May 12, 2020

FYP PART a Report

Control of rolling-balancing mechanical system – disk on disk

Timothy Dunn

c3234207

The University of Newcastle

# Abstract

Table of Contents

[1 Introduction 4](#_Toc41309656)

[2 Literature Review? 6](#_Toc41309657)

[3 Modelling and Simulation 7](#_Toc41309658)

[3.1 Euler-Lagrange Method 7](#_Toc41309659)

[3.2 The Disk-on-Disk System 8](#_Toc41309660)

[3.3 Simulation 10](#_Toc41309661)

[4 Control 11](#_Toc41309662)

[4.1 Controller design 11](#_Toc41309663)

[4.2 State estimation 13](#_Toc41309664)

[5 Experiment Apparatus Design 15](#_Toc41309665)

[6 Manufacture 15](#_Toc41309666)

[7 Appendices 16](#_Toc41309667)

[8 References 17](#_Toc41309668)

# Introduction

The stabilisation of an inverted pendulum robot represents an ideal opportunity for a student to implement advanced design, modelling, and control techniques. In conjunction with appropriate electrical and mechanical designs, the derivation of a mathematical model for plant dynamics, alongside the design of a suitable controller and observer, is an invaluable learning experience.

The type II inverted pendulum robot (hereafter ‘inverted pendulum robot’) refers to the configuration of a pendulum robot in which the centre of mass of the system is above the wheel axis of rotation, such as a Segway. A type I robot refers to a cart system with a pendulum mounted to the top. This report and project exclusively refer to a type II configuration. A system of this configuration is inherently unstable, and control requires robust implementation of advanced control methods onboard an embedded system at high control-loop rates.

In order to demonstrate a balancing inverted pendulum robot, the following steps must be undertaken:

* Derive a mathematical model of the plant dynamics
* Determine a suitable design for the inverted pendulum robot
* Determine suitable hardware, such as actuators and sensors, to provide measurements
* Determine optimal configuration of hardware and electronics to achieve desired dynamics
* Commission a printed circuit board with suitable power electronics to act as a motherboard
* Develop a simulation model of the plant in Simulink
* Derive and develop a control model and controller
* Derive and develop an observer model and state estimator
* Conduct experiments to determine plant parameters
* Simulate behaviour of system
* Implement state estimator and controller in embedded C code
* Develop remote interface for communication
* Test and tune system

In order to demonstrate the following capabilities:

* Functioning sensors and actuators
* Remote interface for commanding actuators, reading sensors, and setting reference velocity
* Stabilisation of chassis angle
* Velocity regulation
* Chassis stabilisation on an inclined slope
* Chassis stabilisation during transition between ground plane and inclined plane

Information and resources necessary to complete this task are presented in the MCHA3500 course notes and laboratory exercises. This material is the foundation of the work undertaken and provides the major building blocks of the entire system. Several components of the embedded C implementation and simulation tools are provided as part of the laboratory documents, with adaptation required to suit the type II robot.

Must have note about hand and object here.

# Literature Review?

# Modelling and Simulation

Simulation and modelling are the foundation upon which a mechatronic system is built. A model of a system is a mathematical description of the output behaviour given an input. By taking this model and applying known inputs, the behaviour of the system can be recorded, and a simulation has occurred. If a particular output or system behaviour is desired, then the system must be influenced in such a way that the desired behaviour is realised. Influencing a system in this way is called control. Control without a working model and simulation is impossible.

The modelling of a system is often the first step taken in this process. The physical arrangement of the DoD system is already defined, and the physical phenomena at work are well known. The kinematics of the disk on disk system is adequately captured through use of the Euler-Lagrange modelling method.

## Euler-Lagrange Method

The Euler-Lagrange method is an energy-based method of modelling. The Euler-Lagrange method is often used for rigid body dynamic systems as it can be immediately apparent which components of the system are energy storing elements, and what their associated degrees of freedom are. In the case of the DoD system there are two main energy storing elements: the hand and the object. The driveshaft of the hand may also be considered to be energy storing.

Once the energy storing elements of the system are identified, the kinetic co-energy, , and potential energy, , can be determined. The kinetic co-energy and potential energy must be determined for each energy storing element in each associated degree of freedom. By taking the difference of the kinetic co-energy and the potential energy, the Lagrangian of the system is found [1].

|  |  |
| --- | --- |
|  |  |

where is column vector of position coordinates. The kinetic co-energy can be factored into quadratic form if the component constitutive relationships are linear in velocity[[1]](#footnote-1). The component constitutive relationships describe how a component of the system relates certain magnitudes [2].

|  |  |
| --- | --- |
|  |  |

where is the symmetric positive definite mass matrix.

If the effects of damping are to be considered, the impact of the generalised resistors of the system can be accounted for using the Rayleigh dissipation function . This function is also factored into quadratic form. The dissipation function must satisfy the inequality for all .

|  |  |
| --- | --- |
|  |  |

where is the symmetric positive semi-definite damping matrix.

If the system in question has non-conservative input forces, these can be included in the Euler-Lagrange equation alongside the dissipation forces.

|  |  |
| --- | --- |
|  |  |

By substituting (1), (2) and (3) into (4), the following is found,

|  |  |
| --- | --- |
|  |  |
|  |  |

is grouped to form the centripetal-Coriolis matrix and the gradient of the potential energy function is represented by , giving

|  |  |
| --- | --- |
|  |  |

A model of this form for the DoD system is developed in 4.2.

## The Disk-on-Disk System

A simplified model of the DoD system is considered for the modelling process. This simplification process is based upon several assumptions:

1. Torque transfer between the actuator and the hand is not subject to any dissipation effects
2. The object and hand interact without slipping
3. The hand and object are always in contact
4. The system is always operating near the upright balancing point

The model developed will be unable to perfectly describe the system dynamics if one or more of these assumptions is not met. Figure 1 shows the layout of the DoD system.

Figure . Layout of ideal DoD system.

The energy storing elements and their associated degrees of freedom can be identified via inspection. The hand has a single degree of freedom: the angle of rotation about point P. The object has two associated degrees of freedom: the angle of rotation about its centre and the angle of rotation about the point P. Assumption (A2) allows to be described as a function of and , reducing the system to two degrees of freedom.

There are two scenarios to consider when developing the kinematic model: the case where and the case where .

In the first case, if then

|  |  |
| --- | --- |
|  |  |
|  |  |

Similarly, when

|  |  |
| --- | --- |
|  |  |

This then gives the relationship:

|  |  |
| --- | --- |
|  |  |

The kinetic co-energy of the object can then be described as

|  |  |
| --- | --- |
|  |  |
|  |  |

The conversion between translational and rotational coordinates is most easily accomplished using the single-axis rotation matrices. Alternatively, the parallel axis theorem may be used to find the rotational inertia of the object about the point P and solve directly.

The kinetic co-energy of the hand is

|  |  |
| --- | --- |
|  |  |

The total kinetic co-energy can then be factorised to find the mass matrix with the generalised coordinates .

|  |  |
| --- | --- |
|  |  |
|  |  |

The potential energy is given by

|  |  |
| --- | --- |
|  |  |
|  |  |

There is only one input force, the torque applied to the hand, giving

|  |  |
| --- | --- |
|  |  |

As the mass matrix is not time varying and the kinetic co-energy is not a function of the centripetal-Coriolis matrix is zero. This leaves the final Euler-Lagrange equation as

|  |  |
| --- | --- |
|  |  |

## Simulation

Simulink is a graphical block diagram simulation program. It enables the user to build a model using distinct block components in order to simulate the behaviour of a dynamic system. Simulink is often useful when a system is a combination of several distinct subsystems as it allows the user to add or remove subsystems easily, with interactions being visually displayed.

The DoD system can be built and simulated in Simulink using integration blocks. By rearranging the derived Euler-Lagrange equation so that the second state derivative is the subject of the equation, the system can be set up with sequential integration blocks, allowing the position states and the velocity derivatives to be calculated.

|  |  |
| --- | --- |
|  |  |

These states and derivatives are then used to calculate the next state. The layout of this model is shown in Figure 2 below.

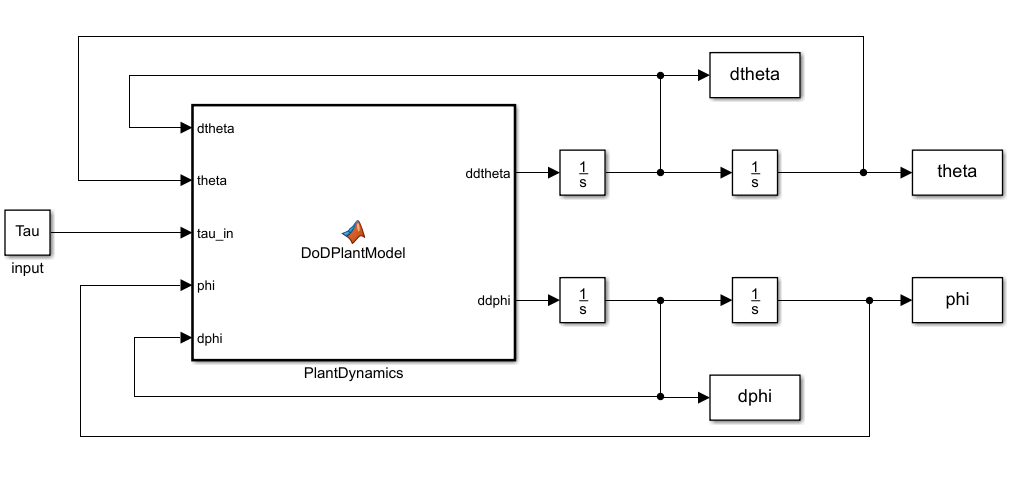


Figure . Layout of Simulink model.

The derived model can now be verified using this simulation. If the object is initially position at small angle offset from the upright balancing position, the expectation is that it would fall toward the ground. Due to A(3) and the absence of energy dissipation, the object is expected to swing with sinusoidal motion without decay. The hand is also expected to rotate due to A(2). Figure 3 below shows the results of running the simulation for these conditions, verifying that the model is a valid representation of the DoD system.



Figure . Plots of hand and object angle and angular velocity.

# Control

The ability to control the output behaviour of a system is often desired. When a system is controlled, the output of the system follows some specified reference input, regardless of the details of the dynamics in the system. Robust control methods allow reference input tracking in the presence of input disturbances, errors introduced to the model by assumptions, and errors introduced in measurements by the sensors [3].

Digital control methods are, by nature, discrete-time problems. Despite continual advancements in technology, calculations must ultimately be performed at specified time intervals. The models so far developed have been in continuous-time and must be discretised in order to develop the discrete-time control algorithms.

## Controller design

The control method used is the Linear-Quadratic Regulator (LQR). LQR requires that the dynamics of the system be wholly described by a set of linear differential equations. The core of the LQR method is minimising a quadratic cost function.

Consider the system

|  |  |
| --- | --- |
|  |  |
|  |  |

which is linear time-invariant and stabilisable. The desired feedback control law is of the form

|  |  |
| --- | --- |
|  |  |

which minimises the cost function

|  |  |
| --- | --- |
|  |  |

This cost function is the sum of the quadratic function in the states and the quadratic function in the inputs. The matrix **Q** influences the control response to the state deviations from zero. If the magnitude of **Q** is large, the control force will increase to minimise the deviation of the state from zero. Similarly, the matrix **R** influences the control effort. If the magnitude of **R** is large, the control effort will be minimised.

The solution to (25) is given when

|  |  |
| --- | --- |
|  |  |

**S** is the solution to the Discrete-time Algebraic Ricatti Equation

|  |  |
| --- | --- |
|  |  |

The closed-loop system is stable if and are stabilisable, and and are detectable [4].

Fortunately, MATLAB has an inbuilt function to calculate the discrete-time LQ control gain **K** given the continuous-time state space equations. The control system toolbox functions ‘lqrd’ and ‘c2d’ use the method outline above and the zero-order hold approximation, respectively.

A continuous-time state space representation of the system described in (20) is found via the following process. The generalised positions are chosen for the state vector . By choosing , (20) can be rearranged to

|  |  |
| --- | --- |
|  |  |

which is the continuous-time form of (22).

There is only one output to be regulated, the velocity of the hand , making the output regulation equations trivial.

|  |  |
| --- | --- |
|  |  |

To regulate this output to a desired setpoint, , discrete-time reference feedforward is applied. This involves the introduction of a non-zero steady state and input into the control law derived above.

|  |  |
| --- | --- |
|  |  |

As the desired output is . Therefore, the following must be satisfied

|  |  |
| --- | --- |
|  |  |
|  |  |

which can be rearranged into a matrix equation

|  |  |
| --- | --- |
|  |  |

The solution is linear in r, allowing to be set to and to be set to . The equation can then be solved to find

|  |  |
| --- | --- |
|  |  |

This gives the control law

|  |  |
| --- | --- |
|  |  |

The entire control law, consisting of the reference feedforward terms and the state feedback terms, can be implemented with the structure shown in Figure 4 [4].

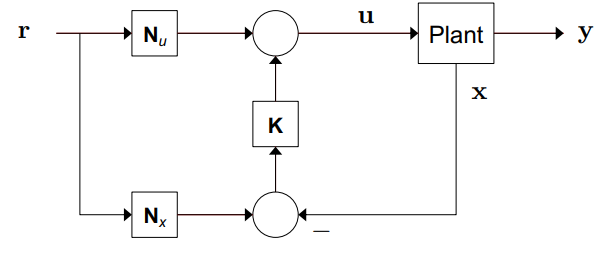


Figure . Linear-Quadratic with Reference Feedforward control structure [4].

The linear control method shown here is restricted in both application and performance. This control method is not intended to be the final control method implemented but serves as adequate foundation to determine physical system parameters for mechanical design. It also allows the state estimator to be implemented and tested with minimal complications.

SS model – then LQG (MPC?)

## State estimation

Kalman filter state estimator and computer vision

# Experiment Apparatus Design

# Manufacture

# Appendices

# References

[1] A. Fairclough, "Lab 2: Euler-Lagrange modelling," in *MCHA3500 Lab Notes*, ed: The University of Newcastle, 2019.

[2] T. Perez, *Engineering System Dynamics (Modelling, Analysis, and Simulation)*, T. Perez, ed.: The University of Newcastle, AUSTRALIA, 2013.

[3] G. F. Franklin, J. D. Powell, and M. L. Workman, *Digital control of dynamic systems*. Addison-wesley Menlo Park, CA, 1998.

[4] D. C. Renton, "MCHA3500 – Review of LQG," *Mechatronics Design 1 Course Notes*: The University of Newcastle, Australia, 2019.

1. Linearity in velocity occurs when the magnitudes are significantly below the speed of light. [↑](#footnote-ref-1)