-44}$$

which describes an $(n-p-k)$-dimensional, affine hyperplane in $\mathcal{S} \in \mathbb{R}^{n-p}$ that is defined by a set of k linear equations. This affine hyperplane describes the constraint space that the system will evolve on, a subspace of $\mathcal{S}$. We observe here that as matrix $\mathcal{D}_s(\boldsymbol{r})\boldsymbol{Q_s}$ maps $\boldsymbol{v_s}$ to the constrained space described in (3-44), the same way matrix $\boldsymbol{A_s}^\top(\boldsymbol{r})$ maps $\boldsymbol{v_s}$ to the invariant set defined in (3-42) as the vector:

$$\boldsymbol{v_\omega} = \boldsymbol{A_s}^\top(\boldsymbol{r})\boldsymbol{v_s} \in \mathbb{R}^k \tag{3-45}$$

and since these spaces are the orthogonal complement of each other, we know that vector $\boldsymbol{v_s}$ will go to zero if both $\boldsymbol{v_\alpha} \to \boldsymbol{0}$ and $\boldsymbol{v_\omega} \to \boldsymbol{0}$ is achieved. We have already showed that $\boldsymbol{v_\alpha} \to \boldsymbol{0}$ is feasible for the quadratic choice of V_{ds} in (3-41). The concept now is to use the unconstrained variable $\boldsymbol{r}$ in order to drive $\boldsymbol{v_\omega}$ to zero as well meaning that $\boldsymbol{s^\omega} \to \boldsymbol{s^*}$ and thus $\boldsymbol{s} \to \boldsymbol{s^*}$. This is possible by the following quadratic choice:

$$V_{dr} = \frac{1}{2}\boldsymbol{v_\omega}^\top\boldsymbol{Q_r}\boldsymbol{v_\omega} \tag{3-46}$$

with $\boldsymbol{Q_r} \in \mathbb{R}^{k\times k}$ a constant symmetric matrix serving tuning purposes. Substituting this expression into (3-40) yields the following result:

$$\frac{\partial V_{dr}}{\partial \boldsymbol{r}} = \boldsymbol{0} \Rightarrow \frac{\partial V_{dr}}{\partial \boldsymbol{v_\omega}}\frac{\partial \boldsymbol{v_\omega}}{\partial \boldsymbol{r}} = \boldsymbol{0} \Rightarrow \boldsymbol{v_\omega}^\top\boldsymbol{Q_r}\frac{\partial \boldsymbol{v_\omega}}{\partial \boldsymbol{r}} = \boldsymbol{0} \Rightarrow \frac{\partial^\top \boldsymbol{v_\omega}}{\partial \boldsymbol{r}}\boldsymbol{Q_r}\boldsymbol{v_\omega} = \boldsymbol{0} \tag{3-47}$$

The system will be stabilized at $\boldsymbol{v_\omega} = \boldsymbol{0}$, and since $\boldsymbol{v_\alpha} = \boldsymbol{0}$ can be driven to zero we conclude that we get $\boldsymbol{v_s} = \boldsymbol{0}$ meaning that $\boldsymbol{s} \to \boldsymbol{s^*}$. Note that $\boldsymbol{v_\alpha} = \boldsymbol{v_\alpha}(\boldsymbol{s}, \boldsymbol{r})$, $\boldsymbol{v_\omega} = \boldsymbol{v_\omega}(\boldsymbol{s}, \boldsymbol{r})$ meaning that both the desired potentials are functions of both the constrained and unconstrained variables implying a coupling of the systems via the potential components of the control law (3-14). However, due to orthogonality each desired potential V_{ds} and V_{dr} leads to potential

forces that act only on their respective variables. More specifically, the control action on $\mathcal{S}$ is the term $\mathcal{D}_s^\top(\boldsymbol{r})\boldsymbol{Q_s}\boldsymbol{v_s} \in \mathbb{R}^{n-p-k}$ and while a function of $\boldsymbol{r}$ it only acts on the $\boldsymbol{s}$ variables. Similarly, the control action on $\mathcal{R}$ is given by $\frac{\partial V_{dr}}{\partial \boldsymbol{r}} = \frac{\partial^\top \boldsymbol{v_\omega}}{\partial \boldsymbol{r}}\boldsymbol{Q_r}\boldsymbol{v_\omega}$ and while it is a function of $\boldsymbol{s}$ acts only on $\boldsymbol{r}$.

Remark: Notice that since $\boldsymbol{v_\omega} = \boldsymbol{A_s}^\top(\boldsymbol{r})\boldsymbol{v_s}$, as $\boldsymbol{s}$ approaches $\boldsymbol{s}^*$ then $\boldsymbol{v_\omega}$ will also become arbitrarily small. In order to circumvent this problem we can define:

$$\boldsymbol{v_{s_n}} = \frac{\boldsymbol{v_s}}{\|\boldsymbol{v_s}\|} \tag{3-48}$$

And then redefine $\boldsymbol{v_\omega}$ as:

$$\boldsymbol{v_\omega} = \boldsymbol{A_s}^\top(\boldsymbol{r})\boldsymbol{v_{s_n}} \in \mathbb{R}^k \tag{3-49}$$

which ensures that the control action for the unconstrained variables will have the proper direction but it will be invariant of how close the system is to $\boldsymbol{s}^*$.

Having achieved $\boldsymbol{s} \to \boldsymbol{s}^*$ (i.e. stabilizing the constrained variables $\boldsymbol{s}$ which are in general difficult to handle), we can shift our attention to the unconstrained variables $\boldsymbol{r}$. The unconstrained variables $\boldsymbol{r}$ are not stabilized on the desired equilibrium $\boldsymbol{r}^*$ since they were used so far to stabilize the other variables. Now that $\boldsymbol{s} = \boldsymbol{s}^*$ we can switch to the desired potential function for $\boldsymbol{r}$ expressed as:

$$V_{dr} = \frac{1}{2}(\boldsymbol{r} - \boldsymbol{r}^*)^\top \boldsymbol{Q_r}(\boldsymbol{r} - \boldsymbol{r}^*) \tag{3-50}$$

and since the $\boldsymbol{r}$ variables are not hindered by constraints they can be stabilized to the desired equilibrium $\boldsymbol{r}^*$. Note that the aforementioned control choices lead to asymptotic stabilization which is more of theoretical interest since $\boldsymbol{s} \to \boldsymbol{s}^*$ typically takes infinitely long time. For this reason, we can attain $\boldsymbol{r} \to \boldsymbol{r}^*$ by triggering the switch when the norms $\|\boldsymbol{s} - \boldsymbol{s}^*\|$ and $\left\|\dot{\tilde{\boldsymbol{s}}}\right\|$ are small enough (i.e. setting stopping criteria). Thus we can define the desired potential to stabilize $\boldsymbol{r}$ as:

$$V_{dr} = \begin{cases} \frac{1}{2}\boldsymbol{v_\omega}^\top \boldsymbol{Q_r}\boldsymbol{v_\omega} \\ \frac{1}{2}(\boldsymbol{r} - \boldsymbol{r}^*)^\top \boldsymbol{Q_r}(\boldsymbol{r} - \boldsymbol{r}^*) \quad \text{if } \|\boldsymbol{s} - \boldsymbol{s}^*\| < s^d, \quad \left\|\dot{\tilde{\boldsymbol{s}}}\right\| < \dot{s}^d \end{cases} \tag{3-51}$$

Example 3 (Continued): Full State Stabilization

In this simple example we can provide some insight on the proposed solution giving a graphical explanation. Consider the knife edge system which has been used so far with $n = 3$ degrees of freedom (where (q_1, q_2) define its position in this plane, q_3 its orientation) and the following energy components:

$$V = 0, \quad \boldsymbol{M} = \left[\begin{array}{cc|c} m & 0 & 0 \\ 0 & m & 0 \\ \hline 0 & 0 & I \end{array}\right]$$

In the constraint velocity space we also have the following components describing the system:

$$V = 0, \quad \tilde{\boldsymbol{M}} = \begin{bmatrix} m & 0 \\ 0 & I \end{bmatrix}, \quad \boldsymbol{S} = \begin{bmatrix} \cos q_3 & 0 \\ \sin q_3 & 0 \\ 0 & 1 \end{bmatrix}, \quad \tilde{\boldsymbol{F}} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \boldsymbol{Y} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

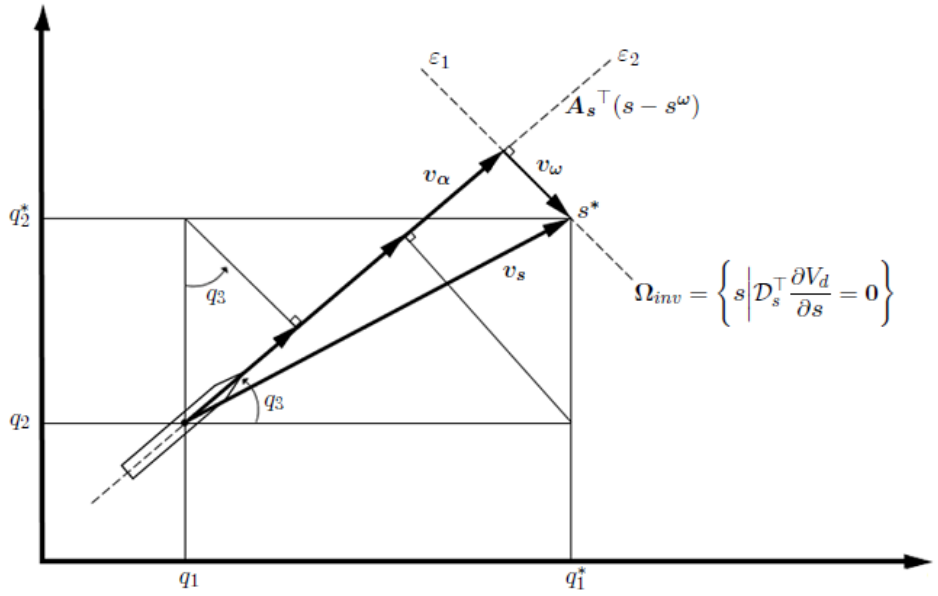


Figure 3-2: Full-state stabilization of the knife edge example

First we investigate if the system can be decomposed as in section 3-2-3. We can set $\boldsymbol{s} = (q_1, q_2)^\top$ and $\boldsymbol{r} = (q_3)$ satisfying the first assumption (3-22). Moreover, the second assumption expressed in (3-23) is also satisfied since the constraint equation is:

$$\begin{pmatrix} \sin q_3 & -\cos q_3 & 0 \end{pmatrix} \dot{\boldsymbol{q}} = 0 \Rightarrow \boldsymbol{A_s}(\boldsymbol{r})\dot{\boldsymbol{s}} = 0$$

with:

$$\mathcal{D}_s(\boldsymbol{r}) = \begin{pmatrix} \cos q_3 \\ \sin q_3 \end{pmatrix}$$

From the mass matrix we have $\boldsymbol{M_s} = \begin{bmatrix} m & 0 \\ 0 & m \end{bmatrix}$, $\boldsymbol{M_{rs}} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$ and $\boldsymbol{M_r} = I$ thus satisfying the third assumption that the mass matrix is not a function of $\boldsymbol{s}$. In addition we have $\mathcal{D}_s^\top \boldsymbol{M_{rs}} = \mathbf{0}$ which means that the system can be successfully decoupled.

This particular system is fully actuated in the constrained space since $\tilde{\boldsymbol{F}}$ is invertible. Thus, the matching conditions vanish. For simplicity, we can then choose $\boldsymbol{M_d} = \tilde{\boldsymbol{M}}$, $\boldsymbol{J} = \mathbf{0_{n-k}}$ and $\boldsymbol{\tau_d} = \mathbf{0}$ and a block diagonal damping matrix $\boldsymbol{K_v}$. Thus the control law (3-14) takes the simpler form:

$$\boldsymbol{\tau} = -\boldsymbol{S}^\top \frac{\partial H_d}{\partial \boldsymbol{q}} - \boldsymbol{K_v y_d} = -\mathcal{D}_s^\top \frac{\partial V_{ds}}{\partial \boldsymbol{s}} - \frac{\partial V_{dr}}{\partial \boldsymbol{r}} - \boldsymbol{K_{vs}}\dot{\boldsymbol{s}} - \boldsymbol{K_{vr}}\dot{\boldsymbol{r}}$$

We choose the desired potentials V_{ds} and V_{dr} as described in this section with $\boldsymbol{Q_s} = \boldsymbol{I_{n-p}}$ and $\boldsymbol{Q_r} = \boldsymbol{I_k}$ for simplicity:

$$V_{ds} = \frac{1}{2}(\boldsymbol{s} - \boldsymbol{s}^*)^\top (\boldsymbol{s} - \boldsymbol{s}^*)$$

$$V_{dr} = \begin{cases} \frac{1}{2}\boldsymbol{v_\omega}^\top \boldsymbol{v_\omega} \\ \frac{1}{2}(\boldsymbol{r} - \boldsymbol{r}^*)^\top (\boldsymbol{r} - \boldsymbol{r}^*) \quad \text{if } \|\boldsymbol{s} - \boldsymbol{s}^*\| < s^d, \quad \left\|\dot{\tilde{\boldsymbol{s}}}\right\| < \dot{s}^d \end{cases} \tag{3-52}$$

then the invariant set takes the form:

$$\mathcal{D}_s^{\top}(\boldsymbol{r})\boldsymbol{Q_s}(\boldsymbol{s}-\boldsymbol{s^*}) = \mathbf{0_{n-p-k}} \Rightarrow (q_2 - q_2^*) = -\cot q_3(q_1 - q_1^*)$$

Which represents a straight line in the $q_1 - q_2$ plane denoted as ε_1. From equation (3-44) we then have:

$$\boldsymbol{A_s}^{\top}(\boldsymbol{r})\,(\boldsymbol{s^\omega} - \boldsymbol{s}) = \mathbf{0_k} \Rightarrow (q_2 - q_2^\omega) = \tan q_3(q_1 - q_1^\omega)$$

Which represents another straight line in the $q_1 - q_2$ plane denoted as ε_2. Notice that if λ_1, λ_2 are the respective slopes we have:

$$\lambda_1\lambda_2 = -\cot q_3 \tan q_3 = -1$$

Implying that $\varepsilon_1 \perp \varepsilon_2$. The vectors $\boldsymbol{v_\alpha}$ and $\boldsymbol{v_\omega}$ we are trying to drive to zero are given as:

$$\boldsymbol{v_\alpha} = \mathcal{D}_s^{\top}\boldsymbol{v_s} = \cos q_3(q_1 - q_1^*) + \sin q_3(q_2 - q_2^*)$$

$$\boldsymbol{v_\omega} = \boldsymbol{A}_s^{\top}\boldsymbol{v_s} = \sin q_3(q_1 - q_1^*) - \cos q_3(q_2 - q_2^*)$$

The aforementioned are illustrated in Figure 3-2. The potential forces from the chosen quadratic desired potentials can be interpreted as the linear spring forces depicted in this figure: We have a linear spring ($\frac{\partial V_{ds}}{\partial \boldsymbol{s}}$) which acts on the constraint space and drives $\boldsymbol{s} \to \boldsymbol{s}^\omega$ and another linear spring ($\frac{\partial V_{dr}}{\partial \boldsymbol{r}}$) which acts on the invariant set and drives $\boldsymbol{s}^\omega \to \boldsymbol{s}^*$. After driving $\boldsymbol{s} \to \boldsymbol{s}^*$ we can switch V_{dr} and drive also $\boldsymbol{r} \to \boldsymbol{r^*}$ thus achieving full-state stabilization $\boldsymbol{q} \to \boldsymbol{q}^*$.

Chapter 4

Distributed IDA-PBC for Nonholonomic Systems

In this chapter we aim to extend the single-agent results derived so far to a distributed control scheme. More specifically, in the following sections we are attempting to bridge the single-agent results of Chapter 3 with the distributed version of Interconnection and Damping Assignment Passivity-Based Control (IDA-PBC) which was introduced in [1, 2]. This chapter is based on the original work in [1, 2] and its summary in [3]. We include this part not only for completeness but also because the system equations are different due to the constrains which leads to different results as well.

4-1 Towards a Distributed Solution

4-1-1 Local and Cooperative Variables

In order to extend the method presented so far to a distributed setting, further assumptions need to be posed. We begin by splitting the generalized coordinates $\boldsymbol{q} \in \mathbb{R}^n$ into $\boldsymbol{\theta} \in \mathbb{R}^{n-l}$ and $\boldsymbol{x} \in \mathbb{R}^l$. In the distributed setting, agents should cooperatively control variable $\boldsymbol{x}$ while locally stabilizing their local variable $\boldsymbol{\theta}$. Following this separation, we define a new variable $\boldsymbol{z}(\boldsymbol{q}) = \boldsymbol{z}(\boldsymbol{x}, \boldsymbol{\theta}) \in \mathbb{R}^l$ such that $\boldsymbol{z} \to \boldsymbol{z}^*$ and $\boldsymbol{\theta} \to \boldsymbol{\theta}^*$ implies that $\boldsymbol{x} \to \boldsymbol{x}^*$. For particular definitions of $\boldsymbol{z}(\boldsymbol{q})$ we can ensure that the matching conditions are never violated, which is essential in the shaping of the cooperative potential in Section 4-2. With this new definition of variables, interest goes to systems for which the potential can be written as:

$$V_d(\boldsymbol{q}) = V_l(\boldsymbol{q}) + V_c(\boldsymbol{z}(\boldsymbol{q})) \tag{4-1}$$

Herein, $V_l(\boldsymbol{q})$ is the *locally stabilizing potential function* and $V_c(\boldsymbol{z}(\boldsymbol{q}))$ is the *cooperative potential function*. The latter function is free as long as $V_d(\boldsymbol{q})$ remains positive definite in a neighbourhood near the goal $\boldsymbol{q}^*$. In the control law, the partial derivative of the potential

function $\frac{\partial V_d(\boldsymbol{q})}{\partial \boldsymbol{q}}$ appears. After posing assumption (4-1) for the potential energy, this derivative can be written as:

$$\frac{\partial V_d}{\partial \boldsymbol{q}} = \frac{\partial V_l}{\partial \boldsymbol{q}} + \boldsymbol{\Psi}\frac{\partial V_c}{\partial \boldsymbol{z}} \tag{4-2}$$

with:

$$\boldsymbol{\Psi} = \frac{\partial \boldsymbol{z}}{\partial \boldsymbol{q}} = \left[\frac{\partial \boldsymbol{z_1}(\boldsymbol{q})}{\partial \boldsymbol{q}}, \frac{\partial \boldsymbol{z_2}(\boldsymbol{q})}{\partial \boldsymbol{q}}, \ldots, \frac{\partial \boldsymbol{z_l}(\boldsymbol{q})}{\partial \boldsymbol{q}}\right] = \begin{bmatrix} \frac{\partial \boldsymbol{z_1}(\boldsymbol{q})}{\partial \boldsymbol{q_1}} & \cdots & \frac{\partial \boldsymbol{z_l}(\boldsymbol{q})}{\partial \boldsymbol{q_1}} \\ \vdots & & \vdots \\ \frac{\partial \boldsymbol{z_1}(\boldsymbol{q})}{\partial \boldsymbol{q_n}} & \cdots & \frac{\partial \boldsymbol{z_l}(\boldsymbol{q})}{\partial \boldsymbol{q_n}} \end{bmatrix} \in \mathbb{R}^{n\times l} \tag{4-3}$$

If now V_l is selected such that:

$$\boldsymbol{F}^{\perp}\left(\frac{\partial V}{\partial \boldsymbol{q}} - \boldsymbol{M_d}\boldsymbol{M}^{-1}\frac{\partial V_l}{\partial \boldsymbol{q}}\right) = \boldsymbol{0} \tag{4-4}$$

holds, then the the potential matching condition (3-13) can be satisfied by requiring:

$$\boldsymbol{F}^{\perp}\boldsymbol{M_d}\boldsymbol{M}^{-1}\boldsymbol{\Psi} = \boldsymbol{0}_{(n-m)\times l} \tag{4-5}$$

Note that requiring (4-4) and (4-5) to hold is more conservative than for the potential (3-13) to hold. Nevertheless this strategy ensures that the matching equations are always satisfied, while $V_c(\boldsymbol{z}(\boldsymbol{q}))$ is free in $\boldsymbol{z}(\boldsymbol{q})$. Therefore any positive definite potential is admissible as V_c, while the existing single-agent solutions can be used for local stabilization and kinetic energy shaping. With the aforementioned considerations, the control law (3-14) can be rewritten by categorizing the appearing terms as:

$$\boldsymbol{\tau} = \boldsymbol{\sigma} - \boldsymbol{\Phi}\frac{\partial V_c}{\partial \boldsymbol{z}} - \boldsymbol{K_v}\boldsymbol{y_d} + \boldsymbol{\tau_d} \tag{4-6}$$

where the term $\boldsymbol{\sigma} \in \mathbb{R}^m$ achieves energy shaping for stabilization:

$$\boldsymbol{\sigma} = \left(\tilde{\boldsymbol{F}}^{\top}\tilde{\boldsymbol{F}}\right)^{-1}\tilde{\boldsymbol{F}}^{\top}\left(\boldsymbol{S}^{\top}\frac{\partial \tilde{H}}{\partial \boldsymbol{q}} - \boldsymbol{M_d}\tilde{\boldsymbol{M}}^{-1}\frac{\partial\left(\frac{1}{2}\tilde{\boldsymbol{p}}^{\top}\boldsymbol{M_d}^{-1}\tilde{\boldsymbol{p}} + V_l\right)}{\partial \boldsymbol{q}} - \boldsymbol{Y}\frac{\partial \tilde{H}}{\partial \tilde{\boldsymbol{p}}} + \boldsymbol{J}\frac{\partial H_d}{\partial \tilde{\boldsymbol{p}}}\right) \tag{4-7}$$

Matrix $\boldsymbol{\Phi} \in \mathbb{R}^{m\times l}$ denotes the matrix that ensures that the matching conditions are satisfied, while the cooperative potential remains free:

$$\boldsymbol{\Phi} = \left(\tilde{\boldsymbol{F}}^{\top}\tilde{\boldsymbol{F}}\right)^{-1}\tilde{\boldsymbol{F}}^{\top}\boldsymbol{M}_d\boldsymbol{M}^{-1}\boldsymbol{S}^{\top}\boldsymbol{\Psi} \tag{4-8}$$

4-1-2 Connection With Passive Configuration Decomposition

Consider the Passive Configuration Decomposition in Section 3-2-3 where we are interested in systems whose generalized coordinates $\boldsymbol{q}$ can be decomposed as $\boldsymbol{q} = \begin{pmatrix}\boldsymbol{s}^{\top} & \boldsymbol{r}^{\top}\end{pmatrix}^{\top}$, $\boldsymbol{s} \in \mathbb{R}^{n-p}$, $\boldsymbol{r} \in \mathbb{R}^p$ and assume that the necessary assumptions made in that section hold. The unconstrained variables $\boldsymbol{r}$ are used by the agent so that the constrained variables $\boldsymbol{s}$ can be stabilized first and then, by switching the control law, they can be stabilized subsequently

to an arbitrary position. This means that the unconstrained variables $\boldsymbol{r}$ need to act as *local* variables until the constrained variables are stabilized. After this task is completed, we have the freedom to choose if the $\boldsymbol{r}$ variables are going to remain local or if they will become cooperative variables as well. In the latter scenario, they need to be communicated via the network too.

Let's consider the general case in which we want to stabilize all variables $\boldsymbol{q} = \begin{pmatrix} \boldsymbol{s}^\top & \boldsymbol{r}^\top \end{pmatrix}^\top$ cooperatively. Since stabilization of the constrained variables $\boldsymbol{s}$ happens by properly driving the unconstrained variables $\boldsymbol{r}$ first, the latter need information on variables $\boldsymbol{s}$ of the other agents since these constitute the goal in a cooperative task. Then, the $\boldsymbol{r}$ variables can be cooperatively stabilized if shared after the switching of V_{dr} in (3-51) takes place. The unconstrained variables $\boldsymbol{r}$ then act as cooperative variables for both goals. In case that we are not interested in cooperative stabilization of the $\boldsymbol{s}$ variables (i.e. they are not shared), $\boldsymbol{r}$ act first as a local variables driving $\boldsymbol{s} \to \boldsymbol{s}^*$ and then after switching they can be stabilized cooperatively.

We introduce here the switching function sw_h defined as:

$$sw_h = \begin{cases} 1 & \text{if } \|\boldsymbol{s} - \boldsymbol{s}^*\| < s^d \text{ and } \left\|\dot{\tilde{\boldsymbol{s}}}\right\| \leq \dot{\tilde{s}}^d \\ 0 & \text{otherwise} \end{cases} \tag{4-9}$$

where s^d and $\dot{\tilde{s}}^d$ are sufficiently small switching criteria introduced for practical reasons as discussed in Section 3-2-4. This switch turns on the "holonomic" action in a constraint system when the constrained variables have been stabilized successfully. We can then partition the local potentials as:

$$V_l(\boldsymbol{q}) = V_{ls}(\boldsymbol{q}) + (1 - sw_h)V_{lr}(\boldsymbol{q}) \tag{4-10}$$

while the cooperative potential can partitioned as:

$$V_c(\boldsymbol{q}) = V_{cs}(\boldsymbol{z_s}) + sw_h V_{cr}(\boldsymbol{z_r}) \tag{4-11}$$

where the subscripts "s" and "r" in the cooperative variables $\boldsymbol{z}$ denote the dependence on either the unconstrained or the constrained variables respectively. For a nonholonomic system, the local potential V_{lr} will serve the nonholonomic purpose of the unconstrained variables $\boldsymbol{r}$ and when this is complete, it will vanish leaving control of these variables to the cooperative potential $V_{cr}(\boldsymbol{z_r})$ if necessary. Note that the local potential $V_{ls}(\boldsymbol{q})$ may contain another subset of variables $\boldsymbol{s}$ not used as cooperative variables but as variables that need to be stabilized locally for example the pendulum angle of an inverted mobile pendulum [13]. In case of a holonomic system we can simply set $sw_h = 1 \quad \forall t$ which will yield the original IDA-PBC control law.

4-2 Distributed IDA-PBC for Nonholonomic Systems

4-2-1 Total System Description

Consider N agents, where each agent i with n_i generalized coordinates, k_i Pfaffian constraints and m_i control inputs is described by the Hamiltonian dynamics (3-1) as:

$$\begin{cases} \begin{bmatrix} \dot{\boldsymbol{q}}_{\boldsymbol{i}} \\ \dot{\tilde{\boldsymbol{p}}}_{\boldsymbol{i}} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{\boldsymbol{n_i}} & \boldsymbol{S}_i(\boldsymbol{q_i}) \\ -\boldsymbol{S_i}^{\top}(\boldsymbol{q_i}) & \boldsymbol{Y}_i(\boldsymbol{q_i}, \tilde{\boldsymbol{p}}_i) \end{bmatrix} \begin{bmatrix} \frac{\partial \tilde{H}_i}{\partial \boldsymbol{q_i}}(\boldsymbol{q_i}, \tilde{\boldsymbol{p}}_i) \\ \frac{\partial \tilde{H}_i}{\partial \tilde{\boldsymbol{p}}_i}(\boldsymbol{q_i}, \tilde{\boldsymbol{p}}_i) \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{n_i \times m_i} \\ \tilde{\boldsymbol{F}}_i(\boldsymbol{q_i}) \end{bmatrix} \boldsymbol{\tau_i} \\ \tilde{\boldsymbol{y}}_{\boldsymbol{i}} = \tilde{\boldsymbol{F}}_{\boldsymbol{i}}{}^{\top}(\boldsymbol{q_i}) \dfrac{\partial \tilde{H}_i}{\partial \tilde{\boldsymbol{p}}_{\boldsymbol{i}}}(\boldsymbol{q_i}, \tilde{\boldsymbol{p}}_{\boldsymbol{i}}) \\ \tilde{H}_i(\boldsymbol{q_i}, \tilde{\boldsymbol{p}}_{\boldsymbol{i}}) = \dfrac{1}{2} \tilde{\boldsymbol{p}}_{\boldsymbol{i}}{}^{\top} \tilde{\boldsymbol{M}}_{\boldsymbol{i}}{}^{-1}(\boldsymbol{q_i}) \tilde{\boldsymbol{p}}_{\boldsymbol{i}} + V_i(\boldsymbol{q_i}) \end{cases} \tag{4-12}$$

Each of the dynamic parameters and dimensions may be different for each agent which allows the multi-agent system to be *heterogeneous*. Moreover, some agents may be not imposed to any nonholonomic constraints in which case $\boldsymbol{S_i} = \boldsymbol{I_{n_i}}$, $\boldsymbol{Y_i} = \mathbf{0}_{\boldsymbol{n_i}}$, $\tilde{\boldsymbol{M}}_{\boldsymbol{i}} = \boldsymbol{M_i}$ and $\tilde{\boldsymbol{F}}_{\boldsymbol{i}} = \boldsymbol{F_i}$ thus the dynamics take the original holonomic form (for example found in [2]). We can write the dynamics of the total system, consisting of these N agents, as:

$$\begin{cases} \begin{bmatrix} \dot{\bar{\boldsymbol{q}}} \\ \dot{\bar{\boldsymbol{p}}} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{\bar{\boldsymbol{n}}} & \bar{\boldsymbol{S}}(\bar{\boldsymbol{q}}) \\ -\bar{\boldsymbol{S}}^{\top}(\bar{\boldsymbol{q}}) & \bar{\boldsymbol{Y}}(\bar{\boldsymbol{q}}, \bar{\boldsymbol{p}}) \end{bmatrix} \begin{bmatrix} \frac{\partial \bar{H}}{\partial \bar{\boldsymbol{q}}}(\bar{\boldsymbol{q}}, \bar{\boldsymbol{p}}) \\ \frac{\partial \bar{H}}{\partial \bar{\boldsymbol{p}}}(\bar{\boldsymbol{q}}, \bar{\boldsymbol{p}}) \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{\bar{\boldsymbol{n}} \times \bar{\boldsymbol{m}}} \\ \bar{\boldsymbol{F}}(\boldsymbol{q}) \end{bmatrix} \bar{\boldsymbol{\tau}} \\ \bar{\boldsymbol{y}} = \bar{\boldsymbol{F}}^{\top}(\bar{\boldsymbol{q}}) \dfrac{\partial \bar{H}}{\partial \bar{\boldsymbol{p}}}(\bar{\boldsymbol{q}}, \bar{\boldsymbol{p}}) \\ \bar{H}(\bar{\boldsymbol{q}}, \bar{\boldsymbol{p}}) = \dfrac{1}{2} \bar{\boldsymbol{p}}^{\top} \bar{\boldsymbol{M}}^{-1}(\bar{\boldsymbol{q}}) \bar{\boldsymbol{p}} + \bar{V}(\bar{\boldsymbol{q}}) \end{cases} \tag{4-13}$$

where the bar notation denotes composed system components explicitly given by:

$$\bar{n} = \sum_{i=1}^{N} n_i, \quad \bar{m} = \sum_{i=1}^{N} m_i, \quad \bar{k} = \sum_{i=1}^{N} k_i, \quad \bar{V} = \sum_{i=1}^{N} V_i$$

$$\bar{\boldsymbol{M}} = \begin{bmatrix} \tilde{\boldsymbol{M}}_{\boldsymbol{1}} & & \\ & \ddots & \\ & & \tilde{\boldsymbol{M}}_{\boldsymbol{N}} \end{bmatrix}, \quad \bar{\boldsymbol{F}} = \begin{bmatrix} \tilde{\boldsymbol{F}}_{\boldsymbol{1}} & & \\ & \ddots & \\ & & \tilde{\boldsymbol{F}}_{\boldsymbol{N}} \end{bmatrix}, \quad \bar{\boldsymbol{q}} = \begin{bmatrix} \boldsymbol{q_1} \\ \vdots \\ \boldsymbol{q_N} \end{bmatrix}, \quad \bar{\boldsymbol{p}} = \begin{bmatrix} \tilde{\boldsymbol{p}}_{\boldsymbol{1}} \\ \vdots \\ \tilde{\boldsymbol{p}}_{\boldsymbol{N}} \end{bmatrix}$$

$$\bar{\boldsymbol{S}} = \begin{bmatrix} \boldsymbol{S_1} & & \\ & \ddots & \\ & & \boldsymbol{S_N} \end{bmatrix}, \quad \bar{\boldsymbol{Y}} = \begin{bmatrix} \boldsymbol{Y_1} & & \\ & \ddots & \\ & & \boldsymbol{Y_N} \end{bmatrix}, \quad \bar{\boldsymbol{\tau}} = \begin{bmatrix} \boldsymbol{\tau_1} \\ \vdots \\ \boldsymbol{\tau_N} \end{bmatrix}, \quad \bar{\boldsymbol{y}} = \begin{bmatrix} \tilde{\boldsymbol{y}}_{\boldsymbol{1}} \\ \vdots \\ \tilde{\boldsymbol{y}}_{\boldsymbol{N}} \end{bmatrix} \tag{4-14}$$

Following the single-agent case, the IDA-PBC method defines a control input $\bar{\boldsymbol{\tau}}$ that changes the uncontrolled network dynamics (4-14) into the asymptotically stable dynamics:

$$\begin{cases} \begin{bmatrix} \dot{\bar{\boldsymbol{q}}} \\ \dot{\bar{\boldsymbol{p}}} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{\bar{\boldsymbol{n}}} & \bar{\boldsymbol{S}}\bar{\boldsymbol{M}}^{-1}\bar{\boldsymbol{M}}_{\boldsymbol{d}} \\ -\bar{\boldsymbol{M}}_{\boldsymbol{d}}\bar{\boldsymbol{M}}^{-1}\bar{\boldsymbol{S}}^{\top} & \bar{\boldsymbol{J}} - \bar{\boldsymbol{F}}\bar{\boldsymbol{K}}_{\boldsymbol{v}}\bar{\boldsymbol{F}}^{\top} \end{bmatrix} \begin{bmatrix} \frac{\partial \bar{H}_d}{\partial \bar{\boldsymbol{q}}}(\bar{\boldsymbol{q}},\bar{\boldsymbol{p}}) \\ \frac{\partial \bar{H}_d}{\partial \bar{\boldsymbol{p}}}(\bar{\boldsymbol{q}},\bar{\boldsymbol{p}}) \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{\bar{\boldsymbol{n}}\times\bar{\boldsymbol{m}}} \\ \bar{\boldsymbol{F}}(\bar{\boldsymbol{q}}) \end{bmatrix} \bar{\boldsymbol{\tau}}_{\boldsymbol{d}} \\ \bar{\boldsymbol{y}}_{\boldsymbol{d}} = \bar{\boldsymbol{F}}^{\top}(\bar{\boldsymbol{q}}) \dfrac{\partial \bar{H}_d}{\partial \bar{\boldsymbol{p}}}(\bar{\boldsymbol{q}},\bar{\boldsymbol{p}}) \\ \bar{H}_d(\bar{\boldsymbol{q}},\bar{\boldsymbol{p}}) = \dfrac{1}{2}\bar{\boldsymbol{p}}^{\top}\bar{\boldsymbol{M}}_{\boldsymbol{d}}^{-1}(\bar{\boldsymbol{q}})\bar{\boldsymbol{p}} + \bar{V}_d(\bar{\boldsymbol{q}}) \end{cases} \tag{4-15}$$

The desired dynamics for the total system are equivalent to those in the single-agent case and a similar approach will be followed. Splitting the desired total potential energy into local and cooperative components as explained in Section 4-1 yields:

$$\bar{V}_d = \bar{V}_c + \sum_{i=1}^{N} V_{l,i} \tag{4-16}$$

where $V_{s,i}$ are the known potentials V_l for each single-agent solution that serve local stability purposes and $\bar{V}_c$ is a potential to be designed for cooperative objectives.

4-2-2 Matching Conditions and Control Law

Since the composed system description is equivalent to the description of the single-agent, the results of Section 3-1-2 are directly expendable to the multi-agent case. The control law is then expressed as:

$$\bar{\boldsymbol{\tau}}(\bar{\boldsymbol{q}},\bar{\boldsymbol{p}}) = \left(\bar{\boldsymbol{F}}^{\top}\bar{\boldsymbol{F}}\right)^{-1}\bar{\boldsymbol{F}}^{\top}\left(\bar{\boldsymbol{S}}^{\top}\frac{\partial \bar{H}}{\partial \bar{\boldsymbol{q}}} - \bar{\boldsymbol{M}}_{\boldsymbol{d}}\bar{\boldsymbol{M}}^{-1}\bar{\boldsymbol{S}}^{\top}\frac{\partial \bar{H}_d}{\partial \bar{\boldsymbol{q}}} - \bar{\boldsymbol{Y}}\frac{\partial \bar{H}}{\partial \bar{\boldsymbol{p}}} + \bar{\boldsymbol{J}}\frac{\partial \bar{H}}{\partial \bar{\boldsymbol{p}}}\right) - \bar{\boldsymbol{K}}_{\boldsymbol{v}}\bar{\boldsymbol{y}}_{\boldsymbol{d}} + \bar{\boldsymbol{\tau}}_{\boldsymbol{d}} \tag{4-17}$$

The kinetic and potential matching conditions are expressed as:

$$\bar{\boldsymbol{F}}^{\perp}\left(\bar{\boldsymbol{S}}^{\top}\frac{\partial \bar{\boldsymbol{p}}^{T}\bar{\boldsymbol{M}}^{-1}\bar{\boldsymbol{p}}}{\partial \bar{\boldsymbol{q}}} - \bar{\boldsymbol{M}}_{\boldsymbol{d}}\bar{\boldsymbol{M}}^{-1}\bar{\boldsymbol{S}}^{\top}\frac{\partial \bar{\boldsymbol{p}}^{T}\bar{\boldsymbol{M}}_{\boldsymbol{d}}^{-1}\bar{\boldsymbol{p}}}{\partial \bar{\boldsymbol{q}}} - 2\bar{\boldsymbol{Y}}\bar{\boldsymbol{M}}^{-1}\bar{\boldsymbol{p}} + 2\bar{\boldsymbol{J}}\bar{\boldsymbol{M}}_{\boldsymbol{d}}^{-1}\bar{\boldsymbol{p}}\right) = \mathbf{0}_{\bar{\boldsymbol{n}}-\bar{\boldsymbol{k}}-\bar{\boldsymbol{m}}} \tag{4-18a}$$

$$\bar{\boldsymbol{F}}^{\perp}\left(\bar{\boldsymbol{S}}^{\top}\frac{\partial \bar{V}}{\partial \bar{\boldsymbol{q}}} - \bar{\boldsymbol{M}}_{\boldsymbol{d}}\bar{\boldsymbol{M}}^{-1}\bar{\boldsymbol{S}}^{\top}\frac{\partial \bar{V}_d}{\partial \bar{\boldsymbol{q}}}\right) = \mathbf{0}_{\bar{\boldsymbol{n}}-\bar{\boldsymbol{k}}-\bar{\boldsymbol{m}}} \tag{4-18b}$$

Note that $\bar{\boldsymbol{\tau}}(\bar{\boldsymbol{q}},\bar{\boldsymbol{p}})$ can in general be a function of all coordinates in the system. Since we seek for a distributed solution, this information is not necessarily available for agent i. Some of the information has to be communicated over a network between agents. This network can be modelled by a communication graph $\mathcal{G}$ termed here as the *position graph*. In Appendix A a brief summary of graph theory is presented. We assume that the communication among agents is delay free since network effects are beyond the scope of this work. The interested reader is refereed to [4] for further information on the topic.

4-2-3 Control Objectives

The overall objective is to have each agent i stabilize its own local coordinates $\boldsymbol{\theta_i} \in \mathbb{R}^{n_i - l}$ at $\boldsymbol{\theta_i^*}$ while achieving a desired stationary formation between the agents in $\boldsymbol{x_i} \in \mathbb{R}^l$ coordinates. The latter objective is presented as a formation in $\boldsymbol{z_i} \in \mathbb{R}^l$. The aforementioned objectives can be grouped as individual local and cooperative objectives. Local objectives include stationary formation and local stabilization and are expressed respectively as:

$$\lim_{t\to\infty} \left\|\dot{\boldsymbol{q}}_{\boldsymbol{i}}(t)\right\| = \mathbf{0}, \quad \forall i = 1, ..., N \tag{4-19}$$

$$\lim_{t\to\infty} \left\|\boldsymbol{\theta_i}(t) - \boldsymbol{\theta_i^*}(t)\right\| = \mathbf{0}, \quad \forall i = 1, ..., N \tag{4-20}$$

Cooperative objectives include reaching a desired inter-agent formation with or without leader. These objectives are expresses as:

$$\lim_{t\to\infty} \left\|\boldsymbol{z_i}(t) - \boldsymbol{z_j}(t) + \boldsymbol{z_{ij}^*}\right\| = \mathbf{0}, \quad \forall (i,j) | \boldsymbol{\mathcal{A}_{ij}} > \mathbf{0} \tag{4-21}$$

$$\lim_{t\to\infty} \left\|\boldsymbol{z_i}(t) - \boldsymbol{z_i^*}(t)\right\| = \mathbf{0}, \quad \forall i | \boldsymbol{\mathcal{B}_i} > \mathbf{0} \tag{4-22}$$

where $\boldsymbol{\mathcal{A}_{ij}}$ denotes the adjacency matrix and $\boldsymbol{\mathcal{B}_i}$ the leader matrix. These matrices describe the network topology and are further explained in Appendix A. Equation (4-21) expresses the *formation forming* objective among the agents. For the special case of $\boldsymbol{z_{ij}^*} = \mathbf{0}$, $\forall (i,j) | \boldsymbol{\mathcal{A}_{ij}} > \mathbf{0}$ the problem is termed as *consensus* or *synchronization*. Equation (4-22) describes the formation objective for leader agents where $\boldsymbol{z_i^*}$ denotes their goal configuration. Lastly, if $\boldsymbol{\mathcal{B}_i} = \mathbf{0}$, $\forall i = 1, ..., N$ then no leaders exist in the network. In order to achieve the aforementioned objectives suitable choices for the design components $\bar{V}_c$ and $\bar{\boldsymbol{K}}_{\boldsymbol{v}}$ should be made.

Selection of the cooperative component $\bar{V}_c$ depends on the cooperative objectives described in (4-21) and (4-22). In [2] its selection is proposed as a quadratic function of the form:

$$\begin{aligned} \bar{V}_c = &\frac{1}{4} \sum_{i=1}^{N} \sum_{j=1}^{N} \left(\boldsymbol{z_i} - \boldsymbol{z_j} + \boldsymbol{z_{ij}^*}\right)^T \boldsymbol{\mathcal{A}_{ij}} \left(\boldsymbol{z_i} - \boldsymbol{z_j} + \boldsymbol{z_{ij}^*}\right) \\ &+ \frac{1}{2} \sum_{i=1}^{N} \left(\boldsymbol{z_i} - \boldsymbol{z_i^*}\right)^T \boldsymbol{\mathcal{B}_i} \left(\boldsymbol{z_i} - \boldsymbol{z_i^*}\right) \end{aligned} \tag{4-23}$$

Note that $\bar{V}_c > 0$ and $\bar{V}_c = 0$ if and only if the objective is reached. The Jacobian of the potential function is:

$$\frac{\partial \bar{V}_c}{\partial \bar{\boldsymbol{z}}} = \begin{bmatrix} \boldsymbol{\mathcal{B}_i} \left(\boldsymbol{z_1} - \boldsymbol{z_1^*}\right) + \sum_{j=1}^{N} \boldsymbol{\mathcal{A}_{1j}} \left(\boldsymbol{z_1} - \boldsymbol{z_j} + \boldsymbol{z_{1j}^*}\right) \\ \vdots \\ \boldsymbol{\mathcal{B}_N} \left(\boldsymbol{z_N} - \boldsymbol{z_N^*}\right) + \sum_{j=1}^{N} \boldsymbol{\mathcal{A}_{Nj}} \left(\boldsymbol{z_N} - \boldsymbol{z_j} + \boldsymbol{z_{Nj}^*}\right) \end{bmatrix} \tag{4-24}$$

which is only zero at the desired formation. The Jacobian for each agent i depends not only on its own coordinates, but also on the coordinates of its neighbours introducing interaction among them. These inter-agent terms describe the energy contained in the interconnections.

Last but not least, a proper damping matrix should be selected in order to achieve the stationary formation objective expressed in (4-19) making the system asymptotically stable.

This matrix enters the control law through the product $\bar{\boldsymbol{K}}_{\boldsymbol{v}}\bar{\boldsymbol{y}}_{\boldsymbol{d}}$. Hence, its block diagonal terms describe local damping, while its off-diagonal blocks describe inter-agent damping. Since the latter requires that the agents exchange velocities consider this information exchange over a velocity graph, let it be $\mathcal{G}^v$ so that it is not necessarily the same as the position graph $\mathcal{G}$. Then consider to partition $\bar{\boldsymbol{K}}_{\boldsymbol{v}}$ as:

$$\bar{\boldsymbol{K}}_{\boldsymbol{v}} = \boldsymbol{\mathcal{L}}^{\boldsymbol{v}} + \boldsymbol{\mathcal{B}}^{\boldsymbol{v}} > \mathbf{0} \tag{4-25}$$

where $\boldsymbol{\mathcal{L}}_{\boldsymbol{v}}$ is the Laplacian matrix and $\boldsymbol{\mathcal{B}}_{\boldsymbol{v}}$ denotes the leader matrix of the velocity graph (see Appendix A). Through this choice the positivity condition on the damping matrix is translated to conditions on the network. Given the partitioning in (4-25) the damping term in the control law can be expressed as:

$$-\bar{\boldsymbol{K}}_{\boldsymbol{v}}\bar{\boldsymbol{y}}_{\boldsymbol{d}} = -\bar{\boldsymbol{K}}_{\boldsymbol{v}}\dot{\bar{\boldsymbol{q}}} = \begin{bmatrix} -\boldsymbol{\mathcal{B}}_{\mathbf{1}}^{\boldsymbol{v}}\dot{\boldsymbol{q}}_{\mathbf{1}} - \sum_{j=1}^{N}\boldsymbol{\mathcal{A}}_{\mathbf{1}\boldsymbol{j}}^{\boldsymbol{v}}\left(\dot{\boldsymbol{q}}_{\mathbf{1}} - \dot{\boldsymbol{q}}_{\boldsymbol{j}}\right) \\ \vdots \\ -\boldsymbol{\mathcal{B}}_{\boldsymbol{N}}^{\boldsymbol{v}}\dot{\boldsymbol{q}}_{\boldsymbol{N}} - \sum_{j=1}^{N}\boldsymbol{\mathcal{A}}_{\boldsymbol{N}\boldsymbol{j}}^{\boldsymbol{v}}\left(\dot{\boldsymbol{q}}_{\boldsymbol{N}} - \dot{\boldsymbol{q}}_{\boldsymbol{j}}\right) \end{bmatrix} \tag{4-26}$$

Two implementation examples that lead to asymptotic convergence are:

1. All agents are leader on the velocity graph.
2. The velocity graph is connected and there is at least one leader on the velocity graph.

The first example requires that all agents can measure their velocity, but does not require inter-agent communication therefore locally satisfy the damping condition. The second example introduces inter-agent damping by using a velocity graph. Note that there are more possible network topologies that satisfy the damping conditions.

4-2-4 Distributed IDA-PBC Solutions

In this section a summary of the solution to the distributed IDA-PBC problem is presented. This summary covers solely the underactuated solutions, since these solutions contain the fully actuated solutions as a special case. For the solution of the distributed IDA-PBC problem it is essential that single-agent solutions exist so that $V_{s,i}$, $\boldsymbol{J}_{\boldsymbol{i}}$ and $\boldsymbol{M}_{\boldsymbol{d},\boldsymbol{i}}$ are known for each subsystem.

Consider combining the single-agent IDA-PBC solutions of all agents in a system as:

$$\begin{aligned} \bar{\boldsymbol{M}}_{\boldsymbol{d}} &= \begin{bmatrix} \boldsymbol{M}_{\boldsymbol{d},\mathbf{1}} & & \\ & \ddots & \\ & & \boldsymbol{M}_{\boldsymbol{d},\boldsymbol{N}} \end{bmatrix}, \quad \bar{\boldsymbol{J}} = \begin{bmatrix} \boldsymbol{J}_{\mathbf{1}} & & \\ & \ddots & \\ & & \boldsymbol{J}_{\boldsymbol{N}} \end{bmatrix} \\ \bar{\boldsymbol{K}}_{\boldsymbol{v}} &= \begin{bmatrix} \boldsymbol{K}_{\boldsymbol{v},\mathbf{1}} & & \\ & \ddots & \\ & & \boldsymbol{K}_{\boldsymbol{v},\boldsymbol{N}} \end{bmatrix}, \quad \bar{\boldsymbol{F}}^{\perp} = \begin{bmatrix} \boldsymbol{F}_{\mathbf{1}}{}^{\perp} & & \\ & \ddots & \\ & & \boldsymbol{F}_{\boldsymbol{N}}{}^{\perp} \end{bmatrix} \end{aligned} \tag{4-27}$$

It can be shown that using these definitions, the kinetic energy matching condition of the multi-agent systems (4-13) and (4-15) separates into N independent Partial Differential Equations (PDEs) given by:

$$\boldsymbol{F_i}^{\perp}\left(\boldsymbol{S_i}^{\top}\frac{\partial \tilde{\boldsymbol{p}}_{\boldsymbol{i}}{}^{T}\tilde{\boldsymbol{M}}_{\boldsymbol{i}}{}^{-1}\tilde{\boldsymbol{p}}_{\boldsymbol{i}}}{\partial \boldsymbol{q_i}}-\boldsymbol{M_{d,i}}\tilde{\boldsymbol{M}}_{\boldsymbol{i}}{}^{-1}\boldsymbol{S_i}^{\top}\frac{\partial \tilde{\boldsymbol{p}}_{\boldsymbol{i}}{}^{T}\boldsymbol{M_{d,i}}{}^{-1}\tilde{\boldsymbol{p}}_{\boldsymbol{i}}}{\partial \boldsymbol{q_i}}-2\boldsymbol{Y_i}\tilde{\boldsymbol{M}}_{\boldsymbol{i}}{}^{-1}\tilde{\boldsymbol{p}}_{\boldsymbol{i}}+2\boldsymbol{J_i}\boldsymbol{M_{d,i}}{}^{-1}\tilde{\boldsymbol{p}}_{\boldsymbol{i}}\right)$$
$$=\boldsymbol{0_{n_i-k_i-m_i}}\quad \forall i=1,\ldots,N \tag{4-28}$$

These PDEs correspond to the single-agent kinetic matching condition (3-12) and hence are satisfied by the use of existing single-agent solutions. Similarly to the single agent case, separating the multi-agent potential energy matching condition gives:

$$\boldsymbol{F_i}^{\perp}\left(\boldsymbol{S_i}^{\top}\frac{\partial V_i}{\partial \boldsymbol{q_i}}-\boldsymbol{M_{d,i}}\tilde{\boldsymbol{M}}_{\boldsymbol{i}}{}^{-1}\boldsymbol{S_i}^{\top}\frac{\partial \bar{V}_d}{\partial \boldsymbol{q_i}}\right)=\boldsymbol{0_{n_i-k_i-m_i}}\quad \forall i=1,\ldots,N \tag{4-29}$$

In order to satisfy both local and cooperative objectives while satisfying the potential matching condition consider the partitioning of the potential energy as:

$$\bar{V}_d=\bar{V}_c+\sum_{i=1}^{N}V_{s,i} \tag{4-30}$$

Similarly to the single agent case consider to structure $\bar{V}_c$ as:

$$\bar{V}_c(\bar{\boldsymbol{z}}(\bar{\boldsymbol{q}}))=\bar{V}_c\left(\bar{\boldsymbol{z}}_{\boldsymbol{1}}\left(\bar{\boldsymbol{q}}_{\boldsymbol{1}}\right),\ldots,\bar{\boldsymbol{z}}_{\boldsymbol{N}}\left(\bar{\boldsymbol{q}}_{\boldsymbol{N}}\right)\right) \tag{4-31}$$

so that the partial derivatives to local coordinates $\boldsymbol{q_i}$ satisfy:

$$\frac{\partial \bar{V}_c}{\partial \boldsymbol{q_i}}=\boldsymbol{\Psi_i}\frac{\partial \bar{V}_c}{\partial \boldsymbol{z_i}} \tag{4-32}$$

with $\boldsymbol{\Psi_i}$ denoting the transpose Jacobian:

$$\boldsymbol{\Psi_i}=\frac{\partial^T \boldsymbol{z_i}}{\partial \boldsymbol{q_i}}=\begin{bmatrix}\frac{\partial \boldsymbol{z_{1,i}}(\boldsymbol{q})}{\partial \boldsymbol{q_{1,i}}} & \cdots & \frac{\partial \boldsymbol{z_{l,i}}(\boldsymbol{q})}{\partial \boldsymbol{q_{1,i}}}\\ \vdots & & \vdots\\ \frac{\partial \boldsymbol{z_{1,i}}(\boldsymbol{q})}{\partial \boldsymbol{q_{n_i,i}}} & \cdots & \frac{\partial \boldsymbol{z_{l,i}}(\boldsymbol{q})}{\partial \boldsymbol{q_{n_i,i}}}\end{bmatrix}\in\mathbb{R}^{n_i\times l} \tag{4-33}$$

The PDEs in (4-29) can then also be partitioned after substitution of (4-30) in (4-29) as:

$$\boldsymbol{F_i}^{\perp}\left(\boldsymbol{S_i}^{\top}\frac{\partial V_i}{\partial \boldsymbol{q_i}}-\boldsymbol{M_{d,i}}\tilde{\boldsymbol{M}}_{\boldsymbol{i}}{}^{-1}\boldsymbol{S_i}^{\top}\frac{\partial V_{s,i}}{\partial \boldsymbol{q_i}}-\boldsymbol{M_{d,i}}\tilde{\boldsymbol{M}}_{\boldsymbol{i}}{}^{-1}\boldsymbol{S_i}^{\top}\boldsymbol{\Psi_i}\frac{\partial \bar{V}_c}{\partial \boldsymbol{z_i}}\right)=\boldsymbol{0_{n_i-k_i-m_i}}\quad \forall i=1,\ldots,N \tag{4-34}$$

The single-agent solutions ensure that:

$$\bar{\boldsymbol{F}}_{\boldsymbol{i}}{}^{\perp}\boldsymbol{M_{d,i}}\tilde{\boldsymbol{M}}_{\boldsymbol{i}}{}^{-1}\boldsymbol{S_i}^{\top}\boldsymbol{\Psi_i}=\boldsymbol{0_{n_i-k_i-m_i}}\quad \forall i=1,\ldots,N \tag{4-35}$$

Therefore:

$$\bar{\boldsymbol{F}}_{\boldsymbol{i}}{}^{\perp}\boldsymbol{M_{d,i}}\tilde{\boldsymbol{M}}_{\boldsymbol{i}}{}^{-1}\boldsymbol{S_i}^{\top}\frac{\partial \bar{V}_c}{\partial \boldsymbol{q_i}}=\bar{\boldsymbol{F}}_{\boldsymbol{i}}{}^{\perp}\boldsymbol{M_{d,i}}\tilde{\boldsymbol{M}}_{\boldsymbol{i}}{}^{-1}\boldsymbol{S_i}^{\top}\boldsymbol{\Psi_i}\frac{\partial \bar{V}_c}{\partial \boldsymbol{z_i}}=\boldsymbol{0_{n_i-k_i-m_i}}\quad \forall i=1,\ldots,N \tag{4-36}$$

And also:

$$\boldsymbol{F_i}^{\perp}\left(\boldsymbol{S_i}^{\top}\frac{\partial V_i}{\partial \boldsymbol{q_i}}-\boldsymbol{M_{d,i}}\tilde{\boldsymbol{M}}_{\boldsymbol{i}}^{-1}\boldsymbol{S_i}^{\top}\frac{\partial V_{s,i}}{\partial \boldsymbol{q_i}}\right)=\boldsymbol{0}_{n_i-k_i-m_i}\quad\forall i=1,\ldots,N \tag{4-37}$$

Meaning that the potential matching condition is also satisfied and hence the matching conditions are satisfied. The control law for agent i is formulated by substituting quadratic potential (4-23) as well as the single agent solutions (4-27) into (4-17) resulting in:

$$\boldsymbol{\tau_i}=\boldsymbol{\sigma}_i+\boldsymbol{\Phi}_i\left(\boldsymbol{\mathcal{B}}_i\left(\boldsymbol{z}_i^*-\boldsymbol{z}_i\right)+\sum_{j=1}^{N}\boldsymbol{\mathcal{A}}_{ij}\left(\boldsymbol{z}_j-\boldsymbol{z}_i-\boldsymbol{r}_{ij}^*\right)\right)-\boldsymbol{\mathcal{B}}_i^v\boldsymbol{y}_{d,i}+\sum_{j=1}^{N}\boldsymbol{\mathcal{A}}_{ij}^v\left(\boldsymbol{y}_{d,j}-\boldsymbol{y}_{d,i}\right) \tag{4-38}$$

where $\boldsymbol{\sigma_i}\in\mathbb{R}^{m_i}$ denotes the locally stabilising control for agent i, given by:

$$\boldsymbol{\sigma_i}=$$
$$\left(\tilde{\boldsymbol{F}}_{\boldsymbol{i}}^{\top}\tilde{\boldsymbol{F}}_{\boldsymbol{i}}\right)^{-1}\tilde{\boldsymbol{F}}_{\boldsymbol{i}}^{\top}\left(\boldsymbol{S_i}^{\top}\frac{\partial\tilde{H}_i}{\partial \boldsymbol{q_i}}-\boldsymbol{M_{d,i}}\tilde{\boldsymbol{M}}_{\boldsymbol{i}}^{-1}\boldsymbol{S_i}^{\top}\frac{\partial\left(\frac{1}{2}\tilde{\boldsymbol{p}}_{\boldsymbol{i}}^{\top}\boldsymbol{M_{d,i}}^{-1}\tilde{\boldsymbol{p}}_{\boldsymbol{i}}+V_{s,i}\right)}{\partial \boldsymbol{q_i}}-\boldsymbol{Y_i}\frac{\partial\tilde{H}_i}{\partial\tilde{\boldsymbol{p}}_{\boldsymbol{i}}}+\boldsymbol{J_i}\frac{\partial H_{d,i}}{\partial\tilde{\boldsymbol{p}}_{\boldsymbol{i}}}\right) \tag{4-39}$$

with:

$$\boldsymbol{\Phi_i}=\left(\tilde{\boldsymbol{F}}_{\boldsymbol{i}}^{\top}\tilde{\boldsymbol{F}}_{\boldsymbol{i}}\right)^{-1}\tilde{\boldsymbol{F}}_{\boldsymbol{i}}^{\top}\boldsymbol{M_{d,i}}\tilde{\boldsymbol{M}}_{\boldsymbol{i}}^{-1}\boldsymbol{\Psi_i} \tag{4-40}$$

the matrix for agent i that ensures that the matching conditions are satisfied, while the cooperative potential remains free.

Chapter 5

Collision Avoidance

While the developed methodology presented so far can achieve multi-agent coordination it does not guarantee collision avoidance between an agent and the physical obstacles that exist in its environment or among agents themselves. The problem of collision avoidance is far from trivial and has drawn a lot of attention in the research community for decades. In this chapter we aim to present a simple method to achieve dynamic collision avoidance in the local environment of each agent and show how it can be implemented in accordance with the results for nonholonomic systems derived in the previous chapters.

5-1 The Collision Avoidance Problem

5-1-1 Brief Overview

Numerous approaches serving collision avoidance have been developed so far. Many of them rely on fundamentally different frameworks. Following [24] two initial distinctions can be made in collision avoidance methods:

- Motion Planning Techniques
- Obstacle Avoidance Techniques

Motion planning techniques consider the collision avoidance problem as a motion planning problem to begin with, by designing collision free trajectories in the configuration space of the agent. This approach has the advantage of giving complete and global solutions but when the surroundings are unknown and unpredictable these techniques often fail. Moreover, the obstacle representation in the configuration space can prove to be really difficult for complex obstacle geometries when also the rotational, three-dimensional motion of an agent is taken into account. Last but not least, path planning methods usually aim to present a higher level control strategy and do not accomplish path following directly. Nevertheless approaches such the Probabilistic Roadmap (PRM), Rapidly Exploring Random Tree (RRT) and their optimal versions, have yielded successful collision free path planning [25, 26, 27]. Moreover, there exist control schemes such as Model Predictive Control (MPC) that integrate the entire motion planning problem (path generation and path following) simultaneously.

Obstacle avoidance techniques have the objective to move an agent towards a target location free of collisions with the obstacles collected by the sensors during motion execution. The advantage of reactive obstacle avoidance is to compute motion by introducing the sensor information within the control loop, used to adapt the motion to any contingency incompatible with initial plans. The main disadvantage of considering the reality of the world during execution is locality.

A taxonomy of obstacle avoidance techniques is described in [24]. The different methods in literature can be distinguished as:

1. Heuristic methods which were the first techniques used to generate motion based on sensors.
2. Methods of physical analogies which assimilate the obstacle avoidance problem. Among those, the Artificial Potential Fields method is known for its widespread use along with some of its variants e.g. the Navigation Functions.
3. Methods of subsets of controls which compute an intermediate set of motion controls and next they choose one of them as a solution. They can be further distinguished in two types:
   - Methods that compute a subset of motion directions (Vector Field Histogram (VFH), Obstacle Restriction Method (ORM), Steer Angle Field Approach (SAFA)
   - Methods that compute a subset of velocity controls (Dynamic Window Approach (DWA), Velocity Obstacles Velocity Obstacles (VO), Optimal Reciprocal Collision Avoidance Optimal Reciprocal Collision Avoidance (ORCA)
4. Methods that use high level information (nearness diagram navigation)

Among these approaches, we are going to elaborate on the method of Artificial Potential Fields (APF) since it has inherent similarities to the extension of Interconnection and Damping Assignment Passivity-Based Control (IDA-PBC) presented in the previous chapters. More details on the method and its implementation in this work will follow in the subsequent sections.

5-1-2 Problem Definition

We begin with the problem definition based on [3]. Let $\mathcal{W}$ denote the *workspace* of N robots, $\mathcal{V}_i(\boldsymbol{q_i}) \subset \mathcal{W}$ be the volume occupied by agent i with configuration $\boldsymbol{q_i}$ and $\overline{\mathcal{V}}$ denote the volume occupied by all agents as a function of their configuration:

$$\bar{\mathcal{V}}(\bar{\boldsymbol{q}}) \triangleq \bigcup_{i=1}^{N} \mathcal{V}_i\left(\boldsymbol{q_i}\right) \tag{5-1}$$

The free configuration space of agent i from all the other agents $j \neq i$ is described as follows:

$$\mathcal{F}_i^{\text{agents}}(\bar{\boldsymbol{q}}) = \left\{ \boldsymbol{q_i} \middle| \bigcup_{i=1, j\neq i}^{N} \mathcal{V}_i\left(\boldsymbol{q_i}\right) \cap \mathcal{V}_j\left(\boldsymbol{q_j}\right) = \emptyset \right\} \tag{5-2}$$

In a similar manner we express the volume that is occupied by an obstacle i at time t as $\mathcal{O}_i(t) \subset \mathcal{W}$ and denote the total space occupied by N_{obs} obstacles by:

$$\bar{\mathcal{O}}(t) \triangleq \bigcup_{i=1}^{N_{obs}} \mathcal{O}_i(t) \tag{5-3}$$

The free configuration space of agent i from all the obstacles is described as follows:

$$\mathcal{F}_i^{\text{obs}}(t) = \left\{ \boldsymbol{q_i} | \mathcal{V}_i\left(\boldsymbol{q_i}\right) \cap \bar{\mathcal{O}}(t) = \emptyset \right\} \tag{5-4}$$

Thus, for each agent i the free configuration space from both other agents and obstacles can be defined as the set:

$$\mathcal{F}_i(\bar{\boldsymbol{q}}, t) = \mathcal{F}_i^{\text{obs}}(t) \cap \mathcal{F}_i^{\text{agents}}(\bar{\boldsymbol{q}}) \tag{5-5}$$

The control objectives described in (4-19)-(4-22) with the derived control law (4-38) then need to be achieved under the condition:

$$\boldsymbol{q_i} \in \mathcal{F}_i(\bar{\boldsymbol{q}}, t) \quad \forall i \in N, \quad \forall t \tag{5-6}$$

so that collision avoidance is achieved successfully.

5-2 The Artificial Potential Field Method

5-2-1 Basic Concept

The APF method [28] relies on the idea that the agent moves in a field of forces that result from an artificial potential field. This potential field is usually the sum of an *attractive field* that moves the agent towards the control objective and several *repulsive fields* that prevent the agent from colliding with other agents or obstacles in its workspace. We have already seen that in the distributed IDA-PBC framework we design a *desired potential* function $\bar{V}_c$ so that the agents achieve their cooperative tasks (cooperative potential). Thus, in our case the cooperative function is utilized as the attractive component of the APF method. Since $\bar{V}_c$ in (4-23) is free in $\boldsymbol{z}$ (as described in Section 4-1), it can be extended by incorporating a repulsive field as follows:

$$\bar{V}_c = \bar{V}_{att} + \bar{V}_{rep} \tag{5-7}$$

where $\bar{V}_{att}$ denotes the attractive potential such as the quadratic potential described in (4-23) and $\bar{V}_{rep}$ denotes the repulsive potential field used for collision avoidance. The attractive potential field is assumed to be of quadratic form as it is the case in (4-23). The repulsive potential filed should be designed so that it creates a potential barrier around the obstacles' surfaces and become negligible beyond them. In the classical potential field method [28], the repulsive potential field for each obstacle is described by the Force Inducing an Artificial Repulsion from the Surface (FIRAS) function described as:

$$V_{rep_i}\left(\mathcal{O}_i\right) = \begin{cases} \frac{1}{2}\eta\left(\frac{1}{\rho} - \frac{1}{\rho_0}\right)^2, & \text{if } \rho \leq \rho_0 \\ 0, & \text{if } \rho > \rho_0 \end{cases} \tag{5-8}$$

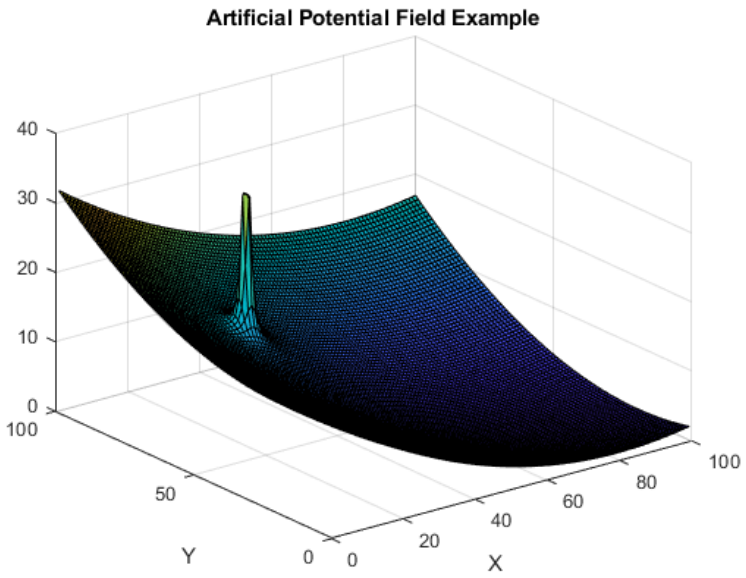


Figure 5-1: An example of the resulting potential field from the summation of an attractive quadratic field and the repulsive FIRAS Function

where $\eta \in \mathbb{R}$ denotes a scaling factor of the field, $\rho \in \mathbb{R}$ is the shortest distance to the obstacle and $\rho_0 \in \mathbb{R}$ is the radius of effect of the potential field. A simple representation of the function described in (5-8) is illustrated in Figure 5-1. The total repulsive field is simply the sum of all the repulsive fields from every obstacle:

$$\bar{V}_{rep} = \sum_{i=1}^{N_{obs}} V_{rep}^{i}\left(\mathcal{O}_i\right) \tag{5-9}$$

Thus, the total potential field in (5-7) can be designed as described here so that the control tasks (4-19) - (4-22) and the task of collision avoidance are integrated in the expression of $\bar{V}_c$ and solved by the same control law (4-38).

5-2-2 Limitations

The method of APF is generally easy to implement and as discussed in the previous section it seems particularly suitable to use in the context of IDA-PBC. However the method suffers from well-known limitations. A systematic criticism of the method is presented in [29] which includes the following [30]:

1. Trap situations due to local minima
2. No passage between closely spaced obstacles
3. Oscillations in the presence of obstacles
4. Oscillations in narrow passages

Moreover, another limitation presented in [30] with a possible solution is the problem of Goals Non-Reachable with Obstacles Nearby (GNRON). Among the aforementioned, the major problem is non-convexity of the total potential field: the interaction of the attractive

field and the repulsive fields often creates stationary points (either local minima or saddle points) which lead to failure.

With the issues discussed so far, it is clear that this method has a *local* perspective of the robot environment. This implies that it can probably be considered as an integral part of a higher level motion planner that is able to find simpler collision-free trajectories in complex environments and leave the task for fast, real time collision avoidance in the local environment to the APF method. For the purpose of local collision avoidance another favourable attribute of APF is that it can be extended to handle dynamic environments, i.e. environments in which the obstacles are in motion, something particularly useful for cooperative tasks in multi-agent systems as these agents will often follow intersecting trajectories. An extension of the APF with dynamic obstacles is presented in [31].

5-3 Proposed Repulsive Field

5-3-1 Design of the Repulsive Field

In this section we will design a repulsive field based on [31] in order to achieve inter-agent dynamic collision avoidance. As mentioned earlier, we will use this repulsive field to achieve collision avoidance in the local environment of the agent rather than to present a complete global motion planner. As stated in [31], in a highly dynamic environment where both the target and the obstacles are moving, the situations where the configuration of the obstacles and target keeps static are rare. To solve the problem of local minima in this case the simplest method is to keep the agents under the influence of the virtual forces as usual and wait for the obstacles or the target to change its motion.

We begin with the assumption that each agent knows the position, momenta and shape of the neighboring agents which are in the range of possible collision. In order to simplify the approach, we assume that the volume occupied by each agent is a sphere with its center and radius so that it encompasses the true volume of the agent. This assumption leads to a more conservative solution since some free space might be assumed as occupied. Nonetheless, according to [32] this is not limiting as approximations of the real objects can be realized with a number of spheres [33]. This approach greatly simplifies the collision avoidance task. Since the spheres are invariant of rotations, the scope of collision avoidance can be reduced to the subset of the transnational components (transnational displacements, velocities etc.).

Let $\boldsymbol{c_i} = \boldsymbol{c_i}(\boldsymbol{z_i})$ denote the center of each sphere encompassing agent i and ρ_{0_i} its radius. In many cases where the cooperative variables denote the position of an agent we might as well have $\boldsymbol{c_i} = \boldsymbol{z_i}$ otherwise this center can denote the coordinates of another part of the agent e.g. the center of the links of an agent-manipulator which we want them to be avoided by the rest of the agents. In this subsection, let also $\boldsymbol{p_i}$ denote the transnational component of the momenta of each agent, that is, the component that leads to the translational motion of the spheres.

The magnitude of the relative momenta between two agents is given by:

$$p_{ij} = \left(\boldsymbol{p_i} - \boldsymbol{p_j}\right)^\top \boldsymbol{n_{ij}} \tag{5-10}$$

where $\boldsymbol{n_{ij}}$ denotes the unit vector the points from the center of agent i to the center of agent j. If $p_{ij} \leq 0$ then the agents are moving away from each other and no avoidance motion is needed. If $p_{ij} > 0$ then the agents are moving towards each other and collision avoidance is needed. Consider the case where the agents move towards each other. Assume that for each agent a maximum translational force of magnitude F_{max_i} can be applied in order to reduce its momentum. Ignoring its peer's intentions for an avoiding maneuver, the distance that agent i will travel before p_{ij} reduces to zero is:

$$\rho_{m_{ij}} = \frac{p_{ij}^2}{2m_i F_{max_i}} \tag{5-11}$$

Let also $\rho_{ij} = \rho_{ij}(\boldsymbol{z_i}, \boldsymbol{z_j})$ denote the distance between the centers $\boldsymbol{c_i}$ and $\boldsymbol{c_j}$ of agents i, j and $\rho_{0_{ij}} = \rho_{0_i} + \rho_{0_j}$ the sum of the radius of each pair of agents. The repulsive field can then be defined as follows:

$$V_{rep_{ij}}(\boldsymbol{z_i}, \boldsymbol{z_j}, p_{ij}) = \begin{cases} 0 & \text{if } \rho_{ij} - \rho_{m_{ij}} \geq \rho_{0_{ij}} \text{ or } p_{ij} \leq 0 \\ \eta_{ij}\left(\frac{1}{\rho_{ij}-\rho_{m_{ij}}} - \frac{1}{\rho_{0_{ij}}+\rho_{m_{ij}}}\right)^2 & \text{if } 0 < \rho_{ij} - \rho_{m_{ij}} < \rho_{0_{ij}} \text{ and } p_{ij} > 0 \\ \text{not defined} & \text{if } p_{ij} > 0 \text{ and } \rho_{ij} < \rho_{m_{ij}} \end{cases} \tag{5-12}$$

In case that $\rho_{ij} - \rho_{m_{ij}} \geq \rho_{0_{ij}}$ or $p_{ij} \leq 0$ the two agents are either far enough or they are moving away from each other thus collision avoidance is not needed. In case that $0 < \rho_{ij} - \rho_{m_{ij}} < \rho_{0_{ij}}$ and $p_{ij} > 0$ the agents are moving towards each other and they are at collision proximity meaning that collision avoidance is necessary. In the last case where we have $p_{ij} > 0$ and $\rho_{ij} < \rho_{m_{ij}}$ collision avoidance cannot be avoided since the agents move towards each other and the current distance ρ_{ij} is less than the minimum distance that the agent will travel with full breaking power. In that case, we might as well have the repulsive potential activated in order to at least reduce the severity of the unavoidable collision. Notice also that in the case that the repulsive field is activated we have:

$$\lim_{\rho_{ij} \to \rho_{m_{ij}}} \eta_{ij}\left(\frac{1}{\rho_{ij} - \rho_{m_{ij}}} - \frac{1}{\rho_{0_{ij}} + \rho_{m_{ij}}}\right)^2 \to \infty \tag{5-13}$$

For an easier implementation, we introduce here the switching function $sw_{rep_{ij}}$ defined as follows:

$$sw_{rep_{ij}} = \begin{cases} 0 & \text{if } \rho_{ij} - \rho_{m_{ij}} \geq \rho_{0_{ij}} \text{ or } p_{ij} \leq 0 \\ 1 & \text{otherwise} \end{cases} \tag{5-14}$$

in order to activate or deactivate the repulsive fields around agent i, with $sw_{rep_{ii}} = 0 \quad \forall i \in N$. The total repulsive filed for agent i is then given as:

$$\bar{V}_{rep} = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} sw_{rep_{ii}}\eta_{ij}\left(\frac{1}{\rho_{ij}(\boldsymbol{z_i}, \boldsymbol{z_j}) - \rho_{m_{ij}}(p_{ij})} - \frac{1}{\rho_{0_{ij}} + \rho_{m_{ij}}(p_{ij})}\right)^2 \tag{5-15}$$

The total repulsive filed is then summed to the previously defined cooperative function (4-23) which yields the new cooperative potential:

$$\begin{aligned} \bar{V}_c = &\frac{1}{4}\sum_{i=1}^{N}\sum_{j=1}^{N}\left(\boldsymbol{z_i} - \boldsymbol{z_j} + \boldsymbol{z_{ij}^*}\right)^T \boldsymbol{\mathcal{A}_{ij}}\left(\boldsymbol{z_i} - \boldsymbol{z_j} + \boldsymbol{z_{ij}^*}\right) + \frac{1}{2}\sum_{i=1}^{N}\left(\boldsymbol{z_i} - \boldsymbol{z_i^*}\right)^T \boldsymbol{\mathcal{B}_i}\left(\boldsymbol{z_i} - \boldsymbol{z_i^*}\right) \\ &+ \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} sw_{rep_{ii}}\eta_{ij}\left(\frac{1}{\rho_{ij}(\boldsymbol{z_i}, \boldsymbol{z_j}) - \rho_{m_{ij}}(p_{ij})} - \frac{1}{\rho_{0_{ij}} + \rho_{m_{ij}}(p_{ij})}\right)^2 \end{aligned} \tag{5-16}$$

The control law remains the same as expressed in (4-38). The cooperative component will then be responsible for collision avoidance as it is the part of the control law which is responsible to dictate the motion in the (cooperative) workspace of each agent. In Figure 5-2 a schematic

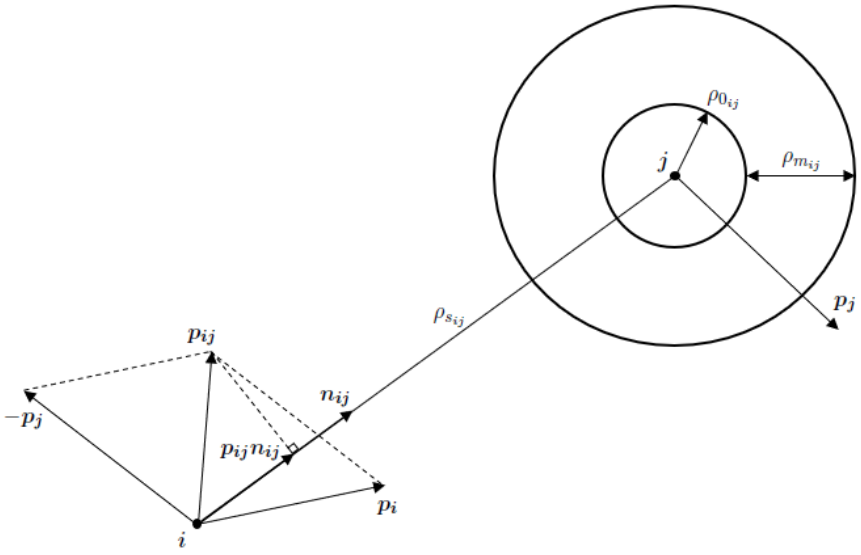


Figure 5-2: A schematic representation of the proposed collision avoidance approach

representation of the aforementioned is presented. Here we highlight the main differences with the approach we were inspired from in [31]. First of all, since we restricted our scope to inter-agent collision avoidance, we use the generalized momenta instead of the velocities of the agents. Although the idea is essentially the same, the momentum contains more information than the velocity as it describes also the inertia and thus the "stubbornness" of each agent movement. This acts as a weighting factor to the mutual collision avoidance maneuvers of the agents: the agent with more momentum will deviate from its course less compared to the agent with less momentum which will be easier to follow an avoiding maneuver. Another important difference is the utilization of the distance of deceleration $\rho_{m_{ij}}$. In our case, this distance is used as an increment to the summed radius of the two agents virtually enlarging the volume of the "true" volume of the two agents encompassed by the virtual spheres so that deceleration stars early enough in order to avoid collision. This increment is radius is illustrated in Figure 5-2 where agent i is illustrated as a single point in 2-D space and perceives agent j as an obstacle incremented by its own size plus the distance $\rho_{m_{ij}}$. Last but not least, notice that the repulsive potential is a function of the generalized momenta of the agents since $p_{ij} = p_{ij}(\boldsymbol{p_i}, \boldsymbol{p_j})$. In the original approach [31], the repulsive force is the gradient of the repulsive potential with respect to both the positions and velocities of the agents. In our approach, since we want to integrate the repulsive potential to the existing distributed control law, the repulsive force results as the gradient of the repulsive potential with respect to the cooperative variables (i.e. coordinates) only.

5-3-2 Connection to this Work and Implementation

The key tool to handle nonholonomic systems in this work, is the freedom to manipulate the constraint-free variables so that stabilization in the constrained space $\mathcal{S}$ is feasible. A repulsive field like the one designed in the previous subsection will lead to a different evolution of the unconstrained variables and the stabilization task will probably fail. Another key insight is that the repulsive field is usually designed in task-space physical coordinates. These coordinates are most of the times the constrained variables $\boldsymbol{s}$ (e.g. the $xy-$ position of a differential robot on a plane). This motivates the design of the repulsive field on the constrained space so that it does not interfere with the unconstrained variables $\boldsymbol{r}$. The repulsive field can be then expressed as:

$$\bar{V}_{rep} = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} sw_{rep_{ii}}\eta_{ij}\left(\frac{1}{\rho_{ij}(\boldsymbol{z_{s_i}}, \boldsymbol{z_{s_j}})} - \frac{1}{\rho_{0_{ij}} + \rho_{m_{ij}}(p_{ij})}\right)^2 \tag{5-17}$$

while ρ_{ij}, $\rho_{0_{ij}}$, and $\rho_{m_{ij}}$ regard the constrained variables $\boldsymbol{s}$. A repulsive field on the constrained space $\mathcal{S}$ will act on the "attracting" vector $\boldsymbol{v_s}$ changing its magnitude and direction. The resulting $\boldsymbol{v_s}$ vector will be affected by the (local) repulsive field in a way to drive the system far from the source of repulsion. The change in magnitude means some deceleration in case the repulsive field is approached head on while the change in direction means that the system wants to follow an evasive maneuver. Recall that the control action for the unconstrained variables $\boldsymbol{r}$ is characterized by the vector:

$$\boldsymbol{v_\omega} = \boldsymbol{A}_s^\top \boldsymbol{v_s}$$

This vector, as already discussed, drives the unconstrained variables $\boldsymbol{r}$ so that the system "aligns" to the attractor (represented by $\boldsymbol{v_s}$) in the constrained space $\mathcal{S}$. Hence, a change of $\boldsymbol{v_s}$ due to the repulsive field will be followed by an appropriate change of vector $\boldsymbol{v_\omega}$ so that the motion can continue as desired. Nevertheless, the problem of local minima still remains a major one and this approach still remains of local interest for fast and dynamic collision avoidance action rather than a complete motion planning approach for nonholonomic systems.

We provide here a practical representation of the potential energy shaping with some tuning parameters as used in the simulations of Chapter 6.

$$\begin{aligned}
\bar{V}_c = \\
&k_s\left[\frac{1}{4}\sum_{i=1}^{N}\sum_{j=1}^{N}\left(\boldsymbol{z_{s_i}} - \boldsymbol{z_{s_j}} + \boldsymbol{z^*_{s_{ij}}}\right)^T \boldsymbol{\mathcal{A}_{ij}}\left(\boldsymbol{z_{s_i}} - \boldsymbol{z_{s_j}} + \boldsymbol{z^*_{s_{ij}}}\right) + \frac{1}{2}\sum_{i=1}^{N}\left(\boldsymbol{z_{s_i}} - \boldsymbol{z^*_{s_i}}\right)^T \boldsymbol{\mathcal{B}_i}\left(\boldsymbol{z_{s_i}} - \boldsymbol{z^*_{s_i}}\right)\right] + \\
&k_r\left[\frac{1}{4}\sum_{i=1}^{N}\sum_{j=1}^{N} sw_{h_i}\left(\boldsymbol{z_{r_i}} - \boldsymbol{z_{r_j}} + \boldsymbol{z^*_{r_{ij}}}\right)^T \boldsymbol{\mathcal{A}_{ij}}\left(\boldsymbol{z_{r_i}} - \boldsymbol{z_{r_j}} + \boldsymbol{z^*_{r_{ij}}}\right) + \frac{1}{2}\sum_{i=1}^{N} sw_{h_i}\left(\boldsymbol{z_{r_i}} - \boldsymbol{z^*_{r_i}}\right)^T \boldsymbol{\mathcal{B}_i}\left(\boldsymbol{z_{r_i}} - \boldsymbol{z^*_{r_i}}\right)\right] + \\
&\eta\left[\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} sw_{rep_{ii}}\left(\frac{1}{\rho_{ij}(\boldsymbol{z_i}, \boldsymbol{z_j}) - \rho_{m_{ij}}(p_{ij})} - \frac{1}{\rho_{0_{ij}} + \rho_{m_{ij}}(p_{ij})}\right)^2\right]
\end{aligned} \tag{5-18}$$

The local potential for the of each agent can be expressed as:

$$V_{l_i}(\boldsymbol{q_i}) = V_{ls_i}(\boldsymbol{q_i}) + (1 - sw_{h_i})k_r V_{lr_i}(\boldsymbol{q_i}) \tag{5-19}$$

with

$$V_{lr_i}(\boldsymbol{q_i}) = \frac{\partial^\top \bar{V}_c}{\partial \boldsymbol{s_i}} \boldsymbol{A_s}(\boldsymbol{r_i}) \boldsymbol{A_s}(\boldsymbol{r_i})^\top \frac{\partial \bar{V}_c}{\partial \boldsymbol{s_i}} \tag{5-20}$$

where k_s, k_r and η are tuning parameters for the cooperative and local potential setting the intensity of the attractive forces in the constrained and unconstrained space as well as the repulsive forces in the constrained space. The switches sw_{h_i} and $sw_{rep_{ij}}$ are designed for each agent as descibed in the previous sections.

Chapter 6

Case Studies

In this chapter we aim to illustrate the applicability and practicality of the results derived throughout this work via simulations. The simulations were carried out in MATLAB and SIMULINK. In the MATLAB environment, we first check the user-defined elements of each system and the elements of the network so that they satisfy necessary properties (agreeable dimensions, symmetry, positive definiteness etc.). Then the systems' dynamics are generated automatically based on user defined information and automatically saved as MATLAB functions. The SIMULINK file, having access to the systems' dynamics, is used as a solver for the Ordinary Differential Equations (ODEs). We have set an automatic solver selection with a variable-step as no in depth study was realized in order to choose a solver. Nonetheless, with these default settings the problems were solved without failure. We begin this chapter by briefly describing the systems used in simulations. Subsequently, four different simulation scenarios elucidate different aspects of the theoretical results.

6-1 Systems Introduction

One of the main purposes of this work is to highlight the practicality of distributed IDA-PBC for nonholonomic systems. Practical applications that are favourable to be addressed with the approach developed in this work include but are not limited to collaborative transportation, exploration of unknown or dangerous terrains, large scale sensing and area monitoring. Such *high-level* tasks can be decomposed into different, simpler *low-level* cooperative group objectives known as consensus, synchronization, or formation problems which are mathematically similar across the wide spectrum of the previously mentioned applications. An example in collaborative transportation could be that of a group of robotic arms that collaboratively grasps objects from a wheeled mobile robot. In area monitoring, we may require the deployment of different wheeled mobile robots that need to reach a specified formation in the area of interest so that sufficient area coverage is achieved. In the following simulations we show how we can fulfill these low-level tasks.

6-1-1 The 3-DoF Manipulator

One particularly useful class of robotic agents for various tasks in the realms of logistics and manufacturing are robotic manipulators. These robots can perform tasks such as picking and placing objects, assembling components or participating in other manufacturing processes. A drawing of a simple 3-DoF manipulator is presented in Figure 6-1. This system has $n = 3$

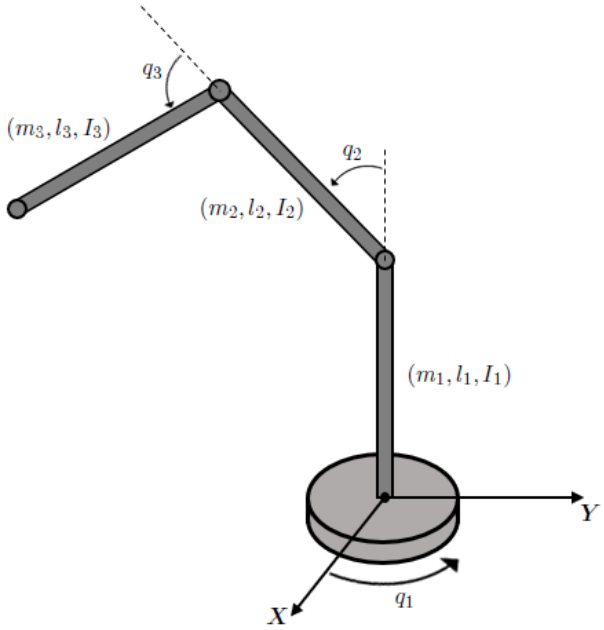


Figure 6-1: A schematic representation of a 3-DoF manipulator

degrees of freedom $\boldsymbol{q} = \begin{pmatrix} q_1 & q_2 & q_3 \end{pmatrix}^\top$, with each component representing a joint angle. The system is holonomic with $k = 0$ constraints and fully actuated with $m = 3$ control inputs. The control inputs are torques that are directly applied on the joint angles. The end-effector of the is robotic arm can move in the subset of three-dimensional space that the length of the robot links allow it to. The mass matrix of this system according to [10] is given as:

$$\boldsymbol{M} = \begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{bmatrix} \tag{6-1}$$

where:

$$\begin{aligned}
M_{11} &= I_{y_2}\sin^2 q_2 + I_{y_3}\sin^2(q_2+q_3) + I_{z_1} + I_{z_2}\cos^2 q_2 + I_{z_3}\cos^2(q_2+q_3) \\
&\quad + m_2\frac{l_2^2}{4}\cos^2 q_2 + m_3\left(l_1\cos(q_2) + \frac{l_3^2}{4}\cos(q_2+q_3)\right)^2 \\
M_{12} &= 0 \\
M_{13} &= 0 \\
M_{21} &= 0 \\
M_{22} &= I_{x_2} + I_{x_3} + m_3 l_1^2 + m_2\frac{l_2^2}{4} + m_3\frac{l_3^2}{4} + 2m_3 l_1\frac{l_3}{2}\cos q_3 \\
M_{23} &= I_{x_3} + m_3\frac{l_3^2}{4} + m_3 l_1\frac{l_3}{2}\cos q_3 \\
M_{31} &= 0 \\
M_{32} &= I_{x_3} + m_3\frac{l_3^2}{4} + m_3 l_1\frac{l_3}{2}\cos q_3 \\
M_{33} &= I_{x_3} + m_3\frac{l_3^2}{4}
\end{aligned} \tag{6-2}$$

With where m_i is the mass of each link and I_{x_i}, I_{y_i}, and I_{z_i} are the moments of inertia about the $x-$, $y-$, and $z-$axes of the i^{th} link frame. Note that several of the moments of inertia of the different links do not appear in this expression. This is because the limited degrees of freedom of the manipulator do not allow arbitrary rotations of each joint around each axis. The potential energy of the system is the sum of the potential energy of each link:

$$\begin{aligned}
V(\boldsymbol{q}) &= m_1 g r_{1z} + m_2 g r_{2z} + m_3 g r_{3z} \\
&= m_1 g\frac{l_1}{2} + m_2 g\left(l_1 + \frac{l_2}{2}\cos q_2\right) + m_3 g\left(l_1 + l_2\cos q_2 + \frac{l_3}{2}\cos(q_2+q_3)\right)
\end{aligned} \tag{6-3}$$

The system is fully-actuated so the input matrix is simply the identity matrix $\boldsymbol{F} = \boldsymbol{I_3}$ and unconstrained thus $\boldsymbol{A} = \boldsymbol{0_3}$. With this set of information the system is fully defined. The system is holonomic and fully-actuated meaning simplifying its control.

6-1-2 The Differential Robot

The differential robot is a nonholonomic system that is frequently met both in academia and the industry. These mobile robots are very compact and agile and virtually all consumer robots on the market today use differential steering primarily for its low cost and simplicity. The movement of this mobile robot is based on two separately driven wheels placed on either side of the robot body. It can change its direction by varying the relative rate of rotation of its wheels and hence does not require an additional steering motion. In practice, additional caster wheels may be added to balance the robot. A drawing of a differential robot is presented in Figure 6-2. This system has $n = 3$ degrees of freedom $\boldsymbol{q} = \begin{pmatrix} q_1 & q_2 & q_3 \end{pmatrix}^\top$, $k = 1$ constraints and $m = 2$ control inputs, the torques on each one of the two wheels. The rotation of each wheel can also be seen as a degree of freedom but usually these are omitted as we are interested

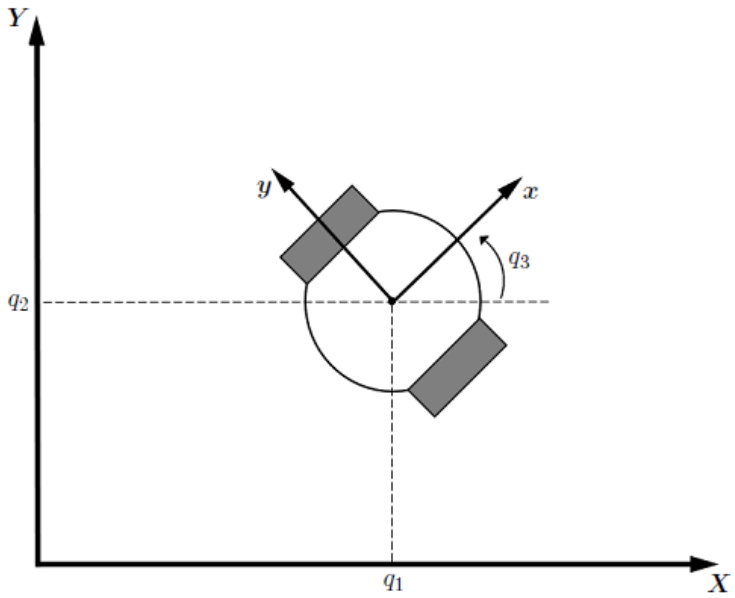


Figure 6-2: A schematic representation of a differential robot

in the planar motion of the robot.The mass matrix of this system is given as:

$$\boldsymbol{M} = \begin{bmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & I \end{bmatrix} \tag{6-4}$$

where m is the mass of the robot and I is the moment of inertia with respect to its planar rotation. Since the system performs planar motion, its potential energy is zero:

$$V(\boldsymbol{q}) = 0 \tag{6-5}$$

The system is actuators are two motors, each one applying a torque at each wheel. This results in a translational force that acts on the longitudinal axis and a torque that rotates the robot. The input matrix is then:

$$\boldsymbol{F}(\boldsymbol{q}) = \begin{bmatrix} \frac{\cos q_3}{r} & \frac{\cos q_3}{r} \\ \frac{\sin q_3}{r} & \frac{\sin q_3}{r} \\ \frac{d}{2} & -\frac{d}{2} \end{bmatrix} \tag{6-6}$$

Where r is the wheel radius and d the distance between the two wheel centers. The system is subjected to the nonholonomic constraint:

$$\sin q_3 \dot{q}_1 - \cos q_3 \dot{q}_2 = 0 \Rightarrow \begin{pmatrix} \sin q_3 & -\cos q_3 & 0 \end{pmatrix} \dot{\boldsymbol{q}} = 0 \Rightarrow$$
$$\boldsymbol{A}^\top(\boldsymbol{q}) = \begin{pmatrix} \sin q_3 & -\cos q_3 & 0 \end{pmatrix}$$

Both the constraint matrix and the mass matrix satisfy the conditions expressed in Section 3-2-3 so that it can be controlled with the derived control laws.

6-2 Scenario 1

Before jumping into the previously mentioned cooperative tasks we first demonstrate here a comparison between our approach and the Passivity Based Switching Control (PBSC) that was introduced in [7]. We aim to show that with our approach we can significantly reduce the time to achieve stabilization in the constrained space under the exact same conditions. For illustration we use the differential robot as it is one of the systems also used in [7]. Note that in [7] a Passivity Based Time-Varying Control (PBVC) is also derived but we will not illustrate these results here. Moreover, here we implement the solution proposed in [7] but not the exact simulation scenarios. Nevertheless, the response of the simulated system is similar to the original one.

The differential robot's constrained coordinates are $\boldsymbol{s} = \begin{pmatrix} q_1 & q_2 \end{pmatrix}^\top$ while the unconstrained one is $r = q_3$ Assume that the robot starts with initial position $\boldsymbol{q_0} = \begin{pmatrix} 1 & 1 & 0 \end{pmatrix}^\top$ and needs to reach the point $\boldsymbol{s^*} = \begin{pmatrix} 4 & 4 \end{pmatrix}^\top$ with an arbitrary orientation $r = q_3$.

According to [7], the desired potential can be designed as:

$$V_1 = \underbrace{\frac{1}{2} k_s (\boldsymbol{s} - \boldsymbol{s^*})^\top \boldsymbol{Q_s} (\boldsymbol{s} - \boldsymbol{s^*})}_{V_s} + \underbrace{\frac{1}{2} k_r (r - r_\sigma)^\top (r - r_\sigma)}_{V_{r_\sigma}} \tag{6-7}$$

where k_s, k_r and $\boldsymbol{Q_s}$ are tuning parameters and r_σ is a constant at which we want r to be stabilized and σ is to embed a switching sequence. Then $\boldsymbol{s}$ will converge to the switching manifold $\mathcal{G}_\sigma \subset \mathcal{S}$:

$$\mathcal{G}_\sigma = \left\{ \boldsymbol{s} \in \mathcal{S} \middle| \mathcal{D}_s^\top (r_\sigma) \frac{\partial V_s}{\partial \boldsymbol{s}} = \boldsymbol{0} \right\} \tag{6-8}$$

With two switching manifolds ($\sigma = 1, 2$) and $r_1 = -\pi/2$, $r_2 = 0$ then the idea is to stabilize r at each of one of the aforementioned manifolds sequentially triggering the switch when stabilization has been practically achieved (when $\boldsymbol{s}$ is close enough to $\mathcal{G}_\sigma$ and small enough velocity). This is by no means an extensive description of the method as we only present what is necessary for the comparison that follows. The interested reader can find more details on this approach along with validating results in [7].

According to what we proposed in Chapter 3 the desired potential can be designed as:

$$V_2 = \underbrace{\frac{1}{2} k_s (\boldsymbol{s} - \boldsymbol{s^*})^\top \boldsymbol{Q_s} (\boldsymbol{s} - \boldsymbol{s^*})}_{V_s} + \underbrace{\frac{1}{2} k_r \left(\frac{\boldsymbol{s} - \boldsymbol{s^*}}{\|\boldsymbol{s} - \boldsymbol{s^*}\|} \right)^\top \boldsymbol{A_s}(r) \boldsymbol{A_s}(r)^\top \left(\frac{\boldsymbol{s} - \boldsymbol{s^*}}{\|\boldsymbol{s} - \boldsymbol{s^*}\|} \right)}_{V_r} \tag{6-9}$$

where $\boldsymbol{A_s}^\top = \begin{pmatrix} \sin q_3 & -\cos q_3 \end{pmatrix}$ according to Section 3-2-3 and Section 6-1-2. The system has mass matrix $\boldsymbol{M} = \boldsymbol{I_3}$ for simplicity and $V = 0$ as it performs a planar motion. The values of the control parameters are presented in Table 6-1. We denote as *system 1* the differential robot controlled as in [7] (existing solution) while *system 2* denotes the differential robot controlled by the proposed solution in this work. Applying the aforementioned potentials and setting the control parameters to the values shown in Table 6-1 we simulate the differential robot under to different subjected to the two different control approaches.

The results of this simulation scenario are illustrated in the following figures along with some brief discussion. In Figure 6-3 the trajectories of the same system for the two different

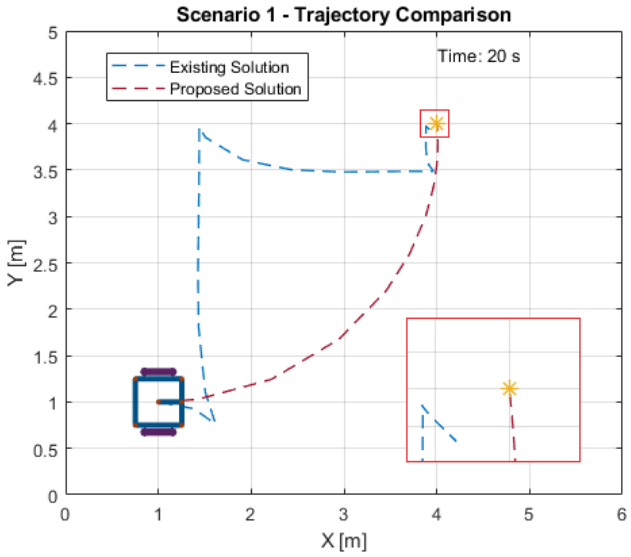
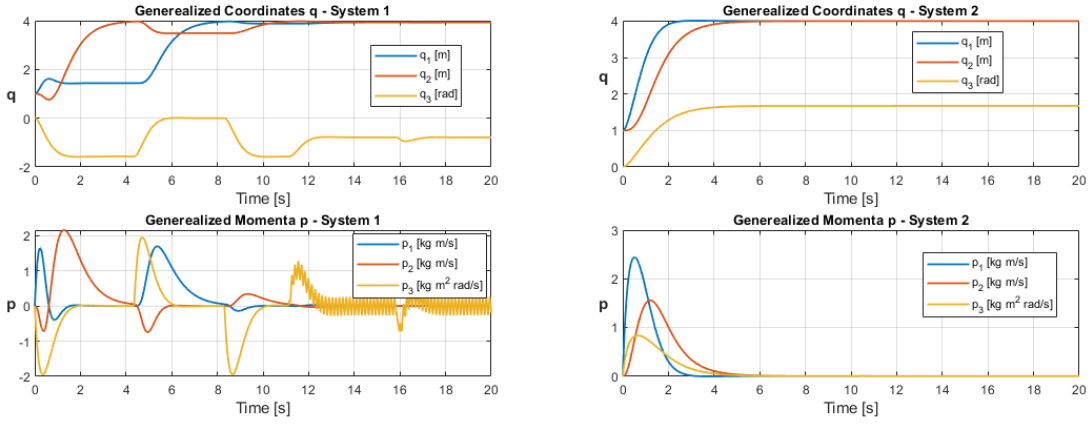


Figure 6-3: Scenario 1: A comparison of the two trajectories

control approaches is illustrated. We see the differential robot in its initial state while its final state is hidden so that the trajectories near the goal are clearly visible. First of all, we observe that with our approach we can directly converge to $\boldsymbol{s}^*$ without having to switch between manifolds. Moreover, we can observe that with the existing solution, since a practical switching criterion needs to be implemented, the system will be stabilized at an offset from the desired configuration (here we have set $s^d = 0.05m$ meaning that every time $\boldsymbol{s}$ is closer than s^d to each manifold the switching is triggered). Thus, the constrained variable $\boldsymbol{s}$ can only

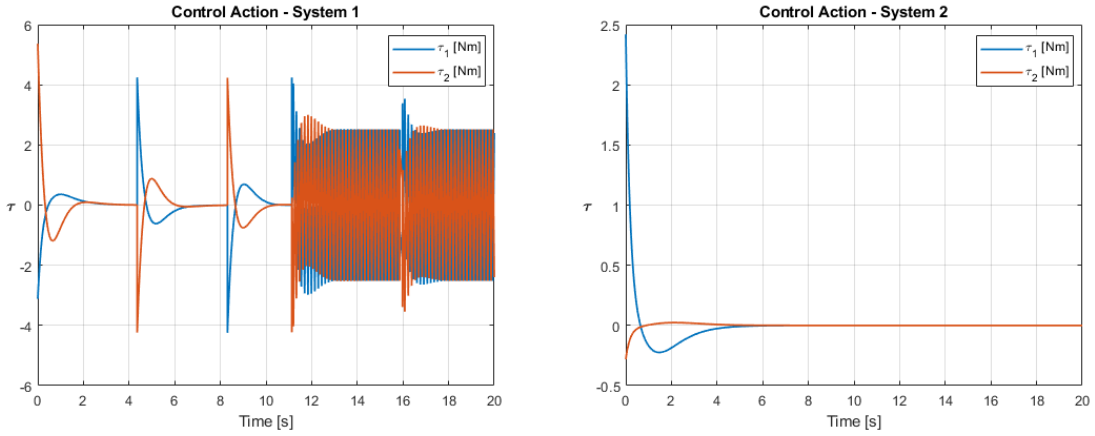


(a) Existing solution **(b)** Proposed solution

Figure 6-4: Scenario 1: The state evolution of the differential robot

reach $\boldsymbol{s}^*$ up to some tolerance. This of course can be set arbitrarily small though convergence time will probably increase considerably.

In Figure 6-4 we can see the evolution of the state of each system. It is clear that under the same conditions we can achieve stabilization of the constrained variables much faster (settling time at around $t = 5s$) compared to the existing solution. Moreover, due to the fact that we do not require any switching, no oscillations are observed in the generalized momenta.



(a) Existing solution **(b)** Proposed solution

Figure 6-5: Scenario 1: The control inputs on the wheels of the differential robot

Regarding the control inputs on the wheels, it is clear that in our approach the control action required is much less and without any odd switching when the system approaches the goal (Figure 6-5). Before we end this section we emphasize here that the solution from

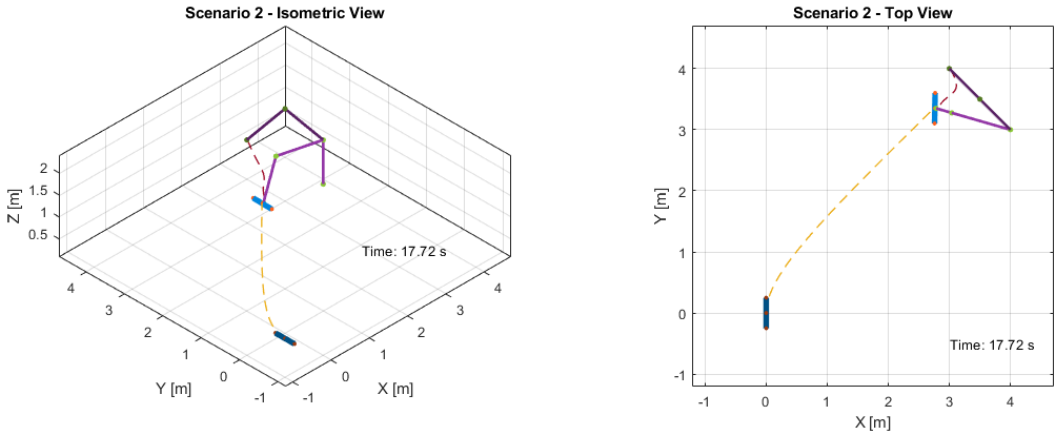
k_s	$\boldsymbol{Q_s}$	k_r	$\boldsymbol{K_v}$
5	$\boldsymbol{I_2}$	9	$5 \cdot \boldsymbol{I_2}$

Table 6-1: Control parameters for Scenario 1

[7] implemented here is similar up to the method and not the exact parameters. Perhaps different values for r_σ could yield a better transient response. Nevertheless, the values used are according to what is proposed in their work and also the resulted trajectories are similar. Note that a PBVC approach is also implemented in their work that results to different trajectories. It is claimed in [7] that the PBVC approach achieves faster stabilization in the constrained space. Even in this approach the trajectories are still not directly leading to the goal as they rather have a spiral-like shape. Note also that our control law in this example is smooth. This does not contradict with Brockett's necessary condition as full-state stabilization is not achieved. Achieving full-state stabilization requires switching from the potential function (6-9) to a another smooth potential that can stabilize the unconstrained variable r subsequently. In order to highlight the difference in the stabilization approach of the constrained variables $\boldsymbol{s} = \begin{pmatrix} q_1 & q_2 \end{pmatrix}^\top$ we ignore the stabilization of the unconstrained variable q_3. Full-state stabilization is illustrated in Section 6-3 (Scenario 2).

6-3 Scenario 2

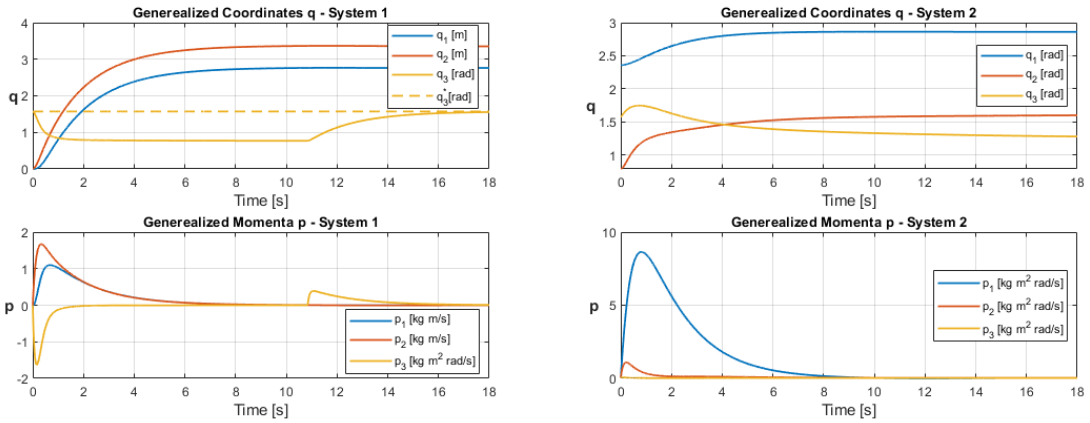
In this scenario we want the nonholonomic knife edge system (description in the examples of Chapter 2 and Chapter 3) and the end-effector of the 3-DoF manipulator to reach consensus. Specifically we want to show that it is possible to address the unconstrained variable of the knife edge system as a local variable. We can first use it to reach consensus and then stabilize it locally at a desired set point thus not needing to exchange it via the network.



(a) Trajectories in Scenario 2 - Isometric View **(b)** Trajectories in Scenario 2 - Top View

Figure 6-6: Scenario 2: The trajectories of the knife edge system and the 3-DoF manipulator

The cooperative variables of the knife edge system is the xyz coordinates of its center while the for the 3-DoF manipulator these are the coordinates of its end-effector. Notice that consensus is reached approximately at time $t = 11s$ and then the control of the knife edge switches to achieve stabilization of the local variable q_3 (orientation) of the knife edge system. Since the unconstrained dynamics have been successfully stabilized, the task of stabilizing the unconstrained variable (orientation) is relatively easy between two stable configurations.



(a) The state evolution of the knife edge system **(b)** The state evolution of the 3-DoF manipulator

Figure 6-7: Scenario 2: The state evolution for the two systems respectively

The transient response is smooth even when the control law switches. However, we observe that since we are stabilizing the variables sequentially, more time is needed for full-state stabilization. This can be seen in Figure 6-8 where consensus is achieved at roughly time $t = 11s$ while full-state stabilization needs additional four seconds.

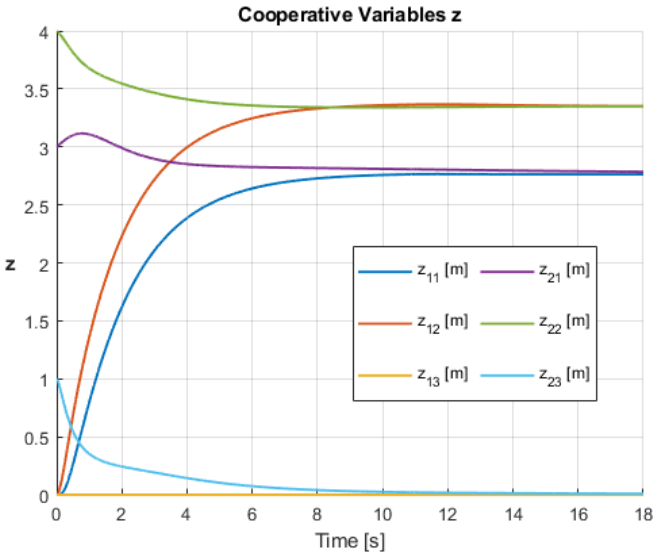


Figure 6-8: Scenario 2: The evolution of the cooperative variables

Overall, we can see that in this simple scenario the consensus problem successfully achieved while the unconstrained variable of the nonholonomic system can be controlled independently leading to full-state stabilization. In Section 6-4 (Scenario 3) we will also demonstrate how this variable can be used as a cooperative variable as well. The control parameters for this scenario can be are illustrated in Table 6-2.

k_s	k_r	η	$\boldsymbol{K_{v_1}}$	$\boldsymbol{K_{v_2}}$
3	2.5	30	$diag(9, 15)$	$diag(90, 16, 16)$

Table 6-2: Control parameters for Scenario 2

6-4 Scenario 3

In this scenario we want to demonstrate how a group of differential robots can reach a desired formation. The robots need to reach this formation with one of the agents being the group leader.

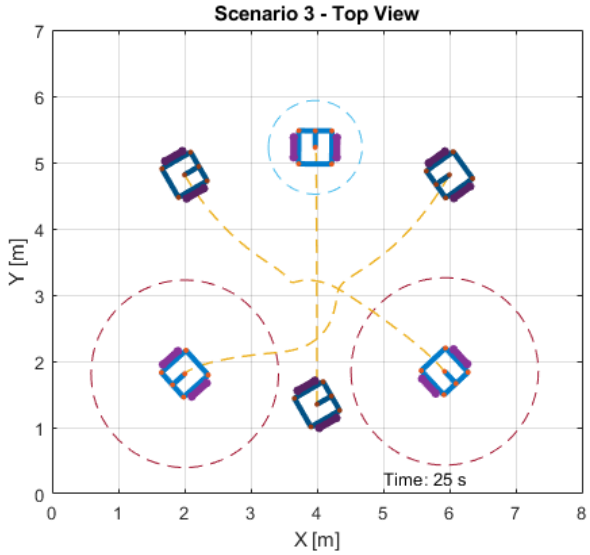
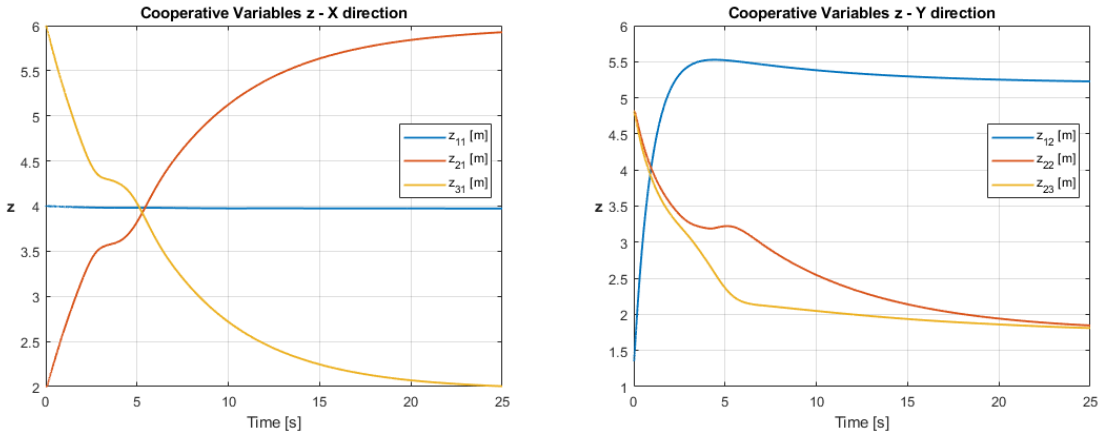


Figure 6-9: Scenario 3: The trajectories followed by the differential robots

Moreover, the simple collision avoidance protocol is implemented as well to show how these agents can perform avoiding maneuvers while reaching for their goal configurations. In contrast to Scenario 2, here the unconstrained variables act at first as a local variables for stabilization in the constraint space of each agent and subsequently as a cooperative variables so that formation can be achieved in the full state-space (relative positions and orientation).



(a) The state evolution of the knife edge system **(b)** The state evolution of the 3-DoF manipulator

Figure 6-10: Scenario 3: The evolution of the cooperative variables (x and y coordinates)

These trajectories are illustrated in Figure 6-9 with the initials positions of the agents in a darker shade than the final configurations. The agents start from an equilateral triangle formation and need to form an equilateral triangle again, mirrored to the first one with respect to the horizontal axis passing through the triangle's center of gravity. When they reach that formation, they also need to be oriented symmetrically and point to a direction away from the triangle ($\frac{3\pi}{4}$ relative angle to each other).

Notice that the leader robot which starts at the very bottom reaches its target configuration without encountering the other agents. The trajectories of other two agents on the other hand intersect and the collision avoidance maneuver is clear in the same figure. In Figure

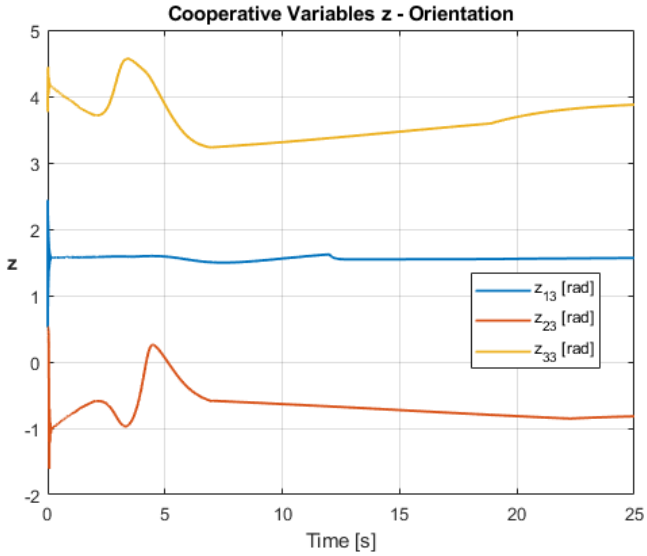


Figure 6-11: Scenario 3: The evolution of the cooperative variables (orientation) of the agents

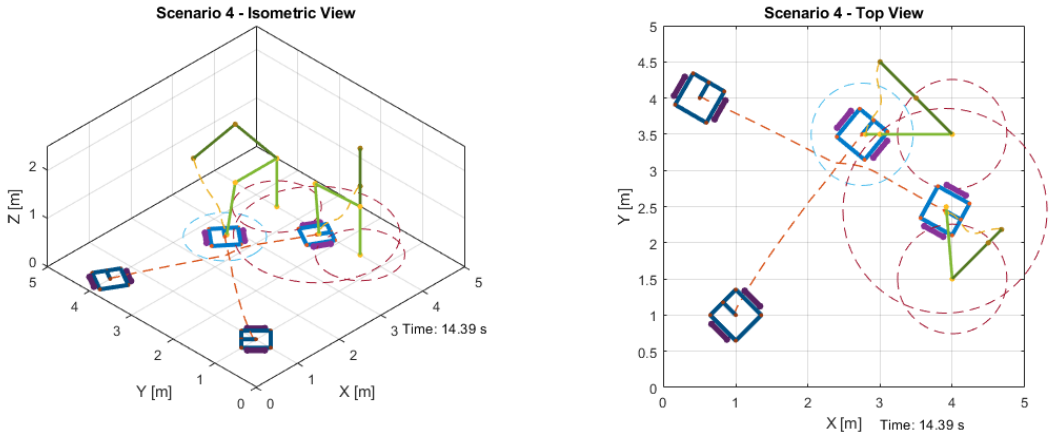
6-10 we can see the x and y coordinates of the agents. In the x direction, the leader does not move while the other agents reach a relative position of $x_{12} = 4m$ which is the original side of the triangle. In the y direction, the leader reaches the target position while the other two agents reach consensus at the desired distance from the leader. Lastly, in Figure 6-11 we can observe how the robots reach a formation in the unconstrained variables (orientation) as well. The switching is visible at approximately time $t = 18s$ for the orientation of agent 3. Notice also that between times $t = 18s$ and $t = 18s$, the orientations of agents 2 and 3 change so that an avoiding maneuver can be achieved thus proving that the collision avoidance protocol in Chapter 5 can work well with the stabilization results derived in Chapter 3. The control parameters for this scenario can been in Table 6-3.

k_s	k_r	η	$\boldsymbol{K_{v_1}}$	$\boldsymbol{K_{v_2}}$	$\boldsymbol{K_{v_3}}$
5	10	37	$diag(60, 60)$	$diag(60, 60)$	$diag(60, 60)$

Table 6-3: Control parameters for Scenario 3

6-5 Scenario 4

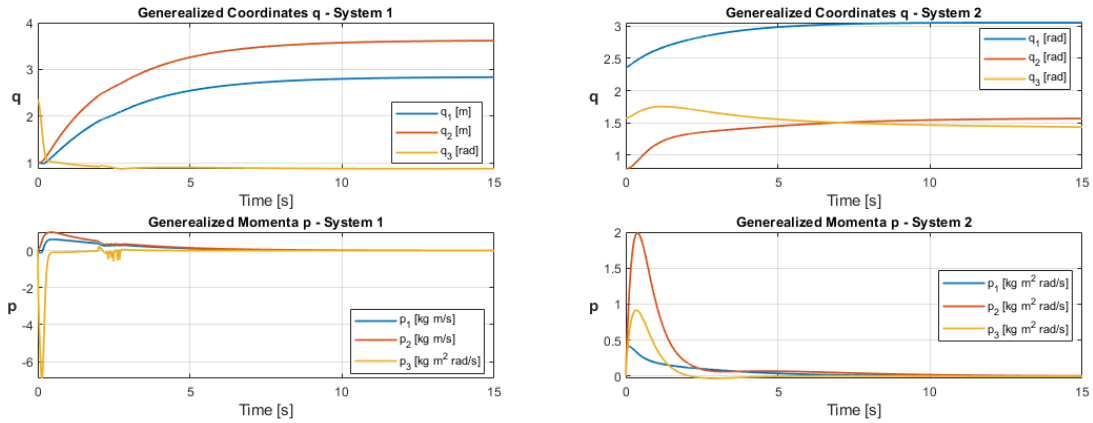
Finally, we present a scenario that simulates a practical application in logistics. We assume that a couple of manipulators are responsible to grasp packages which are transported by differential robots and lie on top of the latter. Each differential robot needs to approach the vicinity of its counterpart manipulator while avoiding collision with the other mobile robot and the base of the manipulators. Likewise the end-effector of each manipulator needs to approach the respective differential robot.



(a) Trajectories in Scenario 4 - Isometric View **(b)** Trajectories in Scenario 4 - Top View

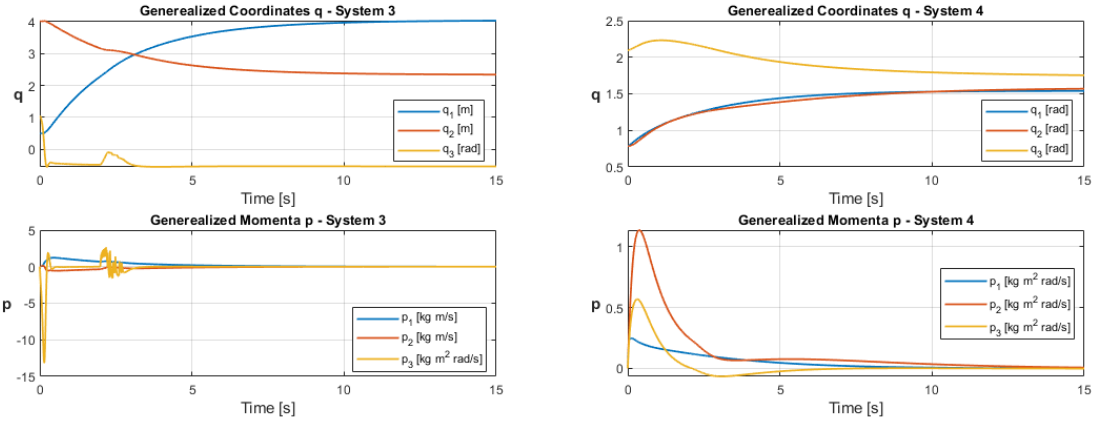
Figure 6-12: Scenario 4: The trajectories of the differential robots and manipulators

The light blue dashed circle seen in Figure 6-12 denotes a desired clearance around differential robot 1 (starting in xy position $(1, 1)$). The red dashed circle denote the effective radius of the repulsive fields caused by the other agents. In Figure 6-12b the avoiding maneuver of differential robot 2 (starting in xy position $(0.5, 4)$) can also be seen.



(a) The state evolution of differential robot #1 **(b)** The state evolution of manipulator #1

Figure 6-13: Scenario 4: The state evolution for the first set of cooperative systems



(a) The state evolution of differential robot #2 **(b)** The state evolution of manipulator #2

Figure 6-14: Scenario 4: The state evolution for the second set of cooperative systems

In Figures 6-13 and 6-14 we can see the evolution of the state of each system. Notice how around time $t = 2sec$ the two mobile robots meet and collision avoidance is activated. This causes the two robots to almost stop instantaneously as the repulsive forces cause deceleration.

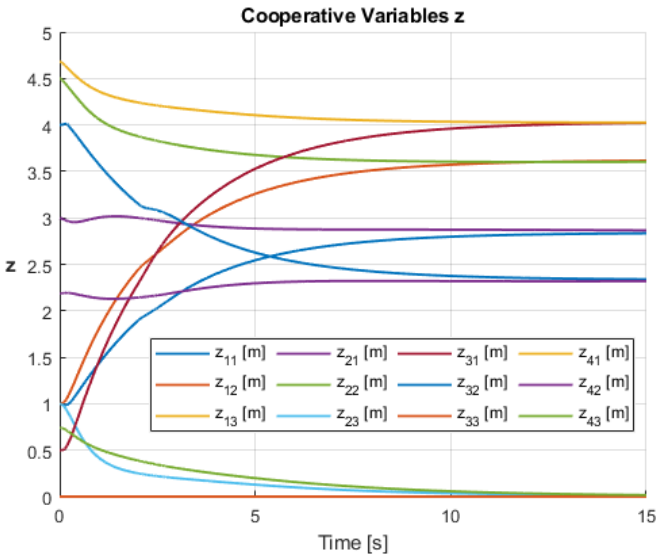


Figure 6-15: Scenario 4: The evolution of the cooperative variables of all systems

The intense oscillations that are visible in the generalized momenta are due to the switch we have used for the implementation of dynamic collision avoidance: when the momenta approach close to zero values then collision avoidance does not pose a considerable danger and the repulsive field is deactivated temporarily until the agents start building up speed.

k_s	k_r	η	$\boldsymbol{K_{v_1}}$	$\boldsymbol{K_{v_2}}$	$\boldsymbol{K_{v_3}}$	$\boldsymbol{K_{v_4}}$
3	9	30	$diag(10, 60)$	$diag(36, 19, 19)$	$diag(10, 60)$	$diag(36, 19, 19)$

Table 6-4: Control parameters for Scenario 4

In Figure 6-15 we can see that consensus is much faster when stabilization of the unconstrained variables is not required although collision avoidance in this extreme scenario is a major hindrance to the fulfillment of the cooperative task. The control parameters for this scenario can be seen in Table 6-4.

Chapter 7

Conclusions and Recommendations

In this chapter we conclude this work with a brief summary in Section 7-1 and a discussion reflecting on its contributions and limitations in Section 7-2. In Section 7-3 we discuss about potential improvements and recommendations for future research.

7-1 Summary

In this work we have extended the novel, distributed control method proposed in [2] to the widely applicable class of nonholonomic mechanical systems. Specifically we have presented step-by-step how the dynamics of such systems can be transformed to an equivalent port-Hamiltonian expression suitable to be addressed with Interconnection and Damping Assignment Passivity-Based Control (IDA-PBC) following [12]. We then presented in detail how these transformed dynamics can be shaped via IDA-PBC and derived the transformed control laws and matching conditions that result from the so-called system matching. In order to complete the task of control and stabilization of nonholonomic mechanical systems we first apply the Passive Configuration Decomposition proposed in [7] to the closed-loop port-Hamiltonian system model which is instrumental for the stabilization of the constrained generalized coordinates. We conclude the topic of stabilization of nonholonomic systems by proposing a novel desired potential that is able to stabilize the full state of such systems with an evident improvement in the transient response. The single-agent results are then extended to a multi-agent setting following the top-down distributed approach proposed in [2]. This approach which originally allows distributed cooperation of heterogeneous and underactuated mechanical systems is now extended to encompass the practically relevant class of nonholonomic mechanical systems proposing a unified cooperative control law for systems with completely different dynamics and limitations (under-actuation, constraints, etc.). Last but not least, a simple inter-agent collision avoidance protocol is implemented which enhances the applicability of this work. The derived control approach was tested in various simulations and successfully addressed the desired objectives in each simulation scenario.

7-2 Discussion and Conclusions

In this work we addressed the problem of distributed control for nonlinear, nonholonomic mechanical systems. The problem we posed in the introduction was stated as:

"Develop a distributed control method which allows a group of non-identical, possibly nonholonomic mechanical systems achieve cooperative objectives while locally avoiding collision with each other."

Reflecting on the results of this work we discuss how this problem was eventually approached and what are the implications of this approach. First of all, port-Hamiltonian modelling allows for the cooperation among heterogeneous agents rendering it more generic and widely-applicable compared to system specific methods that exist for multi-agent systems. From a control perspective, the framework of IDA-PBC allows for control across the wide spectrum of motion of these systems rather than a small region around a desired setpoint which is most usual with conventional linear methods. Moreover, respecting the original system dynamics avoids the use of high feedback gains and leads to more robust and cost-effective controllers. Compared to optimization-based methods, the advantage of an energy-based control method like IDA-PBC is less computational complexity as there is no need to solve non-convex optimization problems with less certainty about solution convergence and online computation feasibility. Nonetheless, the simplicity of the method comes with limitations such as reaching targets in finite time, imposing hard constraints or achieving a desired form of optimality with respect to desired metrics. Among nonlinear control methods, IDA-PBC allows control of underactuated systems which are often met in practical applications (e.g. quad-rotors, overhead cranes etc.). The results in [2] show how these systems can be controlled in a distributed fashion, following cooperative tasks while they stabilize their local dynamics with existing single-agent solutions. In this work we further extend the applicability of the method to the class of nonholonomic systems which are traditionally controlled at a kinematic level and often with feedback linearization techniques that are system specific, or coordinate transformations to achieve non-smooth dynamics which are again most of the times system specific and detach any physical intuition on the resulting system. The desired potential proposed for the stabilization of the constrained variables in this work is expressed in a generic form encompassing a wide class of nonholonomic systems that meet rather mild assumptions and demonstrates significantly improved transient response. The collision avoidance problem with a limited scope to inter-agent collision avoidance was addressed with a dynamic variation of the Artificial Potential Fields (APF). Implementation of this method was straightforward in the context of IDA-PBC since they share intrinsic similarities and allowed for collision avoidance in a dynamic environment with moving obstacles. The efficacy of the aforementioned was tested in various simulation scenarios. However, experimental work is essential in order to fortify the validity of these results. The lack of experimental results indeed gives rise to research questions and future recommendations to enhance this work by withdrawing simplifying assumptions. Improvements and recommendations are discussed in the next section.

7-3 Recommendations

7-3-1 Controllability Analysis

In Chapter 3 we showed that under some assumptions nonholonomic mechanical systems imposed to Pfaffian constraints can have their full state-space stabilized to desired configurations. This however should be permissible by the nonholonomic constraints (2-2). In order to answer this question a proper controllability analysis should be followed using tools from differential geometry and nonlinear control theory, more specifically Frobenius' theorem and nonlinear controllability [10]. This study will shed light on the applicability of the method on different nonholonomic systems based on their controllability properties, whether they are for example Small-Time Locally Accessible (STLA) or Small-Time Locally Controllable (STLC). This will allow for testing the method on more practically interesting nonholonomic systems such as robotic automobiles or space robots with nonholonomic constraints arising from conservation of angular momentum.

7-3-2 Robustness and Robustification

Throughout this work we have assumed ideal conditions for many different aspects. We assume that the agent dynamics and parameters are completely known, no disturbances occur and we also assume that communication is realized via an ideal network with uninterrupted communication. However, the real world is full of uncertainties that can severely deteriorate the performance of the derived control laws. Generally speaking there various sources of uncertainties. In the context of this work we can distinguish between uncertainties related to the network and uncertainties that are related to the dynamics of the agents.

In [4], a novel r-Passivity-Based Control method was proposed that can successfully address uncertainties that stem from network effects such as communication delays and packet dropouts. An interesting research question is to investigate the compatibility between this work and the work in [4] so as to see if a merged solution can be formulated allowing for control of nonholonomic systems in the framework of r-Passivity with robustness to network effects. Nevertheless, we point out here that this task might be challenging as the proposed method in [4] addressed fully-actuated systems while nonholonomic systems are underactuated by definition.

The models that describe mechanical systems are usually accurate since they rely on simple principles like Newton's law of motion and other realistic assumptions. However, the parameters in these models are often uncertain and sometimes impossible to measure while dominant uncertainty usually stems from inertial parameters. Parameter uncertainty can deteriorate performance and compromise stability. A possible solution to this problem is parameter estimation relying on the fact the the differential equations describing mechanical systems, albeit highly nonlinear, are in fact linear to the parameters. The differential equations can be expressed as:

$$\boldsymbol{\zeta}(t) = \boldsymbol{\chi}^{\top}\boldsymbol{\xi}(t) \tag{7-1}$$

where $\boldsymbol{\zeta}$ (observation) and $\boldsymbol{\xi}$ (regressor) denote parameter independent terms that can be measured and $\boldsymbol{\chi}$ (regressand) denotes the lumped up parameters which can be estimated as the solution to a Least Squares problem if persistent excitation and a significant amount of data is gathered and processed properly with filters and robust differentiators.

7-3-3 Collision Avoidance

The APF method we implemented for collision avoidance is favourable for its simplicity and promptness that allows for highly dynamic environments and its scalability which renders it suitable for multi-agent systems. However, as already discussed in Section 5-2-2, the method has well-known limitations with the occurrence of local minima perhaps being the major one. One way to circumvent this problem, as also proposed in [4] is to combine this low-level collision avoidance approach with a higher-level motion planners that can compute collision free paths and set them as reference trajectories to the lower-level controller. Moreover, as also stated in [4], integration of the derived low-level control with higher level planning methods could lead to novel applications that go beyond the formation of configurations.

7-3-4 Critical Damping Assignment

As we briefly described the method of IDA-PBC in Section 3-1 we made clear that several parameters need to be tuned to either solve the matching conditions or achieve the desired transient response. While tuning the parameters for the different simulation scenarios, it was noticed that tuning the damping matrix $\boldsymbol{K_v}$ for each agent played a significant role in the transient response of the total system. Moreover, since this matrix does not appear in the matching conditions, it's a free design parameter. Notice that when analyzing the response of mechanical systems or other second order systems in control theory, one important metric is the damping ratio ζ, a measure describing how rapidly the oscillations decay when systems exhibit oscillatory behavior. When a system is critical damped ($\zeta = 1$), it returns to equilibrium in the minimum amount of time. Since in critical damping the damping coefficient is a function of the inertial and restoring characteristics of the system, it is interesting to investigate if $\boldsymbol{K_v}$ can also be expressed as a function of the desired mass matrix $\boldsymbol{M_d}(\boldsymbol{q})$ and the desired potential $V_d(\boldsymbol{q})$ in a way that can lead to "critical damping" in higher dimensions. Since critical damping usually refers to linear systems, an approximation of the stiffness matrix is then:

$$\boldsymbol{K}(\boldsymbol{q}) = \frac{\partial^2 V_d(\boldsymbol{q})}{\partial \boldsymbol{q} \partial \boldsymbol{q}} \tag{7-2}$$

with the damping matrix for critical damping expressed for example as:

$$\boldsymbol{K_v}(\boldsymbol{q}) = 2\boldsymbol{F}(\boldsymbol{q})^\top \Big(\boldsymbol{K}(\boldsymbol{q})\boldsymbol{M}(\boldsymbol{q})\Big)^{\frac{1}{2}} \boldsymbol{F}(\boldsymbol{q}) \tag{7-3}$$

inspired by the simple, single degree of freedom linear oscillator.

Appendix A

Brief Graph Theory

In this appendix only some few definitions are given. For more details the reader is referred to [1]. The simplest description of the interconnection structure of a network is provided by

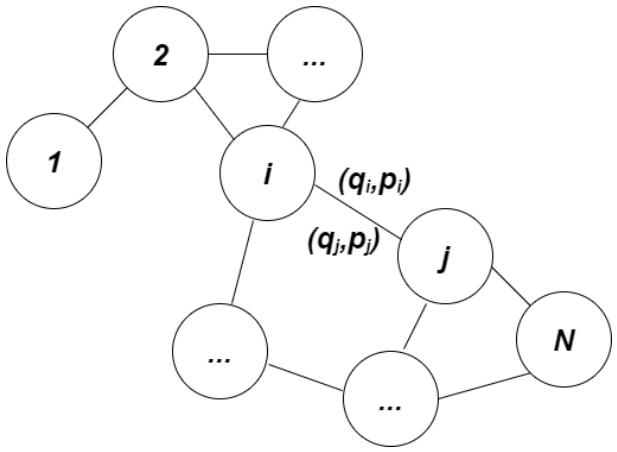


Figure A-1: A simple undirected network graph with the neighboring agents exchanging their state $(\boldsymbol{q}, \boldsymbol{p})$ in order to achieve the formation and inter-agent collision avoidance tasks

a graph, which formalizes the patterns of linkages between different units, or nodes.

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ denote a graph with vertices $\mathcal{V}$ and edges $\mathcal{E} = (\mathcal{V} \times \mathcal{V})$. Every vertex $v \in \mathcal{V}$ represents a mechanical system, while every edge $e_{ij} = (v_i, v_j) \in \mathcal{E}$ represents a one-way communication link between two systems. Graphs can be classified as either *directed* or *undirected* and *weighted* or *unweighted*.

The set of neighbours of vertex i, denoted by N_i, is the set of vertices v_j for which $e_{ij} \in \mathcal{E}$. If for every edge $e_{ij} = (v_i, v_j) \in \mathcal{E}, e_{ij} = (v_j, v_i) \in \mathcal{E}$ then graph $\mathcal{G}$ is undirected. It is directed otherwise.

A path of length r is a sequence $(v_0, v_1, ..., v_r)$ of distinct vertices such that:

$$(v_i, v_{i+1}) \in \mathcal{E} \quad \forall i = 0, 1, ..., r-1 \tag{A-1}$$

A graph is termed *strongly connected* if there exists a path between every pair of vertices.

For each pair of edges $(i,j),(j,i)$ between agents, there is an associated symmetric weight matrix $\boldsymbol{\mathcal{A}_{ij}} = \boldsymbol{\mathcal{A}_{ji}} > \mathbf{0_l}$ known as the *adjacency matrix* and defined as:

$$\boldsymbol{\mathcal{A}_{ij}} = \begin{bmatrix} \mathcal{A}_{ij,11} & \mathcal{A}_{ij,12} & \cdots & \mathcal{A}_{ij,1\ell} \\ \mathcal{A}_{ij,12} & \mathcal{A}_{ij,22} & & \\ \vdots & & \ddots & \\ \mathcal{A}_{ij,1\ell} & & & \mathcal{A}_{ij,\ell\ell} \end{bmatrix} \tag{A-2}$$

If there are no edges between vertex i and j, then $\boldsymbol{\mathcal{A}_{ij}} = \boldsymbol{\mathcal{A}_{ji}} = \mathbf{0_l}$ Self-edges are not allowed, such that $\boldsymbol{\mathcal{A}_{ii}} = \mathbf{0_l}$

The adjacency components $\boldsymbol{\mathcal{A}_{ij}}$ can be used to construct a symmetric *Laplacian matrix* $\boldsymbol{\mathcal{L}} \in \mathbb{R}^{Nl \times Nl}$ that encodes the network topology, defined as:

$$\boldsymbol{\mathcal{L}} = \begin{bmatrix} \sum_{j=1}^{N} \boldsymbol{\mathcal{A}_{1j}} & -\boldsymbol{\mathcal{A}_{12}} & \cdots & -\boldsymbol{\mathcal{A}_{1N}} \\ -\boldsymbol{\mathcal{A}_{12}} & \sum_{j=1}^{N} \boldsymbol{\mathcal{A}_{2j}} & & \\ \vdots & & \ddots & \\ -\boldsymbol{\mathcal{A}_{1N}} & & & \sum_{j=1}^{N} \boldsymbol{\mathcal{A}_{Nj}} \end{bmatrix} \tag{A-3}$$

If agent i is a leader, it has an associated leader weight $\boldsymbol{\mathcal{B}_i} = \boldsymbol{\mathcal{B}_i}^T > \mathbf{0_l}$. If agent j is a follower, its leader weight satisfies $\boldsymbol{\mathcal{B}_i} = \mathbf{0_l}$. The leader matrix is defined as:

$$\boldsymbol{\mathcal{B}} = \begin{bmatrix} \boldsymbol{\mathcal{B}_1} & & \\ & \ddots & \\ & & \boldsymbol{\mathcal{B}_N} \end{bmatrix} \geq \mathbf{0_{N\ell}}, \quad \boldsymbol{\mathcal{B}}_i = \begin{bmatrix} \boldsymbol{\mathcal{B}_{i,11}} & \cdots & \boldsymbol{\mathcal{B}_{i,1\ell}} \\ \vdots & & \vdots \\ \boldsymbol{\mathcal{B}_{i,1\ell}} & \cdots & \boldsymbol{\mathcal{B}_{i,\ell\ell}} \end{bmatrix} \tag{A-4}$$

If the graph is connected and there is at least one leader (at least one $\boldsymbol{\mathcal{B}_i} > \mathbf{0}$), the matrix $\boldsymbol{\mathcal{L}} + \boldsymbol{\mathcal{B}} > \mathbf{0_N}$ is symmetric positive definite.

Bibliography

[1] L. Valk, "Distributed Control of Underactuated and Heterogeneous Mechanical Systems: Using Passivity-Based Control by Interconnection and Damping Assignment as a Tool for Distributed Control Design and Human-Machine Interaction," Master's thesis, Delft University of Technology, 2018.

[2] L. Valk and T. Keviczky, "Distributed control of heterogeneous underactuated mechanical systems," *IFAC-PapersOnLine*, vol. 51, no. 23, pp. 325–330, 2018.

[3] O. de Groot, "Distributed IDA-PBC for Mechanical Systems: A survey on communication delays, collision avoidance and kinematic constraints." Literature Review (not published), Delft University of Technology, 2018.

[4] O. de Groot, "Cooperative r-Passivity-Based Control: Development of a Multi-Agent Passivity-Based Control Scheme with Robustness towards Network Effects," Master's thesis, Delft University of Technology, 2019.

[5] Y. Cao, W. Yu, W. Ren, and G. Chen, "An overview of recent progress in the study of distributed multi-agent coordination," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 427–438, 2012.

[6] R. Ortega, M. W. Spong, F. Gómez-Estern, and G. Blankenstein, "Stabilization of a class of underactuated mechanical systems via interconnection and damping assignment," *IEEE Transactions on Automatic Control*, vol. 47, no. 8, pp. 1218–1233, 2002.

[7] D. Lee and K. Y. Lui, "Passive configuration decomposition and passivity-based control of nonholonomic mechanical systems," *IEEE Transactions on Robotics*, vol. 33, no. 2, pp. 281–297, 2016.

[8] D. T. Greenwood, *Advanced Dynamics*. Cambridge University Press, 2003.

[9] S. Natsiavas, *Applied Dynamics*. Ziti Publications, 1996.

[10] R. M. Murray, Z. Li, S. S. Sastry, and S. S. Sastry, *A mathematical introduction to robotic manipulation*. CRC press, 1994.

[11] A. Van Der Schaft, "Port-hamiltonian systems: an introductory survey," in *Proceedings of the international congress of mathematicians*, vol. 3, pp. 1339–1365, Citeseer, 2006.

[12] A. V. D. Schaft and B. Maschke, "On the hamiltonian formulation of nonholonomic mechanical systems," *Reports on Mathematical Physics*, vol. 34, no. 2, pp. 225 – 233, 1994.

[13] V. Muralidharan, M. T. Ravichandran, and A. D. Mahindrakar, "Extending Interconnection and Damping Assignment Passivity-Based Control (IDA-PBC) to underactuated mechanical systems with nonholonomic Pfaffian constraints: The mobile inverted pendulum robot," in *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, pp. 6305–6310, 12 2009.

[14] B. Brogliato, R. Lozano, B. Maschke, and O. Egeland, *Dissipative Systems Analysis and Control: Theory and Applications*. Springer, 2007.

[15] G. Blankenstein, "Matching and stabilization of constrained systems," in *Proceedings of the Mathematical Theory of Networks and Systems*, 2002.

[16] K. M. Lynch, A. M. Bloch, S. V. Drakunov, M. Reyhanoglu, and D. Zenkov, "Control of nonholonomic and underactuated systems," in *The Control Systems Handbook: Control System Advanced Methods, Second Edition*, pp. 967–1002, CRC Press, 2010.

[17] A. Astolfi, "Discontinuous control of nonholonomic systems," *Systems & control letters*, vol. 27, no. 1, pp. 37–45, 1996.

[18] R. W. Brockett, "Asymptotic stability and feedback stabilization," in *Differential Geometric Control Theory*, pp. 181–191, Birkhauser, 1983.

[19] K. Fujimoto, S. Sakai, and T. Sugie, "Passivity based control of a class of hamiltonian systems with nonholonomic constraints," *Automatica*, vol. 48, no. 12, pp. 3054–3063, 2012.

[20] A. M. Bloch, "Nonholonomic mechanics," in *Nonholonomic Mechanics and Control*, pp. 207–276, Springer, 2003.

[21] D. Lee, "Passivity-based switching control for stabilization of wheeled mobile robots," in *Robotics: Science and Systems*, 2007.

[22] H. Yang and D. Lee, "Cooperative grasping control of multiple mobile manipulators with obstacle avoidance," in *IEEE International Conference on Robotics and Automation*, pp. 836–841, 2013.

[23] J. P. Ostrowski, "Computing reduced equations for robotic systems with constraints and symmetries," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 1, pp. 111–123, 1999.

[24] B. Siciliano and O. Khatib, *Springer Handbook of Robotics*. Berlin, Heidelberg: Springer-Verlag, 2007.

[25] T. Lozano-Pérez and M. A. Wesley, "An algorithm for planning collision-free paths among polyhedral obstacles," *Communications of the Association for Computing Machinery*, vol. 22, no. 10, pp. 560–570, 1979.

[26] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566–580, 1996.

[27] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *The International Journal of Robotics Research*, vol. 30, no. 7, pp. 846–894, 2011.

[28] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *International Conference on Robotics and Automation*, vol. 2, pp. 500–505, IEEE, 1985.

[29] Y. Koren, J. Borenstein, *et al.*, "Potential field methods and their inherent limitations for mobile robot navigation.," in *International Conference on Robotics and Automation*, vol. 2, pp. 1398–1404, 1991.

[30] S. S. Ge and Y. J. Cui, "New potential functions for mobile robot path planning," *IEEE Transactions on Robotics and Automation*, vol. 16, no. 5, pp. 615–620, 2000.

[31] S. S. Ge and Y. J. Cui, "Dynamic motion planning for mobile robots using potential field method," *Autonomous Robots*, vol. 13, no. 3, pp. 207–222, 2002.

[32] P. Fiorini and Z. Shiller, "Motion planning in dynamic environments using velocity obstacles," *The International Journal of Robotics Research*, vol. 17, no. 7, pp. 760–772, 1998.

[33] J. O'Rourke and N. Badler, "Decomposition of three-dimensional objects into spheres," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 3, pp. 295–305, 1979.

Glossary

List of Acronyms

PBSC	Passivity Based Switching Control
PBVC	Passivity Based Time-Varying Control
IDA-PBC	Interconnection and Damping Assignment Passivity-Based Control
r-PBC	r-Passivity Based Control
ODEs	Ordinary Differential Equations
PDEs	Partial Differential Equations
MPC	Model Predictive Control
PRM	Probabilistic Roadmap
RRT	Rapidly Exploring Random Tree
VFH	Vector Field Histogram
ORM	Obstacle Restriction Method
SAFA	Steer Angle Field Approach
DWA	Dynamic Window Approach
APF	Artificial Potential Fields
VO	Velocity Obstacles
ORCA	Optimal Reciprocal Collision Avoidance
FIRAS	Force Inducing an Artificial Repulsion from the Surface
GNRON	Goals Non-Reachable with Obstacles Nearby
STLA	Small-Time Locally Accessible
STLC	Small-Time Locally Controllable