Contributions to Motion Planning and Orbital Stabilization

Case studies: Furuta Pendulum swing up, Inertia Wheel oscillations and Biped Robot walking

PEDRO XAVIER MIRANDA LA HERA

Department of Applied Physics and Electronics
UMEÅ UNIVERSITY, SWEDEN
Umeå, October 2008
Thesis for the degree of licentiate of engineering in Applied Electronics at Umeå University
Vetenskaplig uppsats för avläggande av teknologi licentiatesexamen i tillämpad elektronik vid Umeå Universitet

ISSN 1654-5419:3
ISBN 978-91-7264-665-0

Copyright © Pedro Xavier Miranda La Hera, 2008.

Author e-mail: xavier.lahera@tfe.umu.se
Typeset in Latex by Pedro X. La Hera
Printed by Print & Media, Umeå University, Umeå, October 2008
Contributions to Motion Planning and Orbital Stabilization

Case studies: Furuta Pendulum swing up, Inertia Wheel oscillations and Biped Robot walking

Pedro Xavier Miranda La Hera

Robotics and Control Lab, Umeå University, Sweden

ABSTRACT

Generating and stabilizing periodic motions in nonlinear systems is a challenging task. In the control system community this topic is also known as limit cycle control. In recent years a framework known as Virtual Holonomic Constraints (VHC) has been developed as one of the solutions to this problem. The aim of this thesis is to give an insight into this approach and its practical application.

The contribution of this work is primarily the experimental validation of the theory. A step by step procedure of this methodology is given for motion planning, as well as for controller design. Three particular setups were chosen for experiments: the inertia wheel pendulum, the Furuta pendulum and the two-link planar pendulum. These under-actuated mechanical systems are well known benchmarking setups for testing advanced control design methods.

Further application is intended for cases such as biped robot walking/running, human and animal locomotion analysis, etc.

Keywords: Under-actuated Systems, Orbital Exponential Stability, Motion Planning, Virtual Constraints, Walking robots, Limit Cycle Walking.
Preface

Parts of the contributions presented in this thesis have previously been accepted to conferences or submitted to journals. Here is the whole list of the papers with the author’s contributions:

▷ **Pedro La Hera**, Leonid B. Freidovich, Anton S. Shiriaev, Uwe Mettin, “Swinging up the Furuta Pendulum via stabilization of a planned trajectory: Theory and Experiments,” *provisionally accepted to the Mechatronics Journal.*

▷ Leonid Freidovich, **Pedro La Hera**, Uwe Mettin, Anders Robertsson, Anton Shiriaev, Rolf Johansson, "Stable Periodic Motions of Inertia Wheel Pendulum via Virtual Holonomic Constraints", submitted to *The International Journal of Robotics Research*

▷ **Pedro La Hera**, Leonid B. Freidovich, Anton S. Shiriaev, Uwe Mettin, "Orbital stabilization of a pre-planned periodic motion to swing up the Furuta Pendulum. Theory and Experiments”, submitted to *The International Conference on Robotics and Automation 2009*

▷ **Pedro La Hera**, Uwe Mettin, Simon Westerberg, Anton S. Shiriaev, "Cascade Control Design for Hydraulically Actuated Cranes, Based on System Identification”, submitted to *The International Conference on Robotics and Automation 2009*

▷ **Pedro La Hera**, Andrej Zanhar, Uwe Mettin, Simon Westerberg, Anton Shiriaev, "Identification-based Modeling and Control of Hydraulically actuated Forestry Cranes”, *The Swedish Conference in Control*
PREFACE

System - Reglernöte, Sweden 2008


- Uwe Mettin, Pedro La Hera, Leonid Freidovich, Anton Shiriaev,
"How springs can help to stabilize motions of underactuated systems with weak actuators", *Proceedings of the 47th IEEE conference on Decision and Control, Cancun, Mexico, December 9-11 2008*

# Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abstract</td>
<td>i</td>
</tr>
<tr>
<td>Preface</td>
<td>iii</td>
</tr>
</tbody>
</table>

## I INTRODUCTION

1 INTRODUCTION 3

1.1 Motivation ........................................ 3
1.2 Thesis work ........................................... 6

## II PRELIMINARIES

2 PRELIMINARIES 9

2.1 VIRTUAL HOLONOMIC CONSTRAINTS APPROACH 11

2.1.1 Motion Planning Algorithm ....................... 13
2.1.2 Controller design ................................ 19

2.2 REAL-TIME IMPLEMENTATION ISSUES .................. 22

2.2.1 Hardware .......................................... 22
2.2.2 Sampling .......................................... 22
2.2.3 DC motors electromotive torque constant ........ 23
2.2.4 Velocity estimation ................................. 24
2.2.5 Friction Identification and Compensation ........ 25
## CONTENTS

2.2.6 Numerical Solution of the Riccati Equation . . . . . . . . . 27
2.2.7 Projection of Riccati Solution . . . . . . . . . . . . . . . 28
2.2.8 Model Parameter Identification . . . . . . . . . . . . . . . 29

### III Design of Periodic Motions for the Inertia Wheel Pendulum 31

3 Inertia Wheel Pendulum 33

3.1 Inertia wheel pendulum dynamics . . . . . . . . . . . . . . . 34
3.2 Motion planning . . . . . . . . . . . . . . . . . . . . . . . . . . 35
3.3 Controller design . . . . . . . . . . . . . . . . . . . . . . . . . 37
3.4 Experimental Results . . . . . . . . . . . . . . . . . . . . . . . 40
3.4.1 Friction Compensation . . . . . . . . . . . . . . . . . . . 41
3.4.2 DC motors electromotive torque constant . . . . . . . . . 41
3.4.3 Oscillations Around Downward equilibrium . . . . . . . . 42
3.4.4 Oscillations Around upright equilibrium . . . . . . . . . 44
3.5 Conclusions . . . . . . . . . . . . . . . . . . . . . . . . . . . . 45

### IV Can we make a pendulum to swing up by following a predefined trajectory? Virtual Holonomic Constraint Design 47

4 Furuta Pendulum 49

4.1 Introduction . . . . . . . . . . . . . . . . . . . . . . . . . . . . 49
4.1.1 Preliminaries: Dynamics of the Furuta Pendulum . . . . . 51
4.1.2 Preliminaries: Problem Formulation . . . . . . . . . . . . 53
4.1.3 Preliminaries: Planning a Motion via Imposing Constraint on $\phi$ and $\theta$ . . . . . . . . . . . . . . . . . . . . . . 53
4.2 Main Results . . . . . . . . . . . . . . . . . . . . . . . . . . . . 54
4.2.1 Planning New Homoclinic Curves for Swing Up . . . . . 54
4.2.2 Choosing the synchronization Function $\Phi(\cdot)$ . . . . . 58
4.3 Results of Computer Simulations . . . . . . . . . . . . . . . . . 61
## CONTENTS

4.3.1 Switching law and stabilizing controller simulation . . . . . 63
4.4 Sampling, Real-Time implementation and Experimental Results  . 65
  4.4.1 Sampling . . . . . . . . . . . . . . . . . . . . . . . . . . . . 66
  4.4.2 Velocity estimation . . . . . . . . . . . . . . . . . . . . . . 66
  4.4.3 Friction Compensation . . . . . . . . . . . . . . . . . . . . . 67
  4.4.4 Swing Up experimental results . . . . . . . . . . . . . . . . 68
  4.4.5 Switching law and stabilizing controller experimental results 69
4.5 Conclusions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 70

V First steps into walking bipeds 73

5 A simple walking case: the compass biped 75
  5.1 Hybrid systems . . . . . . . . . . . . . . . . . . . . . . . . . . 76
  5.2 Compass biped modeling . . . . . . . . . . . . . . . . . . . . . 78
    5.2.1 Rigid body collisions . . . . . . . . . . . . . . . . . . . . . 80
    5.2.2 The hybrid Model . . . . . . . . . . . . . . . . . . . . . . . 82
  5.3 Passive Walking . . . . . . . . . . . . . . . . . . . . . . . . . . . 83
  5.4 Definition of constraint polynomials . . . . . . . . . . . . . . . . 83
  5.5 Hybrid zero dynamics . . . . . . . . . . . . . . . . . . . . . . . . 87
    5.5.1 Designing constraint functions to achieve invariance of the
         hybrid zero dynamics . . . . . . . . . . . . . . . . . . . . . . . . 89
    5.5.2 Application of Optimization to design stable walking . . . 90
    5.5.3 Example on the Compass Biped . . . . . . . . . . . . . . . . 92
    5.5.4 Results of Optimization . . . . . . . . . . . . . . . . . . . . . 94
  5.6 Controller Design . . . . . . . . . . . . . . . . . . . . . . . . . . 95
  5.7 Conclusions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 97

VI Appendix 99

A Modeling the Furuta Pendulum 101
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>B Model Parameters Estimation</td>
<td>105</td>
</tr>
<tr>
<td>B.1 Arm and Motor Parameters Identification</td>
<td>106</td>
</tr>
<tr>
<td>B.2 Pendulum Parameters Identification</td>
<td>106</td>
</tr>
<tr>
<td>C LQR controller Design</td>
<td>109</td>
</tr>
<tr>
<td>D Controllability - Furuta Pendulum</td>
<td>111</td>
</tr>
<tr>
<td>E Proof of theorem 3</td>
<td>113</td>
</tr>
</tbody>
</table>
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>(a) Asimo robot from Honda, (b) Hubo robot from Korea.</td>
<td>4</td>
</tr>
<tr>
<td>1.2</td>
<td>Passive compass biped. Figure from the Steve Collin’s Homepage</td>
<td>5</td>
</tr>
<tr>
<td>1.3</td>
<td>Passive walker with knees. Figure from the Nagoya Institute of Technology</td>
<td>6</td>
</tr>
<tr>
<td>2.1</td>
<td>Two-Link manipulator.</td>
<td>12</td>
</tr>
<tr>
<td>2.2</td>
<td>Evolution of motion from point 1 to 2 for the planar two-link manipulator.</td>
<td>15</td>
</tr>
<tr>
<td>2.3</td>
<td>Arms motion $\phi$ as a function of $\theta$.</td>
<td>15</td>
</tr>
<tr>
<td>2.4</td>
<td>Two dimensional manifold containing periodic orbits.</td>
<td>16</td>
</tr>
<tr>
<td>2.5</td>
<td>Analysis of an equilibrium point in terms of amplitude.</td>
<td>18</td>
</tr>
<tr>
<td>2.6</td>
<td>Coulomb + viscous friction.</td>
<td>26</td>
</tr>
<tr>
<td>2.7</td>
<td>Friction compensation scheme.</td>
<td>27</td>
</tr>
<tr>
<td>2.8</td>
<td>Representation of a periodic solution.</td>
<td>29</td>
</tr>
<tr>
<td>3.1</td>
<td>Schematic diagram of the system.</td>
<td>34</td>
</tr>
<tr>
<td>3.2</td>
<td>Results of the motion planning for downward oscillations: (a) Desired phase portrait $\theta(t)$ vs. $\dot{\theta}(t)$, (b) Desired trajectory for $\theta(t)$ as function of time, (c) Ideal constraint function $\phi$ vs $\theta$, (d) Desired phase portrait $\phi(t)$ vs. $\dot{\phi}(t)$.</td>
<td>38</td>
</tr>
<tr>
<td>Section</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>3.3</td>
<td>Results of the motion planning for upright oscillations: (a) Desired phase portrait $\theta(t)$ vs. $\dot{\theta}(t)$, (b) Desired trajectory for $\theta(t)$ as function of time, (c) Ideal constraint function $\phi$ vs $\theta$, (d) Desired phase portrait $\phi(t)$ vs. $\dot{\phi}(t)$. 39</td>
<td></td>
</tr>
<tr>
<td>3.4</td>
<td>Inertia Wheel Pendulum. 40</td>
<td></td>
</tr>
<tr>
<td>3.5</td>
<td>Results of experimental validation for downward oscillations: (a) Desired vs. Experimental validation of $\theta(t)$ vs. $\dot{\theta}(t)$, (b) Ideal vs. experimental constraint function $\phi$ vs $\theta$; (c) Desired vs. Experimental validation of $\phi(t)$ vs. $\dot{\phi}(t)$, (d) Input signal $u(t)$. 43</td>
<td></td>
</tr>
<tr>
<td>3.6</td>
<td>Results of experimental validation for upright oscillations: (a) Desired vs. Experimental validation of $\theta(t)$ vs. $\dot{\theta}(t)$, (b) Ideal vs. experimental constraint function $\phi$ vs $\theta$; (c) Desired vs. Experimental validation of $\phi(t)$ vs. $\dot{\phi}(t)$, (d) Input signal $u(t)$. 44</td>
<td></td>
</tr>
<tr>
<td>4.1</td>
<td>The Furuta Pendulum built at the Department of Applied Physics and Electronics, Umeå University. 52</td>
<td></td>
</tr>
<tr>
<td>4.2</td>
<td>Two homoclinic orbits of the equilibrium at $\theta = 0$ are shown on the phase portrait of (4.4). One of them intersects the line ${\dot{\theta} = 0}$ at $a_1$, an other at $a_2$. In (b) an exemplified periodic trajectory orbiting two homoclinic curves is depicted. 56</td>
<td></td>
</tr>
<tr>
<td>4.3</td>
<td>(a) The phase portrait for (4.4) when condition 3) of Lemma 1 is satisfied. (b) The phase portrait for (4.4) when condition 3) of Lemma 1 is not valid. 61</td>
<td></td>
</tr>
<tr>
<td>4.4</td>
<td>The values of $F(\pi, K)$ and $F(2\pi, K)$ versus $K_n = K \cdot \frac{p_5}{p_2 + p_5}$, blue and red respectively. As seen, they have different signs for most of positive values of $K_n$, while around zero, $K_n &lt; 0.01$, their signs are the same (negative). 62</td>
<td></td>
</tr>
<tr>
<td>4.5</td>
<td>(a) Phase portrait for (4.4) with (4.9). The trajectory marked with bold line represents the periodic solution to be controlled later; (b) Solutions of the system (4.4) around $\theta_0 = 0$ depicted as functions of time. The solution to be stabilized (bold line) has a period $T \approx 4.0454$ sec. 63</td>
<td></td>
</tr>
</tbody>
</table>
4.6 Results of numerical simulations: (a) $\theta(t)$ vs. $\dot{\theta}(t)$ of the desired cycle; (b) Simulation of the behavior of $\theta(t)$ vs. $\dot{\theta}(t)$ for the closed loop system with initial conditions (4.15); (c) Arctangent relation of $\phi$ with respect of $\theta$ as expected from (4.9); (d) periodic trajectory of $\theta$ as a function of time. .......................... 64

4.7 The switching region specified as a range of $\theta$ and $\dot{\theta}$. .................. 65

4.8 (a) The phase portrait of the system (4.4) around the upward equilibrium at $\theta = 0$; (b) Swing up proposed and local stabilizing controller, behavior of $\theta$ as function of time. .................. 66

4.9 Estimated friction. .................................................. 68

4.10 Results of experimental validation for the swinging up stage: (a) $\theta(t)$ vs. $\dot{\theta}(t)$ for the desired circle; (b) Experimental validation of the behavior of $\theta(t)$ vs. $\dot{\theta}(t)$ for the closed loop system with initial conditions (4.15); (c) Arctangent relation between $\phi$ and $\theta$; (d) Experimental oscillatory behavior of $\theta$ for the cycle as a function of time. .......................... 69

4.11 (a) The experimental phase portrait of the system for swinging up and stabilizing controller around the upward equilibrium at $\theta = 0$; (b) Experimental behavior of $\theta$ as a function of time. ... 70

5.1 Bouncing ball simulation. The graph on the top shows the velocity profile. The graph on the bottom shows the height of the ball after each consecutive collision with the ground. ............... 78

5.2 Bouncing ball model. ................................. 79

5.3 Compass like biped. The picture is provided in the website of Goswami ................................. 80

5.4 A walking cycle with leg impacts with the ground. ............... 80

5.5 Limit cycles for different slope angles. Plot is provided at the website http://www.ambarish.com/compass.html ........................ 84

5.6 The anatomy of a Beziéér curve. ................................. 85
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.7</td>
<td>Hybrid zero dynamics, where the walking surface is represented by planes that somehow impose a boundary for the continuous dynamics.</td>
<td>88</td>
</tr>
<tr>
<td>5.8</td>
<td>(a) The phase portrait of the walker for complete cycle, i.e. stance and swing phase. The x-axis is $\theta$ and the y-zaxis is $\dot{\theta}$; (b) Virtual Constraint Function, where x-axis is $\theta$ and the y-axis is $\theta_1$.</td>
<td>95</td>
</tr>
<tr>
<td>5.9</td>
<td>States of the transverse linearization exponentially approaches zero.</td>
<td>97</td>
</tr>
<tr>
<td>A.1</td>
<td>Three-dimensional coordinate representation.</td>
<td>102</td>
</tr>
<tr>
<td>B.1</td>
<td>Simulated Response with estimated values vs recorded data.</td>
<td>107</td>
</tr>
</tbody>
</table>
Part I

INTRODUCTION
The work presented here is related to motion planning and control of under-actuated mechanical systems. As a motivation, a short introduction to some challenges in the field follows next.

1.1 Motivation

Let’s consider an attractive example of biped robots to illustrate roughly the topic of study. In the early 70’s the concept of static stability was applied in the first successful creation of bipedal robots. Static stability means that the vertical projection of the Center of Mass stays within the support polygon formed by the feet. It is straightforward to ensure walking stability in this way, but it drastically limits the speed of the walking motions that can be obtained.
CHAPTER 1. INTRODUCTION

In later years, a new approach known as the ‘Zero Moment Point’ (ZMP) concept was proposed. This method is currently used by most humanoid robots. The ZMP specifies the point with respect to which dynamic reaction force at the contact of the foot with the ground does not produce any moment. The stability is ensured (for sufficiently slow motions) with the ZMP-criterion, which constrains the stance foot to remain in flat contact (observe the feet of the robots in Fig. 1.1(a)-(b)) with the floor at all times. This constraint is less restrictive than static walking because the Center of Mass may travel beyond the support polygon. Nevertheless, these robots are still under-achieving in terms of efficiency, disturbance handling, and natural appearance compared to humans.

One of the possible ways to increase the performance of walking is by avoiding the flatness of the contact between the support foot and the ground. That is studied by the so called ‘Limit Cycle Walking’, which is a new approach with fewer artificial constraints. The aim of this approach is to find more efficient, natural, fast and robust walking patterns.

The ‘Limit Cycle Walking’ concept can be shortly exemplified as follows. Human and animal locomotion can be described as having a periodic pattern. An example of this can be readily observed when the legs oscillate to-and-from with a periodic cycle behavior, quite similar to the pendular motion of a clock. This periodic behavior brings the idea of periodic cycles or limit cycles as it has been
1.1. MOTIVATION

formally introduced.

Tad McGeer [31] showed that a planar mechanism, composed by two legs (see Fig. 1.2), could be made to walk stably down a slight slope. No extra energy or control was needed for this action. This system acts like a two coupled pendula. The stance leg acts like an inverted pendulum, and the swing leg acts like a free pendulum attached to the stance leg at the hip. Given sufficient mass distribution, the system will have a stable limit cycle, that is, a nominal trajectory that repeats itself and will return to this trajectory even if perturbed slightly.

McGeer constructed a physical example of this walker, using a special mechanism to fold the swing feet up so that the swing leg would clear the ground at mid-stride to avoid scuffing.

![Passive compass biped. Figure from the Steve Collin’s Homepage](image)

An extension of the two-segment passive walker is the inclusion of knees (see Fig. 1.3), which provide natural ground clearance avoiding the need for any additional mechanisms. McGeer showed that even with knees, the system has a stable limit cycle [32]. A subsequent physical model demonstrated this quite convincingly\(^1\).

\(^1\)A video with such an example is available at
This form of exponentially stable walking opened a new view to the design of controllers for stabilizing walking and running gaits. In the scientific community this type of approach has been named as dynamic walking.

Dynamic walking suggests that it is a matter of a good understanding about the dynamics of locomotion, to design control algorithms that will induce the desired walking by following desired gaits. Different approaches were introduced in literature and practice, e.g [1], [45], [46].

1.2 Thesis work

Inspired on the above facts, the work done in this thesis is directed to the analysis of limit cycle control. Some educational setups are used as platform to study the technique known as the virtual holonomic constraint approach. This method,

http://www.youtube.com/watch?v=CK8IFEGmiKY&feature=related
introduced in works such as [36–39], addresses the problem of achieving orbital stabilization of periodic orbits in under-actuated nonlinear mechanical systems.

The aim of this study is to gain the experience of the theoretical and experimental application of the method. The long term goal of this research is to extend the approach to cases such as motion planning and control design of limit cycle walking, study of human and animal locomotion, etc. In the following chapters, the mathematical essence of the method will be shortly introduced, so that the reader can gain an understanding of this method and how it can be applied in theoretical analysis and in practice.

The thesis is organized as follows. In chapter 2, some preliminaries are given and issues related to real time implementation are considered. In chapter 3, the case of a reaction wheel pendulum is worked out to have an understanding of the ideas presented in previous chapters. In chapter 4, the method is used for designing non trivial trajectories for the case of the Furuta pendulum. In chapter 5, the first case in bipedal walking is presented, taking as example the so-called compass biped. Finally, some discussions and concluding ideas are posted, also future work objectives are stated.
Part II

PRELIMINARIES
2.1 VIRTUAL HOLONOMIC CONSTRAINTS APPROACH

Imposing constraints on a system is simply another way of stating that there are forces present in the problem that cannot be specified directly. They are known in terms of their effect on the motion of the system. Examples of constrained systems are:

- A particle is constrained to move in the x-y plane, the equation of constraint is $z = 0$, the constraint is holonomic.

- A particle is constrained to move on a circle in the x-y plane, the equations of constraints are $z = 0, x^2 + y^2 - r^2 = 0$, the constraints are holonomic.
Consider now the example of the planar double pendulum depicted in Fig. 2.1, where $\phi$ and $\theta$ are used to represent the angular position of the links. Imagine that a desired motion would be to drive the system from the initial position one to the mirror position two (see notation in Fig. 2.1), and do this repeatedly back and forth. Moving from one position to another periodically can be related to constraining the motion of the links for that particular oscillatory motion, even though the constraint is not physically present (i.e. the links can freely rotate). If this could be achieved by some feedback control torques, we have a virtually constrained system.

![Two-Link manipulator](image)

**Figure 2.1:** Two-Link manipulator.

In the case of a fully actuated systems, it is clear that a finite time trajectory could be specified and stabilized asymptotically by a controller. In the case of under-actuated systems, however, it is not the case. Considering the example of the two link manipulator from above. Imagine that the arm denoted by $\phi$ is actuated by a DC motor, but the second link $\theta$ can freely move. Applying linear control, a solution to this problem might not be achievable. Applying methods such as the virtual holonomic constraint approach [37], however, it has been shown and experimentally validated that such a periodic motion is feasible.
2.1. VIRTUAL HOLONOMIC CONSTRAINTS APPROACH

Starting from this point we will consider the case of under-actuated mechanical systems, which dynamics can be described by the Euler-Lagrange equation

\[
\frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{q}} \right) - \frac{\partial \mathcal{L}}{\partial q} = B(q)u,
\]  

(2.1)

where \( q, \dot{q} \) are vectors denoting the generalized coordinates and velocities\(^1\), and \( u \) is a vector of independent control inputs. The Lagrangian \( \mathcal{L} \) is defined as

\[
\mathcal{L}(q, \dot{q}) = \frac{1}{2} \dot{q}^T M(q) \dot{q} - V(q),
\]  

(2.2)

with \( M(q) \) being a positive-definite matrix of inertias and \( V(q) \) the potential energy. In the coming sections we consider the case when the under-actuation is of degree one, i.e.

\[
\dim u = \dim q - 1.
\]  

(2.3)

and \( B(q) \) has full rank.

2.1.1 Motion Planning Algorithm

Periodic motions could be thought of as constrained motions, where the traveling range of each individual link could be defined by some geometric relations among the generalized coordinates. The main idea of the virtual holonomic constraints approach is to define these geometric relations and impose them eventually by feedback control. Since those constraints are not physically present, hence the term \textit{virtual}. The advantage of imposing virtual constraints into the mechanical design through feedback control rather than physically, opens the possibility to reconfigure the robot by software or electronically instead of mechanically. In that way the robot can achieve different tasks by a matter of the control design.

The next theoretical results analyze this for a particular type of under-actuated mechanical systems. Certain assumptions are done at the first stage of motion planning:

\(^1\)Any set of independent quantities, e.g. \( q_1, q_2, ..., q_s \), which completely define the position of the system with \( s \) degrees of freedom, are called generalized coordinates of the system, and the derivatives are called generalized velocities.
1. it is assumed that the model of the system described by (2.1) is not affected by external disturbances, and the control torque $u$ can be applied in continuous time without any sampling,

2. it is assumed that all state variables are available at every instant of time (e.g. positions, velocities, etc).

None of these assumptions are realistic in the practical sense, and later we will discuss some modifications that are necessary for implementation. However, it is important to point out that the design to be presented in the coming chapter will achieve orbital exponential stability [24, Def. 8.2 and 4.5] and so a certain degree of robustness with respect to modeling error is expected [24, Lemma 9.2].

The constructive tool to plan a periodic motion via the virtual holonomic constraint approach [37] is summarized in the following steps.

**Step 1: Choice of a holonomic constraint**

The starting point of the design is to define a virtual holonomic constraint, with the objective of defining some geometrical relation among the links. A function that reparametrizes the generalized coordinates with respect to some new variable can be defined as

$$q_i = \Phi_i(\theta),$$  \hspace{1cm} (2.4)

where $\theta$ is used to denote the new variable and $i = 1\ldots n$ denotes the index of some degree of freedom. The function $\Phi(\theta)$ represents some linear or nonlinear function of the $\theta$ variable.

As a way to exemplify a conceptual procedure to find a constraint function, let's consider the example of the two-link manipulator mentioned earlier. A natural trajectory to drive the system from position one to two (see Fig. 2.1) is depicted from left to right in Fig. 2.2.

If a trajectory is given, it would be possible to define each link trajectory as a function of time, but according to (2.4) the relation of $\phi$ and $\theta$ according to some new variable is of interest.

Let's choose $\theta$ itself as the new reparametrization variable. The relation of $\phi$ as a function of $\theta$ is shown in Fig. 2.3. It is clearly observed that the arms motion
along the trajectory, in respect of the unactuated link, resembles a linear relation. Therefore, in this case we could propose a linear constraint function defined by

\[ \phi = k \cdot \theta + \phi_{P3}, \]

where the gain \( k < 0 \) and \( k \in \mathbb{R} \), and \( \phi_{P3} \) is the point at which \( \phi = 0 \).

In many cases, these particular type of constraint functions could be designed analytically, but in more complicated cases they could, for example, be extracted from recorded data as presented in [41] or found by an optimization procedure.

**Step 2: Computation of the reduced-order dynamics**
Result of the dynamics (2.1) constrained to (2.4) is a dynamical system living on a two dimensional manifold (see Fig. 2.4) which is defined by $[\theta, \dot{\theta}]$.

Figure 2.4: Two dimensional manifold containing periodic orbits.

In order to find the dynamics of this manifold, we could project the dynamics (2.1) onto the virtual holonomic constraint (2.4), which has to be satisfied irrespective of the control input. This can be often done in a straightforward way by substituting $q_i$, $\dot{q}_i$ and $\ddot{q}_i$ from (2.4) into the unactuated equation of (2.1). The result is a differential equation of the particular form

$$\alpha(\theta_*)\ddot{\theta}_* + \beta(\theta_*)\dot{\theta}_*^2 + \gamma(\theta_*) = 0,$$  \hspace{1cm} (2.5)

with $\alpha(\theta_*)$, $\beta(\theta_*)$ and $\gamma(\theta_*)$ being calculated as follows. Rewriting the controlled Euler Lagrange equation (2.1) in the next form,

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = B(q)u,$$  \hspace{1cm} (2.6)

and the derivatives of the constraint function (2.4) as

$$\dot{q} = \Phi'(\theta)\dot{q},$$  \hspace{1cm} (2.7)

$$\ddot{q} = \Phi''(\theta)\dot{q}^2 + \Phi'(\theta)\ddot{q},$$  \hspace{1cm} (2.8)

by combining (2.6) and (2.8), we obtain

$$M(q)\left(\Phi''(\theta)\dot{q}^2 + \Phi'(\theta)\ddot{q}\right) + C(q, \dot{q})\left(\Phi'(\theta)\dot{q}\right) + G(q) = B(q)u.$$  \hspace{1cm} (2.9)
2.1. VIRTUAL HOLONOMIC CONSTRAINTS APPROACH

There exist a row matrix $B^\perp(q)$, such that $B^\perp(q)B(q)u^* = 0 \forall q$, so that $\alpha(\theta_*)$, $\beta(\theta_*)$ and $\gamma(\theta_*)$ can be calculated as

\[
\begin{align*}
\alpha(\theta) &= B^\perp(\Phi(\theta))M(\Phi(\theta))\Phi'(\theta), \\
\beta(\theta) &= B^\perp(\Phi(\theta))(C(\Phi(\theta), \Phi'(\theta))\Phi'(\theta) + M(\Phi(\theta))\Phi''(\theta)), \\
\gamma(\theta) &= B^\perp(\Phi(\theta))G(\Phi(\theta)).
\end{align*}
\tag{2.10}
\]

Solutions of the reduced dynamics (2.5) might be closed orbits of different amplitudes and periods as shown in Fig. 2.4. A key issue is that, once the system is projected into the virtual holonomic constraint, dependence of time disappears as seen in (2.5). Later some examples will be given to show such calculations for specific systems.

**Step 3: Search for an achievable periodic solution**

Periodic cycles or orbits are implied by the existence of a center at the equilibrium point. The equilibria of the reduced dynamics (2.5) are defined by the solutions of the equation $\gamma(\theta_0) = 0$.

**Theorem 1.** Consider the auxiliary linear system\(^2\)

\[
\ddot{z} + \left[ \frac{d}{d\theta} \frac{\gamma(\theta)}{\alpha(\theta)} \right]_{\theta=\theta_0} z = 0,
\tag{2.11}
\]

if this linear system has a center at $z = 0$, then the nonlinear system (2.5) also has a center at the equilibrium $\theta = \theta_0$. Furthermore, the existence of small cycles around some equilibrium $\theta = \theta_0$ are guaranteed [39] provided that

\[
f(\theta_0) = \frac{\gamma'(\theta_0)}{\alpha(\theta_0)} > 0
\tag{2.12}
\]

and in opposite, the existence of a saddle is guaranteed if $f(\theta_0) < 0$.

**Step 4: Integral function**

A particular property of a solution $[\theta_*, \dot{\theta}_*]$ of (2.5) is stated in the next theorem.

**Theorem 2.** Suppose the function $\alpha(\theta_*)$ has only isolated zeros. If the solution of (2.5), with initial conditions $\theta(0) = \theta_0$ and $\dot{\theta}(0) = \dot{\theta}_0$, exists and is continuously

\(^2\)It is the linearization of (2.5) at $q = q_0$ if $\alpha(\theta_*)$, $\beta(\theta_*)$ and $\gamma(\theta_*)$ are $C^1$-smooth functions.
differentiable, then along this solution the function [39]

\[ I(\theta, \dot{\theta}, \theta_0, \dot{\theta}_0) = \dot{\theta}^2 - \Psi(\theta_0, \theta) \left[ \dot{\theta}_0^2 - \int_{\theta_0}^{\theta} \Psi(s, \theta_0) \frac{2\gamma(s)}{\alpha(s)} ds \right], \tag{2.13} \]

with

\[ \Psi(\theta_0, \theta_1) = \exp \left\{ -2 \int_{\theta_0}^{\theta_1} \frac{\beta(\tau)}{\alpha(\tau)} d\tau \right\} \tag{2.14} \]

preserves its zero value.

Consider the case when \( \theta = \theta_0 \) is a center as shown in Fig. 2.5. Every trajectory has a maximum amplitude along the \( \theta \) axis, which has been denoted as \( A + \theta_0 \) in Fig. 2.5. There exist also a maximum amplitude before entering to the next equilibrium, denoted as \( \hat{A} + \theta_0 \) in Fig. 2.5.

\[
\gamma(\theta_o) = 0 \text{ or } \alpha(\theta_o) = 0
\]

**Figure 2.5:** Analysis of an equilibrium point in terms of amplitude.

In order to compute the period of oscillations for a given periodic solution, we can solve the nonlinear equation (see Fig. 2.5)

\[ I(x, 0, A + \theta_0, 0) = \int_{A+\theta_0}^{x} \exp \left\{ -2 \int_{s}^{x} \frac{\beta(\tau)}{\alpha(\tau)} d\tau \right\} \frac{\gamma(s)}{\alpha(s)} ds = 0, \tag{2.15} \]

and so the period of oscillation can be calculated as [24, Exercise 2.18]

\[ T = \sqrt{2} \int_{x}^{A+\theta_0} \frac{d\tau}{\sqrt{I(x, 0, \tau, 0)}}. \tag{2.16} \]
2.1. VIRTUAL HOLONOMIC CONSTRAINTS APPROACH

Although such calculations could be tedious for complex systems, in certain cases it is possible to define trajectories with predefined amplitudes and periods by applying (2.15)-(2.16), as it will be shown later. This calculation is applicable also to the case when we search parameters of a possible constraint function (e.g. polynomial constraint) [27].

2.1.2 Controller design

Starting from this point, one possible way of designing a feedback controller will be explained. The objective is to guarantee orbital exponential stability in closed loop for a periodic trajectory preplanned as so far presented. In order to present the somewhat simplicity of this design, the key idea introduced in [37] will be explained\(^3\), avoiding entering into deep mathematical details.

One advantage of using virtual holonomic constraints functions, defined as in (2.4), is that it is possible to define a new set of coordinates that can be written as

\[
y_i = q_i - \Phi_i(\theta),
\]

(2.17)

opening the possibility to use different ideas of control to exponentially drive the state \(y\) to zero. Thus, \(y = 0\) implies that the constraint is being kept invariant (i.e. the actuated links are being slaved to the motion by the constraint function). The key idea of [37] is to design a regulator for the auxiliary time-periodic linear system

\[
\dot{\zeta} = A(t)\zeta + B(t)\delta v, \quad \text{where} \quad \zeta = [\delta I, \delta y, \delta \dot{y}]^T,
\]

(2.18)

which describes the linearization of the dynamics transverse to the desired cycle.

In the above equation, \(\delta v\) is a scalar control action related to the second derivative of (2.17) as \(\ddot{y} = v\), \(\delta y\) is the linear part of the deviation from the constraint (2.4) defined as (2.17), \(\delta \dot{y}\) is the linear part of the deviation of \(\dot{y}\) from zero, and the variable \(\delta I\) is the linear part of the deviation from zero of the conservative quantity (2.13)-(2.14). The transverse linearization, and the matrices \(A(t)\) and \(B(t)\), are found using the following results.

---

\(^3\)The author encourage interested readers to get in acquaintance with the work presented in [37] for complementary details, since main equations will be used but not derived.
Theorem 3. With \( x_0 \) and \( y_0 \) being some constants, the time derivatives of the function \( I(\theta, \dot{\theta}, x_0, y_0) \) defined by (2.13)-(2.14), calculated along a solution \([\theta(t), \dot{\theta}(t)]\) of the system (2.5), can be computed as

\[
\frac{d}{dt} I = \dot{\theta} \left\{ \frac{2}{\alpha(\theta)} u - \frac{2\beta}{\alpha(\theta)} I \right\}. \tag{2.19}
\]

Proof. see Appendix E.

The dynamics transverse to a trajectory, defined by a solution \([\theta(t), \dot{\theta}(t)]\) of the reduced dynamics (2.5) is defined as follows,

\[
\frac{d}{dt} I = 2\dot{\theta} \alpha(\theta) \left[ g_y(\cdot)y + g_\dot{y}(\cdot)\dot{y} + g_v(\cdot)v - \beta(\theta) I(\cdot) \right], \tag{2.20}
\]

\[
\ddot{y} = v, \tag{2.21}
\]

where the notation \((\cdot) = (\theta, \dot{\theta}, \ddot{\theta}, y, \dot{y})\).

The \(T\)-periodic matrix functions \(A(t) \in R^{(2n-1) \times (2n-1)}\) and \(B(t) \in R^{(2n-1) \times (n-1)}\) have the form

\[
A(t) = \begin{bmatrix}
a_{11}(t) & a_{12}(t) & a_{13}(t) \\
0_{(n-1) \times 1} & 0_{(n-1) \times (n-1)} & I_{(n-1)} \\
0_{(n-1) \times 1} & 0_{(n-1) \times (n-1)} & 0_{(n-1) \times (n-1)}
\end{bmatrix}, \quad B(t) = \begin{bmatrix}
b_1(t) \\
0_{(n-1) \times (n-1)} \\
I_{(n-1)}
\end{bmatrix},
\]

where the \(T\)-periodic functions \(a_{11}(t), a_{12}(t), a_{13}(t),\) and \(b_1(t)\) are calculated as shown in [37, p. 1169].

If the pair of matrices \(A(t)\) and \(B(t)\) are controllable over the period, there exists a stabilizing solution \(R(t)\) of the Riccati equation

\[
\dot{R}(t) + A(t)^T R(t) + R(t) A(t) + G(t) = R(t) B(t) \Gamma^{-1} B(t)^T R(t), \tag{2.22}
\]

\(\forall t \in [0, T]\), with \(G(t)\) being a \(n\)-by-\(n\) positive definite matrix, and \(\Gamma > 0\), such that the linear system (2.18) is exponentially stabilized by the following feedback law

\[
v = -\Gamma^{-1} B(t)^T R(t) \zeta \tag{2.23}
\]

Finally, it follows from [37, Appendix C, p. 1174] that the input \(u\) for the system, defined by (2.1) is applied according to the feedback transformation

\[
u = \eta(y, \theta)^{-1} \left[ v - \xi(y, \theta, \dot{y}, \ddot{\theta}) \right] \tag{2.24}
\]
2.1. VIRTUAL HOLONOMIC CONSTRAINTS APPROACH

where \( \eta(y, \theta) \) and \( \xi(y, \theta, \dot{y}, \dot{\theta}) \) are functions calculated as follows [37, Appendix C, p. 1174].

Rewriting (2.17) as

\[
q_i = h(y_1, \ldots, y_{n-1}, \theta) + \Phi_i(\theta), \tag{2.25}
\]

by taking the derivatives of the above equation

\[
\dot{q} = L(\theta, y) \begin{bmatrix} \dot{y} \\ \dot{\theta} \end{bmatrix}, \tag{2.26}
\]

\[
\ddot{q} = L(\theta, y) \begin{bmatrix} \ddot{y} \\ \ddot{\theta} \end{bmatrix} + N(\theta, \dot{\theta}, y, \dot{y}), \tag{2.27}
\]

where \( N \) is a \((n \times 1)\) matrix, and \( L \) is defined as

\[
L(\theta, y) = \begin{bmatrix} I_{n-1}, 0_{(n-1) \times 1} \\ grad h \end{bmatrix} + [0_n, \Phi'] \tag{2.28}
\]

where \( grad h = [\partial h/\partial y_1, \ldots, \partial h/\partial y_n, \partial h/\partial \theta] \). Taking the second derivative of \( \ddot{q} \) from (2.9),

\[
\ddot{q} = M(q)^{-1} [B(q)u - C(q, \dot{q})\dot{q} - G(q)], \tag{2.29}
\]

it is possible to equalize the second derivatives from (2.27) and (2.29) as follows

\[
L(\theta, y) \begin{bmatrix} \ddot{y} \\ \ddot{\theta} \end{bmatrix} + N(\theta, \dot{\theta}, y, \dot{y}) = M(q)^{-1} [B(q)u - C(q, \dot{q})\dot{q} - G(q)], \tag{2.30}
\]

and grouping terms, we have

\[
\ddot{y} = \eta(y, \theta)u + \xi(y, \theta, \dot{y}, \dot{\theta}), \tag{2.31}
\]

substituting \( \ddot{y} \) by \( v \), the input to the partial feedback transformation is

\[
u = \eta(y, \theta)^{-1} \left[ v - \xi(y, \theta, \dot{y}, \dot{\theta}) \right].
\]

where the terms of \( \eta(y, \theta) \) and \( \xi(y, \theta, \dot{y}, \dot{\theta}) \) are found as

\[
\eta(y, \theta) = B^\top L^{-1} M^{-1} B,
\]

\[
\xi(y, \theta, \dot{y}, \dot{\theta}) = B^\top L^{-1} \left\{ M^{-1} (-C - G) - N \right\}.
\]
2.2 REAL-TIME IMPLEMENTATION ISSUES

A general overview of real-time implementation will be discussed.

2.2.1 Hardware

For real-time implementation the laboratories of the Control System Group at Umeå University are equipped with the next hardware. Rotation of actuated links are controlled by DC-motors. Angular position in all cases are measured by encoders of different resolutions. In later examples some details according to the particular setup will be added.

Each setup is connected to a dSPACE board 1104 used to run the real-time application. The user interface consists of a PC equipped with ControlDesk (©dSPACE GmbH), used for on-line communication with the dSPACE board. Matlab/Simulink (©Mathworks) is the software used for the implementation of algorithms as well as simulations. Simulink enables the generation and deployment of production C code for use in real-time embedded systems. The compiled code is loaded into the dSPACE’s processor. The program loaded contains a control function, which is executed according to a sampling period.

2.2.2 Sampling

So far, for this design, it has been assumed that disturbances are negligible and that the input signal as well as measured states are realizable at every instant of time. However, the dSPACE board used, as well as any other digital hardware, works with an internal processor which runs according to an internal frequency generator. For this reason the feedback control law must be sampled, meaning that we have to apply for \( t : k h < t \leq (k+1) h \) the control signal

\[
u_r(t) = u[k] \overset{def}{=} \text{Sat}_\psi (u(k h)), \tag{2.32}
\]

where \( k \) is a natural number, the small constant \( h > 0 \) is the sampling time and \( u(k h) \) is the value according to (2.24) at \( t = k h \). \( \text{Sat}_M (\cdot) \) is the standard satu-
ration function with the cut-off level $M = \psi$, which is related to the limitation of the available control torque.

The encoders provide sampled and quantized values. As a consequence, according to the encoders resolution, for $k \leq t \leq (k+1)$ we have available:

$$
\theta[k] = \frac{2 \pi}{\rho_\theta} \left[ \frac{\rho_\theta}{2 \pi} \theta(k \cdot h) \right]_Z, \quad q[k] = \frac{2 \pi}{\rho_\theta} \left[ \frac{\rho_\theta}{2 \pi} q(k \cdot h) \right]_Z,
$$

where $[\cdot]_Z$ denotes the whole part, and $\rho_n$ corresponds to the encoders resolution either at the motor shaft $q$ or the unactuated joint $\theta$.

### 2.2.3 DC motors electromotive torque constant

In our design, the input $u$ to the system described by (2.1) represents the amount of input torque in Newtons meter $Nm$ in the rotational case, and $N$ in the translational one. When DC motors are used, however, the input given by the electronic hardware is often voltage or current. Therefore, it is required to know a priori the electro-mechanical relation between input current and torque produced, in order to apply the correct input level. This relation is known as the electromotive torque constant denoted as $K_{DC}$ and is measured in $Nm/V$. The inverse of this DC gain $K_{DC}$ will be pre-multiplied to the input signal (4.16) in all cases for implementation. Two experimental approaches are proposed for the estimation of this gain $K_{DC}$.

Whenever the actuated link moves in the horizontal plane with no effect of gravity (e.g. Furuta Pendulum, cart pendulum, etc.), the unactuated link can be detached making it possible to isolate the actuated link. The electromotive torque constant can be estimated using a dynamometer attached at the end of the actuated link along the horizontal plane. By applying input voltage of different levels the corresponding force can be measured. The torque is calculated by multiplying this force by the distance from the motor axis to the end of the link. It is commonly the case that the relation between them follows a linear function from which the slope represents $K_{DC}$ (an example of this will be shown later).

---

$^4$Actually, there is also some smoothing done by the Simulink interface.
The second case is when the actuated link hangs around the vertical axis, and so the influence of gravity is present. If the unactuated link is detached it is possible to analyze the motor and link. According to Newton’s second law, the mechanical system follows a relation described by

\[ J\ddot{q}(t) = K_{DC}u(t) - \zeta \sin(q(t)), \]

where \( J \) corresponds to the inertia of the link and \( \zeta \) denotes the weight of the link in proportion to the distance to the center of mass (it can be always calculated). By applying a ramp signal of very slow slope (i.e. \( \dot{q}(t) \approx 0 \)) it is possible to rewrite the previous relation as

\[ 0 = K_{DC}u(t) - \zeta \sin(q(t)), \]

and by having the calculated value of \( \zeta \), the electromotive torque constant can be expressed as

\[ K_{DC} = \frac{\zeta \sin(q(t))}{u(t)}. \]

### 2.2.4 Velocity estimation

Since encoders provide the position states of the setup, a way to estimate the velocity states will be proposed. For the feedback control law, estimates of angular velocities are required. The estimation can be done applying a standard high-gain observer [24].

A high gain observer (HGO) is a fast nonlinear (or linear) full order observer with high observer gains chosen via pole placement. For this differentiating scheme, saturation is needed in order to protect the plant from peaking [24]. Applying [24, Sec. 14.5]

\[
\begin{bmatrix}
\hat{\dot{x}}_1 \\
\hat{\dot{x}}_2
\end{bmatrix} = \begin{bmatrix}
\hat{\dot{v}}_1 \\
\hat{\dot{v}}_2
\end{bmatrix} + \begin{bmatrix}
2(y_1 - \hat{x}_1)/\varepsilon_1 \\
2(y_2 - \hat{x}_2)/\varepsilon_2
\end{bmatrix},
\]

\[
\begin{bmatrix}
\hat{\dot{v}}_1 \\
\hat{\dot{v}}_2
\end{bmatrix} = f_0(\hat{x}_1, \hat{x}_2, \hat{\dot{q}}, \hat{\dot{\theta}}) + \begin{bmatrix}
(y_1 - \hat{x}_1)/\varepsilon_1^2 \\
(y_2 - \hat{x}_2)/\varepsilon_2^2
\end{bmatrix},
\]

\[
\hat{\dot{q}} = \text{Sat}_{M_1} (\hat{\dot{v}}_1), \quad \hat{\dot{\theta}} = \text{Sat}_{M_2} (\hat{\dot{v}}_2),
\]

(2.34)
where the parameters $\varepsilon_{1,2} > 0$ are to be tuned,

$$
\begin{bmatrix}
    y_1 \\
    y_2
\end{bmatrix} =
\begin{bmatrix}
    q \\
    \theta
\end{bmatrix} + w(t)
$$

with

$$
w(t) =
\begin{bmatrix}
    q[k] - q(t) \\
    \theta[k] - \theta(t)
\end{bmatrix}
$$

is the measured output with the ‘measurement noise’ defined using (2.33) $\forall t : kh < t \leq (k + 1) h$, and either

$$
f_0(q, \theta, \dot{q}, \dot{\theta}) = 0
$$

or $f_0(q, \theta, \dot{q}, \dot{\theta})$ is a nominal model defined by (2.1).

In the discretized case, if the sampling period is set to be $h = 0.00085$, then the observer parameters are taken as

$$
\varepsilon_1 = 20 h \quad \text{and} \quad \varepsilon_2 = 10 h,
$$

while $M_1 = M_2 = 20 \text{ rad/s}$, which corresponds to a reasonable level of upper bounds on the speeds.

### 2.2.5 Friction Identification and Compensation

The friction torque can be modeled as a nonlinear static mapping. Its approximation can be used for partial compensation of the friction present in the actuated link. The friction compensation adopted in this thesis considers the Coulomb and viscous friction model, which can be expressed as

$$
\begin{align*}
    u^q_{friction} &= u^q_c + u^q_v
\end{align*}
$$

where $u^q_c$ denotes the Coulomb friction and $u^q_v$ the viscous friction. The Coulomb friction [19] is empirically simple, at least as a first approximation, since it depends on the algebraic sign of the velocity when there is relative motion,

$$
\begin{align*}
    u^q_c &= F^q_c \cdot \text{sign}(\dot{q}),
\end{align*}
$$

In the 19th century the theory of hydrodynamics was developed leading to expressions for the friction force caused by the viscosity of lubricants. The term viscous friction is used for this phenomena, and is normally described as

$$
\begin{align*}
    u^q_v &= b_q \cdot \dot{q},
\end{align*}
$$

\[5\text{In our case it is always the case that implementations are made at a sampling period } T = 0.0008 \text{ s.}\]
CHAPTER 2. PRELIMINARIES

A combination of these two friction phenomena can be plotted as shown in Fig. 2.6, where the x-axis represents the velocity \( \dot{q} \) and the y-axis represents the input torque or voltage level.

![Figure 2.6: Coulomb + viscous friction.](image)

The friction identification techniques are divided in two stages to identify each component of (2.36). Estimation of the velocity is done applying (2.34)-(2.35).

First, a ramp signal of some velocity is given as input signal. The level of Coulomb friction corresponds to the value at which the system starts showing some motion. Since in many practical cases it is not expected to have symmetric Coulomb friction, we should change (2.37) to

\[
u^q_c = \begin{cases} 
    u^q_{c,+}, & \text{if } \dot{q} > 0 \\
    u^q_{c,-}, & \text{if } \dot{q} < 0
\end{cases} \quad (2.39)
\]

Second, by applying step signals of different amplitudes (this could be done in closed loop as well), a map relating input voltage with link velocity can be done as seen in Fig. 4.9. The slope of this linear approximation represents the \( b_q \) component of the viscous friction (2.38).

This linear mapping of friction can be given as a feed-forward term to the controller for friction compensation [19] as shown in Fig. 2.7.
2.2. REAL-TIME IMPLEMENTATION ISSUES

2.2.6 Numerical Solution of the Riccati Equation

On-line solution of (2.22) is not feasible due to the complexity of calculations. Therefore a numerical method approach needs to be consider in such a case. In this work, the periodic Riccati equation (2.22) is solved numerically applying the technique presented in [43, Theorem 2]:

- Solve the following initial value problem:

\[
\dot{Z}(t) = \begin{bmatrix} 0_3 & I_3 \\ -I_3 & 0_3 \end{bmatrix} \begin{bmatrix} -G(t) & A^T(t) \\ A(t) & B(t)B^T(t)/\Gamma \end{bmatrix} Z(t), \quad Z(0) = I_6,
\]

where \( I_n \) is the \( n \times n \) identity matrix and \( 0_n \) is the \( n \times n \) zero matrix, over the period.

- Let the columns of the \( 6 \times 3 \) matrix \( Z_0 \) be the basis vectors of the invariant stable subspace in the kernel of \( Z(T) \) and solve the linear matrix ordinary differential equation

\[
\begin{bmatrix} \dot{X}_1(t) \\ \dot{X}_2(t) \end{bmatrix} = \begin{bmatrix} 0_3 & I_3 \\ -I_3 & 0_3 \end{bmatrix} \begin{bmatrix} -G(t) & A^T(t) \\ A(t) & B(t)B^T(t)/\Gamma \end{bmatrix} \begin{bmatrix} X_1(t) \\ X_2(t) \end{bmatrix} ,
\]

initiated at \( \begin{bmatrix} X_1(0) \\ X_2(0) \end{bmatrix} = Z_0. \)
• Compute the stabilizing solution of (2.22) as

\[ R(t) = -X_2(t)X_1^{-1}(t). \]

The necessity to accurately solve a linear matrix periodic differential equation with the same number of stable and unstable eigenvalues of the transition matrix is a challenging numerical problem. Some more numerically reliable techniques are under consideration [18, 22].

The weighting matrix \( G(t) \) in some cases can be treated as a time variant matrix of the form

\[ G(t) = \text{diag}\{f^*(t), \kappa_y, \kappa_{\dot{y}}\}, \]

where the function \( f^*(t) \) is defined as

\[ f^*(t) = \frac{1}{2} \sqrt{\dot{\theta}_z^2 + \ddot{\theta}_z^2}, \quad (2.40) \]

and is related to an approximation for the coefficient of proportionality between \( I(\theta, \dot{\theta}, \theta_0, \dot{\theta}_0) \) and the distance from any particular point \((\theta, \dot{\theta})\) to the desired cycle [38]. The values for \( \kappa_y \) and \( \kappa_{\dot{y}} \) can be chosen somehow arbitrarily according to the application (as it is done in LQ design).

### 2.2.7 Projection of Riccati Solution

In order to use the stabilizing solution of \( R(t) \) for any \( \theta(t) \) closed to the desired trajectory \( \theta^*(t) \), we have to introduce an operator which projects the points of \([\theta, \dot{\theta}]\) onto a curve defined by the trajectory \([\theta^*, \dot{\theta}^*]\).

Considering the case shown in Fig. 2.8, if we define the angle \( \varsigma \) as the location of states along the trajectory defined by \( \theta^* \) and \( \dot{\theta}^* \) as,

\[ \varsigma = \tan^{-1} \left( \frac{-\dot{\theta}}{\theta} \right), \quad (2.41) \]

where \( \varsigma \) gives the angular location of the states \((\theta, \dot{\theta})\) with respect of the trajectory. Considering that the trajectory has a clockwise direction, the value of (2.41) will be
positive within the range \{0, \pi\}, while from \{\pi, 2\pi\} will be negative. Therefore, the projected time\(^6\) can be defined as

\[
t = \begin{cases} 
\frac{\varsigma}{2\pi/T}, & \text{if } \varsigma > 0 \\
\frac{\varsigma + 2\pi}{2\pi/T}, & \text{if } \varsigma < 0
\end{cases}
\]

(2.42)

where the period \(T\) depends on the trajectory chosen from (2.5) or calculated by (2.16).

2.2.8 Model Parameter Identification

In many cases, parameters of models will need to be identified throughout experiments. Many different methods can be considered at this stage. Nonlinear least square method gives close approximations to real parameters (real in the sense of closeness to calculated values from CAD models), and is simple in the programming sense. The Optimization Toolbox from Matlab/Simulink can be applied in such a case. Later some examples will be given.

\(^6\)This projection can be only used when the trajectory is around one particular equilibrium. For the case when trajectories encircle more than one equilibrium a different projection should be chosen.
Part III

Design of Periodic Motions for the Inertia Wheel Pendulum
In this chapter an analysis of the virtual holonomic constraints approach for the case of the inertia wheel pendulum will be fully worked out. The main idea is to exploit the design technique presented in chapter 2.1.1, in order to plan a class of stabilizable periodic motions in the system and achieve the experimental validation of the feedback control design. Performance will be analyzed mainly by experimental results.

Here we deal with a particular example of the class of controlled mechanical system with under-actuation degree one. The system consists of a free planar rotational pendulum and a symmetric disk, attached to its end and directly controlled by a DC-motor. The main theoretical contribution is applying the method, presented so far, to tune the parameters of a linear holonomic constraint, to generate oscillations of a given amplitude and period, clarifying specially the use of (2.16).
CHAPTER 3. INERTIA WHEEL PENDULUM

3.1 Inertia wheel pendulum dynamics

Dynamics of an inertia wheel pendulum (see Fig. 3.1) can be described [2, 3] by the system of two differential equations, originating from Lagrangian mechanics

\[
\begin{align*}
\mathbf{p}_1 & \ddot{\theta} + \mathbf{p}_2 \ddot{\varphi} + \mathbf{p}_3 \sin \theta &= 0, \\
\mathbf{p}_2 & \ddot{\theta} + \mathbf{p}_2 \ddot{\varphi} &= u,
\end{align*}
\]

(3.1)

where \( \theta \) is the absolute angle of the pendulum, counted clockwise from the vertical downward position; \( \varphi \) is the absolute angle of the disk; \( p_1, p_2, p_3 \) are positive physical parameters, depending on the geometric dimensions and the inertia-mass distribution (we assume that \( p_1 = 4.59695 \cdot 10^{-3} \), \( p_2 = 2.495 \cdot 10^{-5} \), and \( p_3 = 0.35481 \), see [2]); and \( u \) is the controlled torque, applied to the disk.

![Figure 3.1: Schematic diagram of the system.](image)

The objective is stated as follows: make the pendulum and wheel to follow a linear constraint functions to achieve stable oscillations of predefined amplitude and period.
3.2 Motion planning

Going step by step as presented in chapter 2.1.1 we have,

**Step 1: Choice of a holonomic constraint**

Since we are interested in defining a special motion of the pendulum it makes sense to introduce the constraint

\[ \varphi = f(\theta) \overset{def}{=} k\theta + \varphi_0, \]  

(3.2)

where \( f(\theta) \) is a twice continuously differential function, and \( \varphi_0 = -k\bar{\theta} \) and \( \bar{\theta} \in \{0, \pi\} \).

**Step 2: Computation of the reduced-order dynamics**

The dynamics, projected into this constraint (under assumption that there exists a control law that makes it invariant) can be described by

\[ \alpha(\theta_*) \ddot{\theta}_* + \beta(\theta_*) \dot{\theta}_*^2 + \gamma(\theta_*) = 0, \]  

(3.3)

which is always integrable, provided \( \alpha(\theta_*) \neq 0 \) [37], and

\[ \begin{align*}
\alpha(\theta) &= p_1 + p_2 f'(\theta), \\
\beta(\theta) &= 2p_2 f''(\theta), \\
\gamma(\theta) &= p_3 \sin(\theta),
\end{align*} \]  

(3.4)

Applying the constraint function defined by (3.2), the reduced-order dynamics (3.3) takes the form of the dynamics of the standard uncontrolled physical pendulum

\[ (p_1 + kp_2) \ddot{\theta}_* + p_3 \sin \theta_* = 0 \]  

(3.5)

**Step 3: Search for an achievable periodic solution**

It can be shown that (3.3) has infinitely many equilibria, solutions of \( \gamma(\theta) = 0 \), given by the formula

\[ \theta = \theta_i \overset{def}{=} \pi i, \]
where \( i \) is an arbitrary integer. These equilibria correspond to the upward equilibrium if \( i \) is odd and to the downward equilibrium if it is even. It is shown in [37] that the equilibrium is a center, provided
\[
\omega^2 \overset{\text{def}}{=} \frac{\gamma'(\theta_i)}{\alpha(\theta_i)} = \frac{(-1)^i p_3}{p_1 + p_2 f'(\pi i)} > 0
\]  
and a saddle, provided the inequality above is reversed. It is not hard to see from (3.6) and (3.2), that for oscillations around the downward equilibrium, \( \bar{\theta} = 0 \), we should take
\[
k > -\frac{p_1}{p_2}
\]
and for oscillations around the upward equilibrium, \( \bar{\theta} = \pi \), we need
\[
k < -\frac{p_1}{p_2}.
\]

**Step 4: Periodicity of the trajectory**

In order to compute the period of oscillations for a given periodic solution, we have to solve the nonlinear equation
\[
I(x, 0, \bar{\theta} + A, 0) = \int_{\bar{\theta} + A}^{x} \exp \left\{ - \int_{s}^{x} \frac{2 \beta(\tau)}{\alpha(\tau)} d\tau \right\} \frac{\gamma(s)}{\alpha(s)} ds = 0
\]
for the biggest \( x \) in the open half-plane \( x < \bar{\theta} \) and with this solution calculate (for an idea see, e.g., [24, Exercise 2.18])
\[
T = \sqrt{2} \int_{x}^{\bar{\theta} + A} \frac{d\tau}{\sqrt{I(x, 0, \tau, 0)}}.
\]

The conserved quantity applying (2.15)-(2.16) can be written as
\[
I(\theta_*, \dot{\theta}_*, x_0, y_0) = \left( E(\theta_*, \dot{\theta}_*) - E(x_0, y_0) \right) / (p_1 + k p_2),
\]
where
\[
E(\theta, \dot{\theta}) = \frac{(p_1 + k p_2)}{2} \dot{\theta}^2 + p_3 (1 - \cos(\theta))
\]  
(3.7)
is the total energy of the pendulum, described by (3.5). We can choose any amplitude of oscillations from the interval:
\[
0 < A < \pi,
\]
3.3. CONTROLLER DESIGN

$x = \bar{\theta} - A$ (see Fig. 2.5) and the period of oscillations is equal to

$$T = C(\bar{\theta}, A) \sqrt{\frac{2 |p_1 + kp_2|}{p_3}},$$

where

$$C(\bar{\theta}, A) \overset{\text{def}}{=} \int_{\bar{\theta}-A}^{\bar{\theta}+A} \frac{d\tau}{\sqrt{\cos(\tau) - \cos(\bar{\theta} - A)}}$$

(3.8)
can be expressed in terms of an elliptic integral of the first kind. Therefore, we should take

$$k = \left( \frac{T^2 p_3}{2 C^2(0, A)} - p_1 \right) / p_2$$

(3.9)

for the case of $\bar{\theta} = 0$ and

$$k = \left( - \frac{T^2 p_3}{2 C^2(\pi, A)} - p_1 \right) / p_2$$

(3.10)

for the case of $\bar{\theta} = \pi$.

Let’s consider the case of downward motion. Applying the calculations presented above it is required to find the constant $k$, so that a periodic trajectory with $A = 3/4\pi$ and $T = 1.1$ is planned\(^1\). One way to do that is by calculating the coefficient $k$ from (3.9) and (3.8), which results in $k = 2.4507$. In Fig. 3.2 results of this design procedure are shown.

Considering the case of upright oscillations. It would be desired to achieve oscillations with an amplitude $A = \pi/36$ and period $T = 0.48$. Following the design procedure it is concluded that $k = -267.51$. In Fig. 3.3 phase portraits and desired trajectories for this case are shown.

### 3.3 Controller design

Defining the change of variables as shown in (2.17) $(\varphi, \dot{\varphi}, \theta, \dot{\theta}) \mapsto (y, \dot{y}, \theta, \dot{\theta})$, where $y$ is the deviation from the constraint (3.2):

$$y = \varphi - \varphi_0 - k \theta.$$

\(^1\)The amplitude and period of oscillations are chosen in such a way that motor limitations are not reached
Differentiating (3.7) along the trajectories of (3.2), we obtain the classical passivity relation

\[
\frac{d}{dt} E(\theta, \dot{\theta}) = -p_2 v \dot{\theta}.
\] (3.11)

Defining the auxiliary time variant linear system (2.18) as,

\[
\dot{\zeta} = A \zeta + B(t) v, \quad \zeta = \begin{bmatrix} E - E_0 \\ y \\ \dot{y} \end{bmatrix},
\] (3.12)
3.3. CONTROLLER DESIGN

Figure 3.3: Results of the motion planning for upright oscillations: (a) Desired phase portrait \( \theta(t) \) vs. \( \dot{\theta}(t) \), (b) Desired trajectory for \( \theta(t) \) as function of time, (c) Ideal constraint function \( \phi \) vs \( \theta \), (d) Desired phase portrait \( \phi(t) \) vs. \( \dot{\phi}(t) \).

where the matrices \( A(t) \) and \( B(t) \) are calculated as presented in [37, p. 1169],

\[
A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad \text{and} \quad B(t) = \begin{bmatrix} -p_2 \dot{\theta}_*(t) \\ 0 \\ 1 \end{bmatrix}.
\]

It is known that for any matrix \( G = G^T \geq 0 \) and scalar \( \delta \geq 0 \), there exists a positive definite solution \( R(t) \) of the matrix periodic Riccati equation (2.22). Moreover, the feedback control law (2.23) defined as

\[
v = - \begin{bmatrix} -p_2 \dot{\theta} & 0 & 1 \end{bmatrix} R(t) \begin{bmatrix} E(\theta, \dot{\theta}) - E_0 \\ \phi - \varphi_0 - k \theta \\ \dot{\phi} - k \dot{\theta} \end{bmatrix}, \tag{3.13}
\]
ensures that the desired trajectory, $\theta_*(t)$, is an exponentially orbitally stable solution of the closed-loop system (3.1), (3.2).

After straightforward calculations, following the steps presented in chapter 2.1.1, proceeding with partial feedback linearization and the corresponding control transformation (2.30)-(2.31), we obtain the new control input (2.24) defined by

$$u = -\frac{p_2}{p_1 + kp_2}\left((p_3 + kp_2)\sin(\theta) + (p_2 - p_1)v\right),$$  

(3.14)

### 3.4 Experimental Results

The inertia wheel used for experiments is an adaptation of the Mechatronics Control Kit (see Fig. 3.4) from Mechatronic Systems, Inc (for a detailed documentation see [33]). Physical parameters are used as mentioned in section 3.1. The setup is connected to a dSPACE board 1104 used to run the real-time application.

![Figure 3.4: Inertia Wheel Pendulum.](image-url)
3.4. EXPERIMENTAL RESULTS

3.4.1 Friction Compensation

As explained earlier, the friction compensation adopted for implementation considers Coulomb and viscous friction. After some identification tests, it is concluded that the friction torque for the motor can be compensated applying the next input voltage,

\[
u_F = \begin{cases} 
0.03 + 0.001 \cdot \dot{\phi}, & \text{if } \dot{\phi} > 0 \\
-0.032 + 0.001 \cdot \dot{\phi}, & \text{if } \dot{\phi} < 0
\end{cases}
\]  

(3.15)

3.4.2 DC motors electromotive torque constant

Parameters of the setup mentioned in section 3.1, are a copy from the original user manual [33]. Due to reliability on the product and model parameters it is enough to apply least square methods to identify the electromotive torque constant. Velocity in such a case is estimated as presented in section 2.2.4.

If the pendulum is fixed, then dynamics of the motor can be derived from (3.1) as

\[p_2 \ddot{\phi} = K_{DC} \hat{u}, \]  

(3.16)

where the input expressed by (3.14) is related to (3.16) by

\[u = K_{DC} \hat{u}. \]  

(3.17)

Considering that dynamics of the DC motor are affected by frictional forces (3.15), we can rewrite (3.16) as

\[p_2 \ddot{\phi} = K_{DC} \hat{u} - K_{DC} b_\varphi \dot{\phi}. \]  

(3.18)

where \( b_\varphi \) denotes the viscous friction coefficient identified previously, and values can be read from (3.15).

Since the equation above is open loop unstable in respect to the position \( \varphi \), we could for example apply a feedback controller such that

\[\hat{u} = K_p (r - \varphi), \]  

(3.19)
and so it is possible to write the system (3.18) in the closed loop form

\[ p_2 \ddot{\varphi} = K_{DC} K_p (r - \varphi) - K_{DC} b_\varphi \dot{\varphi}, \quad (3.20) \]

which by opening brackets and applying the Laplace transform results in the next transfer function, that relates the reference input with the angular position \( \varphi \)

\[ \frac{\varphi}{r} = \frac{K_{DC} K_p}{p_2 \left( s^2 + \frac{K_{DC}}{p_2} s + \frac{K_{DC} K_p}{p_2} \right)}, \quad (3.21) \]

which resembles a second order transfer function of the form

\[ \frac{\varphi}{r} = \frac{\hat{b}_0}{s^2 + \hat{a}_1 s + \hat{a}_0}, \quad (3.22) \]

where \( \hat{b}_0 = \hat{a}_0 = \frac{K_{DC} K_p}{p_2} \).

By varying the proportional gain \( K_p \) and inducing different input signals it is possible to apply identification methods to identify the transfer function (3.22). With this procedure it was estimated that \( K_{DC} = 0.049^2 \).

### 3.4.3 Oscillations Around Downward equilibrium

In order to solve the Riccati equation (2.22), the matrix \( G \) was defined as

\[ G = \text{diag}([1, 0.5, 0.25]). \quad (3.23) \]

The software was further developed to include all the previous considerations of friction compensation, input signal, DC gain inversion, etc. In Fig 3.5 the experimental results for downward oscillations are presented.

The experimental results (see Fig. 3.5) show that the proposed method is successful in practice. In fact, the desired cycle is closely achieved. Irrespective of variation in initial conditions, experiments show reproducibility and stability.

The mismatch between experimental and desired experiments, e.g. Fig. 3.5(b), is thought to be produced by the following factors: (a) the friction torque at the pendulum shaft is not compensated, (b) discretization of the measurements does

\[ ^2\text{Reference [2] proposes an alternative approach of identification, but in either case the gain is closely similar.} \]
3.4. EXPERIMENTAL RESULTS

not allow us to use sufficiently fast differentiators and leads to introduction of additional delays induced by our observers, and (c) there are presumably input and output delays present in the hardware which are not taken into account in the design. However, it is experimentally verified that the steady-state in the experiment will not be destroyed by these factors. This validates the robustness expected from the design [37].
3.4.4 Oscillations Around upright equilibrium

Same as in the previous case, matrix $G$ is kept same to solve the Riccati equation. In Fig. 3.6 the experimental results are shown. In Fig. 3.6(b) it is seen that the constraint function is fulfilled quite accurate despite the phase portraits (Fig.3.6(a) and Fig.3.6(c)) are not achieved closely. Reasons for this variations are thought to be same as in the downward motion case. Despite of not achieving perfectly the desired cycle, the oscillatory behavior is closely achieved.

Figure 3.6: Results of experimental validation for upright oscillations: (a) Desired vs. Experimental validation of $\theta(t)$ vs. $\dot{\theta}(t)$, (b) Ideal vs. experimental constraint function $\phi$ vs $\theta$; (c) Desired vs. Experimental validation of $\phi(t)$ vs. $\dot{\phi}(t)$, (d) Input signal $u(t)$. 
3.5 Conclusions

This example had the intention to show how the mathematical steps presented in previous chapters can be followed to design periodic motions. Despite of the simplicity of the system dynamics, this type of design can be extended to different type of un-actuated systems. It was shown effectiveness of the method to accurately find the coefficients of a constraint function in order to achieve some desired period and amplitude. This is a very important concept since in many cases, e.g. walking, periodic motions are performed with specific characteristics of amplitude and period.

In the case of implementation, it was shown that taking care of details, such as nonlinearities and accurate states estimation, can help to achieve successful results. In any case, the controller designed also relies on the property of robustness to modeling errors and uncertainty of the states estimation.

The implementation of such algorithm though, relies very much on accurate numerical methods. As explained earlier, the Riccati equation cannot be solved on-line. Therefore the off-line computation of its solution is approximated to polynomial of satisfactory order (30 in our case) for software implementation.

In the next example some more discussions will be given for the case of a more complex system, where numerical methods are more involved.
Part IV

Can we make a pendulum to swing up by following a predefined trajectory? Virtual Holonomic Constraint Design
4.1 Introduction

For decades inverted pendulums have been benchmarking setups for testing advanced control design methods. They have been commonly used in the education of mechanical/electrical/control engineers, by performing experimental studies. Stabilization around an upright unstable equilibrium and swinging up to reach its vicinity could be mentioned as most popular and challenging assignments.

Local stabilization of an unstable equilibrium often admits at least one solution. It is often found using a linearization of the system dynamics around the equilibrium. A controller can be designed to stabilized the linearized model. However, the task of swinging-up has no such an apparent solution, and requires accurate motion planning to reach a vicinity of the chosen equilibrium from a far located
initial point (typically representing the system at rest).

To solve the problem various arguments have been suggested for a number of popular setups: for the Furuta pendulum [9, 15], for the Pendubot [8, 25, 49], for the pendulum on a cart [14, 34, 35], for the Acrobat [30], for the inertia wheel pendulum [40], etc. Most of these approaches use explicitly the hyperbolic structure of the target equilibrium in the following way: open-loop control input is chosen to reshape a stable manifold of a hyperbolic equilibrium to be a homoclinic curve (or a family of homoclinic curves) of a modified dynamics.

This idea is useful because:

- the closure of a homoclinic curve is a compact set in the state space of the system, and

- stabilizing this set pushes the solutions of the closed loop system to visit any neighborhood of the equilibrium infinitely many times.

These general arguments have been applied in planning a very particular class of homoclinic structures created by an appropriate control input. Typically all the degrees of freedom, except one, are kept at target positions, while the remaining degree of freedom follows motions of a physical pendulum. The stable manifold of the upright hyperbolic equilibrium of a physical pendulum consists of two homoclinic curves. They admit the simple description in terms of the original generalized coordinates and velocities: all coordinates, except one, should be equal to some desired constants, their velocities should be zero, the total energy of the system should be equal to the level of energy at the desired equilibrium. As a result, the problem of swinging up gets solved by stabilization of these quantities.

The described method has been successfully validated in simulations and supported by experiments. Nevertheless, this method does not take fully advantages and possibilities of swinging up by creating different homoclinic structures. This turns out to be important for various reasons:

- By stabilizing the homoclinic curves of a physical pendulum, the number of rotations required for reaching a vicinity of the equilibrium, where a local stabilizing controller is activated, cannot be accurately controlled. Experimental setups have often a restricted travel even for the revolute joints. As
4.1. INTRODUCTION

an example, think of cables to an encoder installed on the second joint of the double link pendulum, where only a few rotations of the first link would be allowed before reaching the limits.

- It is not known how to improve the rate of convergence to homoclinic curves of the physical pendulum. When all (except one) degrees of freedom are kept in vicinities of stationary points, controlling them to pump up energy into oscillations of the remaining degree of freedom to recover the behavior of a physical pendulum might be difficult. Limited performance in compensating stiction parts of friction in actuated joints could deteriorate the overall performance, and do not allow to reach a vicinity of the equilibrium where a local stabilizing controller is activated. As known, friction is easier to compensate if joints are in motion.

These and other arguments motivate developing new ideas for planning homoclinic orbits and their orbital stabilization for swinging up inverted pendulums.

This paper suggests such an approach for solving the swing-up problem with the Furuta pendulum taken as example. It is shown how to plan a large family of orbits taking into account the constraints of limited travel. Furthermore, it is shown how to stabilize nearby periodic motions to improve the rate of convergence. The results are supported by simulations, and by successful experimental studies.

The paper is organized as follows: this section is continued by describing the Furuta pendulum model and a formulation of the objectives of the study as well as some preliminary steps to solve the problem. The main theoretical contribution of this work is described in Section 4.2. Section 4.3 presents results of computer simulations, while the results of experiments together with some details of implementation are described in Section 4.4. Concluding remarks are given in Section 4.5.

4.1.1 Preliminaries: Dynamics of the Furuta Pendulum

The Furuta pendulum is a mechanical system with two degrees of freedom, see Fig. 4.1, where \( \phi \) is used to denote the angle of the arms rotation in the horizontal
Figure 4.1: The Furuta Pendulum built at the Department of Applied Physics and Electronics, Umeå University.

plane, and \( \theta \) the angle of the pendulum attached to the end of this arm. The behavior of the pendulum is influenced by the acceleration of the arm, which is actuated by a DC motor. The equations of motion of the Furuta pendulum are [36],

\[
(p_1 + p_2 \sin^2 \theta) \cdot \ddot{\phi} + p_3 \cos \theta \cdot \ddot{\theta} + 2p_2 \sin \theta \cos \theta \cdot \dot{\theta} \cdot \dot{\phi} - p_3 \sin \theta \cdot \dot{\theta}^2 = \tau_{\phi} \quad (4.1)
\]

\[
p_3 \cos \theta \cdot \ddot{\phi} + \left(p_2 + p_5\right) \cdot \ddot{\theta} - p_2 \sin \theta \cos \theta \cdot \dot{\theta}^2 - p_4 \sin \theta = 0 \quad (4.2)
\]

where \( \tau_{\phi} \) is the external torque that allows to control the arm rotation. The constants \( p_1 - p_5 \) are positive and defined by physical parameters of the setup as follows:

\[
p_1 = J + (M + m_p)l_a^2 \ [kg \cdot m^2], \quad p_2 = (M + \frac{1}{2} m_p)l_p^2 \ [kg \cdot m^2] \]
\[
p_3 = (M + \frac{1}{2} m_p)l_p l_a \ [kg \cdot m^2], \quad p_4 = (M + \frac{1}{2} m_p)l_p g \ [Nm], \quad p_5 = \frac{1}{12} m_p l_p^2 \ [kg \cdot m^2].
\]
4.1. INTRODUCTION

Here \( l_a \) is the length of the arm, \( m_a \) is the mass of the arm, \( J \) is the arms inertia, \( l_p \) is the length of the pendulum, \( m_p \) is the mass of the pendulum, \( M \) is the mass of the bob at the end of the pendulum and \( p_5 \) is the inertia of the pendulum around the center of mass.

4.1.2 Preliminaries: Problem Formulation

The objectives are the following:

- Plan feasible motions of the Furuta pendulum to bring the pendulum from a vicinity of the downward equilibrium to a sufficiently small neighborhood of the upright unstable equilibrium;

- Design a feedback controller to orbitally stabilize a suitable preplanned motion.

4.1.3 Preliminaries: Planning a Motion via Imposing Constraint on \( \phi \) and \( \theta \)

Describing all the possible swing-up motions of the Furuta pendulum as a response to some smooth control torques, at first glance, seems to be an intractable task. The next arguments, however, give some insights into searching some of these motions. Suppose that the motion \([\phi_*(t), \theta_*(t)]\) of (4.1)-(4.2) with \( \tau_\phi = \tau_{\phi*}(t) \) is a homoclinic curve of \( \theta \)-dynamics defined for all time moments between \( \pm \infty \), and such that the next two conditions hold

I: \( \lim_{t \to +\infty} \theta_*(t) = 0 \) and \( \lim_{t \to -\infty} \theta_*(t) = 0 \)

II: \( \lim_{t \to +\infty} \phi_*(t) = \phi_{0*} \) and \( \lim_{t \to -\infty} \phi_*(t) = \phi_{0*} \).

Suppose in addition that there exists a \( C^2 \)-smooth function \( \Phi(\cdot) \), such that

\[
\Phi_*(t) = \Phi_*(\theta_*(t)), \quad \forall t.
\]

(4.3)

The first and second time derivatives of \( \phi_*(t) \) are then

\[
\frac{d}{dt} \phi_*(t) = \Phi'_*(\theta_*(t)) \frac{d}{dt} \theta_*(t), \quad \frac{d^2}{dt^2} \phi_*(t) = \Phi''_*(\theta_*(t)) \left[ \frac{d}{dt} \theta_*(t) \right]^2 + \Phi'_*(\theta_*(t)) \frac{d^2}{dt^2} \theta_*(t).
\]
By substitution of these relations and (4.3) into equation (4.2) of the system dynamics, we obtain the second order differential equation

\[
\begin{aligned}
(p_2 + p_5 + p_3 \cos \theta_* \cdot \Phi_*'(\theta_*)) \ddot{\theta}_* + \\
+ \left( p_3 \cos \theta_* \cdot \Phi_*''(\theta_*) - p_2 \sin \theta_* \cos \theta_* \cdot [\Phi'(\theta_*)]^2 \right) \dot{\theta}_*^2 + \\
+ p_4 \sin(-\theta_*) = 0
\end{aligned}
\]

The homoclinic curve(s) \( \theta_*(t) \), described earlier, is/are the solution(s) of this equation. Suppose that neither the function\(^1\) \( \Phi_*(\cdot) \) nor the particular solution – homoclinic curve \( \theta_*(t) \) – are known. The contribution of this paper is to show that existence of homoclinic curves for dynamics of (4.4) can be deduced from a few properties of the coefficients \( \alpha(\cdot) \), \( \beta(\cdot) \) and \( \gamma(\cdot) \). As a result, shaping new homoclinic structures for elaborating swinging-up strategies for the Furuta pendulum becomes feasible.

4.2 Main Results

4.2.1 Planning New Homoclinic Curves for Swing Up

To start the search for swing up motions, consider the relation

\[
\phi(t) = \Phi(\theta(t)), \quad \forall t
\]

which is supposed to be satisfied along this motion, and where the scalar function \( \Phi(\cdot) \) is unknown. If (4.5) is valid, it immediately implies the next relations between angular velocities and acceleration

\[
\frac{d}{dt} \phi = \Phi'(\theta) \frac{d}{dt} \theta, \quad \frac{d^2}{dt^2} \phi = \Phi''(\theta) \left[ \frac{d}{dt} \theta \right]^2 + \Phi'(\theta) \frac{d^2}{dt^2} \theta.
\]

\(^1\)The case of physical pendulums, discussed in the introduction, fits naturally into these settings, with \( \Phi_*(\cdot) \) being equal to the constant \( \phi_{0*} \).
4.2. MAIN RESULTS

Substituting these relations in (4.2), we obtain the second order differential equation (4.4). Important facts on qualitative behavior of its solutions are given in the next statements.

Lemma 1 Let \( \Phi(\cdot) \) be a \( C^2 \)-smooth function defined on \( I_\Phi = (-2\pi - \varepsilon, 2\pi + \varepsilon) \) with \( \varepsilon > 0 \), such that the function \( \alpha(\theta) = p_2 + p_5 + p_3 \cdot \cos \theta \cdot \Phi'(\theta) \) is separated\(^2\) from zero on \( I_\Phi \), then

1) The equilibriums of the dynamical system (4.4), i.e. the solutions of \( \gamma(\theta_{eq}) = 0 \), are

\[ \theta_{eq} = \pi \cdot k, \quad k = -2, -1, 0, 1, 2 \]  \hspace{1cm} (4.6)

2) Consider the equilibrium \( \theta = k\pi, k \in [-2, -1, 0, 1, 2] \), of the dynamical system (4.4). If the constant

\[ \omega_k := \left. \frac{-p_4 \cos \theta}{p_2 + p_5 + p_3 \cdot \cos \theta \cdot \Phi'(\theta)} \right|_{\theta = k\pi} \]  \hspace{1cm} (4.7)

is positive, then this equilibrium is a center. If, in opposite, \( \omega_k \) is negative, then this equilibrium is a saddle.

3) Suppose \( \omega_{-1} > 0, \omega_0 < 0, \omega_1 > 0 \), then the dynamical system (4.4) has two homoclinic curves of the upright equilibrium \( \theta_{eq} = 0 \), see Fig. 4.2(a), if and only if for some \( a_1, a_2 \) with \( a_1 \in (-2\pi, -\pi) \), \( a_2 \in (\pi, 2\pi) \) the integrals

\[ \int_0^{a_1} \left\{ 2 \int_0^s \frac{p_3 \cos \tau \Phi''(\tau) - p_2 \sin \tau \cos \tau \cdot \Phi'(\tau)^2}{p_2 + p_5 + p_3 \cdot \cos \tau \cdot \Phi'(\tau)^2} d\tau \right\} \cdot \frac{p_4 \sin s}{p_2 + p_5 + p_3 \cos s \cdot \Phi'(s)} ds \]

are equal to zeros. Furthermore, these homoclinic curves can be found as the solutions of (4.4) with initial conditions

\[ \theta_s(0) = a_1, \quad \dot{\theta}_s(0) = 0 \quad \text{and} \quad \theta_s(0) = a_2, \quad \dot{\theta}_s(0) = 0. \]  \hspace{1cm} (4.8)

\(^2\)I.e. it is either positive or negative for all \( \theta \in I_\Phi \).
**Figure 4.2:** Two homoclinic orbits of the equilibrium at $\theta = 0$ are shown on the phase portrait of (4.4). One of them intersects the line $\{\dot{\theta} = 0\}$ at $a_1$, another at $a_2$. In (b) an exemplified periodic trajectory orbiting two homoclinic curves is depicted.

*Proof* of 1) is straightforward. Statement of 2) is proved in [39]. To validate 3), we observe that the system (4.4) is integrable [37] and that the integral function $I(\theta(t), \dot{\theta}(t), \theta_0, \dot{\theta}_0)$, defined by

$$I = \dot{\theta}^2(t) - \exp \left\{ -2 \int_{\theta_0}^{\theta(t)} \frac{\beta(\tau)}{\alpha(\tau)} d\tau \right\} \left[ \dot{\theta}_0^2 - \int_{\theta_0}^{\theta(t)} \exp \left\{ 2 \int_{\theta_0}^{s} \frac{\beta(\tau)}{\alpha(\tau)} d\tau \right\} \frac{2\gamma(s)}{\alpha(s)} ds \right],$$

remains zero on the solution $\theta(t)$ of (4.4) started at $[\theta_0, \dot{\theta}_0]$. If we assume that a homoclinic curve $\theta_*(t)$ of the upright equilibrium $\theta_{eq} = 0$ of (4.4) does exist, then

- In the expression $I(\cdot)$ we can let $\theta_0$ and $\dot{\theta}_0$ to converge along the homoclinic curve (in negative time) to the upright equilibrium, i.e.

  $$\theta_0 \rightsquigarrow 0, \quad \dot{\theta}_0 \rightsquigarrow 0$$

- There is a point along a homoclinic curve at which the value of $\dot{\theta}_*(\cdot)$ should change sign, i.e.

  $$\exists t_\bullet: \theta_*(t_\bullet) = a, \quad \dot{\theta}_*(t_\bullet) = 0$$
Therefore, evaluating the function \( I(\cdot) \) at these points of the homoclinic curve, we obtain

\[
0 = \dot{\theta}_e^2(t_*) - e^{-2 \int_{\theta_e(-\infty)}^{\theta_e(t_*)} \frac{\beta(\tau)}{\alpha(\tau)} d\tau} \left[ \dot{\theta}_e^2(-\infty) - \int_{\theta_e(-\infty)}^{\theta_e(t_*)} e^{2 \int_{\tau}^{\infty} \frac{\beta(\tau)}{\alpha(\tau)} d\tau} \frac{2\gamma(s)}{\alpha(s)} ds \right]
\]

or the same

\[
0 = \int_0^a \exp \left\{ 2 \int_0^s \frac{\beta(\tau)}{\alpha(\tau)} d\tau \right\} \frac{2\gamma(s)}{\alpha(s)} ds.
\]

Substituting the expressions for \( \alpha(\cdot), \beta(\cdot), \gamma(\cdot) \) as in (4.4), we arrive to the formula in the statement.

To prove, in opposite, that the upright equilibrium of (4.4) has two homoclinic curves, observe that the condition \( \omega_0 < 0 \) implies that the upright equilibrium is hyperbolic, thus it has one-dimensional stable and unstable manifolds. In turn, the property 3) ensures that both branches of unstable manifolds intersect the line with \( \dot{\theta} = 0 \) at points \( a_1 \) and \( a_2 \). To see that these points indeed belong to homoclinic curves of the equilibrium \( \theta_{eq} = 0 \), it is enough to notice that the phase portrait of the dynamical system (4.4) has the mirror symmetry with respect to the line \( \{ \dot{\theta} = 0 \} \). Thus, the solutions initiated in (4.8) will converge to the upright equilibrium in positive time (along its stable manifold) as they did in negative time (along its unstable manifold).

Swinging up the Furuta pendulum based on stabilizing of homoclinic curves found in Lemma 1 might be challenging. Indeed, to compensate for parametric uncertainties and effects of unmodelled dynamics, experimenting with the Furuta pendulum, the swing up motion should be complemented with a controller. This controller should stabilize a transverse dynamics of the system to the orbit of such a motion in the four dimensional state-space of the Furuta pendulum. Presumably such a controller, if constructed, would be non-robust and of limited use for experimental studies. The conclusion comes from observation that the linearization of the transverse dynamics to each of the homoclinic curves is a linear control system with time varying coefficients, for which stabilization, estimating a rate of convergence and robustness (even for a found controller), is a difficult task.
To overcome the problem, it is suggested to stabilize orbitally a periodic solution of (4.4) rather than the homoclinic orbit itself. This periodic solution should be surrounding, close enough, the homoclinic curves, see Fig. 4.2(b). Exponential orbital stabilization of such a periodic solution, if it exists, can be achieved based on the recently proposed algorithms of [38]. The next statement suggests further mild conditions on the constraint function $\Phi(\cdot)$ of (4.5), to ensure the presence of a strip of periodic solutions surrounding, on the phase portrait, two homoclinic curves of the upright equilibrium from Lemma 1.

**Lemma 2** Suppose that the conditions of Lemma 1 are valid. If the constants $\omega(-2)$ and $\omega_2$ defined by (4.7) are both negative, then any solution $\theta(t)$ of the dynamical system (4.4), with initial conditions at $\dot{\theta}(0) = 0$ and

$$\pi < \theta(0) < 2\pi \quad \text{or} \quad -2\pi < \theta(0) < -\pi$$

is periodic.

*Proof* is based on the fact that the phase portrait of (4.4) is symmetric with respect to $\{\dot{\theta} = 0\}$ axis. Two homoclinic curves of the upright equilibrium, see Fig. 4.2, cannot be attractive or repellent for nearby solutions. This solutions should be bounded by the theorem of *continuous dependence of solutions on initial conditions*. Hence, nearby solutions, i.e. solutions with initial conditions at $\dot{\theta}(0) = 0$ and

$$\pi < \theta(0) < \pi + \delta \quad \text{or} \quad -\pi - \delta < \theta(0) < -\pi$$

are all periodic for small $\delta$ such that $\delta > 0$.

Repeating the same arguments for nearby solutions of established periodic trajectories, we observe that the arguments work unless one of the hyperbolic equilibrium, either at $-2\pi$ or $2\pi$, is reached.

### 4.2.2 Choosing the synchronization Function $\Phi(\cdot)$

To plan a swing up motion for the Furuta pendulum, using Lemmas 1 and 2, we need to find a function $\Phi(\cdot)$ of (4.5) satisfying several properties. To show that
such functions do exist, and that the conditions of Lemmas 1 and 2 are mild and easy to meet, let us present one of them as example. Namely, let us check that Lemmas 1 and 2 hold for the function

\[ \Phi(\theta) = K \cdot \arctan(\theta). \]  

The value of the constant \( K \) limits the interval of possible travel for the \( \phi \)-variable along a swing up motion within \([ -\frac{1}{2} K \pi, \frac{1}{2} K \pi ]\).

With the choice (4.9) the coefficients of (4.4) are:

\[ \begin{align*}
\alpha(\theta) &= p_2 + p_5 + \frac{Kp_3 \cos(\theta)}{(1 + \theta^2)}, \\
\beta(\theta) &= -\frac{K \cos \theta}{(1 + \theta^2)^2} (2p_3 \theta + Kp_2 \sin \theta). 
\end{align*} \]  

(4.10)

Physical parameters for our experimental setup (see Fig. 4.1) are: the length of the arm \( l_a = 0.15 [m] \), the mass of the arm \( m_a = 0.298 [kg] \), the length of the pendulum \( l_p = 0.26 [m] \), the mass of the pendulum \( m_p = 3.2 \cdot 10^{-2} [kg] \) and the mass of the bob at the end of the pendulum \( M = 7.5 \cdot 10^{-3} [kg] \). An estimation of the arms inertia according to CAD drawings is \( J_a = 7.682 \cdot 10^{-4} [kg \cdot m^2] \). For these values the coefficients \( p_1 \)–\( p_5 \) of (4.1) and (4.2) are

\[ \begin{align*}
p_1 &= 1.8777 \cdot 10^{-3}, & p_2 &= 1.3122 \cdot 10^{-3}, \\
p_3 &= 9.0675 \cdot 10^{-4}, & p_4 &= 5.9301 \cdot 10^{-2}, & p_5 &= 1.77 \cdot 10^{-4}. 
\end{align*} \]  

(4.11)

Let us rewrite all the conditions of Lemma 1 and 2 as constraints on the constant parameter \( K \), with the parameters of the model as (4.11):

- **The function \( \alpha(\cdot) \) should be separated from zero on \([-2\pi - \varepsilon, 2\pi + \varepsilon]\) with some \( \varepsilon > 0 \):** The value \( \alpha \left( \frac{\pi}{2} \right) = p_2 + p_5 \) is positive, therefore the condition holds if \( \alpha(\theta) > 0 \) for any \( \theta \in [-2\pi, 2\pi] \). The minimal value of \( \alpha(\cdot) \) on this interval can be found as

\[ \min_{\theta \in [-2\pi, 2\pi]} \{ \alpha(\theta) \} = \min \{ \alpha(-2\pi), \alpha(2\pi), \alpha(\theta_{cr}^{(i)}) \}, \]

where \( \theta_{cr}^{(i)} \) are the critical values, i.e. solutions of the equation

\[ 0 = \alpha'(\theta_{cr}) = -\frac{K \cdot p_3}{1 + \theta_{cr}^2} \left[ \sin \theta_{cr} + \frac{2 \cdot \theta_{cr} \cdot \cos \theta_{cr}}{1 + \theta_{cr}^2} \right], \]
which can be numerically found and are independent of $K$. The condition of positiveness of $\alpha(\cdot)$ within $[-2\pi, 2\pi]$ can be rewritten as the inequality

$$K \geq -38.51 \cdot \frac{p_2 + p_5}{p_3}.$$  \hfill (4.12)

- The values of the next five constants $\omega_k$ with $k = -2, -1, 0, 1, 2$, defined by

$$\omega_k = \left. \frac{-p_4 \cos \theta}{p_2 + p_5 + p_3 \cdot \cos \theta \cdot \frac{K}{1 + \theta^2}} \right|_{\theta = k\pi},$$

see (4.7), should have the following signs

$$\omega_{(-2)} < 0, \quad \omega_{(-1)} > 0, \quad \omega_0 < 0, \quad \omega_1 > 0, \quad \omega_2 < 0.$$  

These conditions are equivalent to only three independent inequalities for the parameter $K$

$$K > -\left(1 + 4\pi^2\right) \frac{p_2 + p_5}{p_3},$$

$$K > -\frac{p_2 + p_5}{p_3},$$

$$K < \left(1 + \pi^2\right) \frac{p_2 + p_5}{p_3}.$$  \hfill (4.13)

- The value of

$$F(a_1, K) = \int_0^{a_1} e^{\left\{2 \int_0^s \frac{p_3 \cos \tau \Phi''(\tau) - p_2 \sin \tau \cos \tau [\Phi'(\tau)]^2}{p_2 + p_5 + p_3 \cdot \cos \tau \cdot \Phi'(\tau)} d\tau \right\}} \cdot \frac{p_4 \sin s}{p_2 + p_5 + p_3 \cos s \cdot \Phi'(s)} ds$$

should be zero for some $a_2 \in (\pi, 2\pi)$ and $a_1 \in (-2\pi, -\pi)$. The fact that this condition is valid (or not) has intuitive geometrical interpretation: the stable and unstable manifolds (seratrixes) of the hyperbolic equilibrium $\theta_{eq} = 0$ are bounded (or not), see Fig. 4.3. A simple sufficient condition for this property is to check the values

$$F(a, K) \quad \text{at} \quad a = -2\pi, -\pi, \pi, 2\pi.$$
4.3. RESULTS OF COMPUTER SIMULATIONS

If for some $K$ both inequalities

$$F(-2\pi, K) \cdot F(-\pi, K) < 0, \quad F(2\pi, K) \cdot F(\pi, K) < 0$$

hold, then the condition holds as well. The values of $F(\pi, K)$ and $F(2\pi, K)$ as functions of $K_n = K \frac{p_3}{p_2 + p_5}$ are shown on Fig. 4.4. As seen, when $K_n$ is around zero $K_n < 0.01$, then both functions $F(\pi, K)$ and $F(2\pi, K)$ are negative, while for most of positive values of $K_n$ on the plot they have different sign. Nevertheless, both functions are difficult to compute accurately for $K_n \geq 6.6$. The behavior of $F(-\pi, K)$ and $F(-2\pi, K)$ is the same. To conclude, condition 3 of Lemma 1 is valid if

$$0.01 \times \frac{p_2 + p_5}{p_3} < K < 6.6 \times \frac{p_2 + p_5}{p_3},$$

(4.14)

where the upper bound in (4.14) can be enlarged.

4.3 Results of Computer Simulations

For planning a motion and simulating the swing-up, the constrained function (4.9) with $K = 1/\pi$ has been chosen. Such value of $K$ satisfies all the conditions of Lemma 1 and 2 described by (4.12)–(4.14). The phase portrait of the corresponding dynamical system (4.4) is shown on Fig. 4.5, where as predicted the upright
CHAPTER 4. FURUTA PENDULUM

Figure 4.4: The values of $F(\pi, K)$ and $F(2\pi, K)$ versus $K_n = K \cdot \frac{p_3}{p_2 + p_5}$, blue and red respectively. As seen, they have different signs for most of positive values of $K_n$, while around zero, $K_n < 0.01$, their signs are the same (negative).

The equilibrium $\theta_{eq} = 0$ has homoclinic curves. Furthermore, homoclinic curves are surrounded by a strip of periodic solutions.

In the following simulation results, the trajectory depicted with bold lines on Fig. 4.5(a)-(b) is chosen as an example of a swinging-up motion and will be orbitally stabilized.

A stabilizing state feedback controller\(^3\) for this cycle has been designed following the steps presented in [37, Appendix C, p. 1174]. The performance of the closed loop system is illustrated on Fig. 4.6(a)-(d), where the target orbit and behavior of the closed loop system solution has initial conditions at

\[
\phi = 0, \quad \dot{\phi} = 0, \quad \theta = \pi, \quad \dot{\theta} = 0,
\]

which is the downward (stable) equilibrium. The simulated closed-loop system

\(^3\)The controller design is not discussed here, since it is beyond the topic of this article and different other references have a detailed description of its derivation, e.g. [36, 37, 39]
4.3. RESULTS OF COMPUTER SIMULATIONS

Figure 4.5: (a) Phase portrait for (4.4) with (4.9). The trajectory marked with bold line represents the periodic solution to be controlled later; (b) Solutions of the system (4.4) around $\theta_0 = 0$ depicted as functions of time. The solution to be stabilized (bold line) has a period $T \approx 4.0454$ sec.

response is in agreement with the theoretical results, i.e. the solution converges to the target cycle with the relation between the angles (see Fig. 4.6) being an arctan function.

4.3.1 Switching law and stabilizing controller simulation

A switching law is needed around the upright equilibrium, in order to swap the roles of the controllers and keep the pendulum balancing. With initial conditions around the downward resting equilibrium, upright is approached first when $\theta = 0$ and $-1.34 < \dot{\theta} < 0$, see Fig. 4.7. In our case, the opposite phase is chosen, in order to keep a fancy motion of the pendulum before the switching. This means that the switching will be performed when $|\phi| < 0.1$, $|\dot{\phi}| < 0.3$, $\dot{\theta}$ is within the interval $[0, 1.34]$ and $[-0.24, 0.24]$ for $\theta$.

In Fig. 4.8 we show results of a simulation of the swing up and the stabilizing controller with the switching proposed. As it is observed, the swing up strategy brings the pendulum close to $\theta = 0$ and then the local stabilizing controller takes over. It is known that a successful switching depends very much on the region of
attract\[^4\] of the local balancing controller. In our case, simulations for the linear feedback controller were done, by varying the initial conditions ($\theta_0, \dot{\theta}_0$), for the interval of switching mentioned before to ensure convergence of $\theta$ to zero.

\[^4\]Note that we have chosen linear stabilizing controller only for simplicity. The region of attraction can be enlarged with other designs, see e.g. [5]
4.4 Sampling, Real-Time implementation and Experimental Results

Let us discuss issues related to the controllers real-time implementation. The Furuta Pendulum (see Fig. 4.1) built in our lab is equipped as follows. Rotation of the arm is controlled by a DC-motor, while the pendulum moves freely in the plane, orthogonal to the arm. The DC motor is a brush-commutated Penta M1, equipped with a servo amplifier SCA-SS-70-10. The angular positions are measured with optical incremental encoders: Eltra EH53A (1024 imp/rev) for the arm’s angle and Wachendorff WDG 40S-2500-ABN-H05-SB5-AAC (2500 imp/rev) for the pendulum’s angle.

The setup is connected to a dSPACE board 1104 used to run the real-time application. The user interface consists of a PC equipped with ControlDesk (©dSPACE GmbH), used for on-line communication with the dSPACE board. Matlab/Simulink (©Mathworks) is the software used for the implementation of algorithms as well as simulations. Simulink enables the generation and deployment of production C code for use in real-time embedded systems. The compiled code is loaded into the dSPACE’s processor. The program loaded contains a control function, which is executed according to a sampling period (0.8 ms in our case).

Figure 4.7: The switching region specified as a range of $\theta$ and $\dot{\theta}$. 
4.4.1 Sampling

So far, for this design, it has been assumed that disturbances are negligible and that the input signal as well as measured states are realizable at every instant of time. However, the dSPACE board used, as well as any other digital hardware, works with an internal processor which runs according to an internal frequency generator. For this reason the feedback control law must be sampled, meaning that we have to apply for \( t : k h < t \leq (k + 1) h \) the control signal to the motor

\[
  u_r(t) = u[k] \overset{def}{=} \text{Sat}_{0.5} (u(k h)),
\]

where \( k \) is a natural number, the small constant \( h > 0 \) is the sampling time and \( u(k h) \) is related to the input torque \( \tau_\phi \) from (4.1) at \( t = k h \). The states \( \theta \) and \( \phi \) are the states measured by the encoders, and \( \text{Sat}_M (\cdot) \) is the standard saturation function with the cut-off level \( M = 0.5 \), which is related to the limitation of the available control torque. The electromotive torque constant expressed in \( Nm/V \), relating the motor input \( u(k h) \) with \( \tau_\phi \), is identified to be 0.51 [36].

4.4.2 Velocity estimation

The velocity, which is part of the feedback states, are estimated by applying a high gain observer [24]. A high gain observer (HGO) is a fast nonlinear (or li-
ear) full order observer with high observer gains chosen via pole placement. For this differentiating scheme, saturation is needed in order to protect the plant from peaking [24]. Applying [24, Sec. 14.5]

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} \dot{v}_1 \\ \dot{v}_2 \end{bmatrix} + \begin{bmatrix} 2(y_1 - \hat{x}_1)/\varepsilon_1 \\ 2(y_2 - \hat{x}_2)/\varepsilon_2 \end{bmatrix},$$

$$\begin{bmatrix} \dot{\hat{v}}_1 \\ \dot{\hat{v}}_2 \end{bmatrix} = f_0(\hat{x}_1, \hat{x}_2, \hat{\phi}, \hat{\theta}) + \begin{bmatrix} (y_1 - \hat{x}_1)/\varepsilon_1^2 \\ (y_2 - \hat{x}_2)/\varepsilon_2^2 \end{bmatrix},$$

(4.17)

$$\hat{\phi} = \text{Sat}_{M_1}(\hat{v}_1), \quad \hat{\theta} = \text{Sat}_{M_2}(\hat{v}_2),$$

where the parameters $\varepsilon_{1,2} > 0$ are to be tuned,

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} \phi \\ \theta \end{bmatrix} + w(t) \quad \text{with} \quad w(t) = \begin{bmatrix} \phi[k] - \phi(t) \\ \theta[k] - \theta(t) \end{bmatrix}$$

is the measured output with the ‘measurement noise’ $\forall t : k h < t \leq (k + 1) h,$ and either

$$f_0(\phi, \theta, \dot{\phi}, \dot{\theta}) = 0$$

(4.18)

or $f_0(\phi, \theta, \dot{\phi}, \dot{\theta})$ is a nominal model defined by (4.1)-(4.2).

In the discretized case, if the sampling period is set to be $h = 0.0008,$ then the observer parameters are taken as

$$\varepsilon_1 = 20 h \quad \text{and} \quad \varepsilon_2 = 10 h,$$

while $M_1 = M_2 = 20 \text{ rad/s},$ which corresponds to a reasonable level of upper bounds on the speeds.

### 4.4.3 Friction Compensation

The friction torque can be modeled as a nonlinear static mapping. Its approximation can be used for partial compensation of the friction present in the arm. The friction compensation adopted here considers Coulomb and viscous friction. Estimation of the velocity is done applying (4.17)-(4.18). Due to delays caused by
the velocity estimation, a performance degradation is expected in the region of the desired orbit where the velocity is close to zero.

Fig. 4.9 presents the static friction identified for our setup. The feed-forward compensation law is defined as:

\[
 u_{fr} = \begin{cases} 
 0.032 - 0.002 \cdot \dot{\phi}, & \text{if } \dot{\phi} > 0 \\
 -0.033 - 0.002 \cdot \dot{\phi}, & \text{if } \dot{\phi} < 0.
\end{cases}
\] (4.19)

![Figure 4.9: Estimated friction.](image)

### 4.4.4 Swing Up experimental results

Recorded experimental results are presented in Fig. 4.10(a)-(d), when the initial conditions are (4.15).

The experimental results show that the proposed swing up method is successful in practice. In fact, the desired cycle is approximately achieved. Irrespective of variation in initial conditions, experiments show reproducibility and stability.

The mismatch between experimental and desired response, see Fig. 4.10(c), is thought to be produced by the following factors: (a) the friction torque at the pendulum shaft is not compensated, (b) discretization of the measurements does not allow us to use sufficiently fast differentiators and leads to introduction of additional delays imposed by our observers, and (c) there are presumably input
4.4.5 Switching law and stabilizing controller experimental results

Considering the switching described earlier, with $|\phi| < 0.1$, $|\dot{\phi}| < 0.3$, $\dot{\theta}$ within the interval $[0, 1.34]$ and $[-0.24, 0.24]$ for $\theta$. Applying a LQR control based on
a linearization of the system dynamics (4.1)-(4.2), Fig. 4.11(a)-(b) shows results applying the closed-loop strategy combining swinging up, switching and local stabilizing controller.

![Figure 4.11](image)

**Figure 4.11:** (a) The experimental phase portrait of the system for swinging up and stabilizing controller around the upward equilibrium at $\theta = 0$; (b) Experimental behavior of $\theta$ as a function of time.

In experiments\(^5\) with different initial conditions, the state $\theta$ converges to the desired steady state as expected. It is worth mentioning that once the switching is done, and the pendulum is stabilized, the friction compensation is deactivated. This is done to avoid the chattering of the input signal due to numerical errors caused when the velocity is close to zero.

### 4.5 Conclusions

We have suggested a new technique for planning swing-up motions for the Furuta pendulum. The arguments are based on an accurate planning of two homoclinic curves for the non-actuated link of the pendulum. For this motion the arm angle and the pendulum angle are regulated by some control input to satisfy a certain geometric relation. If such step in design is successful, we suggest a further mod-

\(^5\)Movies with recorded experiments are available at [http://www.tfe.umu.se/forskning/Control_Systems/Set_Ups/Furuta_Pendulum/Furuta_Pendulum_info.html](http://www.tfe.umu.se/forskning/Control_Systems/Set_Ups/Furuta_Pendulum/Furuta_Pendulum_info.html)
4.5. CONCLUSIONS

Modification of the homoclinic-curve-based-swing-up motion. The modification is to find a nearby periodic solutions to the homoclinic orbits. Such cycle can be robustly stabilized by the method of [37]. The method is illustrated by successful numerical simulations and experimental studies.
Part V

First steps into walking bipeds
A simple walking case: the compass biped

Complexity of actual humanoid robots is rather high. One drawback, however, is the high energy consumption required to perform certain motions. In today’s approaches, human captured data is often applied as reference trajectories for the robot’s motions. In walking for example, the mechanics of the human body makes our walking gait to be optimal in terms of body energy consumption. This does not at all mean, however, that the same leg and joint motion is also efficient for a robot with a different mechanical structure. Hence, simply using human joint trajectories as reference trajectories for robots may result in motion that might look sort of good, but there is no reason to assume that it is efficient as well. Clearly, robot walking based on the ZMP concept would require a high demand of energy
due to the complexity of the hardware involved on imitating the human motion.

Fortunately, energy-efficient trajectories for walking robots do exist. McGeer [31] showed in his remarkable work on passive walking robots, that certain mechanical systems have indeed a natural tendency to walk. As explained earlier, these mechanisms can walk down a shallow slope without actuation, only powered by gravity. The center of mass of these robots does not always remain above the stance foot (as the case of the ZMP), and hence they are not statically stable. Nevertheless, it has been shown, both in theory and in practice, that the walking cycles of these robots are dynamically stable, hence the name dynamic walking.

Inspired by passive dynamic walkers, researchers have looked at various ways of adapting and augmenting the passive dynamic motions in order to obtain more robustness, walking on different slopes (including level floor), and to attain other design goals. Many of these strategies are biologically inspired, as it is the case of Wisse and Van Frankenhuyzen [45], who extended McGeer’s original passive dynamic walker with McKibben muscles to obtain stable, robust, level-floor walking. Others have focused more on the energy conserving properties of passive walkers, and designed their controllers to adjust the energy balance. Spong [46] describes a control law that effectively rotates the apparent gravitational field, thus making the controlled robot move with the same gait on different slopes.

Unfortunately, no matter what control technique is used, the behavior of (almost) passive walkers remains difficult to understand, due to the highly nonlinear, coupled, and generally unstable dynamics, together with the hybrid aspects of switching between left and right foot, and between contact and no-contact.

In this chapter, some ideas towards controlling the case of a simple biped robot applying virtual constraints will be given. To get a brief understanding of the difficulty of this control task, basic steps on modeling hybrid systems will be shortly described.

### 5.1 Hybrid systems

Hybrid systems model nontrivial interactions of continuous (i.e., describable by a set of differential equations) and discrete phenomena (i.e. asynchronous sys-
tems where the state transitions are initiated by discrete events as in automata or finite state machines). This leads to a formulation with several modes of operation, whereby in each mode the behavior of the system is given by difference or differential equations. Hybrid systems arise throughout business and industry in areas such as parallel processing, traffic control, plant process control, telecommunication, automotive systems, biological systems, and robot path planning.

One of the most classical and simple examples to describe a hybrid systems is that of a bouncing ball, a physical system with impacts. Here, the ball (thought of as a point-mass) is dropped from an initial height and bounces off the ground, dissipating its energy after each bounce (see Fig. 5.1). The ball exhibits continuous dynamics between each bounce; however, as the ball impacts the ground, its velocity undergoes an instantaneous change modeled after an inelastic collision. A mathematical description of the bouncing ball follows.

The state is two-dimensional, with $x_1$ being the ball’s height above the floor and $x_2$ being the ball’s velocity. The state space corresponds to $x_1 \geq 0$ and $x_2 \in \mathbb{R}$. The states differential equation is given by

$$\dot{x} = \begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} x_2(t) \\ -\gamma \end{bmatrix} = f(x)$$

where $\gamma$ is the acceleration due to gravity. The differential equation applies in

$$C = \{x_1 > 0, \text{ or } x_1 = 0 \text{ and } x_2 > 0\}.$$  \hspace{1cm} (5.2)

The jump equation for velocity is $x_2^+ = -\mu x_2^- = g(x)$ where $\mu \in (0, 1)$ is a dissipation factor, and the notation $+$ and $-$ denote the events after and before the impact correspondingly. This describes the nature of the inelastic collision. It applies in

$$D = \{x_1 = 0, \text{ and } x_2 \leq 0\}.$$  \hspace{1cm} (5.3)

Figure 5.1 contains a sample trajectory for this system. Visually, the trajectories tend towards the origin.

The model can be also schematized as shown in Fig. 5.2, where the domains for C and D represent the continues and discrete events.
5.2 Compass biped modeling

An example of the compass biped is depicted in Fig. 5.3. For this particular case we consider point masses in the legs and a mass at the hip. Actuation is given only at the hip. The system resembles an under-actuated double pendulum with contact point.

Applying the Euler-Lagrange formulation, we can define the swing phase of the biped dynamics by

\[ M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = B(q)u, \quad (5.4) \]
5.2. COMPASS BIPED MODELING

where the states $q = [\theta_1, \dot{\theta}_1, \theta_2, \dot{\theta}_2]$ and the matrices $M(q), C(q, \dot{q}), G(q)$ are

$$
M(q) = \begin{bmatrix}
mb^2 & -mbl \cos (\theta_2 - \theta_1) \\
-mbl \cos (\theta_2 - \theta_1) & mh l^2 + ml^2 + ma^2
\end{bmatrix},
$$

$$
C(q, \dot{q}) = \begin{bmatrix}
0 & mbl \sin (\theta_2 - \theta_1) \dot{\theta}_2 \\
-mbl \sin (\theta_2 - \theta_1) \dot{\theta}_1 & 0
\end{bmatrix},
$$

$$
G(q) = \begin{bmatrix}
mgb \sin (\theta_1) \\
-(ma + ml + mh) \sin (\theta_2) g
\end{bmatrix}.
$$

In these matrices the physical parameters are the legs mass $m = 5\ kg$, hip mass $mh = 10\ kg$, the leg length $l = 1\ m$ and $b = 0.5\ m$ being the distance from the hip to the center of mass of the leg. The ground slope angle is denoted by $\phi$. The input matrix $B$ is a column vector $[1, -1]^T$. 

Figure 5.2: Bouncing ball model.
5.2.1 Rigid body collisions

When modeling walking robots, or multi-body systems in general, the number of rigid bodies that may come in contact is usually limited. For a bipedal robot, contact normally only occurs between the feet and the ground (see Fig. 5.4); if other parts of the robot touch the ground, walking has clearly failed, and there is usually no need for accurate simulation of the robot falling down and breaking.

The collision model proposed by [47] will be used for this case. One assumption is that the collision with the ground results in no rebound and slipping of the
5.2. COMPASS BIPED MODELING

swing leg. To begin with, we consider an augmented model

\[ M_{aug}(q_{aug})\ddot{q}_{aug} + C_{aug}(q_{aug}, \dot{q}_{aug})\dot{q}_{aug} + G_{aug}(q_{aug}) = B_{aug}(q_{aug})u + \delta F_{impact}, \quad (5.5) \]

where the vector \( q_{aug} = [\theta_1, \dot{\theta}_1, \theta_2, \dot{\theta}_2, x, y, \dot{x}, \dot{y}] \) is an augmented vector with the new states \([x, y]\) being the coordinates of the stance leg in respect to the ground. The external force at the contact point is denoted by \( \delta F_{impact} \). It is assumed that the external force at impact can be modeled by an impulse of short period. Furthermore, it is assumed that the impact changes only the actual velocities, but there is no change in the link positions. Integrating (5.5) over an infinitesimally small period of time denoted by \( t^+ \) and \( t^− \), we obtain

\[ M_{aug}(q_{aug})(\dot{q}_{aug}^+ − \dot{q}_{aug}^−) = \int_{t^−}^{t^+} \delta F_{impact}(t)dt \quad (5.6) \]

having \( \dot{q}_{aug}^+ \) as the new velocity vector after impact and \( \dot{q}_{aug}^− \) right before the impact. The angular position of the links does not change but gets switched, since after the impact the swing leg becomes the stance leg and vice versa. We can define the switching operator as \( q_{aug}^+ = \Delta q_{aug}^− \), where the operator \( \Delta \) is some matrix which defines the relabeling of the legs.

According to [47], analysis is concentrated only at the impact point (end of swing leg), which in our example is defined by the cartesian coordinates

\[ \Psi(q_{aug}) = \begin{bmatrix} x - l \cdot \sin(\theta_2(t)) + l \cdot \sin(\theta_1(t)) \\ y + l \cdot \cos(\theta_2(t)) - l \cdot \cos(\theta_1(t)) \end{bmatrix}, \quad (5.7) \]

and so, the external force at the impact point can be found as

\[ F = \frac{\partial \Psi(q_{aug})}{\partial q_{aug}} \begin{bmatrix} F_y \\ F_x \end{bmatrix}, \quad (5.8) \]

where \( F_y \) and \( F_x \) are the normal and tangential forces applied at the end of the swing leg. Additionally, assuming an inelastic collision with no rebound and slipping, we can formulate that

\[ \frac{\partial \Psi(q_{aug})}{\partial q_{aug}} \dot{q}_{aug}^+ = 0. \quad (5.9) \]
Combining together all equations, it is found that velocities after impact can be computed by the next algebraic matrix system

\[
\begin{bmatrix}
M_{aug} & -\frac{\partial \Psi(q_{aug})}{\partial q_{aug}}^T \\
\frac{\partial \Psi(q_{aug})}{\partial q_{aug}} & 0_{n \times n}
\end{bmatrix}
\begin{bmatrix}
\dot{q}_{aug}^+ \\
F
\end{bmatrix}
= 
\begin{bmatrix}
M_{aug} \dot{q}_{aug}^- \\
0_{n \times 2n}
\end{bmatrix}.
\] (5.10)

Finally, the new states after impact can be denoted as an operator of transformation \( F \), with \( F \) being

\[
F(q_{aug}) = \begin{bmatrix}
\Delta \\
0_{n \times n}
\end{bmatrix}
\begin{bmatrix}
M_{aug} & -\frac{\partial \Psi(q_{aug})}{\partial q_{aug}}^T \\
\frac{\partial \Psi(q_{aug})}{\partial q_{aug}} & 0_{n \times n}
\end{bmatrix}^{-1}
\begin{bmatrix}
M_{aug} \\
0_{n \times 2n}
\end{bmatrix},
\] (5.11)

so that,

\[
\begin{bmatrix}
q^+ \\
\dot{q}^+
\end{bmatrix} = F(q_{aug}) \begin{bmatrix}
q_{aug}^- \\
\dot{q}_{aug}^-
\end{bmatrix}.
\] (5.12)

### 5.2.2 The Hybrid Model

The final model for the compass biped can be expressed as a hybrid system with continues and discrete dynamics, mainly caused by impact effects. Considering the walking surface, given a slope \( \phi \), to be

\[
S = \{ q \in \mathbb{R}^2 : H(q) = \cos(\theta_2 + \phi) - \cos(\theta_1 + \phi) = 0 \},
\] (5.13)

the hybrid model is defined as

\[
\mathcal{H} : \begin{cases}
\dot{x}(t) = f(x(t)) + g(x(t))u(t) & x^-(t) \notin S; \\
x^+ = F \cdot x^- & x^-(t) \in S.
\end{cases}
\] (5.14)

The simulation of the compass biped is as follows. The robot starts at some initial condition defined by the state \( x_0 = [\theta_1(0), \dot{\theta}_1(0), \theta_2(0), \dot{\theta}_2(0)] \). Integration of the continuous dynamics is done until the moment that the states of \( x(t) \) fulfill the walking surface (5.13). The impact model (5.12) is applied at this point making a switch in the angular positions of the robot and calculating new values due to impact for the velocities, giving a set of new initial conditions for the model, and the new integration starts over.
5.3 Passive Walking

A rough solution to find passive gaits for this model is by solving the next optimization problem,

$$\min_{\{\xi, \zeta, \lambda, \varsigma\}} \left\{ \left\| \bar{q}(0^+) - \begin{bmatrix} \xi \\ \zeta \end{bmatrix} \right\|^2 + \left\| \dot{\bar{q}}(0^+) - \begin{bmatrix} \lambda \\ \varsigma \end{bmatrix} \right\|^2 \right\}$$  \hspace{1cm} (5.15)

with the vector $[\xi, \zeta, \lambda, \varsigma]$ being the initial states $x_0$ at the beginning of the motion denoted as $(0+)$. The optimization task is performed as follows. An initial guess is made for the parameters $[\xi, \zeta, \lambda, \varsigma]$, the hybrid model (5.14) is integrated up to getting into the switching surface (5.13), the impact model is applied getting the new values for the states $[\bar{q}(0^+), \dot{\bar{q}}(0^+)]$. If the parameters $[\xi, \zeta, \lambda, \varsigma]$ are equal to the new set of initial conditions, then the periodic trajectory is found.

Examples of limit cycles for different slopes are shown in Fig. 5.5. Those were found as explained in the previous paragraph. These cycles, which represent the so-called $1^{st}$ gaits as explained in the work of [7] and the references there in, represent limit cycles for the system (5.14). Issues related to stability analysis, for the case of passive walking, will not be touched in this thesis, since other references, e.g. [7], provide a deeper and more precise study on that topic.

5.4 Definition of constraint polynomials

The aim of this section is to define a family of constraint polynomials, which will be used later for control design, as shown in previous chapters. The aim is to give this simple system, i.e. the compass biped, the control strategy that would produce walking gaits not only in shallow slopes, but also in level ground and rough terrain.

It is possible to apply polynomials of different order for the re-parametrization of the dynamics, but in order to gain advantage from the polynomial properties, we consider parametric spline functions, i.e. Beziér polynomials.

Beziér curves are an important tool used to model smooth curves that can be scaled indefinitely. It was originally used by Pierre Beziér for CAD/CAM operations at Renault motor company. Beziér curves are the basis of the entire Adobe...
PostScript drawing model which is used in the software products: Adobe Illustrator, Macromedia Freehand and Fontographer.

Examination of Beziér polynomials

Let’s consider the case of a linear polynomial. In order to connect two points by a straight line, we could define the next polynomial

\[ B(t) = (1 - t) \cdot P_0 + t \cdot (P_1), \quad t \in [0, 1], \]

which is equivalent to a linear interpolation. In the previous polynomial, \( P_0 \) and \( P_1 \) are the start and end point respectively. Two properties are clearly seen by this example. At time \( t = 0 \) the polynomial becomes \( B(0) = P_0 \), and at time \( t = 1 \) it is \( B(1) = P_1 \). One important property is that, no matter which order the polynomial has, the start and end points are already defined since the polynomial is defined for the interval \( t \in [0, 1] \).
5.4. DEFINITION OF CONSTRAINT POLYNOMIALS

Generalization

The Beziér curve of degree \( n \) can be generalized as follows. Given points \( P_0, P_1, \ldots, P_n \), the polynomial is

\[
B(s) = \sum_{i=0}^{n} \binom{n}{i} (1 - s)^{n-i} s^i P_i, \quad s \in [0, 1].
\]  

(5.16)

where the points \( P_i \) are called control points. The polygon formed by connecting the Beziér points with lines, starting with \( P_0 \) and finishing with \( P_n \), is called the Beziér polygon (or control polygon). The convex hull of the Beziér polygon contains the Beziér curve [48]. For example, for \( n = 3 \):

\[
B(s) = (1 - s)^3 P_0 + 3s(1 - s)^2 P_1 + 3s^2(1 - s) P_2 + s^3 P_3, \quad s \in [0, 1].
\]

gives somewhat the graph depicted in Fig. 5.6.

![Fig. 5.6: The anatomy of a Beziér curve.](image)

Properties of Beziér splines

Beziér splines enjoy some nice properties. For example

- End points interpolation:

\[
B(0) = P_0
\]
\[ B(1) = P_n. \]

- Convex hull. The curve is contained in the convex hull of its control polygon.
- Symmetry. \( B(s) \) defined by \( \{P_0, ..., P_n\} \equiv B(1-s) \) defined by \( \{P_n, ..., P_0\} \).
- Tangents.
  \[
  \frac{\partial B(0)}{\partial s} = n(P_1 - P_0) \\
  \frac{\partial B(1)}{\partial s} = n(P_n - P_{n-1}).
  \]
- \( k \)'th derivatives: In general,
  \[
  \frac{\partial^{(k)} B(0)}{\partial s^{(k)}} \text{ depends only on } \{P_0, ..., P_k\}. \\
  \frac{\partial^{(k)} B(1)}{\partial s^{(k)}} \text{ depends only on } \{P_n, ..., P_{n-k}\}.
  \]
  At intermediate points \( s \in (0, 1) \), all control points are involved for every derivative.

In general, the reparametrization variable \( \theta \), explained in previous chapters, will not take values within the unit interval as required for the Beziér polynomials. So, in order to apply Beziér functions, to take advantage of their properties, we could project \( \theta \) to be within the convex hull, defined by the control points, as

\[
  s(\theta) = \frac{\theta - \theta^+}{\theta^- - \theta^+}, \quad (5.17)
\]

where \( q_i \) represents the link-\( i \) and \( \theta^+, \theta^- \) represent the initial and final values of \( \theta \) in the robot’s step. According to the normalization it is clear that derivatives of \( B(s) \) will take the form:

\[
  \frac{\partial B(\theta)}{\partial \theta} = \frac{\partial B(\theta)}{\partial s} \frac{\partial s}{\partial \theta}
\]

and so, the tangent properties of the Beziér polynomial will be:

\[
  \frac{\partial B(\theta^+)}{\partial \theta} = \frac{n}{\theta^- - \theta^+} (P_1 - P_0) \\
  \frac{\partial B(\theta^-)}{\partial s} = \frac{n}{\theta^- - \theta^+} (P_n - P_{n-1}).
\]
5.5 Hybrid zero dynamics

Assuming some function $\phi_*(\theta_*(t))$ representing the constraint function (e.g. Bezier polynomial), the hybrid zero dynamics can be found by following the next steps.

**Step 1: Reparametrizing the states of the system**

The reparametrization is done as follows,

\begin{align*}
\theta_{*,1}(t) &= \phi_*(\theta_*(t), P) \\
\theta_{*,2}(t) &= \theta_*(t),
\end{align*}

where $\theta_2$ is chosen as the coordinate for reparametrization\(^1\) and $P$ is a set of polynomial parameters unknown at the moment.

**Step 2: Swing leg zero dynamics**

The swing leg zero dynamics is found by multiplying (5.4) by the matrix

$$T = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix},$$

which obtains the new dynamics with underactuation of degree one, i.e. $n - 1$ actuators, where $n$ is the number of degrees of freedom. By replacing (5.18) into the unactuated equation, the associated zero dynamics is

$$\alpha(\theta_*, P)\dot{\theta}_* + \beta(\theta_*, P)\dot{\theta}_*^2 + \gamma(\theta_*, P) = 0,$$

where $P$ represents the parameters of the constraint polynomials to be identified by some method. The differential equation is a parametric family of two dimensional manifolds of the state space of (5.4). Integral properties, as shown in previous chapters, holds, provided the solution $\theta(t, P)$ is defined for the walking period.

**Step 3: Projection of the discrete dynamics of the hybrid system**

By analysing the impact equation (5.12) as

$$\mathcal{F}([\theta, \dot{\theta}])_{[\theta, \dot{\theta}] \in S_\pm} = F([q, \dot{q}])_{[q = \phi(\theta, P), \dot{q} = \phi'(\theta, P)\dot{\theta}]}$$

\(^1\)The stance leg increases monotonically during each step
results in the projected version of the impact equation (5.12) to the constraint function defined by (5.18). The switching surface is projected as follows,

\[ S_+ = \{ q \in \mathbb{R}^4 : \cos(\theta_2 + \phi) - \cos(\theta_1 + \phi) = 0 \} \cap O_+(p_+^*) \]  
\[ S_- = \{ q \in \mathbb{R}^4 : \cos(\theta_2 + \phi) - \cos(\theta_1 + \phi) = 0 \} \cap O_-(p_-^*) \]

where \( O_+ \) and \( O_- \) are some neighborhoods of the beginning and end of the continues in time part of the gait. The cycle starts at the point \( p_+^* = [q_*(\theta_+), \dot{q}_*(\theta_+)] \) and ends at \( p_-^* = [q_*(\theta_-), \dot{q}_*(\theta_-)] \). The intersections of the zero dynamics states with \( S_+ \) and \( S_- \) are the intervals of straight lines,

\[ \gamma_+ = \{ q \in \mathbb{R}^4 : \theta_1 = \theta_1^*(\theta_+), \theta_2 = \theta_+, \dot{\theta}_1 = \phi'(\theta_+) \dot{\theta}_+ \} \]  
\[ \gamma_- = \{ q \in \mathbb{R}^4 : \theta_1 = \theta_1^*(\theta_-), \theta_2 = \theta_-, \dot{\theta}_1 = \phi'(\theta_-) \dot{\theta}_- \} \]

A graphical representation can be seen in Fig. 5.7.

**Figure 5.7**: Hybrid zero dynamics, where the walking surface is represented by planes that somehow impose a boundary for the continuous dynamics.
5.5. HYBRID ZERO DYNAMICS

5.5.1 Designing constraint functions to achieve invariance of the hybrid zero dynamics

In this section we will apply the properties of the Beziér functions in order to achieve periodic solutions of the hybrid zero dynamics. The steps to follow are:

- Choose the order of the polynomial to be applied. Different order of polynomials result in different motions, and the larger the number of parameters to be estimated the more complicated the problem might become.

- According to the end point interpolation property, it is always the case that the initial and final angular states can be related as:

  \[ q_i(\theta_+ + \theta_-) = P_0, \]
  \[ q_i(\theta_+ - \theta_-) = P_n. \]

- In the same way, the velocities of the links can be related by the tangent property as:

  \[ \dot{q}_i(\theta_+ + \theta_-) = \frac{n}{\theta_+ - \theta_-} (P_1 - P_0) \dot{\theta}_+, \]
  \[ \dot{q}_i(\theta_+ - \theta_-) = \frac{n}{\theta_+ - \theta_-} (P_n - P_{n-1}) \dot{\theta}_-. \]

- Invariance of the parameters to the impact conditions is achieved by analyzing (5.12) as:

  \[ \begin{bmatrix} \theta_+ \dot{\theta}_+ \end{bmatrix} \in \gamma_+, \]
  \[ \begin{bmatrix} \theta_- \dot{\theta}_- \end{bmatrix} \in \gamma_-, \]
  \[ \mathcal{F}([\theta_- \dot{\theta}_-]) = [\theta_+ \dot{\theta}_+]. \]

- From (5.29) we can obtain the values of \( P_1 \) as:

  \[ P_1 = \frac{\dot{q}_i(\theta_+)}{\dot{\theta}_+} \frac{\theta_- - \theta_+}{n} + P_0, \]

  since the values for velocities are calculated by the impact equation and can be known.
5.5.2 Application of Optimization to design stable walking

The previous section defined that no matter the order of the polynomial, there exist a set of 4 parameters that can be related to the initial and final states of the robot’s configuration and velocities. The solution of those equations satisfying the discrete event results in periodic trajectories of the hybrid zero dynamics. Therefore, it is a matter of finding or selecting the rest of the parameters, to achieve some desired motion.

An optimization method (see e.g. [1]) can be applied in order to possibly find gaits which satisfy certain walking properties. Unlike the work presented in [1], in this section a procedure based on [27], which was used for the case of non-hybrid systems, will be derived.

The optimization procedure is defined in the following steps:

**Step 1: Selecting constraint function**
Choose the order of Bezier polynomials to be applied, in order to have a set of $C^2$-smooth functions

$$
\phi(\theta, P) = \{\phi_1(\theta, P), \phi_2(\theta, P), \ldots, \phi_n(\theta, P)\},
$$

for all the degrees of freedom, with $\theta$ being a monotonically increasing variable of the robot’s configuration. Applying (5.27) calculate $P_0$ and $P_n$. Generally the stepping configurations, i.e. initial and final states, are decided a priori.

**Step 2: Zero dynamics**
Compute the zero dynamics (5.20). For each set of parameters $P$, the system (5.20) is integrable, so that the function,

$$
I \left( \theta(T-, P), \dot{\theta}(T-, P), \theta(T+, P), \dot{\theta}(T+, P) \right) = 0
$$

computed from $\alpha(\theta, P), \beta(\theta, P), \gamma(\theta, P)$, as shown earlier, satisfies. In the above equation $T+$ and $T-$ represent the initial and final time of the step period.

**Step 3: Projection of the impact equation**
Apply (5.21) to have a projection of the discrete impact event to the desired trajectory. According to properties defined earlier, we can observe that:

\[
\begin{bmatrix}
P_0^{(i)} \\
\theta_+ \\
\dot{P}_+^{(i)} \\
\dot{\theta}_+
\end{bmatrix}
= F
\begin{bmatrix}
P_n^{(i)} \\
\theta_- \\
\dot{P}_-^{(i)} \\
\dot{\theta}_-
\end{bmatrix}.
\]  

(5.34)

where \(\theta\) has been chosen to be one of the generalized coordinates, and the index \(i\) points to link-\(i\). Applying (5.33) calculate parameters \(P_1^{(i)}\).

**Step 4: Optimization criteria**

Estimate the final velocity \(\dot{\theta}_-\), so that

\[
J(P_k) = \int_{T-}^{T+} \| I \left( \theta(T-, P), \dot{\theta}(T-, P), \theta(T+, P), \dot{\theta}(T+, P) \right) \|_2^2 \, dt
\]

is minimized, and the integral function is calculated as shown in (5.41).

**Step 5: Inequality constraints**

One way to reduce the search space of the optimization problem is by defining constraint boundaries (inequality constraints) that satisfy certain properties and restrict the search within a smaller region. For example, it is known that the swing leg after touching the ground should not rebound or slip, meaning that

\[ F_y > 0, \]

and the ratio of the ground forces, which represents the ground friction, should be a small value less than one, i.e.

\[ \frac{F_x}{F_y} < \mu. \]

Another restriction is that the velocity of the swing leg should not be greater and slower than certain values, defined by the designer, i.e.

\[
\begin{bmatrix}
\dot{x} \\
\dot{y}
\end{bmatrix}
= \begin{bmatrix}
-l \cdot \sin(\theta_2(t)) + l \cdot \sin(\theta_1(t)) \\
l \cdot \cos(\theta_2(t)) - l \cdot \cos(\theta_1(t))
\end{bmatrix}
\begin{bmatrix}
\dot{\theta}_1 \\
\dot{\theta}_2
\end{bmatrix}
< \begin{bmatrix}
\xi \\
\zeta
\end{bmatrix},
\]

where \(\theta\) refers to the joint angle of the swing leg and \(\mu\) is the coefficient of friction.
where $\xi$ and $\zeta$ are some constants defined by the designer.

The last constraint is defined by the input torque required to perform a motion. It was shown that for walking down a slope, a 2-link biped does not need any actuation apart of gravity, but in the case of a higher dimensional robot some sort of actuation will be required. In order to make the robot to walk with a certain speed, or in rough terrain with the minimum required torque, it would be of interest to restrict the torque of certain motion within an interval, that is

$$K_{dc} \cdot u^*(t) < \tau,$$

where $\tau$ is some constant defining the maximum torque applied for a motion, and $K_{dc}$ is the motor’s electromotive constant, explained in previous chapters.

It is less important to ensure some stability related issues [1], since the control design based on transverse linearization is able to control the case of unstable hybrid zero dynamics. These arguments are not further explored in this chapter.

### 5.5.3 Example on the Compass Biped

Following the steps presented so far:

**Step 1: Selecting the constraint function**

For the compass biped, a fourth order Beziér polynomial is chosen as a constraint function, so that

$$\theta_1(s) = (1 - s)^4 P_0 + 4s(1 - s)^2 P_1 + 6s^2 (1 - s)^2 P_2 + \ldots \quad (5.35)$$

$$+ \quad 4s^3 (1 - s) P_3 + s^4 P_4, \quad (5.36)$$

$$\theta_2(s) = \theta(t), \quad (5.37)$$

where $s$ is defined as in (5.17). Considering the robot to have a leg length $l = 1 \text{ m}$, a step length of approximately $0.4 \text{ m}$ is desired. Therefore,

$$\theta_+ = 0.2, \quad \theta_- = -0.2.$$

Since the leg is symmetric (i.e. same leg length for left and right), by the end point interpolation property of Beziér polynomials we have that,

$$\theta_1(\theta_+) = P_0 = \theta_- \quad (5.38)$$
5.5. HYBRID ZERO DYNAMICS

\[ \theta_1(\theta_\pm) = P_4 = \theta_+. \] (5.39)

The velocity prior the impact is defined as

\[ \dot{\theta}_1(\theta_-) = \frac{4}{\theta_- - \theta_+}(P_4 - P_3)\dot{\theta}_-, \]

where \( P_3 \) and \( \dot{\theta}_- \) are values to be identified by optimization, and \( P_4 \) was defined in (5.39).

**Step 2: Projection of the impact equation**

Analyzing the impact equation:

\[
\begin{bmatrix}
P_0 \\
\theta_+ \\
\dot{\theta}_1(\theta_+) \\
\dot{\theta}_+
\end{bmatrix}
= F
\begin{bmatrix}
P_n \\
\theta_- \\
\dot{\theta}_1(\theta_-) \\
\dot{\theta}_-
\end{bmatrix},
\] (5.40)

where \( \dot{\theta}_- \) is to be identified by optimization. In order to calculate \( P_1 \) we apply the solution of \( \dot{\theta}_1(\theta_+) \) as

\[ P_1 = \frac{\dot{\theta}_1(\theta_+)}{\theta_+} \frac{\theta_- - \theta_+}{4} + P_0. \]

**Step 3: Integral function**

The integral function is defined as follows,

\[ I(\theta_-, \dot{\theta}_-, \theta_+, \dot{\theta}_+) = \dot{\theta}_-^2 - \Psi(\theta_+, \theta_-) \left[ \dot{\theta}_+^2 - \int_{\theta_-}^{\theta_+} \frac{2\gamma(s, P)\alpha(s, P)}{\alpha(s, P)} ds \right], \] (5.41)

with

\[ \Psi(\theta_+, \theta_-) = \exp \left\{ -2 \int_{\theta_+}^{\theta_-} \frac{\beta(\tau, P)}{\alpha(\tau, P)} d\tau \right\} \] (5.42)

where the remaining parameter \( P_2 \) not involved in properties described before, will achieve a desired boundary for the torque.
Step 4: Optimization criteria

The value $\dot{\theta}_-$ is estimated from some initial guess, so that the next cost function satisfies:

$$J(P_k) = \int_{T_-}^{T_+} \left\| I \left( \theta_-, \dot{\theta}_-, \theta_+, \dot{\theta}_+ \right) \right\|_2^2 \, dt$$

with the functions $\alpha(\theta, P), \beta(\theta, P), \gamma(\theta, P)$ analyzed with the Beziér polynomial parameters.

5.5.4 Results of Optimization

For this particular case the initial parameter were set as a vector\(^2\),

$$\xi = [P_2, P_3, \dot{\theta}_-] = [0.6, -0.25, -2];$$

and applying the Optimization Toolbox from Matlab the Beziér polynomial final parameters are listed below:

<table>
<thead>
<tr>
<th>Beziér curve parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>P₀</td>
</tr>
<tr>
<td>P₁</td>
</tr>
<tr>
<td>P₂</td>
</tr>
<tr>
<td>P₃</td>
</tr>
<tr>
<td>P₄</td>
</tr>
</tbody>
</table>

and the initial velocity is calculated to be $\dot{\theta}_+ = -1.8246$. The intention of the motion is to find a constraint function that allows the stance leg to do most of the motion, bringing the swing leg fastly in front only at the end, and have a period of approximately 0.6 sec. In this way, we let the system to act as an inverted pendulum. A graph of the limit cycle and constraint function is depicted in Fig. 5.8, where a whole walking period is depicted (i.e. stance phase and swing phase).

Further analysis will not be made since the intention is to show the application of this method to find limit cycles and constraint functions parameters.

\(^2\)Initial guess of parameters can be nontrivial.
5.6 Controller Design

As presented in previous chapters, the aim of the controller is to exponentially drive the states of a change of coordinates

\[ y = \theta_i - \phi(\theta), \]

to zero, by applying a transverse linearization

\[ \dot{\zeta}(t) = A(t)\zeta(t) + B(t)v(t), \quad t \in (T^+, T^-) \]

which represents a linearization of the dynamics of the robot, transverse to the desired cycle, and

\[ \zeta(t) = [I, y, \dot{y}]. \]

In the new coordinates the states of the system and time derivatives, together with the partial feedback linearization were defined in section 2.1.2, from which, applying (2.24) we have that

\[ u = \eta(y, \theta)^{-1} \left[ \delta v - \xi(y, \theta, \dot{y}, \dot{\theta}) \right], \]

where

\[ \eta(y, \theta) = [0 \ 1]L^{-1}(\theta, y)M^{-1}(\theta_1, \theta_2)B(\theta_1, \theta_2), \]
and

$$\xi(y, \theta, \dot{y}, \dot{\theta}) = [0 \ 1]L^{-1}(\theta, y)\{M^{-1}(\theta_1, \theta_2)(-C(\theta_1, \theta_2) - G(\theta_1, \theta_2)) - N(y, \theta, \dot{y}, \dot{\theta})\}.$$  

with $M$, $C$, $G$ and $B$ being the matrices defined by (5.4).

From section 2.1.2 it is also seen that the $3 \times 3$ matrix $A(t)$ will have the components:

$$a_{11}(t) = \frac{-2\dot{\theta}_*(t)\beta(\theta_*(t))}{\alpha(\theta_*(t))}, \quad (5.43)$$

$$a_{12}(t) = \frac{2\dot{\theta}_*(t)g_y(\theta_*(t), \dot{\theta}_*(t), 0, 0)}{\alpha(\theta_*(t))}, \quad (5.44)$$

$$a_{13}(t) = \frac{2\dot{\theta}_*(t)g_y(\theta_*(t), \dot{\theta}_*(t), 0, 0)}{\alpha(\theta_*(t))}, \quad (5.45)$$

and the $3 \times 1$ matrix $B(t)$ will have one component calculated as follows,

$$b_{11}(t) = \frac{2\dot{\theta}_*(t)g_y(\theta_*(t), \dot{\theta}_*(t), 0, 0)}{\alpha(\theta_*(t))}.$$

There exists a stabilizing solution $R(t)$ of the Riccati equation

$$\dot{R}(t) + A(t)^T R(t) + R(t) A(t) + G(t) = R(t) B(t) B(t)^T R(t) / \Gamma, \quad (5.46)$$

for all $t \in [T_+, T_-]$, with $G(t)$ being a $n \times n$ positive definite matrix, and $\Gamma > 0$, such that the linear system is exponentially stabilized by the following feedback law

$$v = -\frac{1}{\Gamma} B(t)^T R(t) \zeta. \quad (5.47)$$

By the application of this controller, it is possible to relax the restrictions imposed by linear controllers and motion planning design [1], requiring to find stable zero dynamics and so on. By applying the property of the integral, that defines somewhat a measurement of distance between the states and the cycle, a more efficient controller can be designed. A plot of simulations showing the evolution of the states of the transverse linearization under feedback control is shown in Fig. 5.9. It is clearly seen that the states approach exponentially zero, achieving steady state around 1 sec.
5.7 Conclusions

By this example, it was demonstrated an application of the virtual holonomic constraints approach for achieving dynamic walking in biped robots. It was shown how to define constraint functions, and some properties of them, which can be of advantage to achieve invariance of the hybrid zero dynamics. The author is confident that several other polynomials, splines, trigonometric polynomials, etc, can be used for this purpose. The intention of this chapter was to show one example of the several that there might exist.

A modification to an earlier introduced controller design was applied, achieving exponentially stable walking motions. The advantage of the use of the full transverse linearization is that it allows us to control a larger family of possible motions, not achieved by linear control, e.g. high gain PD controller. This gives the freedom to perform more complicated motions, or to design less restrictive motions in terms of speed, etc.

Figure 5.9: States of the transverse linearization exponentially approaches zero.
CHAPTER 5. A SIMPLE WALKING CASE: THE COMPASS BIPED

The topic of stability of cycles and such matters has been skipped in this chapter. Works related to robots with more degrees of freedom, the case of exponentially stable running and experimental validations in real robots have been started, and they will hopefully be reported in the near future.
Part VI

Appendix
A Modeling the Furuta Pendulum

Let us denote the generalized coordinates for the arm motion by \( \phi \) and for the pendulum as \( \theta \), according to Fig. A.1. At each instant of time, cartesian coordinates of the tip of the arm, tip of the pendulum and of the center of mass of the pendulum are

\[
x_a = l_a \cos \phi, \quad y_a = l_a \sin \phi, \quad z_a = 0,
\]

\[
x_p = x_a - l_p \sin \phi \sin \theta, \quad y_p = y_a + l_p \cos \phi \sin \theta, \quad z_p = l_p \cos \theta,
\]

\[
x_{cm_p} = x_a - l_p/2 \sin \phi \sin \theta, \quad y_{cm_p} = y_a + l_p/2 \cos \phi \sin \theta,
\]

\[
z_{cm_p} = l_p/2 \cos \theta.
\]

The kinetic energy is

\[
T = \frac{1}{2} J_{a} \dot{\phi}^2 + \frac{1}{2} M(x_p^2 + y_p^2 + z_p^2) + \frac{1}{2} m_p(x_{cm_p}^2 + y_{cm_p}^2 + z_{cm_p}^2) + \frac{1}{2} J_p \dot{\theta}^2
\]

101
APPENDIX A. MODELING THE FURUTA PENDULUM

and the potential energy is

\[ V = Mg y_p + m_p g y_{cmp}, \]

where \( m_a \) denotes the mass of the arm, \( J_a \) inertia of the arm, \( m_p \) the mass of the pendulum, \( M \) the mass of the bob attached at the end of the pendulum and \( J_p \) the inertia of the pendulum.

The Lagrangian, expressed in generalized coordinates can be written as

\[ L = T - V \]

resulting in

\[ L = \frac{1}{2} \left( J_a + (m_p + M) l_a^2 + (M + \frac{1}{4} m_p) l_p^2 \sin \theta^2 \right) \dot{\phi}^2 + \left( M + \frac{1}{2} m_p \right) l_a l_p \cos \theta \dot{\phi} \dot{\theta} + \frac{1}{2} \left( (M + \frac{1}{4} m_p) l_p^2 + J_p \right) \dot{\theta}^2 - (M + \frac{1}{2} m_p) g l_p \cos \theta \]

The Euler-Lagrange equation, for the two generalized coordinates \( \phi \) and \( \theta \)

\[ \frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\phi}} \right) - \frac{\partial L}{\partial \phi} = \tau_\phi \]
\[
\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}} \right) - \frac{\partial L}{\partial \theta} = \tau_\theta,
\]
results in the next differential equations

\[
(J_a + (m_p + M) l_a^2 + (M + \frac{1}{4} m_p) l_p^2 \sin^2 \theta) \ddot{\phi} + (M + \frac{1}{2} m_p) l_a l_p \cos \theta \ddot{\theta} + 2 (M + \frac{1}{4} m_p) l_p^2 \cos \theta \sin \theta \dot{\theta} \dot{\phi} - (M + \frac{1}{2} m_p) l_a l_p \sin \theta \ddot{\theta}^2 = \tau_\phi
\]

\[
(M + \frac{1}{2} m_p) l_a l_p \cos \theta \ddot{\phi} + ((M + \frac{1}{4} m_p) l_p^2 + J_p) \ddot{\theta} - (M + \frac{1}{4} m_p) l_p^2 \cos \theta \sin \theta \dot{\theta}^2 - (M + \frac{1}{2} m_p) g l_p \sin \theta = \tau_\theta,
\]

which describe the dynamics of the Furuta Pendulum.

Introducing the notation given in (4.1.1), we obtain the system described by (4.1)-(4.2).
Using a DC motor for controlling the arm’s angular motion, requires knowledge of its electromechanical characteristics. The external torque \( \tau_\phi \) consists of various components, which can be modeled as

\[
\tau_\phi = K_{DC} \cdot u + F_{fr}^\phi
\]

with \( u \) being the control input and \( F_{fr}^\phi \) being the friction torque. The gain \( K_{DC} \) represents the electromotive torque constant expressed in \( Nm/volt \). An experimental approach to identify such parameters was used. Furthermore, the parameters \( \alpha, \beta, \delta, \gamma, \chi \) from (4.1)-(4.2) need to be identified, since approximations given by (4.1.1) might not be in accordance with the real set up.

The estimation is done in two steps: identification of the arm parameters when the pendulum is detached, and identification of the pendulum parameters when the
arm is fixed.

**B.1 Arm and Motor Parameters Identification**

Due to the complex shape of the arm (see Fig. 4.1), the approximated value \( \alpha \) related with its inertia needs to be verified. For this purpose, the pendulum was detached from the setup. With such modification the equation of motion (4.1) of the arm becomes

\[
\alpha \cdot \ddot{\phi} = K_{DC} \cdot u + F_{fr}^\phi
\]  

(B.1)

The friction torque \( F_{fr}^\phi \) in (B.1) is used as presented in section 4.4.3. The value for the parameter \( \alpha \) can be calculated from (B.1) if it is considered that the Coulomb part of the \( F_{fr}^\phi \) is compensated. This assumption leads to the arm dynamics

\[
\alpha \cdot \ddot{\phi} = K_{DC} \cdot (u - u_C) - K_{DC} \cdot b_\theta \dot{\phi}
\]  

(B.2)

The transfer function from input voltage to arm velocity is defined by

\[
\frac{\dot{\phi}}{u - u_C} = \frac{K_{DC}/\alpha}{s + K_{DC} \cdot b_\theta/\alpha}
\]  

(B.3)

If we apply a step input of some magnitude \( u - u_C = u_0 \), the gain \( K_{DC}/\alpha \) can be calculated from the step response dividing the magnitude of the steady state of the velocity \( \dot{\phi} = \dot{\phi}_0 \) by the input level. The identified value for \( \alpha \) will then be

\[
\alpha = \frac{K_{DC}}{\dot{\phi}/(u - u_C)}
\]  

(B.4)

If \( K_{DC} \) is estimated as in section 4.4, applying (B.4) results in \( \alpha = 0.0034 \).

**B.2 Pendulum Parameters Identification**

This experiment considers only the movement of the pendulum rot with the arm fixed. No input is given, meaning that \( \phi \) and its derivatives are zero, and the free
oscillation of the pendulum is recorded. For this experiment, the model (4.1)-(4.2) is reduced to:

\[(\beta + \chi) \cdot \ddot{\theta} - \delta \sin(\theta) = \tau_\theta\]  \hspace{1cm} (B.5)

where the values of \(\beta, \chi, \delta\) are to be identified and the torque \(\tau_\theta\) consists of friction mainly. This equation can be recognized as the standard equation of the pendulum in which \(\beta + \chi\) would represent the inertia of the pendulum and bob, and \(\delta\) is a parameter somewhat related with the torque produced by gravity.

The pendulum is manually positioned at some initial condition \(\theta(0)\) where it is left free, producing oscillations around the downward position \(\pi\). The torque \(\tau_\theta\) can be modeled as a viscous friction part with the damping parameter \(b_\theta\)

\[(\beta + \chi) \cdot \ddot{\theta} - \delta \sin(\theta) + b_\theta \dot{\theta} = 0\]  \hspace{1cm} (B.6)

Next, the parameters are fit to the data using the a nonlinear least square method approach. A plot showing the simulated response with identified values vs. the recorded data is shown in Fig. B.1. This estimation results in the values \(\beta + \chi = 0.0014 \, [kg \, m^2]\) and \(\delta = 0.0617 \, [Nm]\), from which \(\beta=0.0013\). These parameters are clearly close to the calculated ones in (4.11).

![Figure B.1: Simulated Response with estimated values vs recorded data.](image)
Given the state space description
\[ \dot{x} = Ax + Bu, \]
the LQR problem is defined as finding the controller transfer-matrix L that minimizes the cost function
\[ J_u(x(t_0), t_0) = \int_{t_0}^{\infty} [x(t)^TQx(t) + Ru(t)^2]dt \]
where Q is asymmetric positive-definite matrix and R is a positive constant. In the state-feedback version of the LQR problem, we assume that the whole state x can be measured and therefore it is available for control. Solution to the optimal state-feedback LQR problem that minimizes the cost function \( J_u(x(t_0), t_0) \) is the linear feedback law
\[ u = -L \cdot x \]
where

\[ L = R^{-1} B^T P \]

and \( P \) is the stabilizing solution to the Riccati equation

\[ A^T P + PA - PBR^{-1}B^T P + Q = 0 \]

The design parameters are the weighting matrices \( Q \) and \( R \). \( Q \) is used to allocate penalty weights to the states and \( R \) is used to penalize the control signal. When stabilizing the pendulum, our objective is to make \( \theta \) close to zero (indicating large weights) while we care less about the velocities (smaller penalties). In the case of the state space description of the Furuta pendulum model, the control signal weight \( R \) used is set to one, while the matrix \( Q \) is defined as

\[
Q = \begin{pmatrix}
1 & 0 & 0 & 0 \\
0 & 0.1 & 0 & 0 \\
0 & 0 & 5 & 0 \\
0 & 0 & 0 & 0.1 \\
\end{pmatrix}
\]

This design leads to the state feedback vector

\[ L = (-1.0488 \quad -8.0823 \quad -0.5982 \quad -1.1765) \]
Controllability - Furuta Pendulum

Modifying the derivations presented in [37, Appendix D, p. 1174-1175], it is not hard to show that the pair of matrices $A(t)$ and $B(t)$ are completely controllable over the period if and only if the inequality

$$\frac{T^3}{12} \int_0^T f^2(t) dt + T \int_0^T t f(t) dt \int_0^T f(t) dt > \left( \int_0^T t f(t) dt \right)^2 + \frac{T^2}{3} \left( \int_0^T f(t) dt \right)^2$$

for the function

$$f(t) = \frac{b_1(t)}{e^{\int_0^t a_{13}(\tau) d\tau}} - \int_0^t a_{13}(s) e^{-\int_0^s a_{11}(\tau) d\tau} ds$$

111
is satisfied. For each desired trajectory we can verify this condition numerically in order to ensure solvability of (2.22).
Proof of theorem 3

The time derivative of the function $I(\theta(t), \dot{\theta}(t), x, y)$ along a solution of (2.5) is

$$\frac{d}{dt} I = \dot{\theta} \frac{\partial}{\partial \theta} I + \ddot{\theta} \frac{\partial}{\partial \dot{\theta}} I$$  \hspace{1cm} (E.1)

where

$$\frac{\partial}{\partial \dot{\theta}} I = 2 \dot{\theta}$$  \hspace{1cm} (E.2)

$$\frac{\partial}{\partial \theta} I = \frac{2\gamma(\theta)}{\alpha} - \frac{2\beta(\theta)}{\alpha} [I - \dot{\theta}^2]$$  \hspace{1cm} (E.3)

and $\dot{\theta}$ is defined by (2.5). Therefore,

$$\frac{d}{dt} I = \dot{\theta} \left\{ \frac{2\gamma(\theta)}{\alpha} - \frac{2\beta(\theta)}{\alpha} [I - \dot{\theta}^2] \right\} + \frac{u - \beta(\theta)\dot{\theta}^2 - \gamma(\theta)}{\alpha(\theta)} 2\dot{\theta},$$  \hspace{1cm} (E.4)
\[
\frac{d}{dt} I = \dot{\theta} \left\{ \frac{2}{\alpha(\theta)} u - \frac{2\beta \theta}{\alpha(\theta)} I \right\}. \tag{E.5}
\]


