TÉCNICO LISBOA

Nonlinear control of an inverted pendulum

António Samuel Ávila Balula

Thesis to obtain the Master of Science Degree in

Engineering Physics

Supervisor(s): Prof. Dr. João Manuel Lage de Miranda Lemos
Prof. Dr. Horácio João Matos Fernandes

Examination Committee

Chairperson: Prof. Dr. Luís Filipe Moreira Mendes
Supervisor: Prof. Dr. João Manuel Lage de Miranda Lemos
Member of the Committee: Prof. Dra. Rita Maria Mendes de Almeida Correia da Cunha
Prof. Dr. Bernardo Brotas de Carvalho

September 2016

For those who never cease to be curious about how the world works.

Acknowledgments

I would like to thank my supervisor, Prof. Dr. João Miranda Lemos and co-supervisor, Prof. Dr. Horácio Fernandes, for their inexhaustible patience, wise guidance and time invested during the entire course of this thesis. I feel extremely lucky to have worked under their supervision.

Many people have been, in one form or another, important in the development of this work. In particular, Manuel Ribeiro, by helping set up the experimental apparatus, Rui Dias, for his life-saving soldering skills, João Fortunato, for collaborating in the design and production process of the control board, Prof. Dr. Pedro Sebastião for his help finding the correct way of calculating the error of the nonlinear least square fit, Prof. Dr. João Henriques and João Valério, by sharing their knowledge in the field of numerical methods in optimal control. I would also like to thank IST, in particular the physics and electrotechnical engineering departments, IPFN and INESC-ID for their hospitality, material and financial support.

A kind word goes for all my colleagues and friends who shared the last five years of this unique challenge, and made the hard work more enjoyable. In particular to José Carvalho, Filipe Richheimer, Pedro Ribeiro, Eduardo Neto and Rita Tomaz who in several occasions, during the development of this thesis, have helped with their sharp and honest judgement.

Last, but not the least, I am grateful to my family, for their guidance, advice, and support. A special word to my parents, António and Gilda, who made of my education a priority, and contributed more anyone else for the person I am today.

This thesis has been performed in the framework of projects PTDC/EEI-PRO/0426/2014 and UID/CEC/50021/2013, funded by the Portuguese Foundation of Science and Technology (FCT).

Resumo

Um pêndulo de Furuta é um pêndulo de rotação actuado na sua base por um motor de corrente contínua com desmultiplicação. Dois problemas estão associados a estes dispositivos 1) o processo de inversão do pêndulo desde o seu equilíbrio estável, em que o pêndulo se encontra abaixo do seu eixo, até à posição invertida e 2) manter o equilíbrio na posição invertida. Ambos os problemas são resolvidos com controlo óptimo a partir do princípio de mínimo de Pontryagin. No caso da inversão o problema é não linear e um método numérico é usado para encontrar a solução. Este baseia-se na integração das equações do estado e do coestado e da otimizacao da Hamiltoniana em função da variável manipulada.

Já no caso do controlo no equilíbrio é usado um controlador LQG, ativado numa região do espaço de estados próxima da posição invertida, com velocidade nula. Procede-se a um estudo da região de atração deste controlador na presença de limitações da potência do motor. Apresentam-se os resultados de simulações efectuadas com o modelo identificado do sistema, bem como dados experimentais e os sistemas de controlo e algoritmos desenvolvidos e utilizados.

Palavras-chave: Controlo óptimo, Pêndulo de Furuta, Controlo não linear, princípio de mínimo de Pontryagin.

Abstract

The Furuta pendulum is a rotational pendulum that is actuated at its basis by a direct current motor with a gear. Two control problems associated to it consist of swinging-up the pendulum, in order to move it from the downwards up to the upwards position, and then to equilibrate the pendulum in the unstable, upwards position. In both cases, optimal control methods are used. The swing-up problem is solved by formulating it as an optimal control problem with a convenient cost, that is then solved by using a numerical method to approximate the Pontryagin's necessary conditions. The numerical method relies on the iterative solution of the state equation (forwards), of the adjoint equation (backwards), and on the optimization of the Hamiltonian function with respect to the manipulated variable in a grid of time points.

Different aspects related to this problem are considered, that include the selection of an appropriate cost and numerical procedure details. The equilibration problem is solved with a standard LQG controller that is activated within a region of the state-space that is close to the upwards, zero velocity, state. A numerical study of the attraction region of the LQG equilibrating controller is performed, in order to show that this controller will fulfil its objective, even in the presence of a saturation non-linearity in the actuator. The algorithms and control systems developed and used are described, as well as simulation and experimental results.

Keywords: Optimal control, Furuta Pendulum, Nonlinear control, Pontyagin's minimum principle.

Contents

List of Tables

List of Figures

Glossary

CMOS	Complementary Metal-Oxide-Semiconductor
CPS	Cyber Physical System
DAC	Digital to Analog Converter
DAQ	Data Aquisition System
DH	Denavit–Hartenberg
FBS	Forward Backward Sweep
FP	Furuta Pendulum
GP	Gaussian Process
HO	Harmonic Oscilator
I/O	Input/Output
IC	Integrated Circuit
LED	Light Emiting Diode
LPV	Linear-Parameter Varying
LQE	Linear Quadratic Estimator
LQG	Linear Quadratic Gaussian
LQR	Linear Quadratic Regulator
MPC	Model Predictive Control
NLP	Nonlinear Programming
NN	Neural Network
PCB	Printed Circuit Board
PCI	Peripheral Component Interconnect
PC	Personal Computer
PID	Proportional Integral Derivative Controller
PRBS	Pseudorandom Binary Sequence
PWM	Pulse Width Modulation
QE	Quadrature Encoder
RPi	Raspberry Pi Computer
SP	Simple Pendulum
USB	Universal Serial Bus

dsPIC	Digital Signal Processing Programmable Intelligent Computer
ndf	Number of Degrees of Freedom

Nomenclature

Greek symbols

α Angle of the joint between the base and the horizontal arm, FP.

β Angle of the joint between the horizontal arm and pendulum, FP.

Roman symbols

I_{xx1} Moment of inertia component at the centre of mass along the x axis of the horizontal arm, FP.

I_{xx2} Moment of inertia component at the centre of mass along the x axis of the pendulum, FP.

J_0 Moment of inertia at the base joint of the horizontal arm and pendulum, FP.

J_2 Moment of inertia at the joint of the pendulum, FP.

K_f Torque produced by the motor per current unit, FP.

K_t Counter-electromotive force term, coupling the angular speed and current of the motor, FP.

K_{a1} Friction coeficient between base and the horizontal arm, FP.

K_{a2} Friction coeficient between the horizontal arm and the pendulum, FP.

L_b Electric impendance of the motor (imaginary part), FP.

L_{cm1} Distance from axis of rotation to centre of mass of the horizontal arm, FP.

L_{cm2} Distance from axis of rotation to centre of mass of the pendulum, FP.

L_{e1} Lenght of the horizontal arm, FP.

L_{e2} Lenght of the pendulum, FP.

m_1 Mass of the horizontal arm, FP.

m_2 Mass of the pendulum, FP.

R Electric internal resistance of the motor, FP.

Chapter 1

Introduction

The aim of this thesis is to develop a control strategy for the nonlinear problem of swinging-up and balancing a Furuta Pendulum (FP) at its upwards, unstable equilibrium position. Although this can be seen as an academic problem, this device is illustrative of a wide set of dynamic systems with real-life applications, as discussed in 1.1.

A precise definition of the problem is presented in 1.2, followed by a review of the literature available on the subject in 1.3, the main original contributions of this work in 1.4 and a general outline of this document in 1.5.

1.1 Motivation

Physics has been successful in modelling nature phenomena in a surprising number of orders of magnitude. Although in most cases one can only aspire to observe, in a small, but important, subset, it is possible to actively modify the behaviour of the system. This can be achieved by incorporating computational components that interact with the physical system. These Cyber-Physical systems (CPSs) are an emerging area of research, in which this thesis is integrated.

However, it is not trivial to determine the inputs of a system such that a desired performance is achieved. Apart from open-loop instability, limitations are often posed in the number of variables that can be actuated, and on the state information available. Additionally, a complete, high precision model of the system may not be feasible.

Control theory addresses the above challenges, using feedback, a concept that is also present in natural phenomena, ubiquitous in biology, and a powerful tool in the design of CPSs.

This work deals with systems that can be modelled by a finite number of coupled first-order differential equations [1] for a set of variables that define the state of a system *i.e.*, a set of variables that, if known at an instant of time, and together with knowledge about future excitations, fully determines the future behaviour of the system. A particular case are linear systems, for which powerful analytical tools are available, mostly sustained by the superposition principle.

In the case of nonlinear systems, significant progress has been made in recent decades supported by the increasing computational power available as well as by major advances in theoretical knowledge. They are found in a growing number of state-of-the-art applications, therefore the field is an active area of research. Examples include aerospace, industrial, and medical applications [1, 2].

One of the simplest examples of a nonlinear system is the rotary pendulum (also known as the FP), that is commonly used to illustrate emerging ideas in the field of nonlinear control [3]. It is also a prototype of plants with practical importance, such as robot arms with cylindrical geometry, rotary cranes, or even transport systems of tall objects like large dimension rockets. It is both simple enough for educational purposes and useful for testing the design of modern control techniques [4]. As such, both the swing-up and balancing around the unstable equilibrium position are a well studied control problem.

However, to the author's knowledge, the nonlinear problem of swinging-up the pendulum has never been addressed using optimal control. With this technique the control problem is elegantly formulated as the minimization of a convenient cost, subject to restrictions. The cost function reflects the application requirements, and may yield solutions of minimum time, energy or any other quantity [5].

By studying this system one shall be acquainted with the state-of-the-art control techniques developed, their usefulness in similarly structured problems and to deploy a control technique suitable for this class of devices.

1.2 Objectives

The main objective of this thesis is to design a controller based on Optimal Control methods to swing-up a FP and keep it balanced in the vertical upward position, both in simulation and with a real system. This problem can be subdivided into two main parts:

1. Swing-up the pendulum from the downward to the upward position;

2. Equilibrate the pendulum in the upward position.

While the second objective can be addressed using a linear controller, swinging-up the pendulum is an inherently nonlinear problem. Additionally, a supervising unit must select between the controllers, the one that, at each time, is to remain active at different phases of operation.

The switching between controllers points out the importance of considering stability issues. Indeed, not only the inverted pendulum is open-loop unstable, but switching between different controllers may imply instability if proper care is not taken, even if each controller stabilizes the plant inside the operating region for which it is designed.

The system to be used is the *Inverted Pendulum Experiment* mounted on a *Rotary Servo Base Unit* (figure 1.1), both manufactured by *Quanser* [6]. It consists of

- **base**: a geared servo-mechanism. Comprises an electric motor coupled to a gearbox in a solid aluminium frame. Sensors are available for measuring the position (potentiometer and optical encoder) and speed (tachometer) of the output shaft.

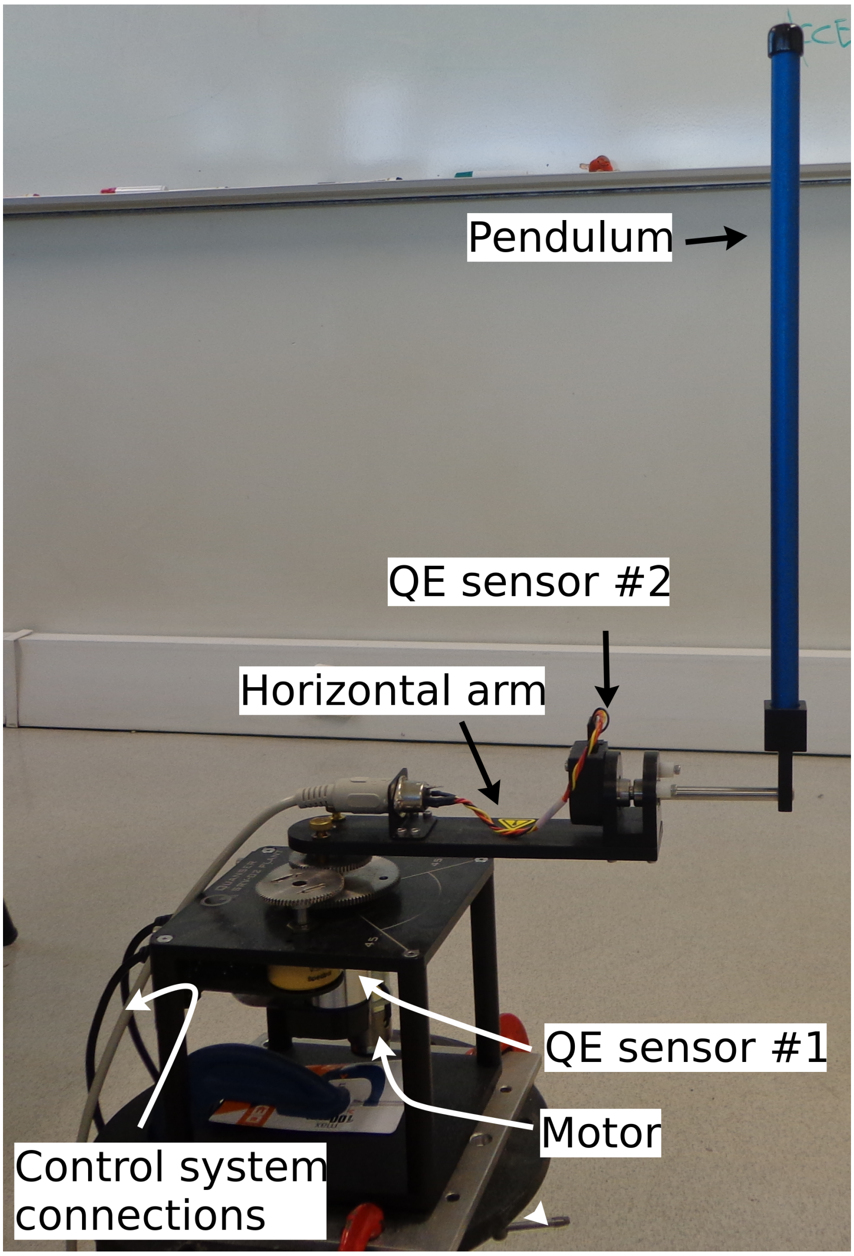


Figure 1.1: Furuta Pendulum experimental apparatus with legends: an *Inverted Pendulum Experiment* mounted on a *Rotary Servo Base Unit*, both manufactured by *Quanser*

- **horizontal arm**: a flat solid aluminium link, mounted on the output shaft of the base with a pivot, and terminating on a metal shaft, whose position is measured by a sensor (digital encoder).

- **pendulum**: a cylindrical aluminium link, mounted on the metal shaft of the horizontal arm.

It is stressed that the control algorithms developed in this thesis are tested not only in simulation but in the actual physical system as well.

1.3 State-of-the-art

This section provides a brief overview of the most relevant methods to perform linear and nonlinear feedback control. The FP is used as case study, since the literature on inverted pendulums control is rich and comprises different types of problems and pendulum structures, including for example simple and multiple planar pendulums with an actuator in the base joint [7], mounted on top of a moving cart [4, 8], and rotating pendulums with several geometric variations. Each of the problems of equilibrating and swinging up the rotating pendulum are addressed.

For a brief history of feedback control see [9, pp. 9-15].

1.3.1 Modelling

Models are a key component to controller design. In addition, they allow to simulate the system and test control approaches in a preliminary phase. In some cases, such as nonlinear Model based Predictive Control (MPC), the control algorithm requires a plant model to be embedded in the decision taken in real time about the value of the manipulated variable. To evaluate global performance, the models needed to simulate the system may be build from first principles – basic laws of mechanics and the geometry of the system, and hence are nonlinear. Although simulation models can be obtained by applying the principles of Newtonian mechanics, it is simpler and much more systematic to use an Euler-Lagrange approach. Furthermore, this approach paves the way to variational methods such as optimal control or to design methods that rely solely on energy shaping.

For controller design, different model types are required, depending on the approach followed. Some methods such as energy shaping, optimal control or nonlinear MPC require nonlinear models. In the swing-up problem the system follows a trajectory in state-space that passes through several operating regions with different dynamics. Indeed, while in the downwards vertical position the pendulum is close to an asymptotically stable equilibrium, the upwards position corresponds to a saddle point. Hence, a model able to tackle the representation of large excursion signal is needed.

In opposition, equilibrating the pendulum in the vertical upwards position requires only a controller that is able to work in a local region, and therefore attaining this objective requires only a linear model. Linear models that are valid around an operation region can be obtained by two ways:

- By Jacobian linearization of the nonlinear model obtained from first principles;
- By identification techniques, in which the linear model parameters are estimated from plant data.

In the present situation, the use of identification techniques is complicated by the fact that the upwards position is open-loop unstable. Hence, if a small disturbance is applied to the input, the output does not respond with small changes, but instead the output signal will grow without bound. To avoid such a problem, one must collect data with a stabilizing controller applied to the plant, and a disturbance signal applied to the manipulated variable.

The above methods fall in the realm of closed-loop identification [10], for which there is a very rich literature, developed mainly in the end of the 1960's, beginning of 1970's, but with significant advances, such as sub-space identification and sparse identification, obtained more recently. The main difficulty is the fact that the Input/Output (I/O) plant data to be used for identification is contaminated by sensor noise that is reinjected in the manipulated variable by the stabilizing controller. To tackle this difficulty one may remove most of this noise with an instrumental variable like method [11]. However, if the noise level is low, no special data pre-processing is needed and a parameter estimation method as Maximum Likelihood or even Least Squares may be used.

Concerning models, one must also refer nonparametric models like neural networks (NN) [12] and Gaussian Processes (GP) [13]. These methods allow the approximation of nonlinear I/O relations (including dynamic relations in the sense that the present sample of the output depends from past samples) from plant data. NNs do this approximation by a combination of sigmoidal functions. GPs rely on gaussian kernels that specify the auto-correlation function of gaussian stochastic processes.

Non-parametric models can be used in situations where there is few knowledge about the stucture of the problem. However, they have serious drawbacks when considering the control of fast open-loop unstable processes like the inverted pendulum. These drawbacks come from the fact that these models imply a heavy computational load, while training them is difficult for open-loop unstable processes. Nevertheless, a significant literature is available concerning their use in control applications, see *e.g.* [14] and references therein.

From another point of view, a model can be classified accordingly to the time domain in which they operate as

- Continuous time
- Discrete time with constant sampling rate
- Discrete time with time varying sampling rate

Continuous time models are directly obtained from the application of basic physical principles, and amount to a set of ordinary differential equations that can be converted to first order with the structure of the state model

$$\dot{x} = f(x, u), \tag{1.1}$$

where $x \in \mathbb{R}^n$ is the state vector, $u \in \mathbb{R}$ is the manipulated variable.

1.3.2 Nonlinear control

Energy control

One of the most common methods to perform the swing-up of an inverted pendulum is energy control. In this case the energy of the system is controlled instead of controlling directly its position and velocity [3]. At the end of the procedure the system converges to a homoclinic orbit that drives the state to the unstable equilibrium position [7]. Dissipative forces are generally not considered.

Energy Shaping and passivity based control

Energy shaping concerns with balances of energy, and is on the essence of passivity based control (PBC). It aims to make the closed-loop system passive, with adequate dissipation, using damping injection [15]. An energy function of the closed loop system is constructed in such a way that it exhibits a minimum at the goal state.

This technique has been used to perform the swing-up of the FP [16], in which feedback is used to transform the dynamics of the FP into the ones of a planar pendulum on a cart plus a gyroscopic force. In this work the proposed control law for the swing-up is tested in simulation and with the real device.

Feedback linearisation

With feedback linearisation the non-linearities of a system are compensated by an appropriate feedback law and state variable transformation [17, 18]. The transformation is required to be a diffeomorfism, ensuring that the transformed and original systems are equivalent. The I/O response of the transformed system is equivalent to the one of a linear system, and can thus be controlled with any of the usual techniques used in linear control.

The transformation of a nonlinear into a equivalent linear system has clear advantages. However, there are also some drawbacks to this approach. Not all systems can be feedback linearised, although it often still possible to perform a partial linearisation. In some cases unstable zero dynamics may arise from the transformation, *i.e.* the existence of states that are not observable, and that may grow without bound. In such an event the control becomes unusable. Additionally, Feedback linearisation requires an accurate model of the system in order to control it, which may not be be feasible.

Control Lyapunov functions

Aleksandr Lyapunov published his work [19] in 1892, which has been applied to the control context by Kalman and Bertram in 1960 [9], and where the direct and indirect (see 2.3.4) Lyapunov's methods are presented to assert the stability of a dynamic system.

Lyapunov direct method states that if there is a function $V(\mathbf{x}) : \mathbf{x} \in \mathbb{R}^n \rightarrow \mathbb{R}$ positive definite, and continuously differentiable in a neighbourhood of $\mathbf{x} = \mathbf{0}$, and if $\dot{V}(x) \leq 0$, then the system is locally stable (locally assimptotically stable if $\dot{V}(x) < 0$).

On systems with control inputs, the existence of a control Lyapunov function implies that exists a control input $u(\mathbf{x}, t)$ able to take the system from any $\mathbf{x} \neq \mathbf{0}$ to the origin. In that case

$$\forall\, \mathbf{x} \neq \mathbf{0},\ \exists\, u :\ \dot{V}(\mathbf{x}, u) = \nabla V(\mathbf{x}) \cdot f(\mathbf{x}, u) < 0, \tag{1.2}$$

and a control law can be designed such that at every instant $V(x)$ decreases [20]. For example, by choosing u that minimizes $\dot{V}(\mathbf{x}, u)$.

Finding a Lyapunov function relies on ingenuity and luck, as there is no general rule to its construction.

Model Predictive Control

Model predictive control (MPC) uses a model of the system to make predictions of future outputs (see [21] and references therein). The finite horizon of discrete predictions is optimized in respect to some objective function *i.e.*, the control input is calculated for k future time instants such that the objective function is minimized. Usually only the first control input is applied, before repeating the optimization in the next iteration, a process known as receding horizon. This allows the measurements produced by sensors to be taken into account. A data fusion algorithm produces an estimate of the current state of the system, which is fed to the optimization process, as the initial condition. As initial guess for the control inputs, the previous result, shifted in time, can be used, which allows a more efficient computation.

MPC has historically been used in processes where the sampling rate is slow enough to allow the computations in real time, for example in some chemical processes, and with simple linear models. The outputs of the method were usually the set point for other controllers, such as PID's. The recent growth in computational power has made its use possible in more complex systems, with high sampling rates.

This technique has been used to perform the swing-up of the FP [22].

Other methods

The list of methods presented above is not exhaustive, several other techniques have been used to perform control on this classes of devices. Other examples are minimum attention control, which explores what is the least computational power needed to perform control, and non-parametric methods, which rely on a statistical description of the system. It is also possible to design a nonlinear controller with a global atraction point at the desired value.

1.3.3 Trajectory planning

It is often convenient to design trajectories offline, *i.e.* to define the path in state space that the system shall perform beforehand. In this situation one is not restrained by the time or computational power required to do such calculations during the process itself, which is particularly useful in complex systems with fast dynamics. The trajectories are designed using a model of the system, by analysing its response to the available inputs. This strategy is used extensively in robotics, where there is the need to plan the motion of mobile robots in confined spaces. Examples include artificial potential fields, Roadmap methods, and Dubbins path [23].

1.3.4 Optimal control

Optimal control deals with the problem of a determining the inputs of a dynamic system that optimize, *i.e.* minimize or maximize, a specified performance index [24]. With an appropriate choice of this cost function, trajectories and control laws can be determined such that they satisfy the design requirements.

The brachystochrone problem proposed in 1697 by Johann Bernoulli is considered to be the cornerstone of optimal control [25]. The brachystochrone is the curve of fastest descent of a body between two points at different heights and subject to a uniform gravitational field. The work of Euler, a student of Bernoulli, and Langrange, eventually led to the development of general techniques and the calculus of variations.

In 1962 Lev Pontryagin published [26] an extension of the calculus of variations subject to inequality constrains, known as Pontryagin's minimum principle (see 3.2). This theory gives necessary conditions for optimality with piecewise, continuous solutions, and led to the development of the field of optimal control.

The linear control case has well known solutions (see 4.1).

1.3.5 Numerical methods in optimal control

The solution of most real-life nonlinear optimal control problems relies on numerical methods. A wide variety of methods have been proposed (see [24] and the references therein for an extensive survey, and [27] for example applications and issues), that can be divided in two major classes – indirect and direct methods. Both share the need to integrate the equations that model the system, but perform the optimization process in two distinct ways.

Direct methods transform the optimal control problem into a nonlinear programming (NLP) problem. A general NLP solver is then employed to find the optimal control.

Indirect methods, in which this work focus, rely on Pontryagin's minimum principle. This approach leads to a multiple-point boundary value problem, whose solutions are extremals, a necessary condition for optimality.

In the realm of indirect methods, forward-backward sweep (FBS) provides a straightforward manner of finding optimal solutions. It relies on the forward integration of the state equations. The terminal conditions of the integration are used to find the initial conditions for the time backwards integration of the costate equations. Repeating this procedure leads to an approximate solution for certain problems.

In this work, FBS was found to be unsuited for solving the problem at hand, and a modification of the method is proposed (see 3.3).

With shooting methods a guess is made for the unknown boundary conditions. The equations are then integrated, either forward or backward in time. The boundary values found with the integration are compared with the initial guesses and used to correct them. The successive application of this methodology leads to an increasingly accurate solution for the problem. An useful analogy to understand this method is the firing of a cannon. A first shot is performed with a initial guess for the tilt angle of the cannon. Not having hit the target, the angle is corrected taking into account the observed error. To overcome difficulties when the time domain is long in comparison with the dynamics of the system, multi-shooting methods have been developed, where the time interval is divided in subintervals. In this case the boundary conditions between each subinterval are set such that the solution is continuous.

Pseudo-spectral methods are a strategy to perform global orthogonal collocation. While local collocation methods maintain the degree of the polynomial and vary the number of time segments, with pseudo-spectral methods the number of time segments is kept fixed, and the degree of the polynomial is allowed to vary. The state is approximated by a projection on an orthogonal base, such a Lagrange polynomials.

1.3.6 Linear control

Linear control is a well established subject, with several technologies available that can be employed on the design of controllers and analysis of the resultant system. A classical proportional–integral–derivative controller (PID) is generally the simplest solution, and the most commonly used in industrial applications. However it does not yield the best possible results and, in some cases, it may not even be able to stabilize the plant. A state space representation, often used in modern control theory, offers more versatility to the design. Both pole placement and a linear quadratic regulator use this methodology.

Linear controllers are often used in a cascade control configuration, where they follow a set point, provided by higher level controller.

PID controller

The PID controller consists of three terms, that take the error $e(t) = r(t) - y(t)$ as an input, and produce the control input to the plant

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt}. \tag{1.3}$$

The proportional term (P) compensates for the current error, and outputs a response proportional to it. The integral term (I) takes into account the past errors, and is useful to compensate for steady state errors. Finally, the differential (D) term produces an anticipatory response, based on the current rate of change of the output [9].

Not all the three terms are required to perform control, and often a P, PI or PD is enough to honour the performance requirements of the controller. The design of the weights can be made manually or with more systematic techniques, such as Ziegler–Nichols's method [28] or other rules that lead to a smaller overshoot.

PID controllers do not rely on a model of the system to be controlled, but only on the measured process variable. Thus, although they are able to control a wide variety of processes with satisfactory results, they are sub-optimal and fail to control systems with large time delays, high order dynamics or strong nonlinearities.

An important aspect is that (1.3) only describes the general structure of the PID. For a pratical application a number of features, some of which are common to other linear controllers, have to be included. These features comprise

- anti-reset of the integral effect, in the presence of actuator limits,
- bumpless control transfer when switching between manual control and automatic control,
- low-pass filtering of the derivative effect

Pole placement and root-locus

For a linear system, the position of the roots (poles and zeros) of its characteristic equation determine its frequency response. A feedback gain alters the position of the roots of the closed-loop characteristic polynomial, and can be calculated such that they are positioned appropriately. Root locus can be used to calculate the effect of loop gain variations in the position of the roots of the closed loop system [9].

Linear Quadratic Gaussian controller

The LQG controller has been applied in this work to the FP (see 4.1 for further detail).

1.3.7 Estimation of asymptotic stability regions

An important consideration with linearised control is the region in which the system is stabilized. Theoretical results allow to compute estimates of the boundaries to the region of convergence [29].

1.3.8 Trajectory control

Different techniques can be used to perform the following of a reference in a non-linear system.

Gain Scheduling

With gain scheduling different controllers are used at different regions of state space, selected by scheduling variables [30]. Designing a gain shelling controller requires linear models of the system, obtained for example with Jacobian linearisation of the non-linear model at different points of operation, or Linear-Parameter Varying (LPV), where the linear model changes with an indicator variable. A family of controllers is designed for the set of linear models, and scheduled to operate when appropriate.

1.4 Original Contributions

This work focuses on the application of optimal control to the problem of making a FP, a nonlinear dynamic system, perform transitions between operating points, and also to maintain the system working at an unstable operating point.

A complete model of the FP is developed. It incorporates the characteristic of the electric motor. The parameters of the model are identified form experimental data, with a modified version of least squares that takes into account known information about the values of the parameters of the system.

A numerical method is dreveloped as a modification of the steepest descent method. This method yields the optimal nonlinear control for the transition between operation points, with improved convergence over the simple forward-backward integration. The method is tested in simulation and experimentally with a real FP, and in simulation with two other devices.

A technique is developed to find convex sets of points in state space in which a controller is able to stabilize a device. This method is applied to the linear controller used in the upwards position of the FP, described by the nonlinear model developed previously.

A custom made printed circuit board and software has been designed to control the FP. In comparison with the previously available hardware at the laboratory it has an overall cost 2 orders of magnitude below, and improved sensing capabilities. This board provides 4 times more precise readings from the measurements taken by the optical encoders, direct measurement of the motor current and angular speeds. It also comprises a standardized interface over a USB port, making it compatible with any modern computer.

1.5 Thesis Outline

In 2 a accurate model of the system is derived and its parameters identified from experimental data. With Jacobian linearisation, continuous and discrete models of the system are found for several operation points.

Trajectories to perform the swing-up of the pendulum are designed in 3, with both ad hoc methods and optimal control. Trajectory following is detailed in 4, where the design of the linear controllers is presented.

In 5, the design of a custom made board for the control of the FP is designed, and compared with the previously available commercial solution, available at the laboratory.

In 6 the main results of this work are presented. Finnaly in 7 the conclusions are provided, as well as a list of future work.

Chapter 2

Model

This chapter presents the linear and non linear models developed for the FP, along with parameter identification.

The non linear model of the FP is constructed from the geometry of the system using Lagrangian mechanics [2.1]. Model parameters are identified from experimental data [2.2] with least squares method using weighting and regularization. Continuous linear models valid around several operating points are calculated by Taylor expansion of the nonlinear model [2.3]. These are then subject to discretisation, with fixed time step [2.3.1]. Stability, controllability and observability are evaluated for each linear model.

2.1 Nonlinear model derivation using Lagrangian mechanics

The energy in the base frame of a N rigid body system can be written as the sum of its kinetic energy $\mathcal{T}$

$$\mathcal{T} = \sum_{i=1}^{N} \left(\frac{1}{2} M_i |\mathbf{v}_i|^2 + \frac{1}{2} \boldsymbol{\omega}_i I_i \boldsymbol{\omega}_i^T \right), \tag{2.1}$$

and potential energy $\mathcal{V}$

$$\mathcal{V} = \sum_{i=1}^{N} M_i \mathbf{r}_i \cdot (-\mathbf{g}), \tag{2.2}$$

where, for each rigid body i, M_i is the mass, $\mathbf{v}_i$ is the velocity of the centre of mass, I_i is the inertia matrix, $\boldsymbol{\omega}_i$ is the angular velocity, $\mathbf{r}_i$ is the position of the centre of mass, and $\mathbf{g}$ is the gravity acceleration.

To calculate explicitly (2.1) and (2.2) from the angles of the joints, a systematic approach from robotics was used: a frame was assigned to each rigid body and transformation matrices T written using the Denavit-Hartenberg (D-H) convention [31, p. 76].

$${}^{i-1}_{i}T = M_{DH}(a_{i-1}, \alpha_{i-1}, d_i, \theta_i) = \begin{bmatrix} c_{\theta_i} & -s_{\theta_i} & 0 & a_{i-1} \\ s_{\theta_i} c_{\alpha_{i-1}} & c_{\theta_i} c_{\alpha_{i-1}} & -s_{\alpha_{i-1}} & -s_{\alpha_{i-1}} d_i \\ s_{\theta_i} s_{\alpha_{i-1}} & c_{\theta_i} s_{\alpha_{i-1}} & c_{\alpha_{i-1}} & c_{\alpha_{i-1}} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{2.3}$$

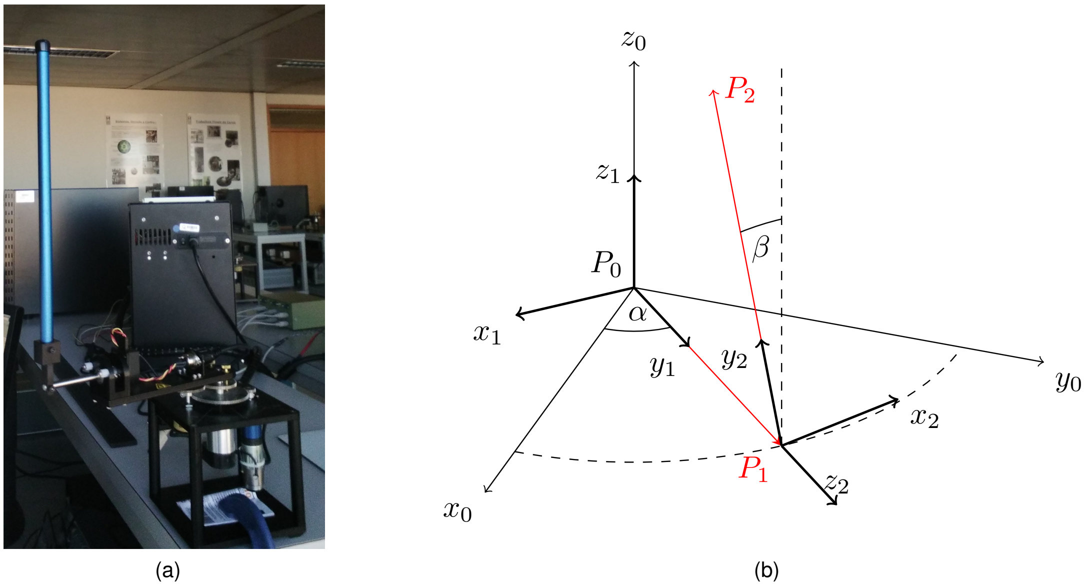


Figure 2.1: Conventions used for angle and frame placement. The physical elements of the experimental apparatus (a) are represented schematically in (b) where the horizontal arm is the red line segment that goes form P_0 to P_1 and the pendulum the one that goes from P_1 to P_2. Angles α and β are the angles of the joints; angle β is measured relative to the dashed vertical line segment that departs from P_1.

where $c_x = \cos x$, $s_x = \sin x$, and a_{i-1},α_{i-1},d_i,θ_i are the D-H parameters. This matrix transforms points in frame i to $i-1$, has the structure of the general transformation

$$ {}^{m}_{n}T = \begin{bmatrix} & {}^{m}_{n}R & & {}^{m}P_{n\ org} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{2.4}$$

and can be inverted with

$$ ({}^{m}_{n}T)^{-1} = \begin{bmatrix} & {}^{m}_{n}R^{T} & & -{}^{m}_{n}R^{T}\ {}^{m}P_{n\ org} \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}^{n}_{m}T, \tag{2.5}$$

where ${}^{m}P_{n\ org} \in \mathcal{M}_{3\times 1}(\mathbb{R})$ describes the translation between frames (these are the coordinates of the origin of frame n measured in frame m) and ${}^{m}_{n}R \in \mathcal{M}_{3\times 3}(\mathbb{R})$ describes the transformation from frame n to m due to the relative angle between them.

A point written in frame n can be rewritten in m as

$$ {}^{m}P = {}^{m}_{n}T(\theta_m, \ldots, \theta_{n-1})\ {}^{n}P = \left(\prod_{k=m}^{n-1} {}^{k}_{k+1}T(\theta_k) \right)\ {}^{n}P. \tag{2.6}$$

In the particular case of the Furuta pendulum, frames were assigned as described in figure 2.1.

Using (2.3)

$$
{}^{0}_{1}T(\alpha) = M_{DH}\left(0,0,0,\alpha(t)-\frac{\pi}{2}\right) = \begin{bmatrix} s_{\alpha(t)} & c_{\alpha(t)} & 0 & 0 \\ -c_{\alpha(t)} & s_{\alpha(t)} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{2.7}
$$

$$
{}^{1}_{2}T(\beta) = M_{DH}\left(0,-\frac{\pi}{2},L_{e1},\beta(t)\right) = \begin{bmatrix} c_{\beta(t)} & -s_{\beta(t)} & 0 & 0 \\ 0 & 0 & 1 & L_{e1} \\ -s_{\beta(t)} & -c_{\beta(t)} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{2.8}
$$

with which points can be expressed in the base frame as

$$
{}^{0}P_1 = {}^{0}_{1}T\,{}^{1}P_1, \tag{2.9}
$$

$$
{}^{0}P_2 = {}^{0}_{1}T\,{}^{1}_{2}T\,{}^{2}P_2, \tag{2.10}
$$

and angular velocities can be written as

$$
\boldsymbol{\omega}_1 = \begin{bmatrix} 0 \\ 0 \\ \dot{\alpha} \end{bmatrix}, \tag{2.11}
$$

$$
\boldsymbol{\omega}_2 = {}^{2}_{1}R\,\boldsymbol{\omega}_1 + \begin{bmatrix} 0 \\ 0 \\ \dot{\beta} \end{bmatrix} = \begin{bmatrix} \dot{\alpha}\ \sin\beta \\ \dot{\alpha}\ \cos\beta \\ \dot{\beta} \end{bmatrix}. \tag{2.12}
$$

Knowing the physical dimensions of the system one can use (2.9) and (2.10) to calculate the positions and velocities of the centres of mass of the rigid bodies of the system. By replacing (2.9) to (2.12) in (2.1) and (2.2) one gets the kinetic and potential energy of the system as a function of its joint angles and angle time derivatives.

Both bodies were considered to have cylindrical symmetry around the y axis. This is consistent with the geometry of the pendulum, and effective for the arm, since it performs only horizontal movements. Due to the cylindrical symmetry and the axis alignment with the principal inertia axis, the inertia matrices are diagonal and $I_{xx} = I_{zz}$. Due to the high height to radius ratio, I_{yy} can be neglected, thus making for $i = \{1,2\}$

$$
I_i = \begin{bmatrix} Ixx_i & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & Ixx_i \end{bmatrix} \tag{2.13}
$$

The Lagrangian $\mathcal{L}$ can readily be computed

$$
\mathcal{L} = \mathcal{T} - \mathcal{V}, \tag{2.14}
$$

Table 2.1: State variables dictionary

State Variable	Variable	Definition
x_1	α	Horizontal arm angle
x_2	$\dot{\alpha}$	Horizontal arm angular speed
x_3	β	Pendulum angle
x_4	$\dot{\beta}$	Pendulum angular speed
x_5	i	Motor Current

and the equations of motion of the system are calculated with the Euler-Lagrange equations

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\alpha}}\right) - \frac{\partial \mathcal{L}}{\partial \alpha} = -K_{a1}\ \dot{\alpha} + K_f\ i, \tag{2.15a}$$

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\beta}}\right) - \frac{\partial \mathcal{L}}{\partial \beta} = -K_{a2}\ \dot{\beta}, \tag{2.15b}$$

where viscous dissipative forces are considered proportional to angular velocities. Explicitly,

$$\begin{cases} \ddot{\alpha}\left(J_0 + J_2 \sin^2\beta\right) + \dot{\alpha}\dot{\beta}\ J_2 \sin(2\beta) + \dot{\beta}^2\ L_{cm2}L_{e1}m_2 \sin\beta - \ddot{\beta}\ L_{cm2}L_{e1}m_2\cos\beta = -\dot{\alpha}K_{a1} + i\ K_f \\ -\ddot{\alpha}\ L_{cm2}L_{e1}m_2\cos\beta - \dot{\alpha}^2\ \frac{1}{2}J_2\sin(2\beta) + \ddot{\beta}\ J_2 = -\dot{\beta}K_{a2} + g\ L_{cm2}\ m_2\sin\beta \end{cases}, \tag{2.16}$$

where $J_0 = I_{xx1} + L_{cm1}^2 m_1 + L_{e1}^2 m_2$ and $J_2 = I_{xx2} + L_{cm2}^2 m2$.

The torque performed on the arm is set proportional to the current in the actuator, whose electric characteristic is given by

$$L_b\frac{di}{dt} + K_t\frac{d\alpha}{dt} + R\ i = u. \tag{2.17}$$

In (3.4), second derivatives are found $\{\ddot{\alpha}, \ddot{\beta}\}$. In order to construct a nonlinear state-space model of the system in the form

$$\dot{x} = f(x, u), \tag{2.18}$$

the variables are substituted with the rules defined on table 2.1,

where $f(x, u)$ is a nonlinear function on $\mathbb{R}^5 \times \mathbb{R}^1 \to \mathbb{R}^5$.

Explicitly

$$\begin{cases} \dot{x}_1 = & x_2 \\ \dot{x}_2 = & \Big\{ - J_2\big[K_{a1}x_2 - K_f x_5 + x_4(L_{cm2}L_{e1}m_2x_4 + 2J_2x_2\cos x_3)\sin x_3\big] \\ & +L_{cm2}L_{e1}m_2\cos x_3\big[- K_{a2}x_4 + (gL_{cm2}m_2 + J_2x_2^2\cos x_3)\sin x_3\big]\Big\}\Big/ \\ & \big[- L_{cm2}^2L_{e1}^2m_2^2\cos^2 x_3 + J_2(J_0 + J_2\sin^2 x_3)\big] \\ \dot{x}_3 = & x_4 \\ \dot{x}_4 = & \Big\{K_{a2}x_4 - gL_{cm2}m_2\sin(x_3)(J_0 + J_2\sin^2 x_3)+ \\ & \cos x_3\big[(-J_0J_2x_2^2 + L_{cm2}^2L_{e1}^2m_2^2x_4^2)\sin x_3 - \\ & J_2^2x_2^2\sin^3 x_3 + L_{cm2}L_{e1}m_2(K_{a1}x_2 - K_f x_5 + J_2x_2x_4\sin(2x_3))\big]\Big\}\Big/ \\ & \big[L_{cm2}^2L_{e1}^2m_2^2\cos^2 x_3 - J_2(J_0 + J_2\sin^2 x_3)\big] \\ \dot{x}_5 = & (-K_t x_2 - Rx_5 + u)/L_b \end{cases} . \tag{2.19}$$

The above equations were found symbolically with the software *Mathematica*. The result was found to be in agreement with [32].

2.1.1 Approximations and limitations

The following dynamic properties of the system were modeled in a simplified manner or ignored entirely:

1. Non-electric dissipative forces were modeled as viscous friction proportional to the angular speed of the joints. Although this term describes the most effective dissipative forces, the addition of other terms could allow a more precise description of this effects. For example, static friction, viscous friction due to the air to body relative speed, and higher order terms of viscous friction could be considered.

2. The system is build of mechanical parts linked by screws and gears. Although mechanical gaps and backlash exist, the coupling between bodies was considered perfect. This features affect the stability and precision of the control loop.

3. The inertia matrices of the system were simplified, as presented in (2.13).

4. In long experiments, the temperature of the actuator increases, changing its electrical characteristic. This effect was not taken in to account.

5. Last, the model is of lumped parameters type. This choice implies to neglect wire capacitance and indutance distributed over space, a reasonable assumption given the range of frequencies considered. In the same vein, mechanical parts are considered to be perfectly rigid.

Although additional terms can be found to describe such effects, in practice they add complexity to an already long model and were found to add significant computational effort to adjust the variables and perform simulations. Since the model was found to comply with experimental data within the required precision, these terms were discarded.

2.1.2 Auxiliary models

For the implementation of new algorithms or when performing certain analysis, it is useful to consider similar but simpler problems that are used for preliminary tests or to gain intuition. As such, the following were considered

Harmonic Oscillator

The harmonic oscillator is a system that, when displaced from equilibrium, is pulled back into it, by a force proportional to the displacement. In the absence of friction, the equation of motion is

$$\ddot{x} + \omega_0^2 x = u. \tag{2.20}$$

This system is very common in nature. For example, it can be used to describe a mass-spring system or a pendulum with small oscillations around its stable fixed point.

Simple inverted pendulum

A pendulum subject to wide movements may only be described by a nonlinear model. Taking the angle reference to be the unstable fixed point, the equation of motion is

$$\ddot{x} - \omega_0^2 \sin x = u. \tag{2.21}$$

These auxiliary problems provide a means to test the algorithms. They require reduced computational loads but provide insight into the more complicated, central problem.

2.2 Parameter estimation

Some of the parameters of the model may be readily measured, while others, such as the friction coefficients or inertia moments, might require disassembling the apparatus to enable a precise measurement. Apart from using the manufacturer standard values for physical quantities, experimental data (from the sensors available in the device) enables the estimation of all relevant parameters.

Parameter estimation from experimental data can be performed with a least squares method. However, due to the high number of free parameters in the model and numerical instabilities for certain sets of parameters, it is convenient to take into account the known information provided by the manufacturer.

Let the model to be identified be written as

$$\dot{x} = f(x, u, \theta), \tag{2.22a}$$

$$y = Cx, \tag{2.22b}$$

where $x \in \mathbb{R}^n$ is the state, $u \in \mathbb{R}$ is the input, $y \in \mathbb{R}^{n_p \times n_y}$ is the matrix of observations obtained from the sensors with n_p points at each of the n_y channel, and $\theta \in \mathbb{R}^{n_\theta}$ is the vector of parameters to estimate.

Assuming that observations follow a normal distribution, the Bayesian estimate is obtained [33] by minimizing J_N with respect to θ:

$$J_N(\theta) = \sum_{i=1}^{n_\theta} \left(\frac{1}{\sigma_i} (\theta_i - \bar{\theta}_i)^2 \right) + \sum_{j=1}^{n_p} \sum_{k=1}^{n_y} \left(\frac{1}{\sigma_{jk}^2} \left(y_{jk} - \hat{y}_{jk}(\theta) \right)^2 \right), \tag{2.23}$$

where the first term is a prior, that penalizes deviations of the parameters from the reference value. The weights σ_i measure the confidence on the value of the prior estimate. The second term accounts for the differences between the model output and the experimental data, for a set of n_p points with dimensions n_y. The weights σ_{jk} measures the confidence on the experimental point j of output channel k.

Note that the system is described by a set of differential equations in the form of a nonlinear model (2.18) for which there is no known closed form solution. For that reason, the differential equation system must be integrated at each iteration of the least squares with a suitable numerical integration method, that provides the numerical values $\hat{\mathbf{y}}_j(\theta) = C\hat{\mathbf{x}}_j(\theta)$ for the same times t_j as the experimental data $\mathbf{y}_j$.

2.2.1 Parameter estimation results

Data acquisition for the identification of the model parameters was performed in three sequential steps

1. The input variable was set to zero and the pendulum moved manually to the upwards, unstable equilibrium position, corresponding to $\beta = 0$, from where it was allowed to move freely.

   This data was first used to validate the model. Simulations performed with the manufacturer parameters were compared with experimental data. Although a qualitative agreement was found, significant quantitative differences were also present. Next, a least squares fit was performed, but making the motor parameters fixed.

2. A PRBS signal was applied to the input, resulting in movements around the stable equilibrium position.

   This experiment allowed for the identification of all parameters, including the motor ones. The parameters obtained from this fit were used to produce a linearised model, valid around the upwards, unstable equilibrium position. A swing-up[1] and a linear[2] closed loop controllers were developed for the system and tested in simulation.

3. The input was controlled by a swing-up and an equilibrium closed loop controllers, taking the pendulum from the downwards to the upwards position.

   This data acquisition setup provided a broader state-space working region then the two previous steps, since in this case the pendulum travels from the downwards to the upwards equilibrium

[1] Swing-up was performed using the ad-hoc strategy proposed in [22] and described in 3.1.2
[2] Equilibrium around the upwards position was performed with a Linear-Quadratic-Gaussian controller, described in 4.1

position while the input is composed by fast and slow changes. The fit performed with this data set (figure 2.2) provides the vector of parameters used in the rest of this work (table 2.2).

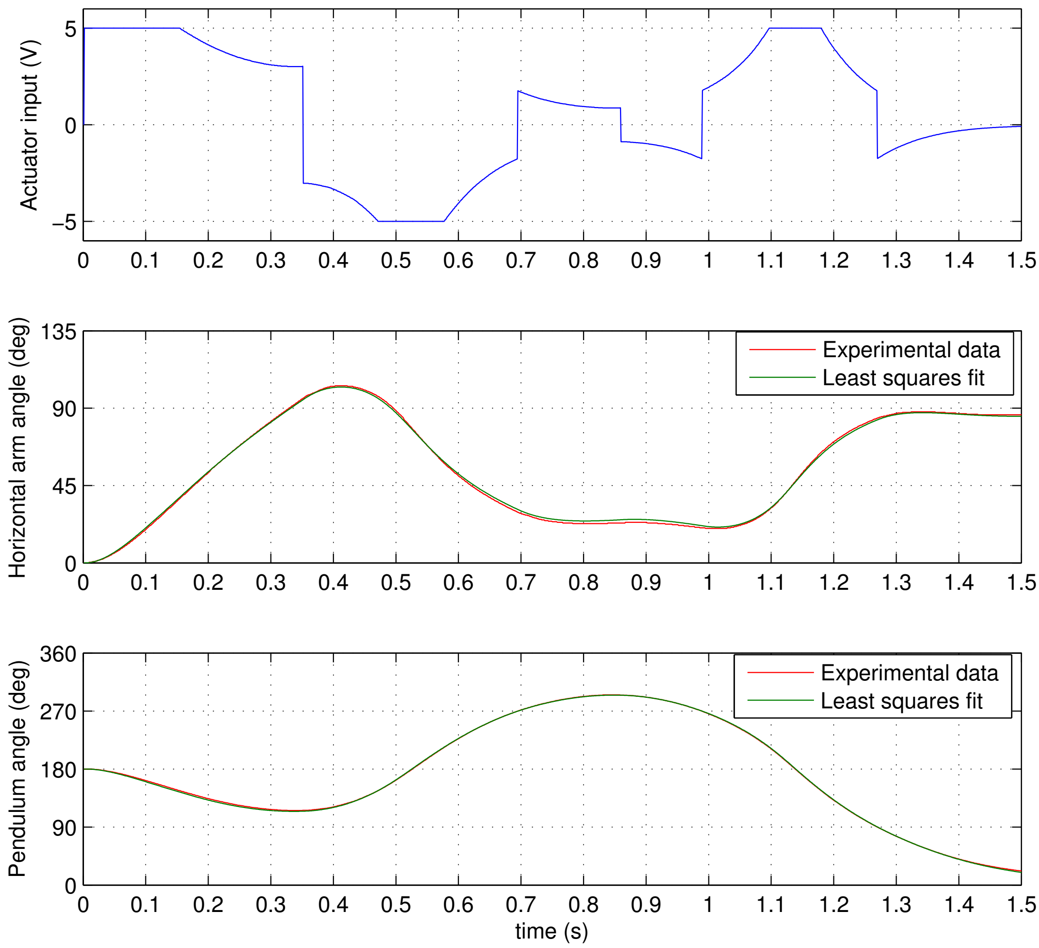


Figure 2.2: Model fitted to experimental data using least-squares method with weighting and regularization, parameters presented in table 2.2. $\chi^2_{ndf} = 8.7$. Error bars omitted for readability: all experimental points have 0.35 deg random errors. Swing-up of the FP using exponentiation 3.1.2, with $n = 2.15$, $k_v = 0.665$.

A useful tool to evaluate the quality of the fit is the χ^2_{ndf}

$$\chi^2_{ndf} = \frac{1}{n_y n_p - n_\theta} \sum_{j=1}^{n_p} \sum_{k=1}^{n_y} \left(\frac{y_{jk} - \hat{y_{jk}}}{\sigma_{jk}} \right)^2, \tag{2.24}$$

which is expected to have a value close to 1.

The value found for $\chi^2_{ndf} = 8.7$ show that 1) the fit exhibits, qualitatively, an agreement to experimental data, and 2) the random errors considered (equal to the resolution of the optical encoders used) are not sufficient to explain the differences observed, therefore there are effects not perfectly modelled, as has been discussed in 2.1.1.

Table 2.2: Identified parameters with least squares method

Parameter	Value	Unit
L_{e1}	227 ± 1	mm
J_0	86.98 ± 0.03	$\mathrm{g \cdot m^2}$
K_{a1}	1.0 ± 0.3	$\mathrm{mN \cdot m \cdot s}$
M_2	309 ± 1	g
L_{cm2}	404 ± 1	mm
J_2	28.37 ± 0.01	$\mathrm{g \cdot m^2}$
K_{a2}	0.136 ± 0.001	$\mathrm{mN \cdot m \cdot s}$
L_b	3.0 ± 0.1	mH
R	2.266 ± 0.002	Ω
K_t	0.696 ± 0.001	$\mathrm{V \cdot s}$
K_f	3.377 ± 0.002	$\mathrm{V \cdot s}$

2.3 Linearised Model

The non-linear system in the form of (2.18) can be linearised by taking the linear terms of the Taylor series around an operation point $(\bar{\mathbf{x}}, \bar{u})$

$$\mathbf{f}(\mathbf{x}, u) \approx \mathbf{f}(\bar{\mathbf{x}}, \bar{u}) + \nabla_{\mathbf{x}} \mathbf{f}\big|_{(\bar{\mathbf{x}}, \bar{u})} (\mathbf{x} - \bar{\mathbf{x}}) + \left.\frac{\partial \mathbf{f}}{\partial u}\right|_{(\bar{\mathbf{x}}, \bar{u})} (u - \bar{u}). \tag{2.25}$$

Let $A = \nabla_{\mathbf{x}} \mathbf{f}\big|_{(\bar{\mathbf{x}}, \bar{u})}$, $B = \frac{\partial \mathbf{f}}{\partial u}\big|_{(\bar{\mathbf{x}}, \bar{u})}$, $\Delta x = \mathbf{x} - \bar{\mathbf{x}}$, and $\Delta u = u - \bar{u}$. The previous equation can be rewritten as

$$\dot{\mathbf{x}} \approx A \Delta \mathbf{x} + B \Delta u + \mathbf{f}(\bar{\mathbf{x}}, \bar{u}). \tag{2.26}$$

Taking the set of parameters identified previously, the linearisation is performed for several $\bar{\mathrm{x}}$, resulting in a family of linear models, where each model is a convenient description of the system only at a vicinity of the point of linearisation. The model used in the equilibrium controller is the linearisation obtained at the state-space origin, *i.e.* the upwards equilibrium position, with zero input. Since at this point $\mathbf{f}(\bar{\mathbf{x}} = 0, u = 0) = 0$, the linear model can be written in the form

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \end{cases}, \tag{2.27}$$

where x, u, and y now denote deviations from the equilibrium, and

$$A = \begin{bmatrix} 0 & 1.0000 & 0 & 0 & 0 \\ 0 & -0.0174 & 20.7861 & -0.0023 & 57.5344 \\ 0 & 0 & 0 & 1.0000 & 0 \\ 0 & -0.0174 & 63.8319 & -0.0071 & 57.4388 \\ 0 & -232.0252 & 0 & 0 & -755.4250 \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 333.3367 \end{bmatrix}. \tag{2.28}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \qquad D = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

At other operating points, only the matrix A and $\mathbf{f}(\bar{\mathbf{x}}, \bar{u})$ change. In particular at $\bar{x} = (0, 0, \pi/2, 0, 0)$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & -0.0089 & -10.5782 & 0 & 29.2797 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -0.0048 & 0 \\ 0 & -232.0252 & 0 & 0 & -755.4250 \end{bmatrix} \quad \mathbf{f}(\bar{\mathbf{x}}, \bar{u}) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 43.0803 \\ 0 \end{bmatrix}, \tag{2.29}$$

and for $\bar{\mathbf{x}} = (0, 0, \pi, 0, 0)$, $\mathbf{f}(\bar{\mathbf{x}}, \bar{u}) = 0$, and

$$A = \begin{bmatrix} 0 & -0.0174 & 20.7861 & 0.0023 & 57.5344 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0.0174 & -63.8319 & -0.0071 & -57.4388 \\ 0 & -232.0252 & 0 & 0 & -755.425 \end{bmatrix}. \tag{2.30}$$

2.3.1 Discretisation

It is useful to describe the system in discrete time since the control is performed by a digital computer.

The continuous linear system of (2.27) may be made discrete for a constant time interval h with

$$\Phi = e^{Ah}, \tag{2.31a}$$

$$\Gamma = \int_0^h e^{A\tau} d\tau B, \tag{2.31b}$$

with which the discrete linear model can be written as

$$\begin{cases} x(t_{k+1}) = \Phi x(t_k) + \Gamma u(t_k) \\ y(t_k) = C x(t_k) + D u(t_k) \\ t_k = t_0 + hk \end{cases}. \tag{2.32}$$

Taking a fixed step size $\Delta t = 0.02\text{s}$ and rounding to the 4th decimal case yields

$$\Phi = \begin{bmatrix} 1 & 0.0172 & 0.0038 & 0.0000 & 0.0012 \\ 0 & 0.7130 & 0.3591 & 0.0038 & 0.0559 \\ 0 & -0.0028 & 1.0124 & 0.0200 & 0.0012 \\ 0 & -0.2874 & 1.2247 & 1.0123 & 0.0563 \\ 0 & -0.2244 & -0.1040 & -0.0010 & -0.0176 \end{bmatrix} \quad \Gamma = \begin{bmatrix} 0.0041 \\ 0.4119 \\ 0.0041 \\ 0.4125 \\ 0.3226 \end{bmatrix}. \tag{2.33}$$
$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \quad D = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

2.3.2 Controllability

A continuous linear system is said to be completely controllable if and only if (iff) from an initial state at the origin, an input function $u(t)$ with $0 < t \leq t_f$ and a finite horizon t_f, an arbitrary $x(t_f) = x_f$ can be achieved.

The linear system (2.27) is completely controllable if the controllability matrix

$$\mathcal{C}[A, B] = \begin{bmatrix} B & AB & A^2B & \cdots & A^{n-1}B \end{bmatrix}, \tag{2.34}$$

has full rank n. An analogue test is valid in the discrete case, replacing $\{A, B\}$ by $\{\Phi, \Gamma\}$.

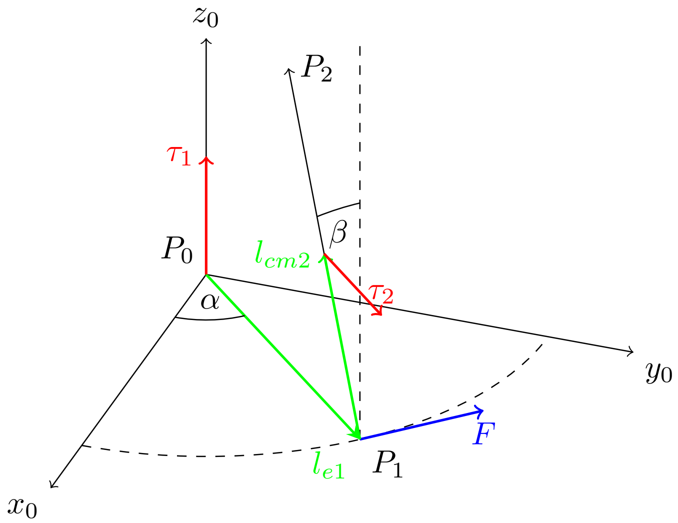


Figure 2.3: Torque transmission in the FP. Conventions are the same as the ones used in figure 2.1b

It is of interest to investigate the physics behind the transmission of torque in this system. When a voltage is applied to the motor, a current is created, and a torque $\boldsymbol{\tau}_1$ is produced at the centre of rotation of the horizontal arm. At the metal shaft where the pendulum is attached, the force created by $\boldsymbol{\tau}_1$ over the arm $\mathbf{l}_{e1}$ is horizontal and perpendicular to both $\mathbf{l}_{e1}$ and $\boldsymbol{\tau}_1$ (figure 2.3), accordingly to the general formula

$$\boldsymbol{\tau} = \mathbf{r} \times \mathbf{F}. \tag{2.35}$$

The torque at the pendulum centre of mass is also given by the above formula, with which the magnitude of $\boldsymbol{\tau}_2$ can be written, as a function of the angle β

$$\tau_2 = l_{cm2} F \cos\beta, \tag{2.36}$$

where r_2 is the distance between the metal shaft and the centre of mass of the pendulum. From the above equations it is clear that the input, the voltage applied to the motor, as a direct action on all the state variables of the system – current of the motor, position and speed of the horizontal arm, and position and speed of the pendulum. There is a clear exception for $\cos\beta = 0$, *i.e.* when the pendulum

Table 2.3: Linearised models eigenvalues, asymptotic stability of the reduced system $(X_2, \cdots, X_5)$, controllability and observability at 5 operating points, where all state variables are set to zero, except for X_3.

X_3	Eigenvalues					A. Stable	Controllable	Observable
	1	2	3	4	5			
0	-737.2	-19.4	$+7.0$	-5.8	0	No	Yes	Yes
$\pi/4$	-743.2	-11.8	-5.3	$+5.0$	0	–	Yes	Yes
$\pi/2$	-746.2	-9.1	-0.0	0	0	–	No	Yes
$3\pi/4$	-743.2	-12.4	$+0.1+5.0i$	$+0.1-5.0i$	0	–	Yes	Yes
π	-737.2	-17.0	$-0.5+6.8i$	$-0.5-6.8i$	0	Yes	Yes	Yes

is horizontal. In this situation the position and the force vectors are parallel, and thus no torque can be applied. Hence, the pendulum variables are not controllable at this point.

These conclusions are in agreement with the tests performed with (2.34) for five linear models (table 2.3).

2.3.3 Observability

A continuous system is said to be completely observable iff the knowledge of $y(t)$ for $0 \leq t \leq t_1$, with t_1 finite, is sufficient for determining the initial state condition $x(0)$.

The linear system (2.27) is observable if the matrix

$$\mathcal{O}(A, C) = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ C \end{bmatrix}, \tag{2.37}$$

has full rank n. An analogue test is valid in the discrete case, replacing A by Φ.

2.3.4 Stability

A fixed point $\bar{\mathrm{x}}$ of a dynamic system is classified as Lyapunov stable if

$$\forall\, \epsilon > 0,\ \exists\, \delta > 0 : ||\mathbf{x}(0) - \bar{\mathbf{x}}|| < \delta \wedge t > 0 \Rightarrow ||\mathbf{x}(t) - \bar{\mathbf{x}}|| < \epsilon, \tag{2.38}$$

and is asymptotically stable if additionally

$$\exists\, \epsilon > 0 : ||\mathbf{x}(0) - \bar{\mathbf{x}}|| < \epsilon \Rightarrow \lim_{t\to\infty} \mathbf{x}(t) = \bar{\mathbf{x}}. \tag{2.39}$$

The asymptotic stability of a nonlinear system can be evaluated from the linearised systems obtained at each fixed point with the Lyapunov indirect method. It is possible to conclude that, given the linearised system in the form of (2.27):

1. if all eigenvalues of A have negative real part, then the fixed point of the non linear system is stable;

2. if at least one eigenvalue of A has positive real part, then the fixed point of the non linear system is unstable;

3. if none of the above apply, *i.e.* if at least one eigenvalue is over the imaginary axis, and there are no eigenvalues with positive real part, then nothing can be concluded.

Eigenvalues were computed for several operation points along with controllability and observability, and asymptotic stability evaluated for the two fixed points of the system (table 2.3). The last eigenvalue is associated with the eigenvector $(1, 0, 0, 0, 0)$. As it is always zero, the dynamics on the state-variable X_1 are neither stable or unstable. This is due to the fact that the system has cylindrical symmetry. Stability analysis was done for the reduced system that excludes this variable.

Chapter 3

Trajectory Planning

This chapter addresses the problem of finding the inputs that make a dynamic system execute a desirable trajectory in state-space, i.e. a trajectory that complies with the constraints and, if possible, optimizes a specified quantity.

Consider as a simple conceptual example a car trip. In this case there are restriction on the initial state and on some of the variables during the trip (speed limit and other traffic rules, vehicle power and dynamics). Assume that the passenger wants to park as close as possible to his final destination but also to minimize fuel consumption. The problem is to find the steering and acceleration inputs that make a desirable trajectory.

Although several strategies can be used to design trajectories, the focus is given to optimal control. This technique provides a clear framework for finding solutions for problems subject to constraints and optimization goals.

The Furuta Pendulum is used as the case study, with the model developed in chapter 2, with which different techniques, numerical methods and minimization goals are explored.

3.1 Ad hoc strategies

This section presents two control strategies developed specifically for the swing-up of a rotary pendulum. They rely on a analysis of the sign of the input needed to add energy to the system, modulated by a function of the state of the system.

3.1.1 Energy Control

In [22] a control law is proposed for the swing-up of the FP, based on energy control, given by

$$u = sat[k_v \ (E - E_0)] \ sign\left(\dot{\beta} \ \cos\beta\right), \tag{3.1}$$

where E is the current energy of the system, E_0 is the energy of the goal state, k_v is the gain of the controller, and β is the angle of the pendulum. The first term defines the amplitude of the input (figure

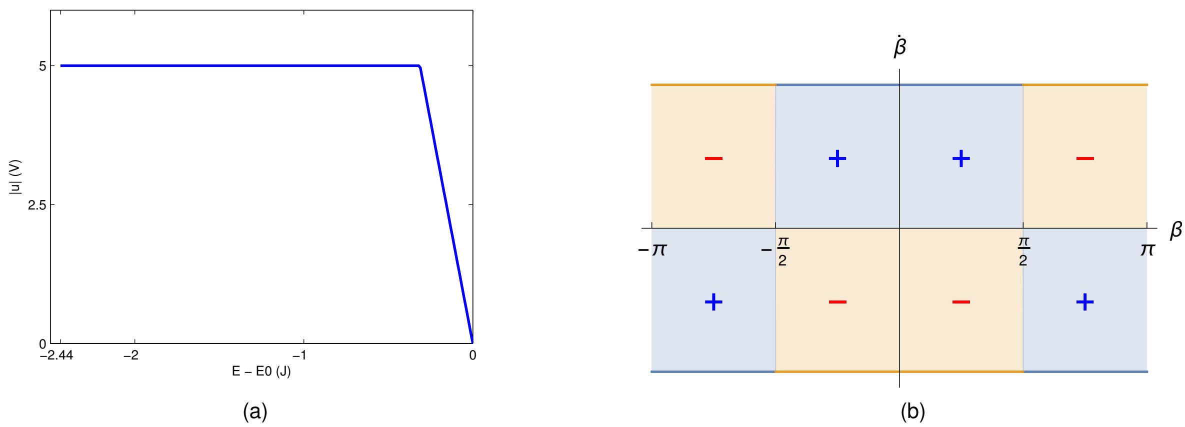


Figure 3.1: Characteristic of the energy control method for swing-up. (a) Amplitude of the input variable as a function of the system energy. (b) Signal of the input variable, as a function of the pendulum angle and angular speed

3.1a). It can be seen as a proportional controller, where the variable is the difference in energy from the current state to the energy of the goal state, by convention set to zero. The amplitude is limited due to physical constraints of the actuator.

The second term defines the sign of the control input, and assures that the input effect is to add energy to the system (figure 3.1b). The term $\cos\beta$ evaluates if the current position of the pendulum is above or under the horizontal position. For $\cos\beta = 0$ the pendulum is horizontal, thus the system is not controllable – no energy can be transmitted to the pendulum (see 2.34). With the additional term $\dot\beta$ the force is applied against the direction of motion of the pendulum when it is under horizontal position and in the same direction when above.

Note that this control law has been designed specifically for the inverted pendulum system, thus being not very general. Additionally it is limited to situations where the energy level of objective state is not degenerate. Although the controller will drive the system to a state with the specified energy, if there are several configurations with the same energy it is not possible to choose between them. This is the case in the Furuta Pendulum, where the energy of the system does not depend on angle of the horizontal arm α. Thus, with this technique there is no control over this variable during the swing-up.

This control law was applied to the Furuta Pendulum in simulation (figure 3.2). It was found that the trajectory is very sensitive to the gain, so the control is not robust to differences between the model and the real device.

3.1.2 Exponentiation of the pendulum position

The control law defined in (3.1) requires calculating the energy of the system. A modified version of the controller is proposed in [22] that takes into account only the state space variables

$$u = sat\left(k_v\ |\beta^n|\right)\ sign\left(\dot\beta\ \cos\beta\right). \tag{3.2}$$

In this case the first term is modified, now taking the angle between the current position and vertical.

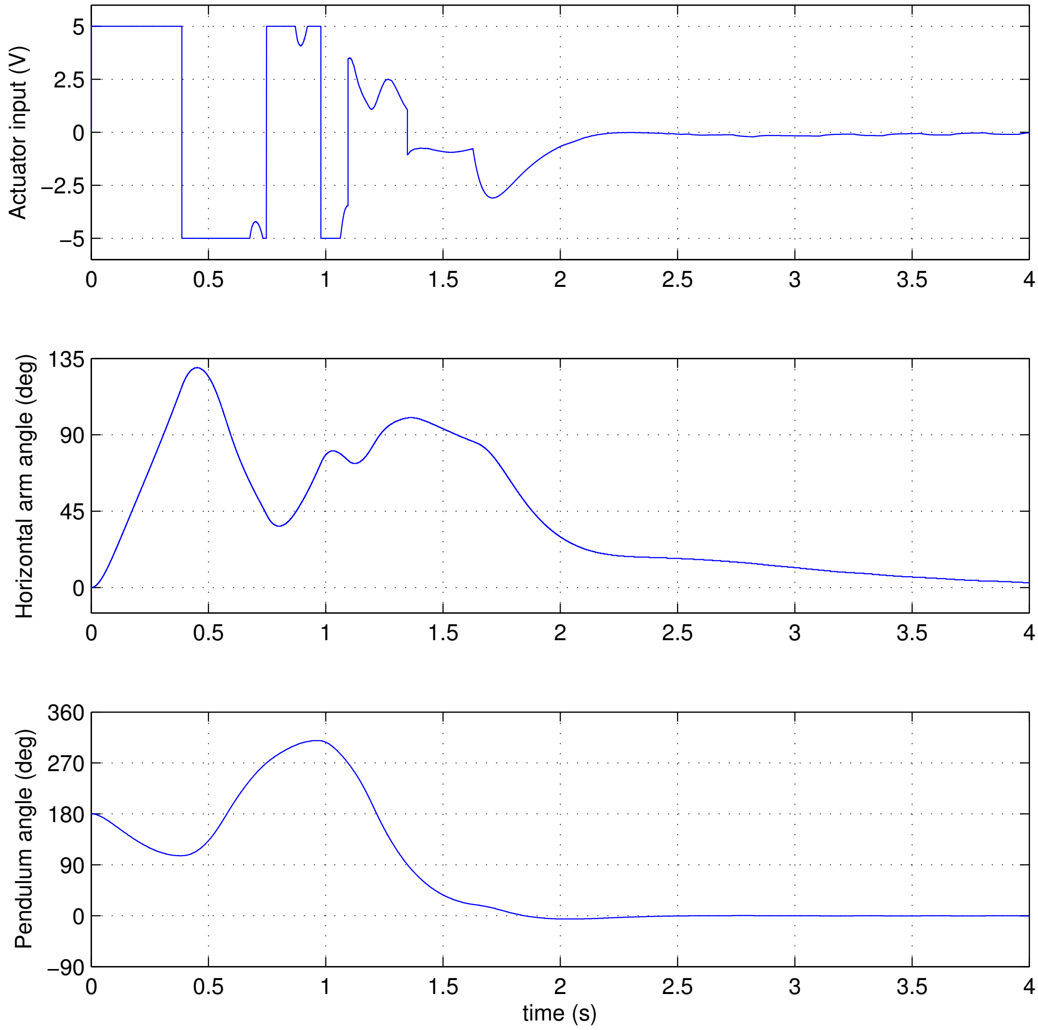


Figure 3.2: Swing-up performed in simulation with energy control, and equilibrium maintained with an LQG controller, after $t = 1.65\mathrm{s}$.

A exponent n is added to the expression, which makes the amplitude of the input greater when the pendulum is far from the upwards position, but smaller when it is closer (figure 3.3a). The authors state that with this law the transition between the nonlinear and stabilizing controller is smother. The second term is unaltered (figure 3.3b).

This control law was applied to the Furuta Pendulum both in simulation and with the real device (figure 2.2). It was found that this control has improved robustness over the energy control strategy, providing a reliable method for performing the swing-up. This allowed for the adjustment of the control parameters by trial and error in simulation, followed by the first successful swing-up manoeuvre achieved with the real device, that allowed for the identification of the parameters using least-squares (see 2.2).

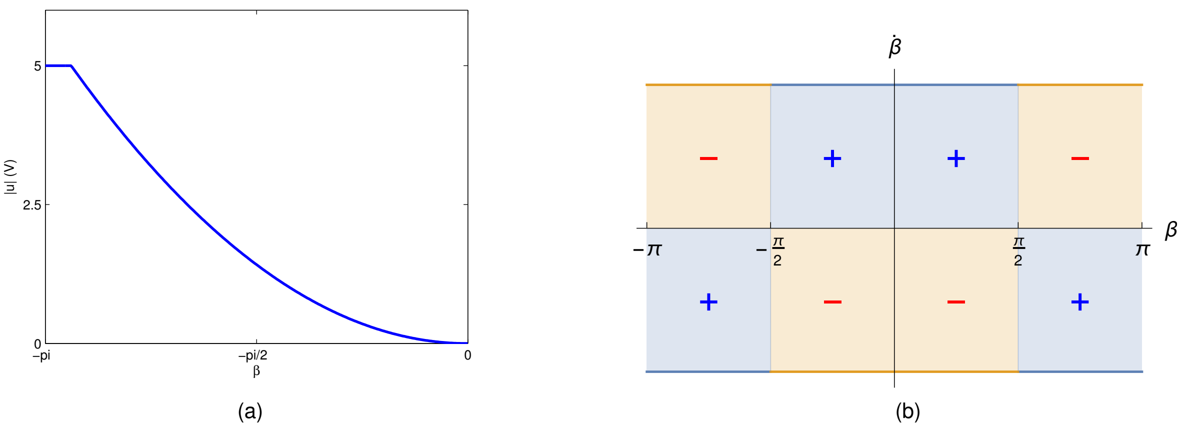


Figure 3.3: Characteristic of the exponentiation method for swing-up. (a) Amplitude of the input variable as a function of the pendulum angle. (b) Signal of the input variable, as a function of the pendulum angle and angular speed.

3.1.3 Energy shaping

In [16] the following control is presented to perform the swing-up

$$u = k_1 \left(x_2 + k_2 \cos \left(x_3 \right) x_4 \right), \tag{3.3}$$

which has been tested successfully both in simulation and with the real device (see 6.2).

3.2 Optimal control (continuous case)

Let $\mathbf{x}(t)$ be the trajectory in state space of the dynamic system with equations of motion

$$\dot{\mathbf{x}} = f[\mathbf{x}(t), u(t)], \tag{3.4}$$

with initial conditions

$$\mathbf{x}(t_0) = \mathbf{x}_i, \tag{3.5}$$

and the cost function $J\left[u(t)\right]$ be in the Bolza form

$$J[u(t)] = \Psi\left[\mathbf{x}(T)\right] + \int_{t_0}^{T} \mathcal{L}\left[\mathbf{x}(t), u(t)\right] dt, \tag{3.6}$$

where $\mathbf{x} \in \mathbb{R}^n$ is the state vector, u is the input variable, that takes values in the space of admissible controls $\mathcal{U}$, T is the terminal time, $\Psi\left[\mathbf{x}(T)\right]$ is the cost term on the terminal state, and $\mathcal{L}\left[\mathbf{x}(t), u(t)\right]$ a function that evaluates the cost during the transient, that is designated either as the Lagrangian or the running cost.

The goal of the optimization process is to minimize the cost function while complying with the constraints

$$\begin{cases} \min & J\left[u(t)\right], \\ s.t. & \dot{\mathbf{x}} = f[\mathbf{x}(t), u(t)], \\ & \mathbf{x}(t_0) = \mathbf{x}_i, \\ & u(t) \in \mathcal{U}, \\ & t \in [t_0, T], \end{cases} \tag{3.7}$$

where the constraints on the terminal state are not set.

Pontryagin's minimum principle gives a necessary condition for the solution of (3.7). Let $\mathcal{H}$ be the Hamiltonian, defined as

$$\mathcal{H}\left[\boldsymbol{\lambda}(t), \mathbf{x}(t), u(t)\right] = \boldsymbol{\lambda}'(t)\ f\left[\mathbf{x}(t), u(t)\right] + \mathcal{L}\left[\mathbf{x}(t), u(t)\right], \tag{3.8}$$

where a co-state $\boldsymbol{\lambda}(t) \in \mathbb{R}^n$ is introduced, and the Hamilton equations hold

$$\frac{d\boldsymbol{\lambda}}{dt} = -\frac{\partial \mathcal{H}}{\partial \mathbf{x}}, \tag{3.9a}$$

$$\frac{d\mathbf{x}}{dt} = +\frac{\partial \mathcal{H}}{\partial \boldsymbol{\lambda}}, \tag{3.9b}$$

where (3.9b) is simply (3.4), and (3.9a) can be expressed as

$$-\dot{\boldsymbol{\lambda}}' = \boldsymbol{\lambda}'\ f_{\mathbf{x}}\left[\mathbf{x}(t), u(t)\right] + \mathcal{L}_{\mathbf{x}}\left[\mathbf{x}(t), u(t)\right], \tag{3.10}$$

with the boundary condition

$$\boldsymbol{\lambda}'(T) = \Psi_{\mathbf{x}}\left[\mathbf{x}(T)\right], \tag{3.11}$$

where the subscript applied to a function represents the partial derivative with respect to the subscript ($\mathcal{F}_{\mathbf{x}} \equiv \nabla_{\mathbf{x}}\mathcal{F}$), and the prime superscript designates the transpose ($\mathcal{M}' \equiv \mathcal{M}^T$). This notation is adopted for simplicity of writing and to avoid confusion with T, the terminal time.

Pontryagin's minimum principle states that the optimal control input $u^*(t)$ minimizes the Hamiltonian $\mathcal{H}$ at every instant $t \in [t_0, T]$. A formal statement of the theorem can be found in [34, p. 94].

Since the input variable is bounded, the minimum can occur in two situations, that have to be evaluated separately

1. at a boundary of $\mathcal{U}$;
2. in the interior of $\mathcal{U}$.

Let the cost function be of the particular form

$$J\left[u(t)\right] = \frac{1}{2}\,\gamma_1\,\mathbf{x}'(T)\,Q_1\,\mathbf{x}(T) + \int_0^T \left(\frac{1}{2}\,\gamma_2\,\mathbf{x}'(t)\,Q_2\,\mathbf{x}(t) + \frac{1}{p}\,\gamma_3\,\left|u(t)\right|^p\right) dt, \tag{3.12}$$

where $\gamma_i \in \mathbb{R}$, and $Q_i \in \mathbb{R}^{n \times n}$ is by convention normalized. Although this form is not general, many problems can be written in this manner. It allows for the description of problems with a quadratic cost on the terminal and transient states, and an arbitrary exponent to the control function. In this case the following equations hold

$$\Psi\left[\mathbf{x}(T)\right] = \frac{1}{2}\,\gamma_1\,\mathbf{x}'(T)\,Q_1\,\mathbf{x}(T), \tag{3.13}$$

$$\Psi_{\mathbf{x}}\left[\mathbf{x}(T)\right] = \gamma_1\,\mathbf{x}'(T)\,Q_1, \tag{3.14}$$

$$\mathcal{L}\left[\mathbf{x}(t), u(t)\right] = \frac{1}{2}\,\gamma_2\,\mathbf{x}'(t)\,Q_2\,\mathbf{x}(t) + \frac{1}{p}\,\gamma_3\,\left|u(t)\right|^p, \tag{3.15}$$

$$\mathcal{L}_{\mathbf{x}}\left[\mathbf{x}(t), u(t)\right] = \gamma_2\,\mathbf{x}'(t)\,Q_2. \tag{3.16}$$

The problem is now a set of $2n$ first-order differential equations. There are n equations of motion in a state-space form (3.4) where the boundary conditions apply at $t = 0$, and n co-state equations (3.10) where the boundary conditions are set for $t = T$.

While for $p = 2$ and linear dynamics this problem has a well known solution, the LQR (see 4.1.1), in general a suitable numerical method must be used to solve this problem.

The weights γ_i define the relative importance given to each component of the cost function. If $\gamma_1 \gg \gamma_2, \gamma_3$, the first term is dominant, and therefore the optimal trajectory will mostly optimize the terminal state, eventually with a very costly control input and disregarding the transient states. This situation reduces the stability of the numerical methods. On the contrary, the terms inside the integral improve the stability of the numerical method, and result in smoother solutions. However, the distance to the desired terminal state increases.

3.2.1 FP Optimization with L_2 ($p = 2$)

When (3.12) has $p = 2$, the cost function is said to result from the L_2 norm. The term on u^2 has the physical characteristics of power, and its integral of energy. Taking the model of the FP (2.19) and using the software *Mathematica* to perform symbolic manipulation, the Hamiltonian can be evaluated from its expression (3.8). For the computation of the optimal control only the terms dependent on u, $\mathcal{H}(u)$, are of interest. These can be written as

$$\mathcal{H}(u) = \frac{\lambda_5}{L_b}u + \frac{1}{2}\gamma_3 u^2, \tag{3.17}$$

which is a function in $\mathcal{U} \to \mathbb{R}$. Pontryagin's minimum principles requires $\mathcal{H}(u)$ to be minimum for every time instant for $u*$ to be optimal. If the minimum is in the interior of $\mathcal{U}$, then it can be found with the zero of the first partial derivative with respect to u (it is a minimum due to convexity of the function, as $\gamma_3 > 0$)

$$\frac{\partial \mathcal{H}(u)}{\partial u} = \frac{\lambda_5}{L_b} + \gamma_3 u = 0, \tag{3.18}$$

where L_b is a parameter of the model, the imaginary part of the motor impedance. This can be rewritten in order to u

$$u^*(t) = -\frac{\lambda_5(t)}{\gamma_3 L_b}, \tag{3.19}$$

as the input variable is bounded, the minimum of $\mathcal{H}(u)$ can also occur at a boundary of $\mathcal{U}$. In general the value at each boundary must be evaluated to find the minimum. The minimum is found for each time instant $t \in [t_0, T]$, which could be done symbolically for simple problems, but is unfeasible in this case. Hence, the time domain must be made discrete, and the optimization performed for each step (figure 3.4).

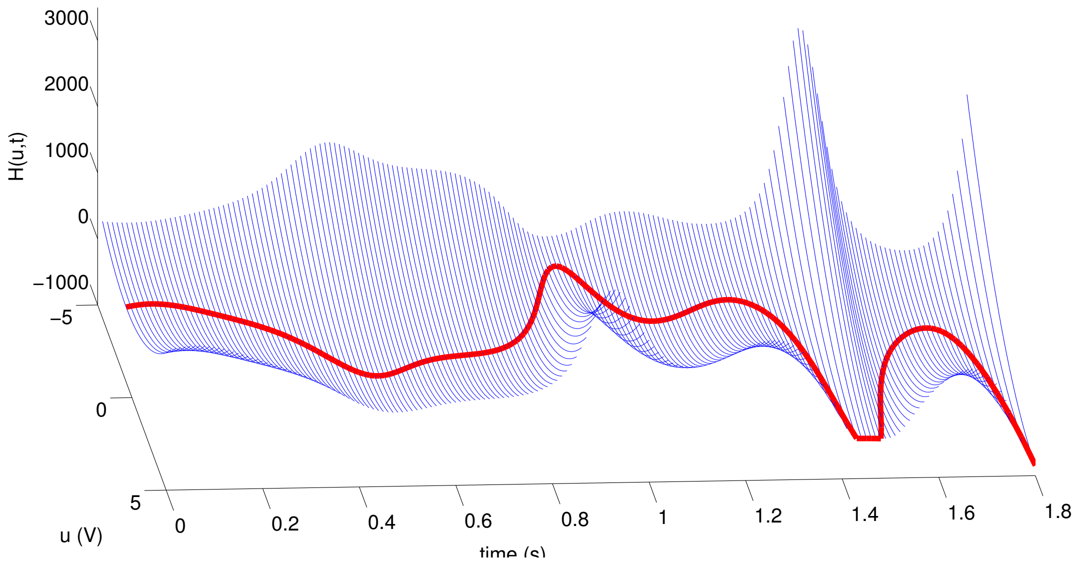


Figure 3.4: Optimization performed for the FP with L_2 norm, at iteration 1000 of the numerical method. The red line shows the optimal value $u^*(t)$, that corresponds to the minimum of $\mathcal{H}(u,t)$ for every time instant.

The optimal control and trajectories, calculated with adequate weights, are approximated by a numerical method (figure 3.5).

It is expected that along a optimal trajectory the Hamiltonian is constant in time. However, due to the characteristics of the problem and the numerical approach for solving the problem, the Hamiltonian is not constant (figure 3.9b). As expected, when the iteration number increases, the deviation decreases. The shape of this function has been used as a qualitative measure of the optimality of the trajectory, along with the evolution of the cost function (figure 3.10).

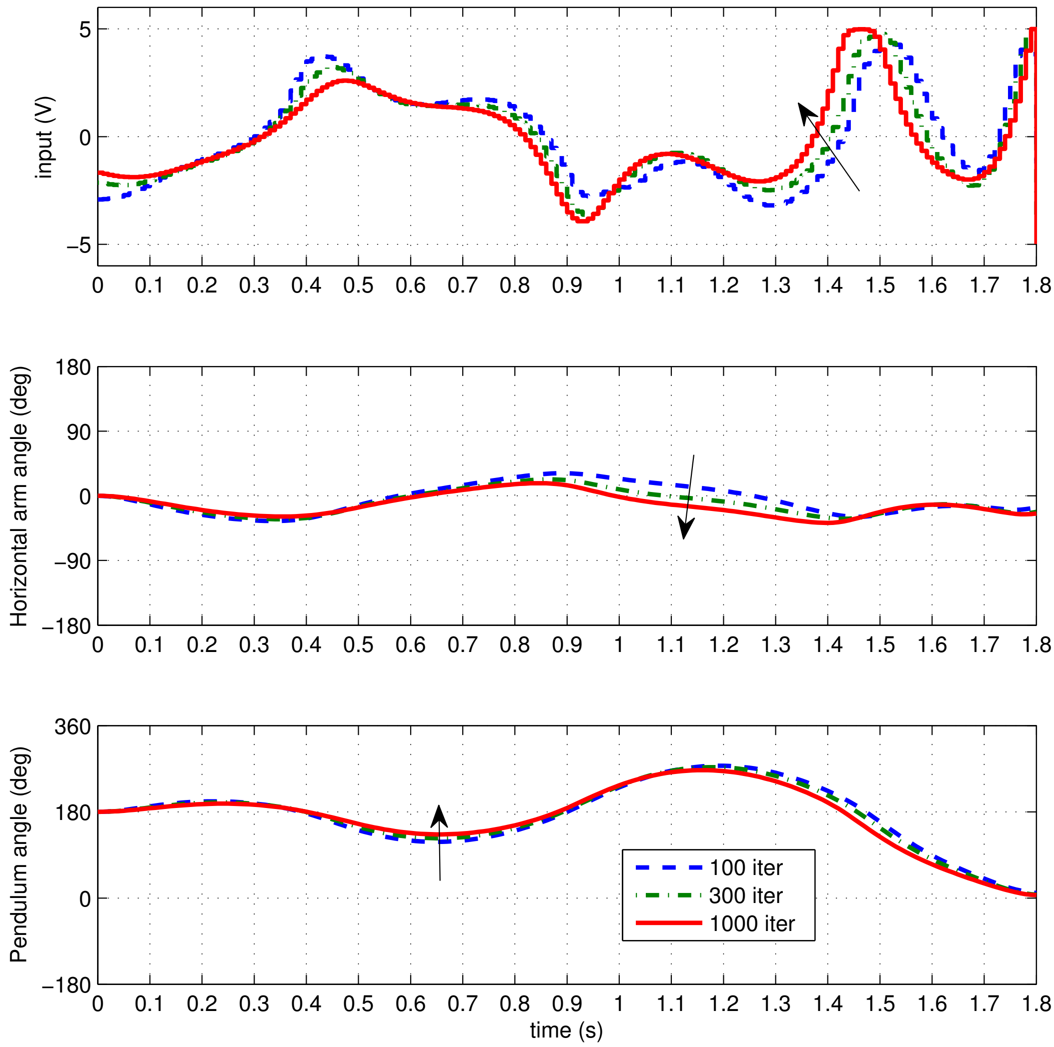


Figure 3.5: Control function and trajectories generated with a L2 norm at different steps of the numerical method.

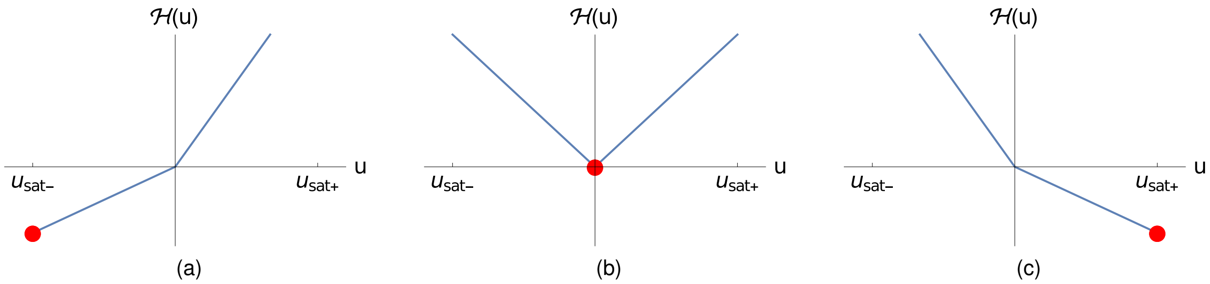


Figure 3.6: Hamiltonian terms dependent on u with L1 norm for $\gamma_3 L_b < \lambda_5$ (a), $-\gamma_3 L_b < \lambda_5 < \gamma_3 L_b$ (b), $\lambda_5 < -\gamma_3 L_b$ (c). The bold point marks the minimum of the function.

3.2.2 FP Optimization with L_1 ($p = 1$)

When (3.12) has $p = 1$ the cost function is said to be of L_1 norm. In this case the Hamiltonian terms dependent on u, $\mathcal{H}(u)$ may be written as

$$\mathcal{H}(u) = \frac{\lambda_5}{L_b} u + \gamma_3 |u| \,. \tag{3.20}$$

The analysis of this case is clearer in a graphical way (figure 3.6). The shape of the cost function at every instant in time is the modulus function, to which a linear function is added. It can have a minimum at the lower and upper bonds, and at zero. In the event where one of the sides of the function becomes horizontal in a finite length of time, the cost function is said to have a singular arc. Note that Pontyagin minimum principle gives only a necessary condition, thus in this situation some additional criteria should be used to find the optimal control input. The expression for u^* is

$$u^* = \begin{cases} u_{sat-}\,, & \gamma_3 L_b < \lambda_5 \\ 0\,, & -\gamma_3 L_b \le \lambda_5 \le \gamma_3 L_b \\ u_{sat+}\,, & \lambda_5 < -\gamma_3 L_b \end{cases} . \tag{3.21}$$

The evolution of $\mathcal{H}$ in optimization procedures is shown in figures 3.7 and 3.8. For $t \in [0.2, 0.4]$ the Hamiltonian is horizontal, and a singular arc occurs. Thus the input variable does not converge to any of the discrete levels defined by (3.21) in this time interval. This effect has been neglected, since the time interval in which it occurs is relatively small.

3.2.3 SP Optimization with L_2 ($p = 2$)

Consider a simple pendulum, actuated at its axis by a torque $u(t)$. The equation of motion is

$$\ddot{\alpha}(t) = \omega_0^2 \sin\alpha(t) + u(t), \tag{3.22}$$

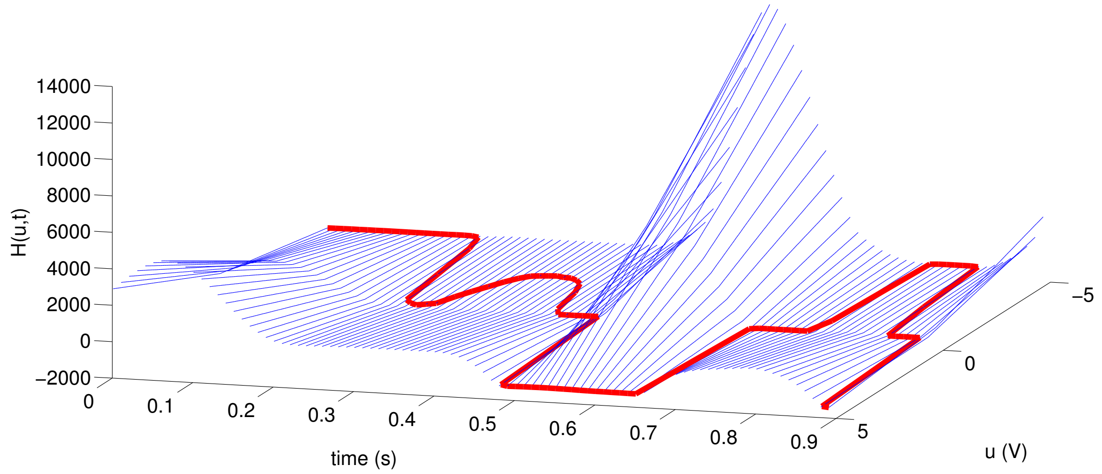


Figure 3.7: Optimization performed for the FP with L1 norm, at iteration 2000 of the numerical method. The red line shows the optimal value $u^*(t)$, that corresponds to the minimum of $\mathcal{H}(u,t)$ for every time instant.

where $\alpha = 0$ at the upwards position. Rewriting in a state space form, with $x_1(t) = \alpha(t)$ and $x_2(t) = \dot{x_1}(t)$

$$\dot{\mathbf{x}}(t) = f\left[\mathbf{x}(t), u(t)\right] = \begin{bmatrix} x_2(t) \\ \omega_0^2 \sin x_1(t) + u(t) \end{bmatrix}. \tag{3.23}$$

The derivative in respect to the state space variables $f_{\mathbf{x}}$

$$f_{\mathbf{x}}\left[\mathbf{x}(t), u(t)\right] = \begin{bmatrix} 0 & 1 \\ \omega_0^2 \cos x_1(t) & 0 \end{bmatrix}. \tag{3.24}$$

For $p = 2$, the Hamiltonianean terms on u, $\mathcal{H}(u)$ may be written as

$$\mathcal{H}(u) = \lambda_2 u + \gamma_3\, u^2, \tag{3.25}$$

and the derivative in respect to u

$$\frac{\partial \mathcal{H}\left[\lambda(t), \mathbf{x}(t), u(t)\right]}{\partial u} = \lambda_2(t) + \gamma_3\, u(t), \tag{3.26}$$

solving for $u(t)$

$$\frac{\partial \mathcal{H}}{\partial u} = 0 \Leftrightarrow u(t) = -\frac{\lambda_2(t)}{\gamma_3}. \tag{3.27}$$

3.2.4 SP Optimization with L_1 ($p = 1$)

In this case the Hamiltonian terms on u, $\mathcal{H}(u)$ may be written as

$$\mathcal{H}(u) = \lambda_2 u + \gamma_3 \left|u\right|. \tag{3.28}$$

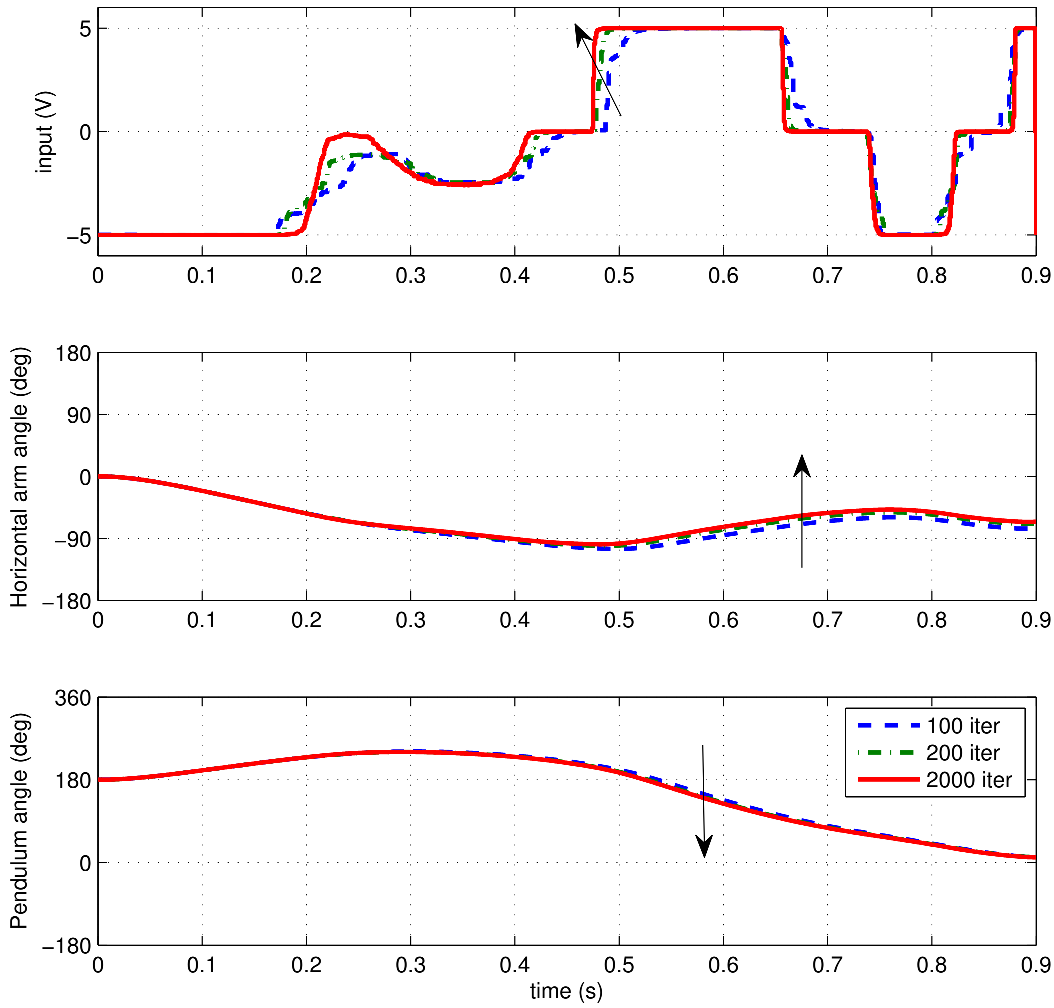


Figure 3.8: Control function and trajectories generated with a L1 norm. Iteration number increases in the direction of the arrow.

The analysis of this case is similar to the one performed in 3.2.2. The expression for u^* is

$$u^* = \begin{cases} u_{sat-} \,, & \gamma_3 < \lambda_2 \\ 0 \,, & -\gamma_3 \leq \lambda_2 \leq \gamma_3 \\ u_{sat+} \,, & \lambda_2 < -\gamma_3 \end{cases} . \tag{3.29}$$

3.3 Numerical Methods in optimal control

The application of Pontryagin minimum principle to a dynamic system described by a system of differential equations results in (3.9) with boundary conditions (3.5), and (3.11). Although there is, in general, no closed form solution, a suitable numerical method can be used to find $\boldsymbol{\lambda}(t), \mathbf{x}(t)$ that satisfy this equations. In this section a indirect method is proposed.

3.3.1 Gradient descent

One of the most straightforward methods to find local minima of a function of multiple variables is the *gradient descent*, also known as *steepest descent*. This is a first-order optimization algorithm where steps are taken in the opposite direction of the gradient and proportionally to its magnitude. On optimal control this method is used in the optimization of the Hamiltonian in respect to the input function. This method is notably useful in cases where the minimum of $\mathcal{H}(u)$ is unknown.

The algorithm may be implemented as follows [35]:

1. An initial guess for the control function $u(t)$ is taken, where the time interval $[t_0, t_f]$ is subdivided into a set of N subintervals. For simplicity, each subinterval can be made of equal length. The input variable $u(t)$ is set to be piecewise-constant for each subinterval;

2. the equations of motion of the system are integrated forward in time;

3. the final value of the state-variables, $x(T)$, is used as initial condition for the backward integration of the co-state;

4. the stopping criteria are evaluated;

5. updated values are set for the control function at $t = t_k$, where $k = 0, \cdots, N$ and $u(t)$ is a piecewise-constant function:

$$u^{(i+1)}(t_k) = u^{(i)}(t_k) - \tau \, \nabla_u \mathcal{H}(t_k); \tag{3.30}$$

6. repeat from 2, until a stopping criteria is met.

3.3.2 Algebraic descent

A method was developed as a modification of the gradient descent method. The modification takes advantage of the existence of a simple algebraic solution for the minimum of $\mathcal{H}(u)$.

A user oriented description of this method is given in appendix A.1, along with the application to a wave energy converter.

The method comprises the same sequential steps as the gradient descent, except for the actualization of the control function, where:

5. for each time instant $t_k, k = 0, \cdots, N$ the Hamiltonian is optimized, i.e. u is set to the value u^* that minimizes $\mathcal{H}(u)$ in a given moment t_k

$$u^{(i+1)}(t_k) = u^*(t_k). \tag{3.31}$$

The update depends on the cost function in use.

Smoothing

The method described above yields rapid convergence for simple problems. However it becomes unstable in problems where there is a high weight on the terminal state, or when the terminal state is very dependent on the input variable, as is the case of unstable systems.

The simplest problem that can illustrate the unstable behaviour is the harmonic oscillator, where the input variable can have both positive and negative values, and the objective is to get the final value as close as possible to zero. The reason for the instability is that altering the input variable at a certain time changes the shape of the state-space trajectory in all following moments, but the optimization is performed for all time intervals before proceeding with another forward-backward integration.

In order to improve stability a smoothing factor was imposed: instead of the full step for u that would minimize $\mathcal{H}(u)$ in that specific time interval, a smaller step is taken in the same direction. In practice this term slows the method, making it more stable.

The smoothing factor is evaluated dynamically. It is a monotonic decreasing function that is updated whenever an iteration yields a higher cost that the record minimum, subject to a small tolerance.

$$u^{(i+1)}(t_k) = \eta u^*(t_k) + (1 - \eta)u^{(i)}. \tag{3.32}$$

Momentum

The addition of the smoothing term makes the algorithm more stable, but also slower. It may become so slow that a stopping criteria is triggered before a convenient solution is found. To improve the converge speed while maintaining stability, a momentum term is added to the method. For each time interval, if two successive iterations are in the same direction, the step is increased as follows

$$u^{(i+1)}(t_k) = \eta u^*(t_k) + (1 - \eta)u^{(i)} + \Delta u^{(i)}. \tag{3.33}$$

Memory

In the event of an increase of the cost function the smoothing factor is updated and the values of the state and co-state variables that provide the cost function record minimum are restored.

Stopping criteria

Three stopping criteria are defined:

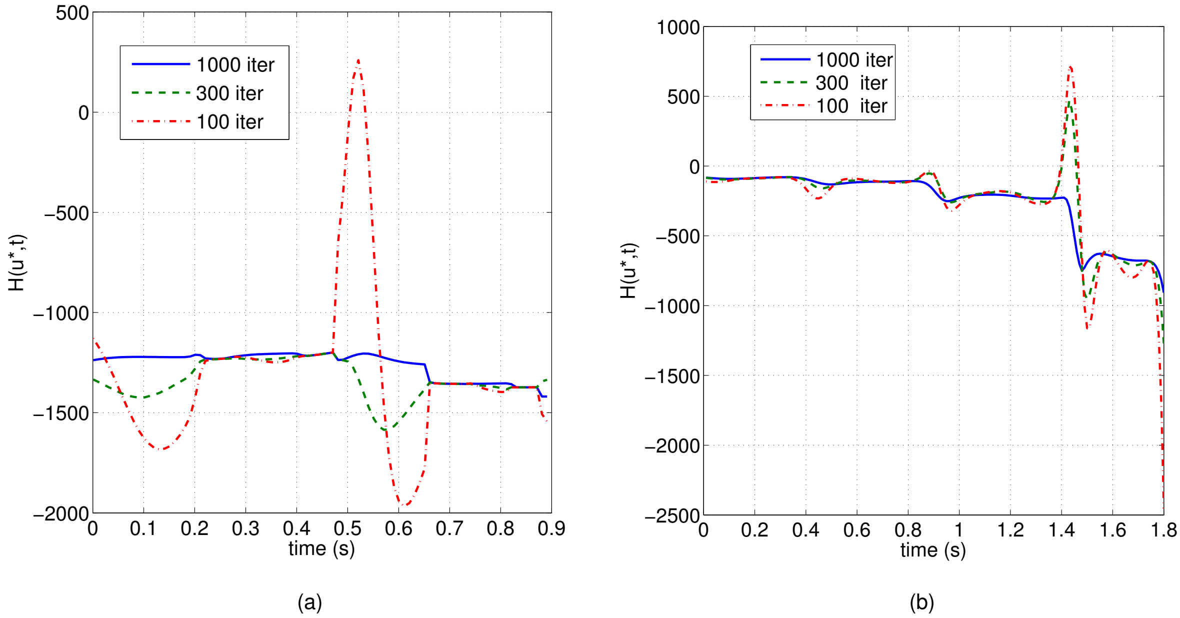


Figure 3.9: Hamiltonian as a function of time at three steps of the optimization process, for the (a) L_1 norm and (b) L_2 norm.

- If the smoothing factor becomes smaller than a certain value;
- If the state-variables have become very large (in which case the method has irremediable diverged);
- If the maximum number of iterations has been reached.

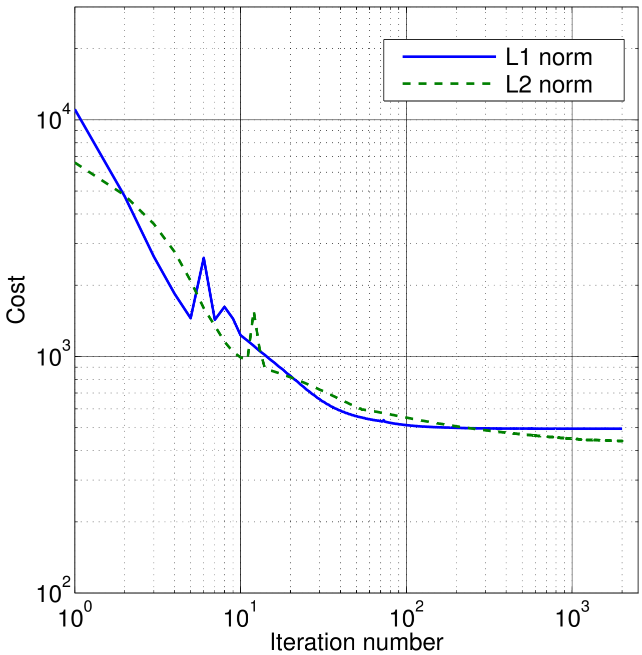


Figure 3.10: Cost J as a function of the iteration number for the optimization process of the FP with L_1 and L_2 norms.

Chapter 4

Trajectory Control

In this chapter optimal control theory is used to design linear controllers for a non linear system. A brief explanation of the theoretical tools is presented, along with an analysis of the effects of the controller parameters. Strategies for the transition between controllers are proposed and tested both in simulation and with the real device.

Linear controllers are used in this work in two circumstances

1. to follow reference trajectories, as the ones created in chapter 3. The controller makes small adjustments to the input function to compensate the differences in behaviour between the model and the real system;

2. to regulate to a constant reference state, in order to keep the system at the desired point of operation, for example balanced around the vertical position. This can been seen as a particular case of the previous item, where the input reference is set to zero.

Since each controller is valid on a limited region of state space, there is the need to design an ensemble of controllers.

4.1 Continuous Linear Quadratic Gaussian Controller

The problem of designing a linear controller with optimal control where

1. the system is described by a linear state space model, either continuous or discrete, with sensors available to measure the output variables,

2. both model and sensor information are subject to additive Gaussian noise,

3. the cost function is defined with a quadratic function on the state space variables and input,

is said to be Linear Quadratic Gaussian (LQG). This is one of the most fundamental problems in optimal control, for which robust tools are available. In the conditions stated, the superposition theorem holds, therefore the design of the LQG controller can be made separately for the two parts which compose it – regulator and estimator[1].

4.1.1 Linear Quadratic Regulator

The goal of the Linear Quadratic Regulator (LQR) algorithm is to find the gain vector K of the state space feedback law

$$u = -Kx, \tag{4.1}$$

that applied to the continuous linear system, defined in 2.3,

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \end{cases}, \tag{2.27}$$

minimizes the cost function

$$J = \int_0^\infty \left(x^T Q_r x + u^T R_r u\right), \tag{4.2}$$

which is a particular case of (3.12), in this case with infinite horizon and quadratic cost. Matrices Q_r and R_r weight the relative importance given to the regulation of the state variables and the cost of the input, and shall be selected taking into account the design goals of the system.

Applying Pontryagin's minimum principle (see 3.2) to the LQR problem, and assuming that $\exists\ P \in \mathbb{R}^{n\times n} : \lambda = Px$, yields the Algebraic Riccati equation

$$A^T P + PA - PBR_r^{-1}B^T P + Q_r = 0. \tag{4.3}$$

from which P can be calculated. The feedback gain K of the controller (figure 4.1) can now be evaluated with

$$K = R_r^{-1}BP. \tag{4.4}$$

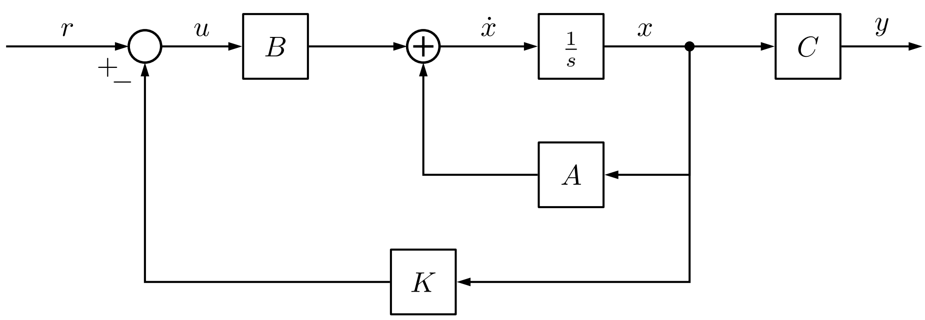


Figure 4.1: Block diagram of a state feedback controller, with D set to zero.

The LQR controller has severable desirable properties, for example, it maintains the order of the

[1] This section is not intended to be a comprehensive reference source, to complement the information on the design of a LQG controller refer to [9], where it is addressed with further detail.

system, as a phase margin of at least 60^o and an infinite gain margin.

4.1.2 Linear Quadratic Estimator

A Linear Quadratic Estimator (LQE), also known as Kalman Filter, produces estimates for the state of the system (figure 4.2), given the model and sensor measurements, both subject to uncertainties. It is a form of data fusion, which yields more precise results than a single measurement and requires small computing power. Additionally, it is the optimal estimator for Gaussian noise.

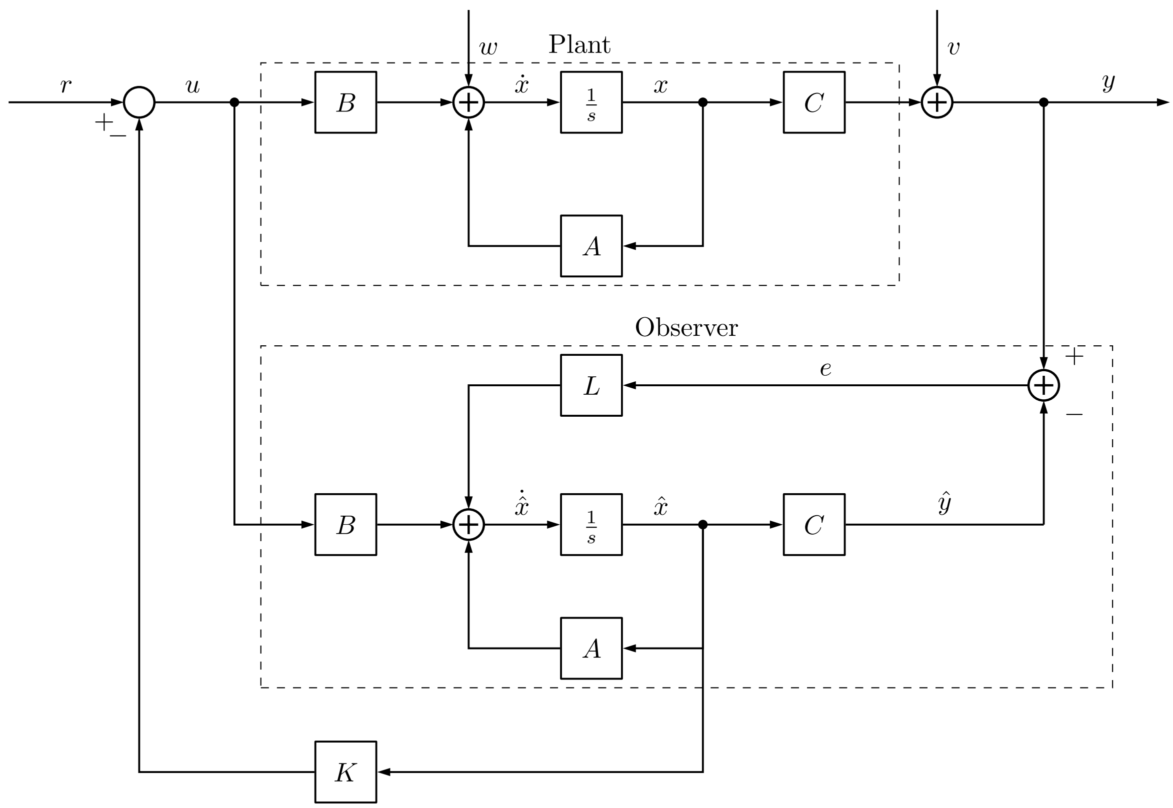


Figure 4.2: Block diagram of a state feedback controller with estimator, with D set to zero.

Let the modelling and the measurements be subject to Gaussian noise

$$\begin{cases} \dot{x} = Ax + Bu + w \\ y = Cx + Du + v \end{cases}, \tag{4.5}$$

with covariances

$$Q_e = E\left[ww^T\right], \tag{4.6}$$

$$R_e = E\left[vv^T\right]. \tag{4.7}$$

The optimal estimator for this problem is the Kalman filter with gains

$$L = \Sigma C^T R_e^{-1}, \tag{4.8}$$

where Σ is the solution of the Riccati equation

$$A\Sigma + \Sigma A^T + Q_e - \Sigma C^T R_e^{-1} C\Sigma = 0, \tag{4.9}$$

and the state estimation $\hat{x}$ is dynamically updated by integrating the differential equation

$$\dot{\hat{x}} = (A - LC)\hat{x} + Bu + Ly. \tag{4.10}$$

4.1.3 Linear Quadratic Gaussian (LQG)

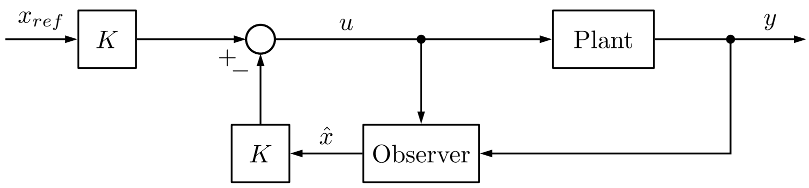


Figure 4.3: Block diagram of a state feedback controller with estimator.

The superposition theorem allows the independent design of the controller and estimator. The LQG problem is the simple combination of both LQR and LQE (figure 4.3). Therefore, substituting (4.1) in (4.10)

$$\begin{cases} \dot{\hat{x}} = (A - BK - LC)\hat{x} - Le \\ u = -K\hat{x} \end{cases}, \tag{4.11}$$

where $e = r - y$.

4.2 Controller design

From an engineering point of view, the LQG provides a complete methodology to design the controller. Several parameters can be used to modify the behavior of the system and can be seen as “manual adjusting knobs” (table 4.1). The designer shall select the set of parameters that best satisfies the design requirements, which make the process iterative.

On the design of the LQG controller, the requirements were:

1. to equilibrate the Furuta pendulum from initial rest conditions for any angle $\beta \in [-12, 12]$ deg;
2. to keep $\alpha \in [-180, 180]$, preventing the electric wire connections to wrap around the gears;
3. to have input signals in the nominal range $u \in [-6, 6]$ V;
4. to maintain equilibrium indefinetly, even when subject to external perturbations.

The controller was designed in *Matlab/Simulink*. Two modes of operation may be selected: 1) the system is simulated (figure 4.4) using the nonlinear model of 2.19 and 2) input is given to the real device

Table 4.1: LQG controller parameters.

Parameter	Description
Q_r	Cost function weight matrix on the state variables, gives relative importance to the regularization of each variable
R_r	Cost function weight matrix on the input function, accounts for the speed of the system and thus the amplitude of the input function
Q_e	Covariance matrix of process noise, accounts for the relative cofidence in the model predictions
R_e	Covariance matrix of measurement noise, accounts for the relative confidence on the sensors measurements

and output obtained from the sensors available. Both blocks, which will be refered to as simulation and experiment, respectively, are interchangeable. Simulation aims to behave as close as possible to experiment, using, for example a zero-order hold in the input variable and sensor discretization.

4.3 Evaluation of the control quality

4.3.1 Stability region

The equilibrium maintained by the linear controller at the upwards position can be perturbed by external forces. The region in which the equilibrium is kept is shown in figure 4.5 for several values of the maximum input amplitude. In this figure the region of convergence is the state space limited by the drawn boundary. Note that state space is of dimension 5, and the plot has dimension 2. Figure 4.7 shows the region of convergence for 3 variables. The initial values of the remaining variables are set to zero, but allow to evolve in time during simulation.

Figures 4.5 and 4.6 show that the area in which the linear controller is able to stabilize the non linear system increase as a function of the maximum input amplitude defined.

4.3.2 Numerical algorithm for determining the stability region

The linear controller valid around the upwards equilibrium points actuates on a non linear system. The stability region is calculated with an original method based on the binary search algorithm, with the assumption that the boundary of the stable region is globally continuous. The algorithm is as follows:

1. A point known to lie inside the stability region is taken as initial guess;

2. A binary search with n iterations is performed within a line segment of length $(2-2^{-(n-1)})\mathrm{step}$. At each iteration the point is tested to be stable or unstable. With this procedure the boundary of the stability region is found with an error of $\leq \mathrm{step}/2^n$;

3. Once the boundary of the stability region is found, a angular binary search with n iterations is performed: points that lie on a circle defined with a convenient norm are tested to be stable or

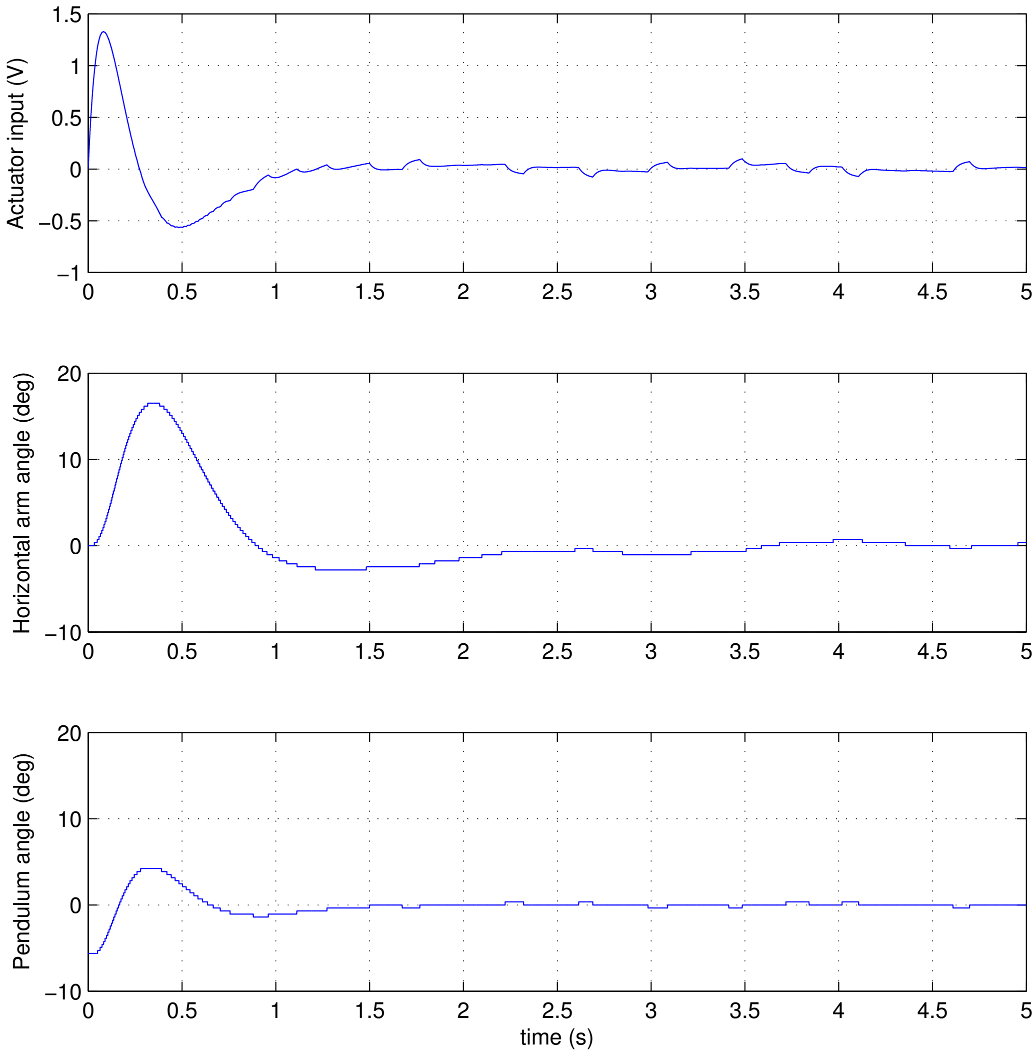


Figure 4.4: Time response of the LQG controller with initial conditions $x = [0\ 0\ \ -0.1\ 0\ 0]$. Results obtained in simulation.

unstable, and the angle updated to provide the best estimate with the available information. The method provides an angle estimate with error $\leq \mathrm{step}/2^n$. The point found is set as the centre of the next angular binary search procedure;

4. The method advances by recursively applying 3 until the boundary is defined in a $360°$ angle around the initial point.

Testing the stability of a point involves simulating the time evolution of the closed loop system, for an adequate amount of time, and to check if at the end it lies within a given tolerance of the set point. For complex systems this may require a considerable amount of time: the plot of figure 4.5 requires about 5 hours of computation.

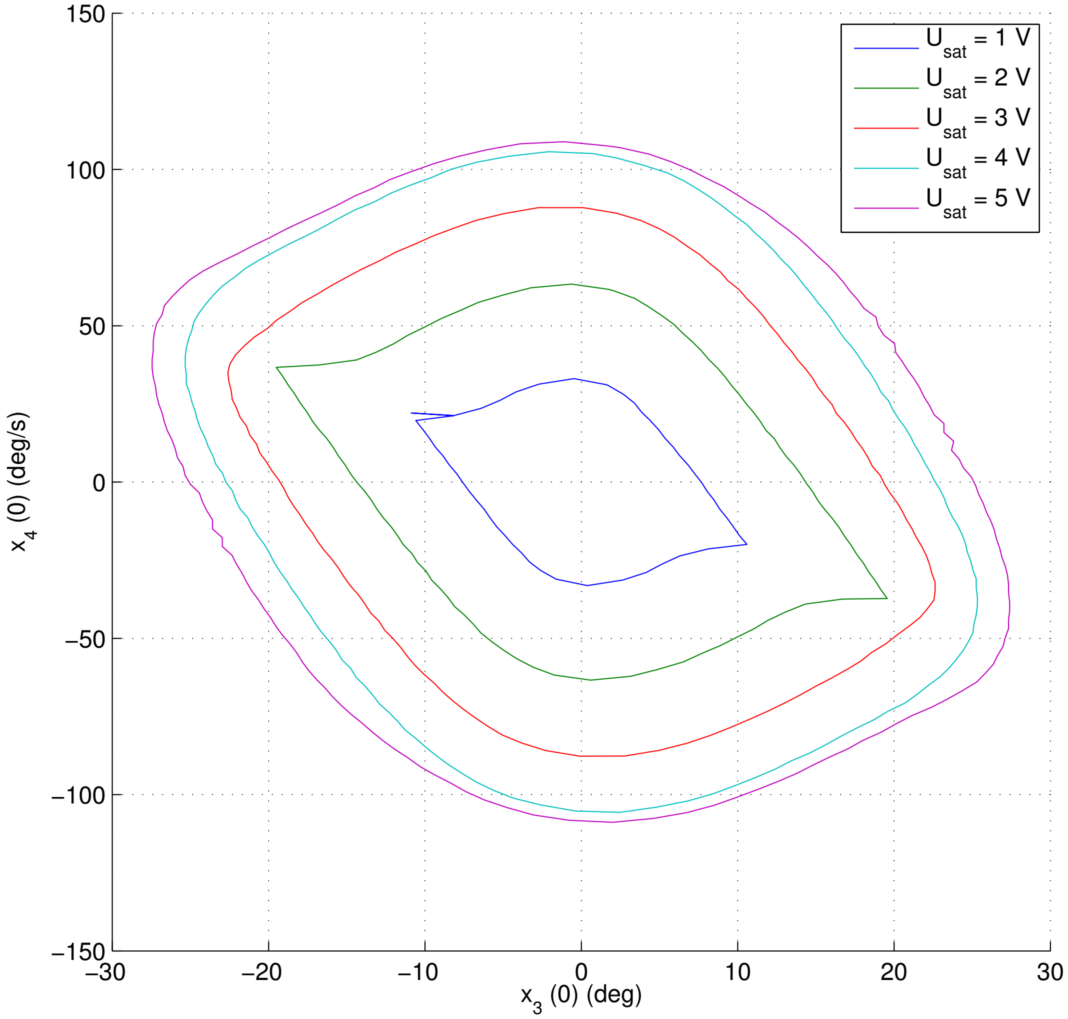


Figure 4.5: FP initial conditions in which equilibrium is achieved, as a function of the allowed amplitude for u, where the lines represent the boundary of the region of attraction. $x_1(0)$, $x_2(0)$ and $x_5(0)$ are set to zero. Simulations are performed with the non linear model using *Matlab/Simulink* and the LQG controller.

4.4 Transition between controllers

The transition between controllers is performed in two different modes of operation:

1. The control for swing-up is applied directly to the device. Only the linear controller around the final point is activated. In the transition between the swing-up and equilibrium modes the control function is made continuous. This approach is used when there is no reference trajectory, and the swing-up is performed in closed loop, for example with the Ad hoc strategies described in 3.1.

2. The reference control is applied to the device, which in turn has a closed loop ensemble of linear controllers which stabilize the difference between the performed and reference trajectories. This is known as a gain scheduling controller.

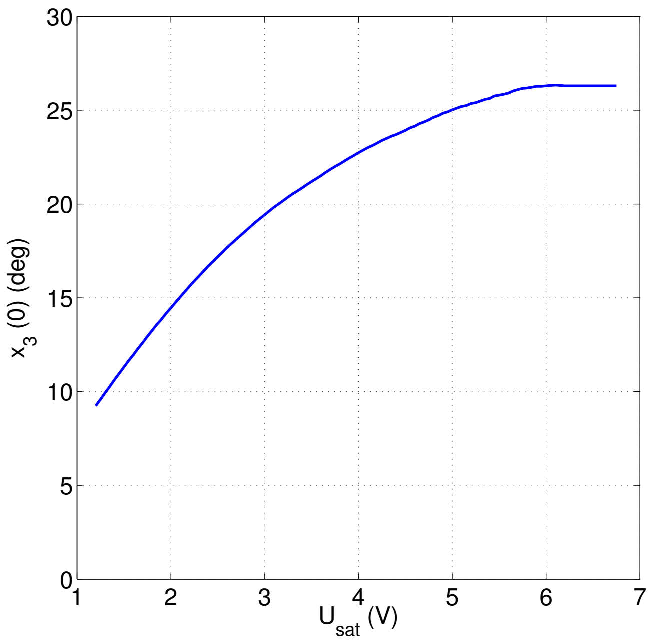


Figure 4.6: FP maximum initial pendulum angle in which equilibrium is achieved, as a function of the allowed amplitude for u. $x_1(0)$, $x_2(0)$, $x_4(0)$, and $x_5(0)$ are set to zero. Simulations are performed with the non linear model using *Matlab/Simulink* and the LQG controller.

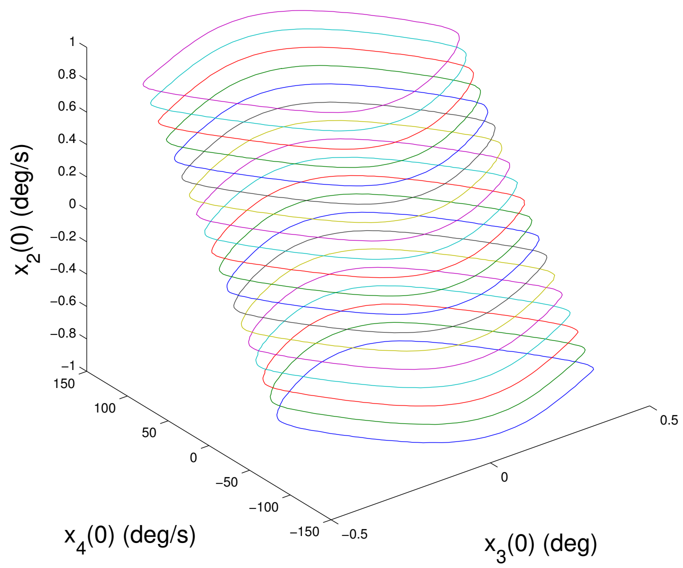


Figure 4.7: FP initial conditions in which equilibrium is achieved, where the lines represent the boundary of the region of attraction. $x_1(0)$ and $x_5(0)$ are set to zero. Simulations are performed with the non linear model using *Matlab/Simulink* and the LQG controller.

4.4.1 Continuous transition between controllers

Consider the linear controller with equations

$$\begin{cases} \dot{\hat{x}} = (A - BK - LC)\hat{x} - Le \\ u = -K\hat{x} \end{cases}, \tag{4.12}$$

and initial state

$$\hat{x}(0) = z. \tag{4.13}$$

In order to make the transition between the swing-up and linear controllers smoother, the input is set to be a continuous function on the transition. For that purpose, the initial estimate of the state of the linear controller $\hat{x}(0)$ was selected as the vector that fulfils the following optimization problem:

$$\begin{cases} \min \quad (\hat{x}(0) - x(0))^2 \\ s.t. \;\; K\hat{x}(0) + u^* = 0 \end{cases}. \tag{4.14}$$

Let u^* be the last input given by the swing-up controller before the commutation

$$u^* = -K\hat{x}(0) = -Kz, \tag{4.15}$$

which may be rewritten by expanding the vector product as a summation over the components

$$\begin{cases} \min \quad \sum_{i=1}^{n} (z_i - x_i(0))^2 \\ s.t. \;\; \sum_{i=1}^{n} K_i\hat{x}_i(0) + u^* = 0 \end{cases}. \tag{4.16}$$

Using the method of Lagrange multipliers, with multiplier λ, this problem is equivalent to

$$\min_{z} \max_{\lambda} \mathcal{L}(z, \lambda), \tag{4.17}$$

where

$$\mathcal{L}(z, \lambda) = \sum_{i=1}^{n} (z_i - x_i(0))^2 + \lambda \left(\sum_{i=1}^{n} K_i z_i + u^* \right). \tag{4.18}$$

Application of the method yields

$$\begin{cases} \frac{\partial}{\partial z_i}\mathcal{L}(z, \lambda) = 2(z_i - x_i(0)) + \lambda K_i = 0 \\ \frac{\partial}{\partial \lambda}\mathcal{L}(z, \lambda) = \sum_{i=1}^{n} K_i z_i + u^* = 0 \end{cases}, \tag{4.19}$$

solving the first equation for z_i and replacing the result in the second

$$\begin{cases} z_i = x_i(0) - \frac{1}{2}\lambda K_i \\ \sum_{i=1}^{n} K_i \left(x_i(0) - \frac{1}{2}\lambda K_i \right) + u^* = 0 \end{cases}. \tag{4.20}$$

Finally, the solution for the Lagrange multiplier and initial state estimative is found

$$\begin{cases} \lambda = 2\frac{\sum_{i=1}^{n} K_i x_i(0)+u^*}{\sum_{i=1}^{n} K_i^2} \\ z_j = x_j(0) - \frac{\sum_{i=1}^{n} K_i x_i(0)+u^*}{\sum_{i=1}^{n} K_i^2} K_j \end{cases} . \tag{4.21}$$

4.4.2 Gain Scheduling controller

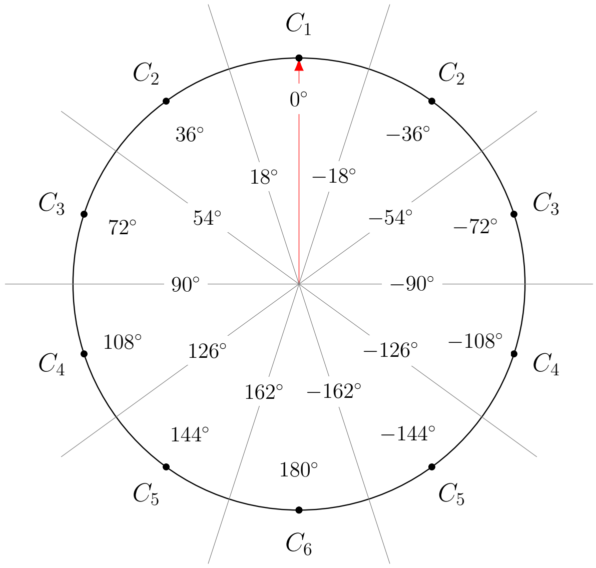


Figure 4.8: Gain scheduling controller regions of operation, as a function of the angle β (x_3). The dots mark the points where Jacobian linearisation is performed, and C_1 to C_6 the controller that operates at that particular region. For example C_1 operates for $\beta \in [-18, 18]$°, and is responsible for keeping the pendulum equilibrated in the upwards position.

The gain scheduling controller implemented in the control of the FP selects one between 6 controllers to be used accordingly to the current angle of the pendulum (figure 4.8). Note that they are disposed symmetrically: the first of the Taylor series is identical for $x_3 = \pm\theta$, and thus is the incremental model as well. The estimate of the full state is provided by a global estimator, that filters sensor data. The controller at the upwards position incorporates a dedicated LQE.

Each LQR controller, which is designed independently, corrects the value of the input variable in order to approximate the behaviour of the real system to the one predicted in the reference trajectory (figure 4.9).

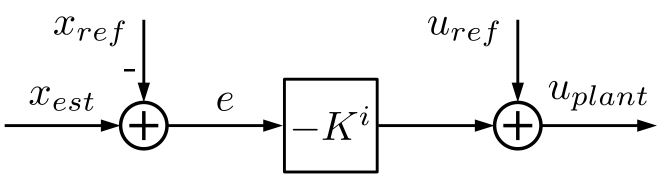


Figure 4.9: Gain scheduling controller architecture. The vector $K^i, i \in \{1, \cdots, 6\}$ is selected accordingly to the current angle of the pendulum.

Results show that the application of the corrections produced by the controller centred at 72 deg are counterproductive, as the probability of a successful swing-up are greatly reduced. This is due to the high gains that this controller exhibits, independently of the weights chosen. A qualitative analysis suggests that the system is *difficult* to control in this region, and this is reflected on the gains obtained with the LQR methodology. The output of this particular controller was chosen not to be applied to the device.

Chapter 5

Control Hardware

This chapter provides a description of the experimental implementation, including a comparison between a commercial and a custom made apparatus used in the laboratory.

Two different control system are employed to test and apply the developed model

1. a commercial, off-the-shelf solution acquired from the manufacturer *National Instruments*. It is composed by a personal computer (PC), running *Matlab/Simulink*, a PCI-6040E board (DAQ) connected to the PC at a PCI port, and an analog amplifier. The sensors – optical quadrature encoders (QE) – at the joints of the FP are connected to the DAQ directly. The actuator signal, produced by a digital to analog converter (DAC) at the DAQ is first amplified by the external analog amplifier before being applied to the motor.

2. a custom made board, designed and produced by the author, capable of controlling the device without external components. It incorporates the functions of the DAC, PC, and analog amplifier of the commercial solution, and provides additional functions such as current measurement and analog filtering. The QE are connected to this board directly.

The PCI-6040E DAQ includes the following characteristics:

- 2 up/down counters/timers, $24\mathrm{bit}$, at a maximum $20\mathrm{MHz}$;
- 2 analog output channels, $12\mathrm{bit}$, sourcing $< 5\mathrm{mA}$;
- 16 analog input channels, $12\mathrm{bit}$, at a maximum $250\mathrm{kS/s}$;

The motivation for the development of the control system derive from a number of shortcomings of the commercial solution, when applied to the FP

- Encoder resolution – the DAQ does not have a native quadrature encoder interface, instead, the up/down counters are used, which reduces the resolution of the sensors by a factor 4. Thus in this configuration the reading of the sensors have 1024 levels per revolution, instead of the 4096 available if adequate hardware was used;

- Unmeasured states – only 3 in 5 of the states variables can be measured directly: the angular positions of both links with quadrature encoders and the speed of the horizontal arm with a tachometer. It is not possible to measure directly the speed of the pendulum, since the timers are already being used as counters, or the current of the motor, since the amplifier does not provide sensing capabilities.

- Cost – the estimated cost of the control electronics is $> 3000\ \mathrm{EUR}$. To this price adds the licenses for the proprietary software compatible with this DAQ;

- Deployment time – using *Matlab/Simulink* the time required to compile and deploy the program is about $60\ \mathrm{s}$;

- Reliability – as the control software grows in complexity, the development environment is subject to frequent errors which often require the restart of the PC. It is clear that a system with this configuration could not be used in a production environment.

In order to overcome the above constraints, the following minimum requirements were defined for a custom made control system

- 2 QE interfaces;

- 2 timers;

- 1 Full H-bridge with current measurement;

- 2 channels of analog filtering with amplification, used for current measurements;

- Microcontroller with digital signal processing capabilities and configurable speed;

- Communication with a computer using a standard USB port, for data logging or interfacing with external software.

5.1 Schematic

The components of the board were chosen to abbey to the defined requirements, but also to provide versatility, while maintaining costs as low as possible. The circuit (see appendix B) has a modular configuration, in which not all components must be soldered, if they are not to be used, or can be turned off by an appropriate choice of the jumper configuration.

5.1.1 Microcontroller

The central component of the board is a dsPIC33FJ128MC804 (dsPIC) microcontroller manufactured by *microchip*, which fulfils the minimum requirements defined. Moreover it has 16-bit architecture and operates at a maximum $40\ \mathrm{MIPS}$, allowing the selection of the necessary maximum speed once the code performance is assessed. The clock reference is provided by a quartz crystal running at $4\ \mathrm{MHz}$, with two load capacitors of $18\ \mathrm{pF}$ attached.

The dsPIC can be programmed with a PICKIT through the header $\mathrm{J8}$. The circuit has been designed to allow the programming by a Raspberry Pi computer (RPi), using its general purpose input/output (I/O) pins, although this feature has not been tested. Reset can be induced using the press button $\mathrm{SW1}$. An input button $\mathrm{SW2}$ is also available, as well as two Light Emiting Diodes (LEDs), connected to the I/O pins of the dsPIC.

Three separate grounds exist in the board – digital, analog and power. They are connected through inductors to minimize the transmission of noise. All digital integrated circuits (ICs) have $100\ \mathrm{nF}$ capacitors connected between the supply voltage and digital ground to decouple digital noise.

5.1.2 Communications

The dsPIC communicates via RS232 protocol with TTL logic levels with the FT232R, at a maximum baudrate of $3\ \mathrm{Mbit/s}$. This IC handles the USB connection and is seen by a connected PC as a regular serial port. Two LEDs signal data being received and transmitted. The μUSB port can be used to power the board, either from a USB port of a PC, or a standard $5\ \mathrm{V}$ USB power supply. Another serial port is connected to the RPi header, providing direct TTL communications with it through the USART.

The dsPIC is also connected to the MAX485 IC, which provides a RS485 serial port, if needed on further developments.

The RPi or a standard computer can be used for human interface either remotely, through web services, or locally, using a console or graphical user interface, for controlling the board operation.

5.1.3 Load driving

The Pulse Width Modulation (PWM) motor control pins of the dsPIC are connected to the DRV8844, a quad half-bridge. This IC provides a maximum $2.5\mathrm{A}$ per channel and allows for the driving of the motor of the FP. The sense pins of this IC are connected to shunt resistors ($\mathrm{R38}$ and $\mathrm{R47}$), used for current measurement, allowing a fast feedback overcharge protection and an indirect measurement of the torque applied, by sensing the flux.

Also present in the board is a ULN2803, an 8-Darlington transistor array, useful for driving external on/off loads such as lighting and relays.

5.1.4 Current measurement and analog filtering

The current across the shunt resistors creates a voltage, according to Ohm's law. The signal is subject to a low pass filter with cutoff frequency of about $330\ \mathrm{Hz}$ and amplified by a factor 22. It uses the voltage reference of $2048\ \mathrm{mV}$ provided by the ADR380ARTZ IC.

5.1.5 Raspberry Pi interface

The board is designed to have a shield such as a RPi, capable of handling web services and data-logging, providing publishing and interface services for humans. The header connects the general purpose I/O pins of the RPi to the board, enabling serial communication, programming of the dsPIC, and power supply sharing.

5.1.6 Power supply

The circuit have been designed with a great flexibility for different power strategies, according to the peripherals connected. Power may be supplied by 4 different means:

1. unregulated $V_{in} \in [8, 50]$ V. This voltage is applied to the external loads of the DRV8844 and ULN2805, but can is limited to $V_{in} \in [6, 25]$ V if the LM1085 is used to provide 5 V to the board. This input is protected with a diode to avoid incorrect polarity.

2. unregulated $V_{in} \in [4.5, 5.5]$ V, if no external loads are used. This is applied to the input of the 3.3 V regulator, LM3940;

3. regulated 5 V over μUSB port

4. regulated 3.3 V, applied directly to the dsPIC.

5.2 Printed Circuit Board

The Printed Circuit Board (PCB) (figure 5.1) was designed with separate zones and associated ground planes according to their functionality: digital, analog and power. It comprises redundant connections: screw terminals and male headers. The format is a standard 3U Eurocard with $100\ \mathrm{mm} \times 160\ \mathrm{mm}$, and is fabricated in a double layer board. Components are mounted in the top layer, except for the female headers that connect to the RPi, that populates the bottom layer.

5.2.1 Fabrication

The PCB was manufactured with the following sequential steps:

1. copper clad board selection and cleaning;

2. manufacturing of layers, through holes and vias with CNC machine;

3. manual drilling of higher diameter holes;

4. soldering of vias;

5. soldering of components;

6. testing.

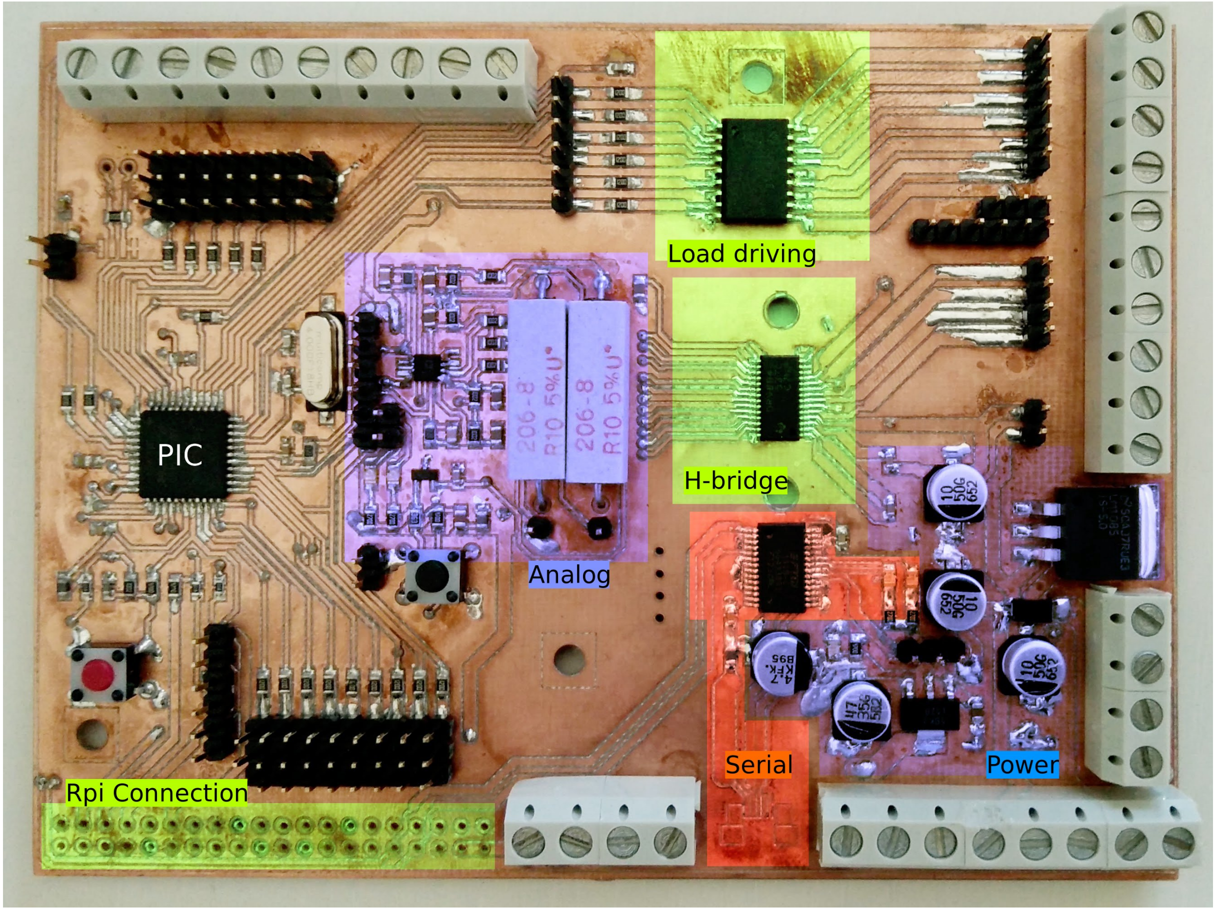


Figure 5.1: Custom made controller board with essential components soldered. The colours identify each functional zone.

Future versions of the board are to be outsourced from steps 1 to 4. Minor errors were detected and corrected for future production batches. The estimated cost per board, including components and PCB fabrication is $< 50\ \mathrm{EUR}$.

5.3 Software

The software implements the algorithm for swing-up using exponentiation of the pendulum position (see 3.1.2) and equilibrium with a LQR controller (see 4.1.1). Note that a LQG was not implemented since all states are measured directly, although the use of a estimator could improve the performance of the control.

The control loop works at $4\ \mathrm{KHz}$, as well as the digital readings from the QE. Analog measurements are performed 5 times per control cycle, *i.e.* at $20\ \mathrm{KHz}$. The samples are averaged which improves the immunity to higher frequency noise. Data is transmitted over the serial port at $100\ \mathrm{Hz}$.

The software developed shows that the developed hardware is capable of performing the swing-up and maintain the FP equilibrated at its unstable, upward position. Experiments performed with the previous apparatus were not repeated, as that would not contribute to a further understanding of the system, but would require extensive porting of the control software developed in *Matlab/Simulink*.

5.3.1 Shortcomings

The greatest disadvantage of the new control system is that it is not integrated in the *Matlab/Simulink* development cycle, and thus changing from simulation to the real device is more difficult. However it is still possible to use the serial port interface on *Matlab/Simulink* to control this board, although this solution introduces additional time overhead to the control cycle.

Chapter 6

Results

In this chapter the main results obtained experimentally are presented and compared with the results of simulations.

6.1 Swing-up with optimal control

The optimal control calculated with the L_2 norm in 3.2.1 is applied to the real device (figure 6.1). The reference input is corrected by the gain scheduling controller, which accounts for the differences between the reference and the output variables, and actuates on the input to approximate the behaviour of the system to the one predicted in simulation. In a similar way, the reference produced with the L_1 norm in 3.2.2, was applied to the real device (figure 6.2).

In both cases, after swing-up is completed, the reference was set to zero for both input and output variables.

State variables are being estimated globally by a filter. Its transient response causes a visible spike at the first $50ms$ in the input variable. Experimental data follows the reference trajectory with small deviations for angles where the pendulum is on a downwards position. At the final swing, and when the pendulum is above horizontal, the gain scheduling controller actuates more aggressively in order to ensure that the pendulum can perform the swing-up manoeuvre. With the L_2 norm this affects the tracking of the reference of the angle of the horizontal arm.

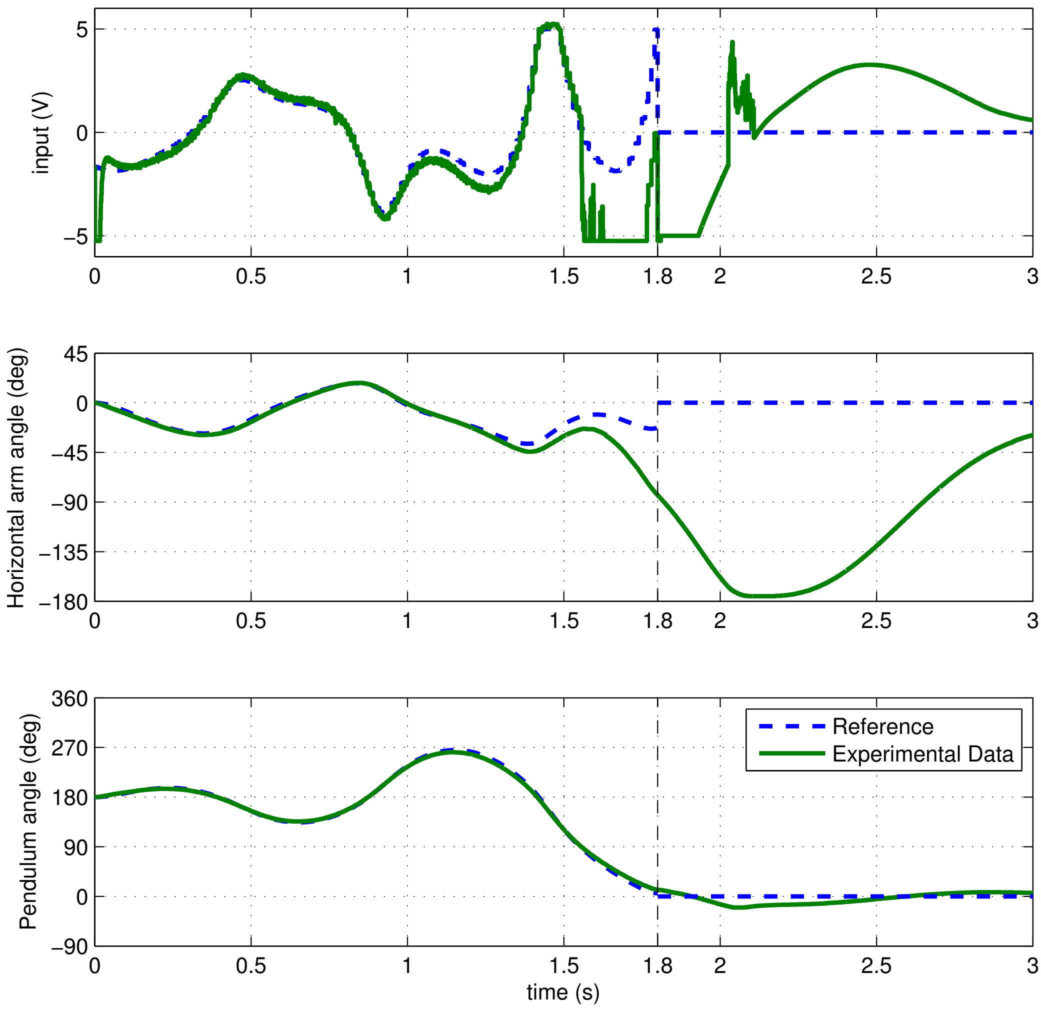


Figure 6.1: Swing-up performed with optimal control (L_2 norm). Input applied from reference and corrected with a gain scheduling controller. After $t = 1.8s$ the references for the input and output are set to zero. Sampling time $1\mathrm{ms}$.

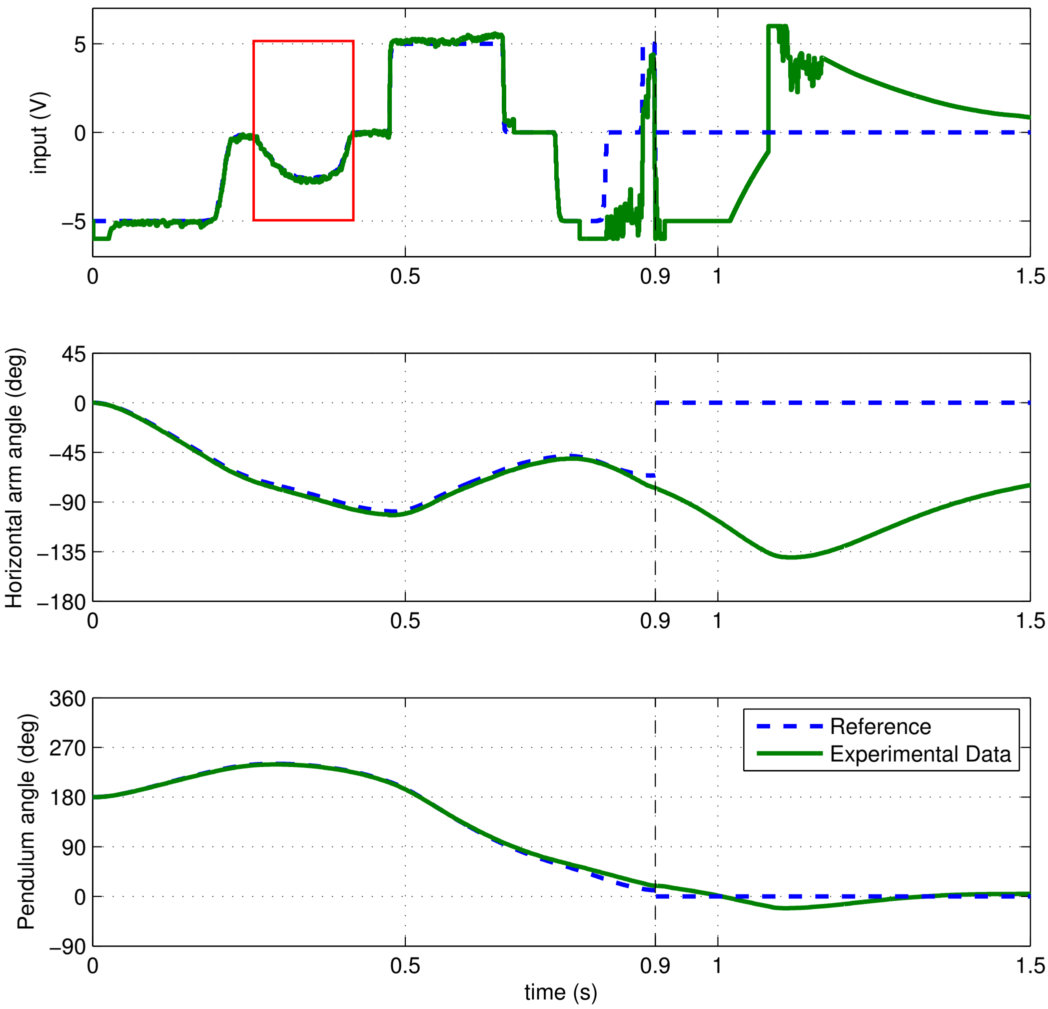


Figure 6.2: Swing-up performed with optimal control (L_1 norm). Input applied from reference and corrected with a gain scheduling controller. After $t = 0.9s$ the references for the input and output are set to zero. Sampling time $1\mathrm{ms}$.

6.2 Swing-up with ad hoc strategies

The swing-up was performed experimentally with two *ad hoc* techniques: energy shaping and exponentiation of the pendulum position. Figure 6.3 shows the results of using energy shaping, where experimental results are compared to the ones obtained in simulation in two ways:

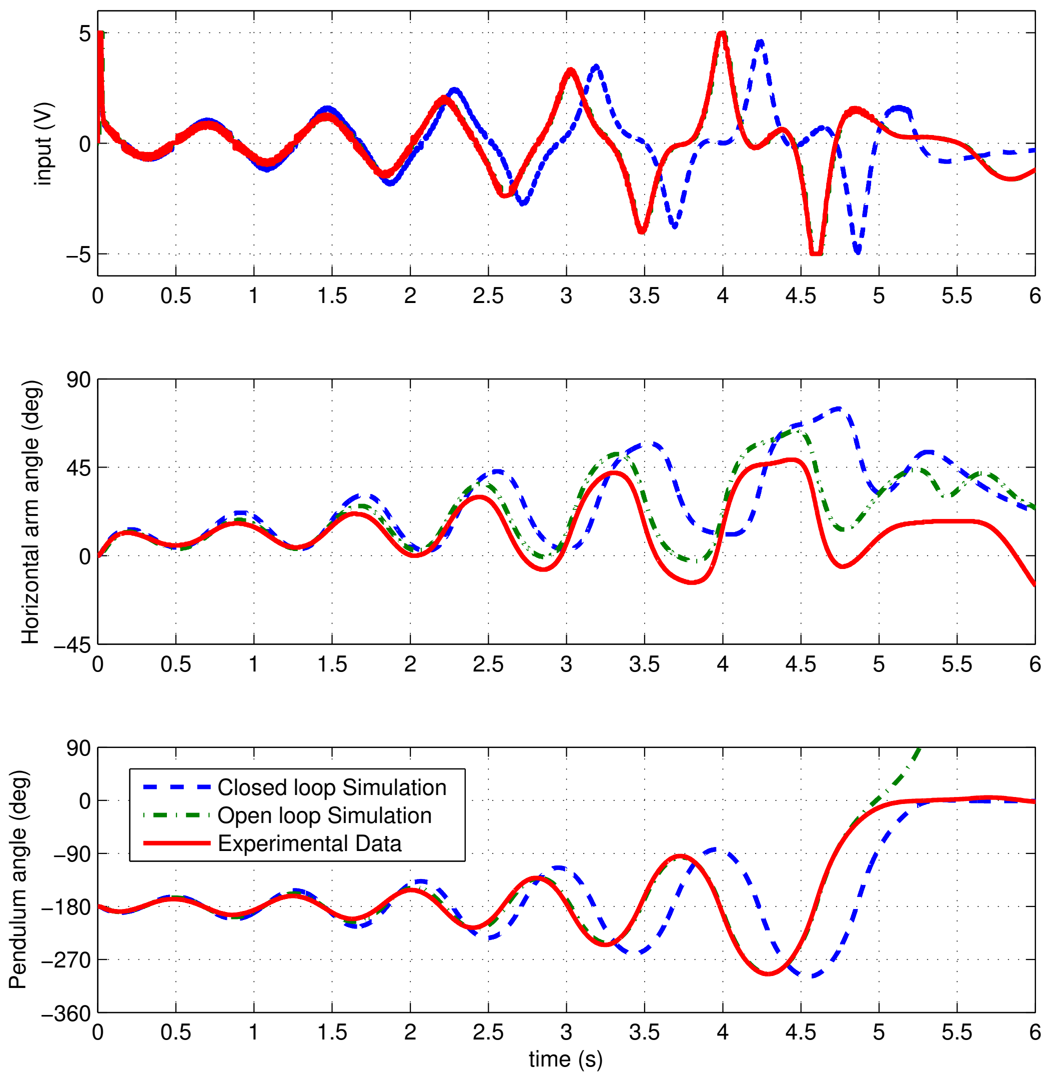


Figure 6.3: Swing-up performed with energy shaping. The experimental data is compared with the results of simulation. In open loop simulation, the input signal of the experimental data is applied to a model of the device, and no feedback control is performed. In closed loop simulation, the control is performed with energy shaping, by using the outputs of the simulated model. Sampling time $1\mathrm{ms}$.

1. by simulating the response of the closed loop system. The input is calculated with the state obtained in simulation and the rule defined in 3.1;

2. by simulating the response of the input applied experimentally. This is referred to as open-loop simulation, as the input is established *a priori*.

The results are used to test the accuracy of the model. The closed loop simulation amplifies the small deviations, and has a response stretched in time. However the open loop simulation is indistinguishable from the response of the real system in the pendulum angle during swing-up, and only diverges after the upwards equilibrium position is achieved.

With exponentiation of the pendulum position the swing-up was achieved with the commercial (figure 6.4) and the custom made (figure 6.5) control systems.

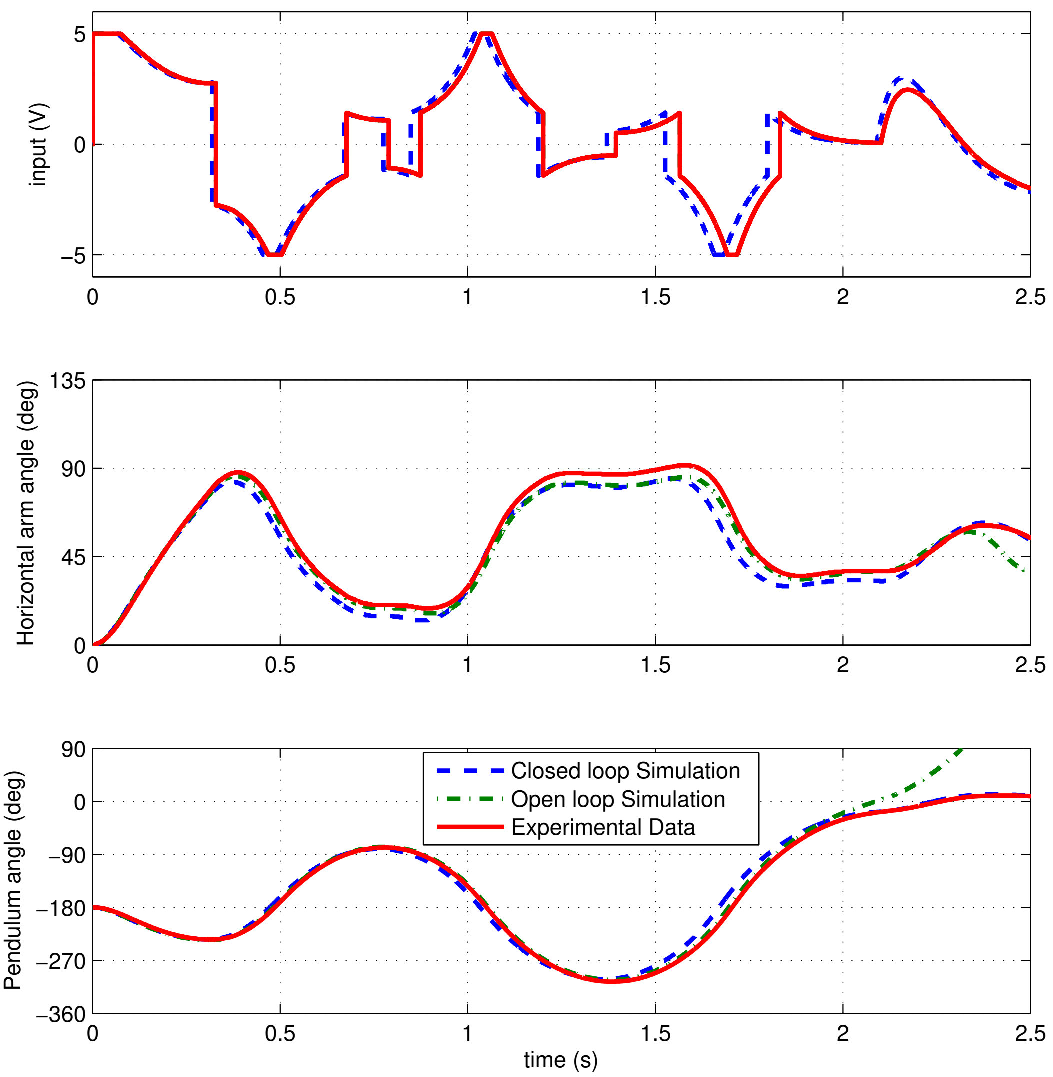


Figure 6.4: Swing-up performed with exponentiation of the pendulum position. The experimental data is compared with the results of simulation. In open loop simulation, the input signal of the experimental data is applied to a model of the device, and no feedback control is performed. In closed loop simulation, the control is performed with the exponentiation of the pendulum position, by using the outputs of the simulated model. Sampling time $1\mathrm{ms}$.

The comparison of the experimental data with the open and closed loop simulations yield similar results. In this case, however, simulations differ less than with the energy shaping swing-up. This is due

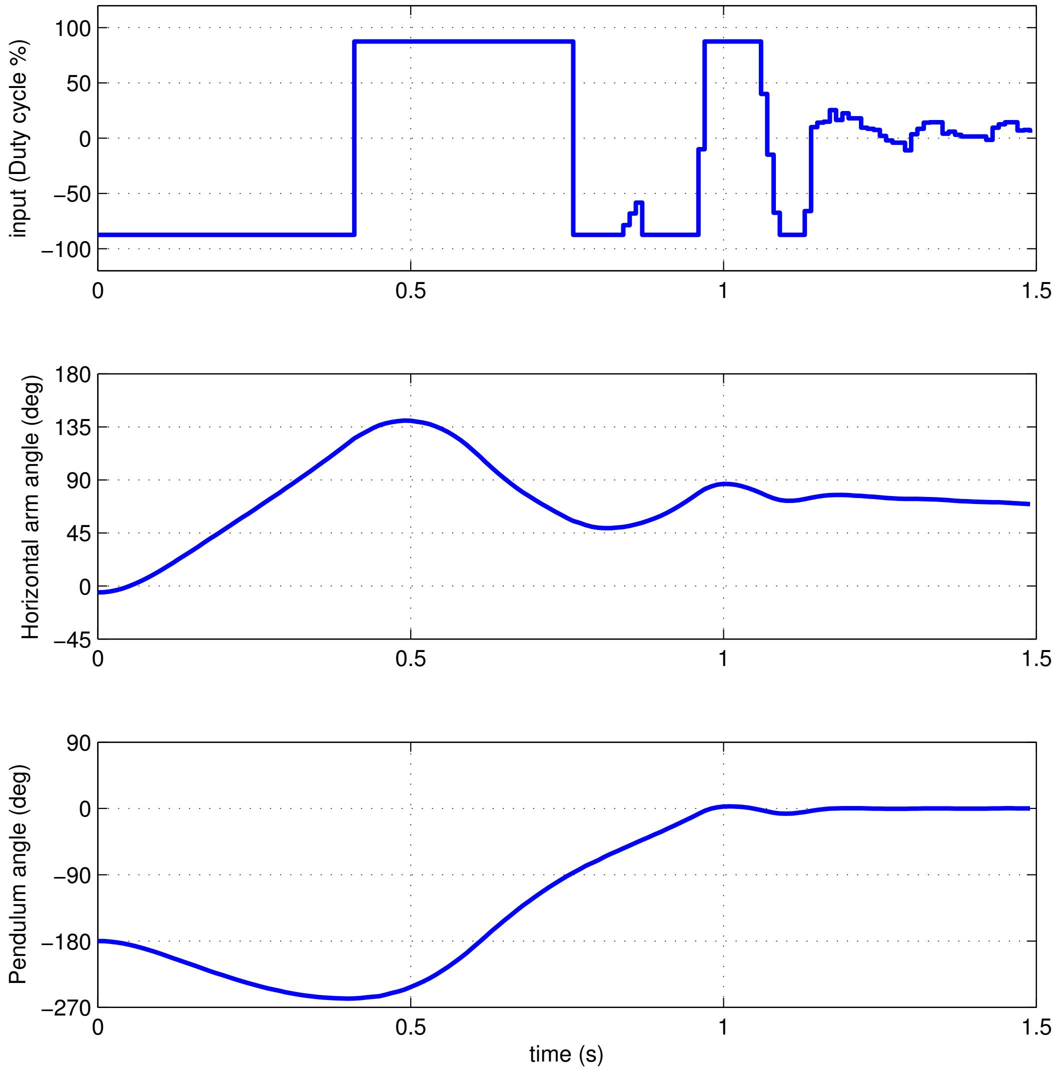


Figure 6.5: Swing-up performed with exponentiation of the pendulum position using the custom made board. Sampling time $10\mathrm{ms}$, control period $250\mu\mathrm{s}$.

to two reasons: 1) The time of swing-up with exponentiation is shorter ($2\ \mathrm{s}$) than with energy shaping ($5\ \mathrm{s}$) and 2) the identification with least-squares has been performed with exponentiation, with similar conditions, thus it is expected that parameters are especially well adjusted to the response that this method generates, since the same frequencies are excited.

Results obtained for the swing-up with the custom made control system have similar characteristics to the ones obtained with the commercial solution (figure 6.5). Equilibrium has also been achieved also shows the equilibrium maintained with a LQR controller.

Table 6.1: Normalized cost of different swing-up strategies

Method	$T(\mathrm{s})$	L_1	L_2
Energy shaping	3.4	359	557
Exponentiation	2.2	323	846
Energy control	2.5	360	1275
Optimal control L_1	0.9	345	1441
Optimal control L_2	1.8	213	452

6.3 Comparison of swing-up methods

Qualitatively, the swing-up performed with the L_2 norm proved to be the smoothest and most reliable of all methods used. The swing-up performed with the L_1 norm was also very reliable, but fast changes of the control result in an aggressive manoeuvre.

The different swing-up methods are compared in terms of cost, using the metrics of optimal control, normalized in respect to the time of simulation (table 6.1).

Results show that the optimal trajectory calculated with $L_i,\ i \in 1, 2$ norm, has a lower cost then the remaining methods, when measured with that specific norm. The similarities of each method with optimal control may be measured with this metric.

6.4 Equilibrium

Equilibrium was maintained with two different LQG controllers (see 4.1), one continuous (figure 6.6) and one discrete (figure 6.7).

Results show that the equilibrium maintained with the discrete controller has smaller variance than the one obtained with the continuous controller. The control is performed by a digital computer, therefore it will always have a discrete nature. While using *Matlbal/Simulink* continuous blocks, the discretisation is handled by the software. With the discrete controller the software only handles basic calculations, and the discretisation is handled by the engineer.

Note that the parameters of each controller have been designed independently, and while the design goals have been the same for both, there is no guarantee that better results could not be achieved with a different choice of parameters.

The equilibrium achieved with the custom made control system (6.8) has lower variance when compared with the control performed with the commercial solution. However, the angle of the horizontal arm has a oscillatory characteristic with a period of about $8\mathrm{s}$.

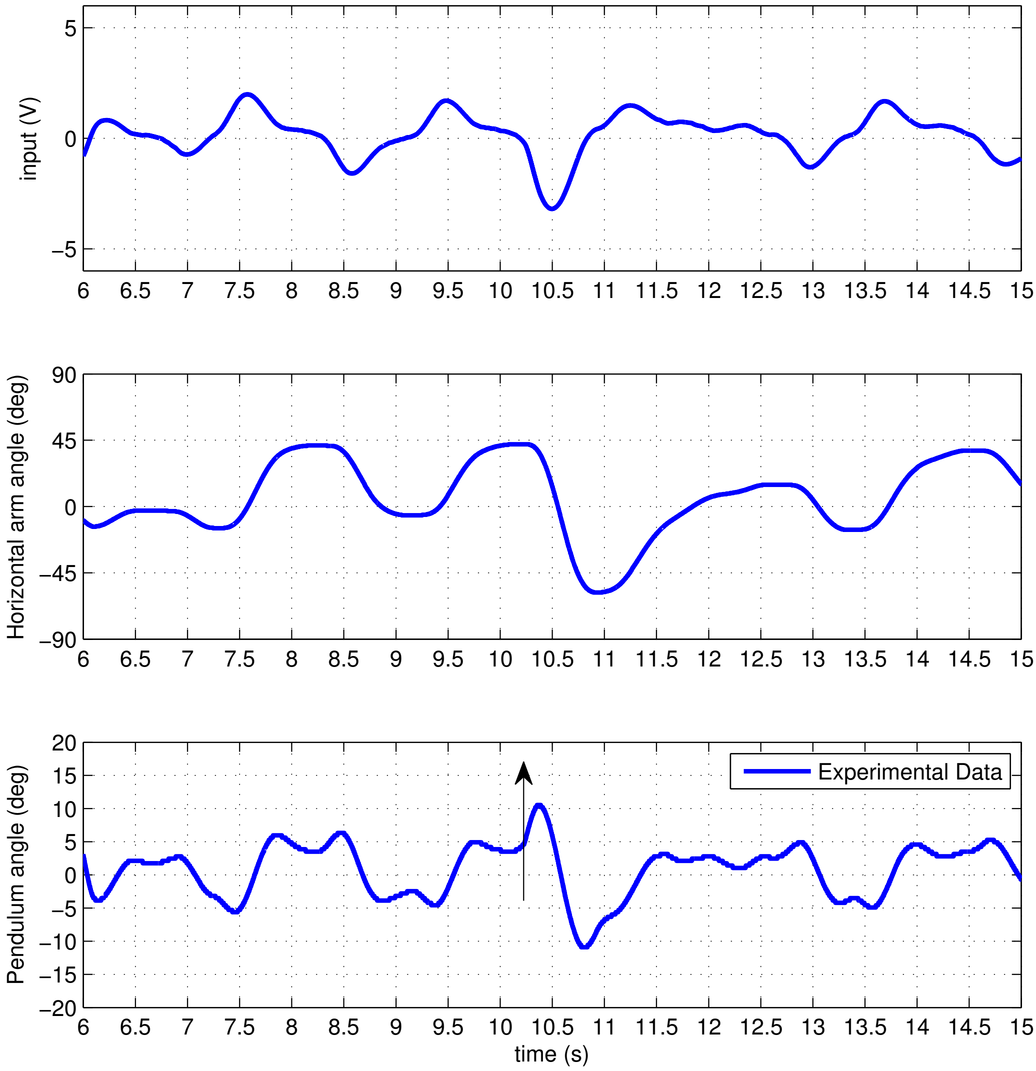


Figure 6.6: Equilibrium maintained with a continuous LQG controller. The arrow indicates an external perturbation applied to the pendulum. Sampling time $1\mathrm{ms}$.

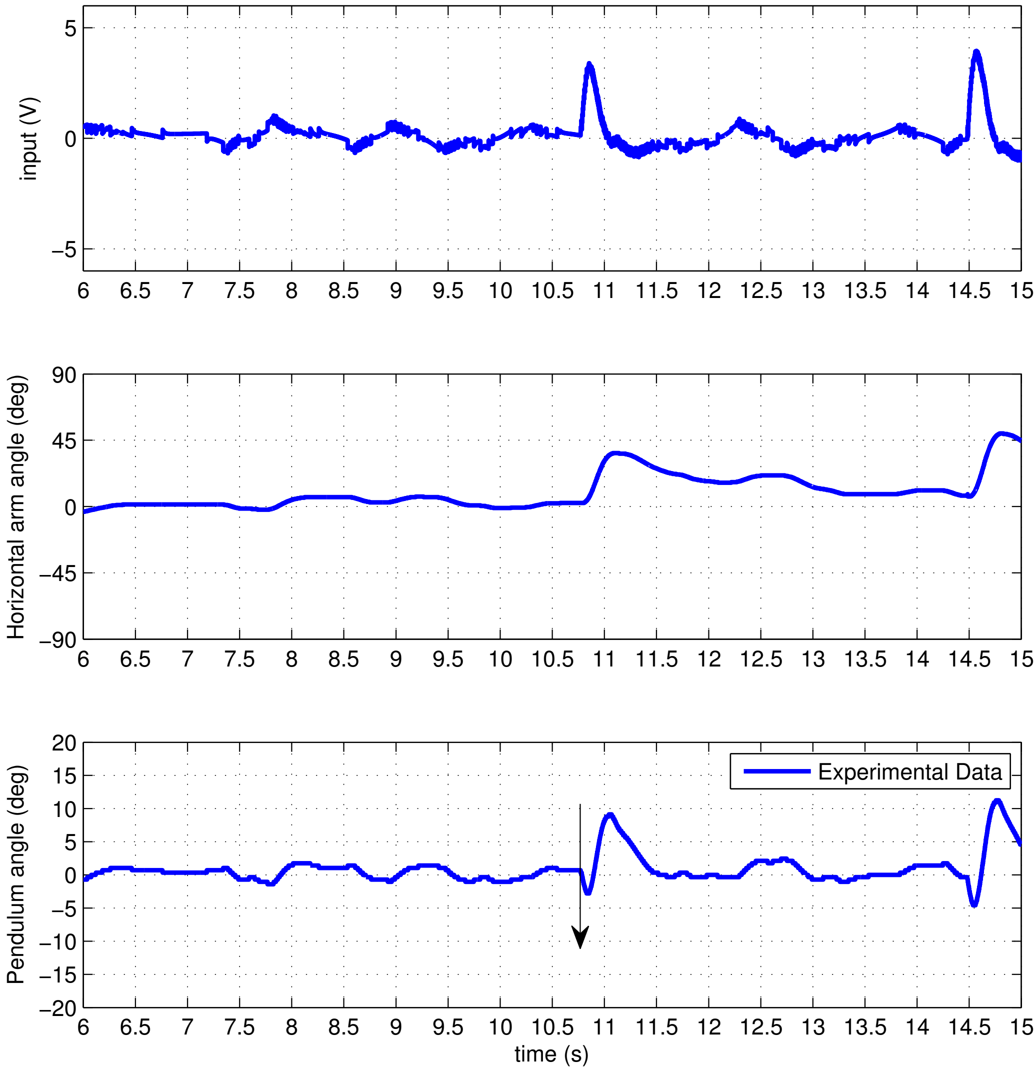


Figure 6.7: Equilibrium maintained with a discrete LQG controller. The arrow indicates an external perturbation applied to the pendulum. Sampling time $1\mathrm{ms}$.

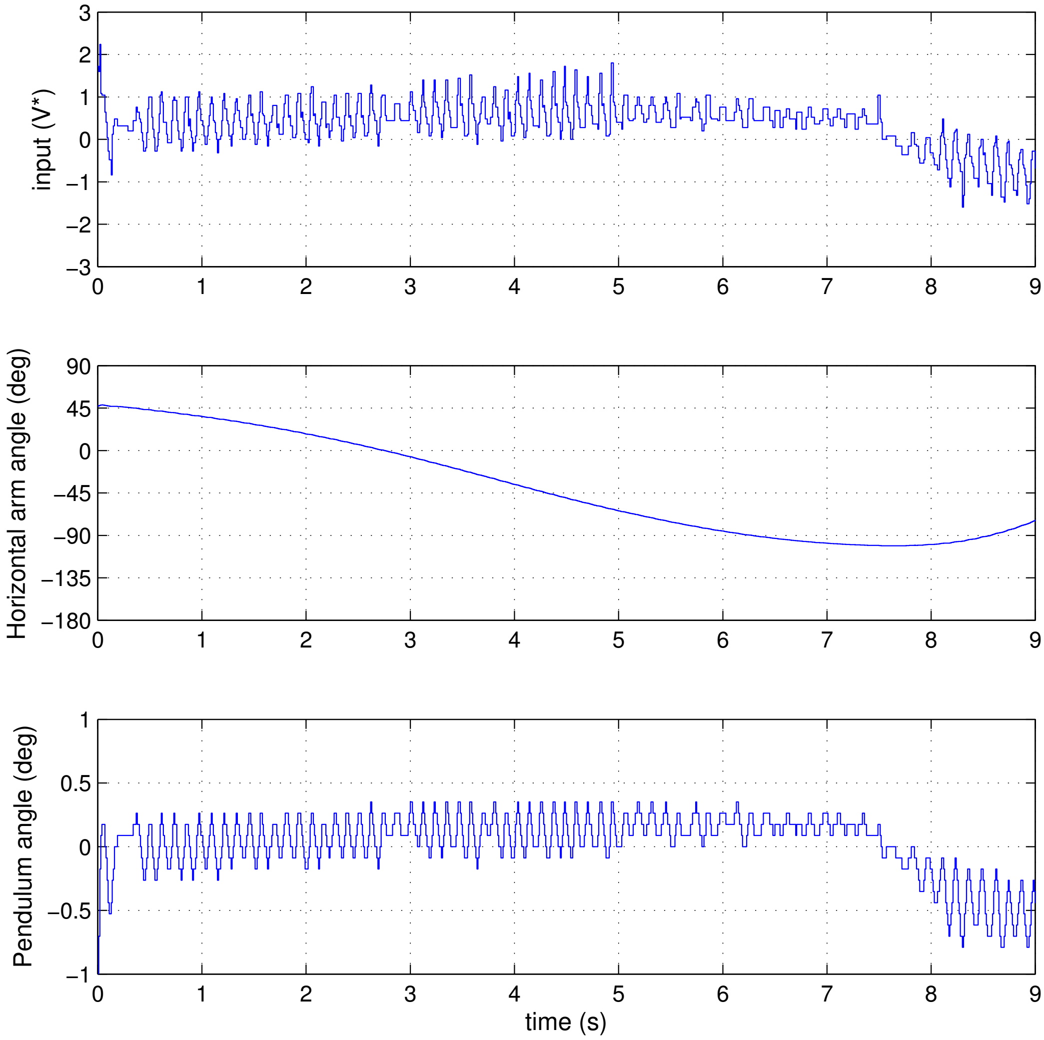


Figure 6.8: Equilibrium maintained with a discrete LQR controller immediately after swing-up, using the custom made control system. Sampling time $10\mathrm{ms}$, control period $250\mu\mathrm{s}$. This data set does not contain external perturbations, besides the initial offset due to the end of the swing-up manoeuvre, which has been omitted due to the different y-magnitudes involved.

Chapter 7

Conclusion

Results show that optimal control is an effective method to calculate state space trajectories for the FP, an under-actuated system with non linear dynamics. The characteristics of the control function and trajectory generated can be altered by an appropriate choice of the cost function structure and weights. Results also suggest that the method can be applied to a wide range of system with different characteristics, if an appropriate numerical method is used.

Results also show that the gain scheduling controller is an effective way to have a nonlinear system follow a reference trajectory. Transitions between controllers can be made smooth, although this requires a trade-off with the speed of reaction.

Contrary to one could expect, when designing an equilibrium controller, it is more difficult to maintain the angle of the horizontal arm within bounds than the pendulum at its upwards position.

7.1 Achievements

A complete model of the system is developed, which captures the dynamics of the FP and actuator. The simulations performed with this model are in agreement with experimental data. The accuracy of the model allows for the design of controllers in a simulated environment, and a hassle-free transition to the real device. Both a global nonlinear and local linear models (continuous and discrete) are produced.

The numerical method for optimal control proposed in this work found solutions not only for the problem of swinging-up the FP, but for other unrelated systems such a wave energy converter. It offers improved stability over other methods such as FBS which failed to converge in both cases.

A technique is proposed and used to find the regions of attraction of a controller, but can also be used in any dynamic system as long as it fulfils the sufficient condition of this region being convex.

Equilibrium is achieved reliably with the use of both a continuous and discrete LQG controllers. They are able to stabilize the plant in the event of external perturbations. They are shown to be limited in their region of attraction by the power of the actuator.The method used to evaluate this region is original and general.

A custom made control system is developed and tested. Results show improved reliability when compared with the previously available commercial solution, which is 2 orders of magnitudes costlier. The new system is more compact and flexible, providing sensing capabilities for all 5 state space variables. The equilibrium achieved with this system has improved variance over the obtained with the commercial solution. This board may be used in the future to control a wide range of devices without additional external components, for example in a classroom environment.

7.2 Future Work

During the development of this work, several divergent lines of research or possible analysis were identified. However, given the time limitations for this thesis development, they have not been addressed:

- Other techniques can be used to model the system, which can be specially useful when there is no information available about its structure, for example neural networks;
- There is a wide number of numerical methods available that allow the calculation of optimal trajectories (and respective control inputs). Since the FP and related simplified systems proved to be a useful test ground, it is of interest to compare the performance of each of this methods systematically, using the results of simulations and experiments with the real device. This comparison could be done for several aspects, for example the cost as a function of the computational power required, or the robustness of the method, and could be done for gradient, shooting, and pseudo-spectral methods for example;
- With an appropriate cost function it should be possible to find minimum time trajectories, which are of interest in many applications;
- The terminal state may be imposed, instead of contributing to the cost function;
- Optimal control can be performed in discrete time, it would be of interest to analyse the differences to the continuous case;
- The numerical method proposed in this work could be applied to different systems, specially to ones where other methods have difficulties;
- A systematic study of the effects of having sensors for all states in the performance of the controller would be of interest.
- A study of the limits of performance of positioning, when equilibrating the pendulum;
- Use of MPC to perform control.

Bibliography

[1] H. K. Khalil and J. Grizzle. *Nonlinear systems*, volume 3. Prentice hall New Jersey, 1996.

[2] J.-J. E. Slotine, W. Li, et al. *Applied nonlinear control*, volume 199. prentice-Hall Englewood Cliffs, NJ, 1991.

[3] K. J. Åström and K. Furuta. Swinging up a pendulum by energy control. *Automatica*, 36(2):287–295, 2000.

[4] Q. Wei, W. P. Dayawansa, and W. Levine. Nonlinear controller for an inverted pendulum having restricted travel. *Automatica*, 31(6):841–850, 1995.

[5] A. E. Bryson and Y.-C. Ho. *Applied optimal control: optimization, estimation and control*. CRC Press, 1975.

[6] *The rotary control lab datasheet*. Quanser, 2014.

[7] I. Fantoni, R. Lozano, M. W. Spong, et al. Energy based control of the pendubot. *IEEE Transactions on Automatic Control*, 45(4):725–729, 2000.

[8] J. P. O. Oliver and V. L. Morales. Control based on swing up and balancing scheme for an experimental underactuated robot. In *Electronics, Robotics and Automotive Mechanics Conference, 2007. CERMA 2007*, pages 512–517. IEEE, 2007.

[9] G. F. Franklin, J. D. Powell, A. Emami-Naeini, and J. D. Powell. *Feedback control of dynamic systems*, volume 2. Addison-Wesley Reading, 1994.

[10] R. Isermann and M. Münchhof. *Identification of Dynamic Systems*. Springer, 2011.

[11] L. Ljung. System identification: Theory for the user. *Englewood Cliffs*, 1987.

[12] P. M. Mills, A. Y. Zomaya, and M. O. Tade. *Neuro-adaptive process control*. Wiley, 1966.

[13] C. E. Rasmussen and C. K. I. Williams. Gaussian processes for machine learning, 2006.

[14] J. Lourenço, J. M. Lemos, and J. S. Marques. Control of neuromuscular blockade with gaussian process models. *Biomedical Signal Processing and Control*, 8(3):244–254, 2013.

[15] R. Ortega, A. J. van der Schaft, I. Mareels, and B. Maschke. *Energy shaping control revisited*, pages 277–307. Springer London, London, 2001. ISBN 978-1-84628-570-7. doi: 10.1007/BFb0110388. URL `http://dx.doi.org/10.1007/BFb0110388`.

[16] S. Nair and N. E. Leonard. A normal form for energy shaping: application to the furuta pendulum. In *Decision and Control, 2002, Proceedings of the 41st IEEE Conference on*, volume 1, pages 516–521. IEEE, 2002.

[17] K. Groves and A. Serrani. Modeling and nonlinear control of a single-link flexible joint manipulator. *APPENDICES*, 2004.

[18] K. Guemghar, B. Srinivasan, P. Mullhaupt, and D. Bonvin. Analysis of cascade structure with predictive control and feedback linearisation. *IEE Proceedings-Control Theory and Applications*, 152(3):317–324, 2005.

[19] A. M. Lyapunov. The general problem of the stability of motion. *International Journal of Control*, 55 (3):531–534, 1992.

[20] E. D. Sontag. Control-lyapunov functions. In *Open problems in mathematical systems and control theory*, pages 211–216. Springer, 1999.

[21] M. Morari and J. H. Lee. Model predictive control: past, present and future. *Computers & Chemical Engineering*, 23(4):667–682, 1999.

[22] P. Seman, B. Rohal'-Ilkiv, M. Juhás, and M. Salaj. Swinging up the furuta pendulum and its stabilization via model predictive control. *Journal of Electrical Engineering*, 64(3):152–158, 2013. ISSN 13353632. doi: 10.2478/jee-2013-0022.

[23] L. E. Dubins. On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. *American Journal of mathematics*, 79(3):497–516, 1957.

[24] A. V. Rao. A survey of numerical methods for optimal control. *Advances in the Astronautical Sciences*, 135(1):497–528, 2009.

[25] H. J. Sussmann and J. C. Willems. 300 years of optimal control: from the brachystochrone to the maximum principle. *Control Systems, IEEE*, 17(3):32–44, 1997.

[26] L. Pontryagin, V. Boltyanskii, R. Gamkrelidze, and E. Mishchenko. The mathematical theory of optimal processes (international series of monographs in pure and applied mathematics). *Interscience, New York*, 1962.

[27] J. T. Betts. Survey of numerical methods for trajectory optimization. *Journal of guidance, control, and dynamics*, 21(2):193–207, 1998.

[28] J. G. Ziegler and N. B. Nichols. Optimum settings for automatic controllers. *trans. ASME*, 64(11), 1942.

[29] R. Genesio, M. Tartaglia, and A. Vicino. On the estimation of asymptotic stability regions: State of the art and new proposals. *Automatic Control, IEEE Transactions on*, 30(8):747–755, 1985.

[30] W. J. Rugh and J. S. Shamma. Research on gain scheduling. *Automatica*, 36(10):1401–1425, 2000.

[31] M. W. Spong, S. Hutchinson, and M. Vidyasagar. *Robot modeling and control*, volume 3. Wiley New York, 2006.

[32] B. S. Cazzolato and Z. Prime. On the dynamics of the furuta pendulum. *Journal of Control Science and Engineering*, 2011:3, 2011.

[33] J. M. Lemos. Bayesian parameter estimation in nonlinear dynamic regression models. *Techical Report*, 26:10, 2009.

[34] H. Schättler and U. Ledzewicz. *Geometric optimal control: theory, methods and examples*, volume 38. Springer Science & Business Media, 2012.

[35] D. E. Kirk. *Optimal control theory: an introduction*. Courier Corporation, 2012.

[36] J. C. C. Henriques, J. M. Lemos, L. Eça, J. N. H. Valério, and L. M. C. Gato. On the use of a discontinuous galerkin method and h-refinement for optimal control. *Unpublished*, 2016.

Appendix A

Numerical method

A.1 User oriented description

This section provides a user oriented description of the numerical method proposed in this work.

Given a cost function in the Bolza form

$$J[u(t)] = \Psi\left[\mathbf{x}(T)\right] + \int_{t_0}^{T} \mathcal{L}\left[\mathbf{x}(t), u(t)\right] dt, \tag{A.1}$$

1. An initial guess for the control function $u(t)$ is taken, where the time interval $[t_0, t_f]$ is subdivided into a set of N subintervals. For simplicity, each subinterval can be made of equal length. The input variable $u(t)$ is set to be piecewise-constant for each subinterval;

2. An initial value is defined for the smoothing factor ($\eta \leq 1$). Values are defined for the momentum term coeficient and actualization coefficient of the smoothing factor;

3. the equations of motion of the system are integrated forward in time

$$\dot{\mathbf{x}} = f[\mathbf{x}(t), u(t)], \tag{A.2}$$

$$\mathbf{x}(t_0) = \mathbf{x}_i; \tag{A.3}$$

4. the final value of the state-variables, $x(T)$, is used as initial condition for the backward integration of the co-state;

$$\boldsymbol{\lambda}'(T) = \Psi_{\mathbf{x}}\left[\mathbf{x}(T)\right], \tag{A.4}$$

$$-\dot{\boldsymbol{\lambda}}' = \boldsymbol{\lambda}'\, f_{\mathbf{x}}\left[\mathbf{x}(t), u(t)\right] + \mathcal{L}_{\mathbf{x}}\left[\mathbf{x}(t), u(t)\right]; \tag{A.5}$$

5. the method stops if any of the stopping criteria are met

   - the smoothing factor is smaller than a certain value;
   - the state-variables are very large (in which case the method has irremediable diverged);

- the maximum number of iterations is reached;

6. updated values are set for the control function at $t = t_k$, where $k = 0, \cdots, N$ and $u(t)$ is a piecewise-constant function

$$u^{(i+1)}(t_k) = \eta u^*(t_k) + (1-\eta)u^{(i)}(t_k) + \Delta u^{(i)}(t_k). \tag{A.6}$$

where

- u^* is the optimal value of the control function, that optimizes $\mathcal{H}(u)$ in a given moment t_k;
- η is the smoothing factor;
- Δu is the momentum term. It is calculated as

$$\Delta u^{(i)} = \delta(u^{(i)} - u^{(i-1)}), \tag{A.7}$$

with $0 \leq \delta \leq 1$. The momentum term is only applied at each element of the vector if the function is evolving consistently in the same direction, *i.e.*

$$(u_k^{(i+i)} - u_k^{(i)} - \Delta u_k^{(i)}) * \Delta u_k^{(i)} > 0; \tag{A.8}$$

7. The cost A.1 is calculated. If it has increased more than a given tolerance, the smoothing factor is actualized

$$\eta = \alpha\eta \tag{A.9}$$

where $0 < \alpha < 1$;

8. repeat from 3.

A.2 Application to a wave energy converter

In [36] a simplified model of a wave energy converter is used. The numerical method proposed in this work has been applied to this problem and yielded the optimal control.

The generator is modelled by a spring-mass-damper system, where a controllable latching system is used to improve efficiency

$$m\ddot{x}(t) + c\dot{x}(t) + kx(t) + Vu(t)\dot{x}(t) = F(t), \tag{A.10}$$

where m is the mass, c the damping coefficient due to the generator, k the stiffness of the spring, that simulates hidrodinamic forces, and $F(t)$ to the excitation force, due to the waves.

The following boundary conditions have been set

$$x(0) = 0, \tag{A.11}$$

$$\dot{x}(0) = 0, \tag{A.12}$$

Equation (A.10) can be rewritten as

$$\ddot{x} = \frac{F - c\dot{x} - kx - Vu\dot{x}}{m}, \tag{A.13}$$

which can be put in a state space form with the substitutions

$$x_0 = \dot{x}, \tag{A.14}$$

$$x_1 = x, \tag{A.15}$$

resulting in the system

$$\dot{x}_0 = \frac{F - cx_0 - kx_1 - Vux_0}{m}, \tag{A.16}$$

$$\dot{x}_1 = x_0. \tag{A.17}$$

The goal is to maximize the energy produced by the generator, *i.e.* to maximize the time integral of power

$$J = E = \int_0^{T_f} cx_0^2 dt. \tag{A.18}$$

By applying Pontryagin's maximum principle, one obtains

$$\mathcal{H} = \frac{F - cx_0 - kx_1 - Vux_0}{m}\lambda_0 + x_0\lambda_1 + cx_0^2 \tag{A.19}$$

and the costate equations

$$g_0 = \frac{c + Vu}{m}\lambda_0 - \lambda_1 - 2cx_0, \tag{A.20}$$

$$g_1 = \frac{k}{m}\lambda_0. \tag{A.21}$$

The application of the numerical method yields the results shown in figure A.1.

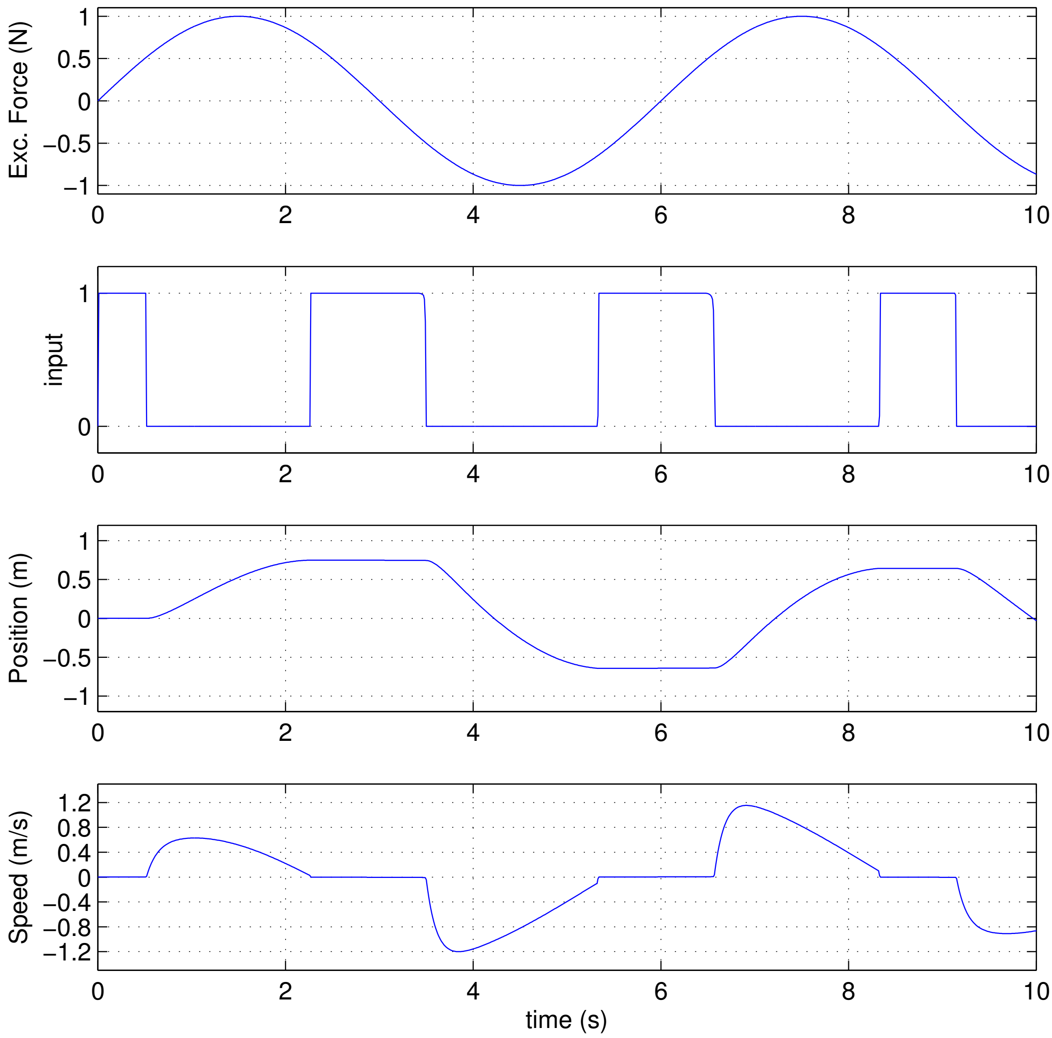
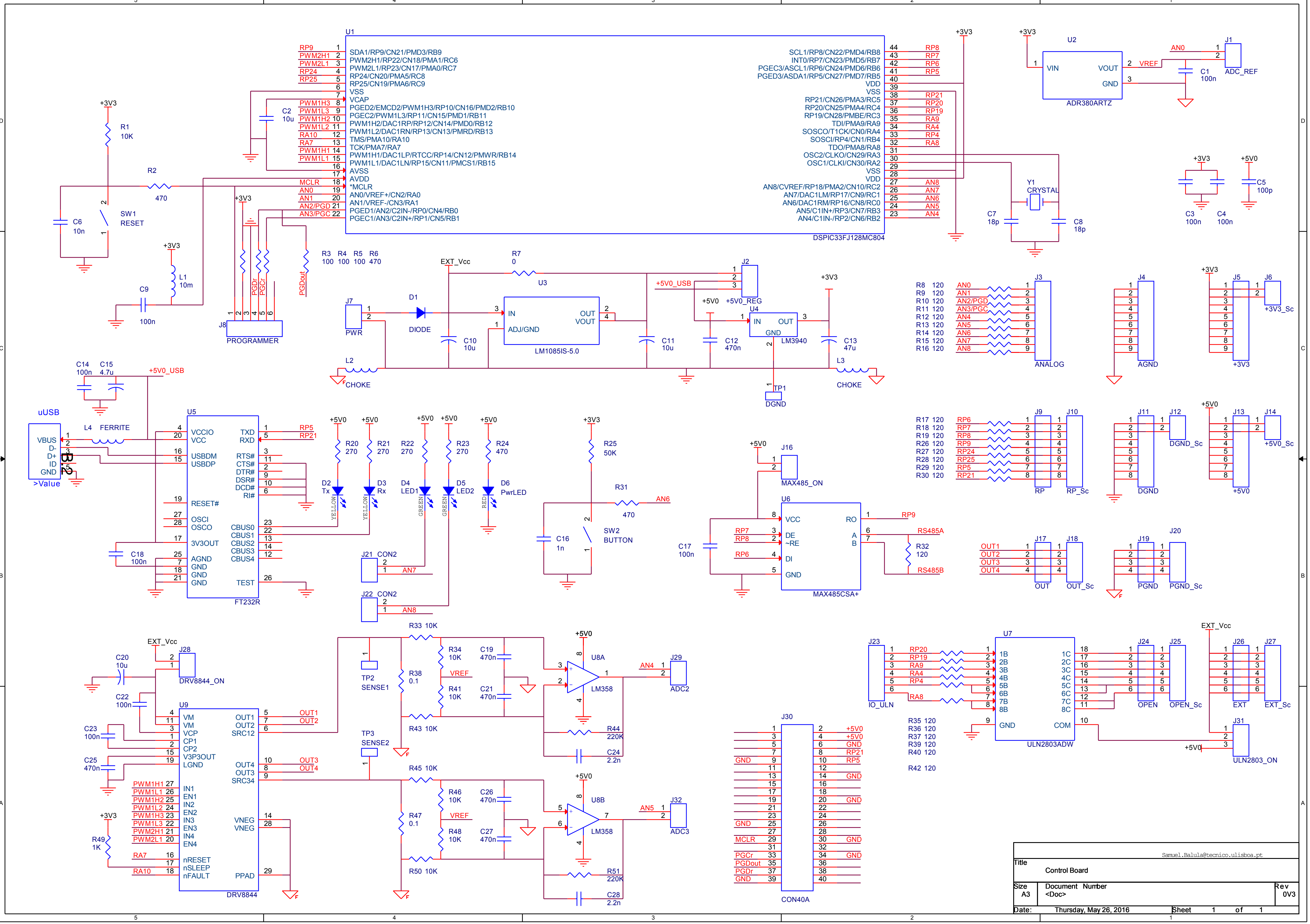


Figure A.1: Optimal control and system response of a spring-mass-damper, a simplified model of a wave energy converter. $f_{ext} = \frac{1}{6}$ Hz, $m = 0.1$ Kg, $c = 1$ Kg s^{-1}, $k = 1$ Kg s^{-2}, $V = 200$ Kg s^{-1}.

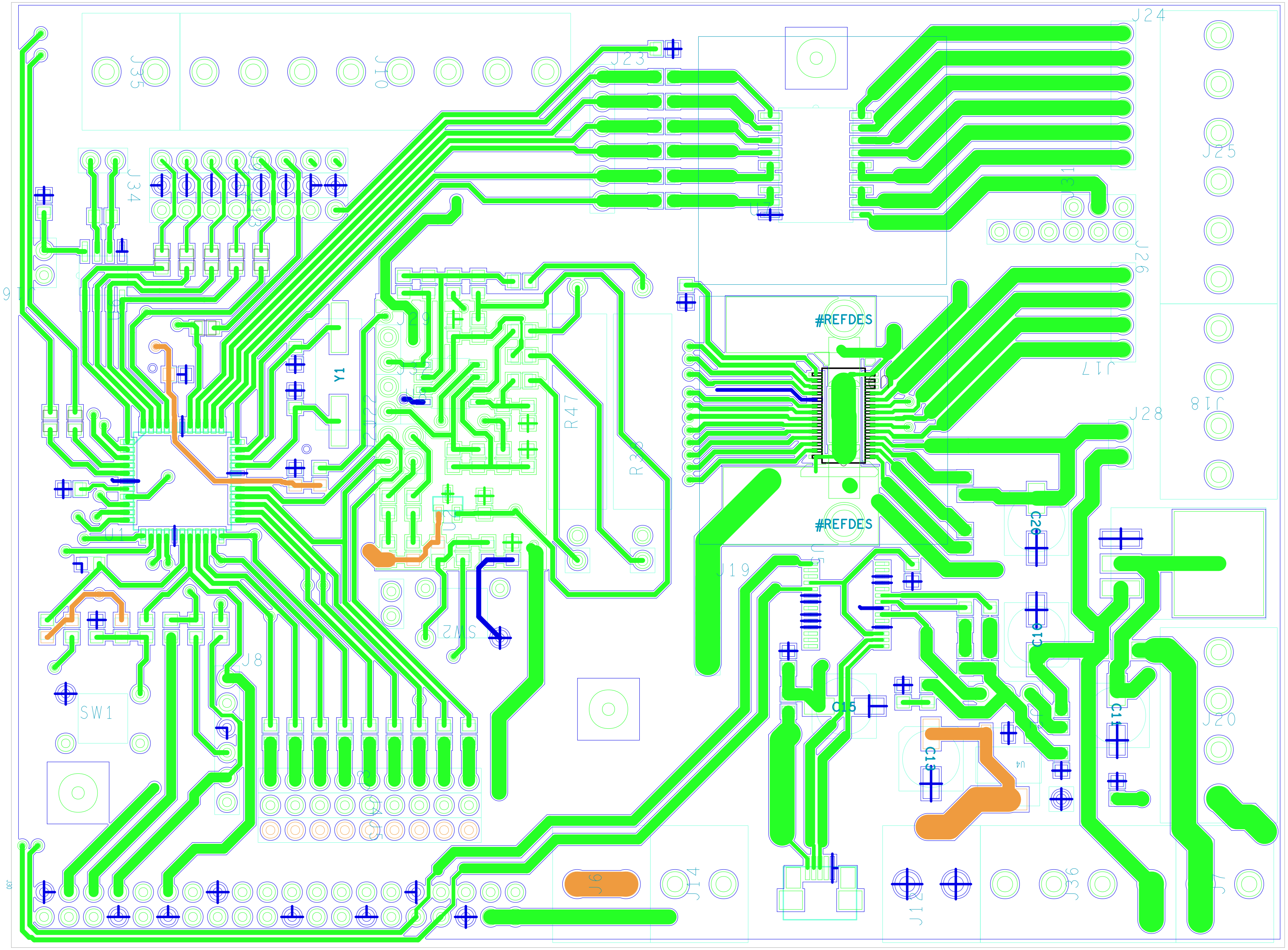
Appendix B

Hardware



U1
DSPIC33FJ128MC804
U2
ADR380ARTZ
J1
ADC_REF
R1 10K
R2 470
SW1 RESET
C6 10n
L1 10m
C9 100n
J8 PROGRAMMER
R3 100 R4 100 R5 100 R6 470
EXT_Vcc
R7 0
U3
LM1085IS-5.0
J7 PWR
D1 DIODE
J2
+5V0_USB
+5V0_REG
U4
LM3940
C10 10u
C11 10u
C12 470n
C13 47u
L2 CHOKE
L3 CHOKE
TP1 DGND
Y1 CRYSTAL
C7 18p
C8 18p
C3 100n
C4 100n
C5 100p
J3 ANALOG
J4 AGND
J5 +3V3
J6 +3V3_Sc
uUSB
L4 FERRITE
C14 100n
C15 4.7u
U5
FT232R
C18 100n
D2 Tx
D3 Rx
D4 LED1
D5 LED2
D6 PwrLED
R20 270 R21 270 R22 270 R23 270 R24 470
R25 50K
R31 470
SW2 BUTTON
C16 1n
J21 CON2
J22 CON2
J16 MAX485_ON
U6
MAX485CSA+
C17 100n
R32 120
J9 RP
J10 RP_Sc
J11 DGND
J12 DGND_Sc
J13 +5V0
J14 +5V0_Sc
J17 OUT
J18 OUT_Sc
J19 PGND
J20 PGND_Sc
J28 DRV8844_ON
C20 10u
C22 100n
C23 100n
C25 470n
R49 1K
U9
DRV8844
TP2 SENSE1
TP3 SENSE2
R33 10K
R34 10K
R38 0.1
R41 10K
R43 10K
C19 470n
C21 470n
U8A LM358
R44 220K
C24 2.2n
J29 ADC2
R45 10K
R46 10K
R47 0.1
R48 10K
R50 10K
C26 470n
C27 470n
U8B LM358
R51 220K
C28 2.2n
J32 ADC3
J30 CON40A
J23 IO_ULN
U7
ULN2803ADW
J24 OPEN
J25 OPEN_Sc
J26 EXT
J27 EXT_Sc
J31 ULN2803_ON
Samuel.Balula@tecnico.ulisboa.pt
Title
Control Board
Size A3
Document Number
<Doc>
Rev 0V3
Date: Thursday, May 26, 2016
Sheet 1 of 1

B.3



B.4