JUNE 4, 2020

# FYP PART A REPORT

## CONTROL OF ROLLING-BALANCING MECHANICAL SYSTEM – DISK ON DISK

TIMOTHY DUNN
C3234207
The University of Newcastle

## Dot-Point Summary

- I solidified my knowledge of energy-based modelling methods
- I applied the Euler-Lagrange method to a new system to develop a mathematical model
- I applied knowledge of inertial reference frames to a new system
- I transformed an unknown system into a form suitable for a known method
- I performed a simulation of a mechanical system using a mathematical model
- I designed a controller for a mechatronic system
- I designed a state estimator suitable for use with mechatronic control
- I applied a method of feature recognition in images
- I applied a method of camera parameter estimation
- I extended the feature recognition method to real-time applications
- I designed an experiment apparatus to test the controller, state estimator and feature recognition process

# 1 Abstract

This project has the objective of researching, modelling, simulating, and implementing control techniques on a rolling-balancing mechanical system. By applying the Euler-Lagrange method to develop a mathematical model, which is then implemented in the Simulink toolbox and controlled using the Linear Quadratic Regulator, this objective is partially achieved. The last objective, yet to be completed, is the implementation of the control techniques using the designed experiment apparatus.

Underactuated mechanical systems are extremely common engineering scenarios with no general solution. Systems such as the disk-on-disk (DoD) present further complications beyond under-actuation. The system is suitable for Linear-Quadratic Regulator (LQR) control schemes to be used. Reference tracking can also be employed using reference feedforward structures.

The DoD system presents complications when measurements of system states is required, particularly the states associated with the unactuated disk. A contactless method of measuring position is employed, utilising colour saturation and blob analysis. The uncertainty associated with the measurements is accounted for with the Kalman Filter.

# Table of Contents

# 2 List of Figures

# 3 Introduction

The DoD system is an underactuated mechanical system of two disks. The top disk, called the object, is situated on top of the bottom disk, the hand. The top disk is free to roll but not translate out of plane with the bottom disk. The bottom disk is actuated.

The goal of the DoD system is to control the position of the top disk. In this case the desired position is the upright balancing position. There is no limitation currently in place for the position of the bottom disk.

Manipulation and control of a system where gripping of the object is possible is relatively straightforward as manipulators can directly influence the position or orientation of the object, called prehensile manipulation. Manipulation of the top disk in the DoD system is non-prehensile manipulation as no such gripping action can occur. Non-prehensile manipulations are common for humans, who will happily push or throw an object to change its position, as they are able to monitor the process visually and adjust movements accordingly. Replicating such movements on a mechatronic system is far more complicated. Complex state estimation must occur for the, usually, nonlinear system alongside high frequency control.

The DoD system presents a system that requires a specific type of non-prehensile manipulation: rolling. Rolling is generally more complex than a pushing or grasping motion due to the nonlinearity of the motion. The DoD system is further broken down to exclude movement in the axial direction, isolating the rolling movement as the only form. Control of this system presents unique insights into understanding non-prehensile motion in the general sense.

This report investigates one method of modelling, simulating, and controlling the disk-on-disk system shown in Figure 1. Energy based methods of modelling are employed to develop a mathematical model, which is then simulated using Simulink. A contactless method of measuring position is developed using image processing and feature recognition. This process can aid in the understanding of the general non-prehensile control problem.

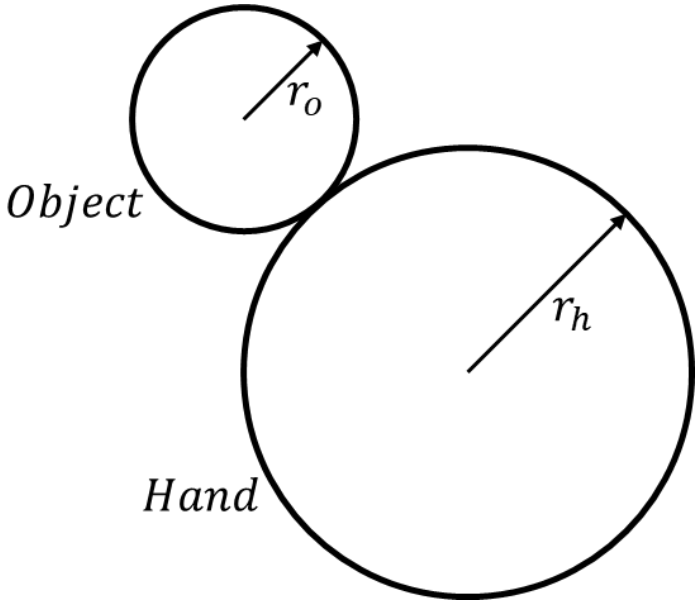


*Figure 1. The disk-on-disk system.*

# 4 Literature Review

The DoD system has been studied by several groups, alongside studies of related problems. [1] investigates the control of a DoD system in the same configuration, where control is achieved through passivity-based methods. Control of the object position and velocity is achieved, and the addition of injection damping allows control of the hand position and velocity. The system is modelled using port-Hamiltonian theory, which differs from the Euler-Lagrange methods applied in this report. This paper presents a much more sophisticated approach to both modelling and control. The computer vision used runs at more than double the frequency of this report, allowing higher quality measurements to be taken. Additionally, the motor used is a continuous rotation servomotor, which reduces the complexity of applying the control law by removing the need for control allocation. The DC motors used in this report do not present such finite control over position.

In [2] the same DoD system is modelled and controlled, both in simulation and experiments. The modelling process expresses the model using the Euler-Lagrange equations, but characterises the movements in terms of arc lengths. Control is achieved with state feedback. Again, the measurement frequency of 800Hz far exceeds the 30Hz used in this report.

These papers show that control of the DoD system is achievable through a range of other methods than the one presented in this report.

# 5 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 DoD system is adequately captured through use of the Euler-Lagrange modelling method.

## 5.1 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, $\mathcal{T}^*$, and potential energy, $\mathcal{V}$, 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 [3].

$$\mathcal{L} = \mathcal{T}^*(\mathbf{f}, \mathbf{q}) - \mathcal{V}(\mathbf{q}) \tag{1}$$

where $\mathbf{q}$ is column vector of position coordinates. The kinetic co-energy can be factored into quadratic form, (2), if the component constitutive relationships are linear in velocity[1]. The component constitutive relationships describe how a component of the system relates certain magnitudes [4].

[1] Linearity in velocity occurs when the magnitudes are significantly below the speed of light.

$$\mathcal{T}^*(\mathbf{q},\dot{\mathbf{q}}) = \frac{1}{2}\dot{\mathbf{q}}^{\mathrm{T}}\mathbf{M}(\mathbf{q})\dot{\mathbf{q}} \tag{2}$$

Where $\mathbf{M(q)}$ 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 $\mathcal{D}$. This function is also factored into quadratic form. The dissipation function must satisfy the inequality $\frac{\partial \mathcal{D}}{\partial \dot{\mathbf{q}}}\dot{\mathbf{q}} \geq 0$ for all $\dot{\mathbf{q}}$.

$$\mathcal{D}(\dot{\mathbf{q}}) = \frac{1}{2}\dot{\mathbf{q}}^{T}\mathbf{D}\dot{\mathbf{q}} \tag{3}$$

Where $\mathbf{D}$ 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.

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}}\right) - \frac{\partial \mathcal{L}}{\partial \mathbf{q}} = \boldsymbol{\tau} - \frac{\partial \mathcal{D}}{\partial \dot{\mathbf{q}}} \tag{4}$$

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

$$\frac{\mathrm{d}}{\mathrm{d}t}(\mathbf{M}(\mathbf{q})\dot{\mathbf{q}}) - \frac{\partial \mathcal{T}^*}{\partial \mathbf{q}} + \frac{\partial \mathcal{V}}{\partial \mathbf{q}} = \boldsymbol{\tau} - \mathbf{D}\dot{\mathbf{q}} \tag{5}$$

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \dot{\mathbf{M}}(\mathbf{q})\dot{\mathbf{q}} - \frac{\partial \mathcal{T}^*}{\partial \mathbf{q}} + \frac{\partial \mathcal{V}}{\partial \mathbf{q}} + \mathbf{D}\dot{\mathbf{q}} = \boldsymbol{\tau} \tag{6}$$

$\dot{\mathbf{M}}(\mathbf{q})\dot{\mathbf{q}} - \frac{\partial \mathcal{T}^*}{\partial \mathbf{q}}$ is grouped to form the centripetal-Coriolis matrix and the gradient of the potential energy function is represented by $\mathbf{g(q)}$, giving

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{D}\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} \tag{7}$$

A model of this form for the DoD system is developed in section 5.2.

## 5.2 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:

(A1) Torque transfer between the actuator and the hand is not subject to any dissipation effects

(A2) The object and hand interact without slipping

(A3) The hand and object are always in contact

(A4) 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 2 shows the layout of the DoD system.

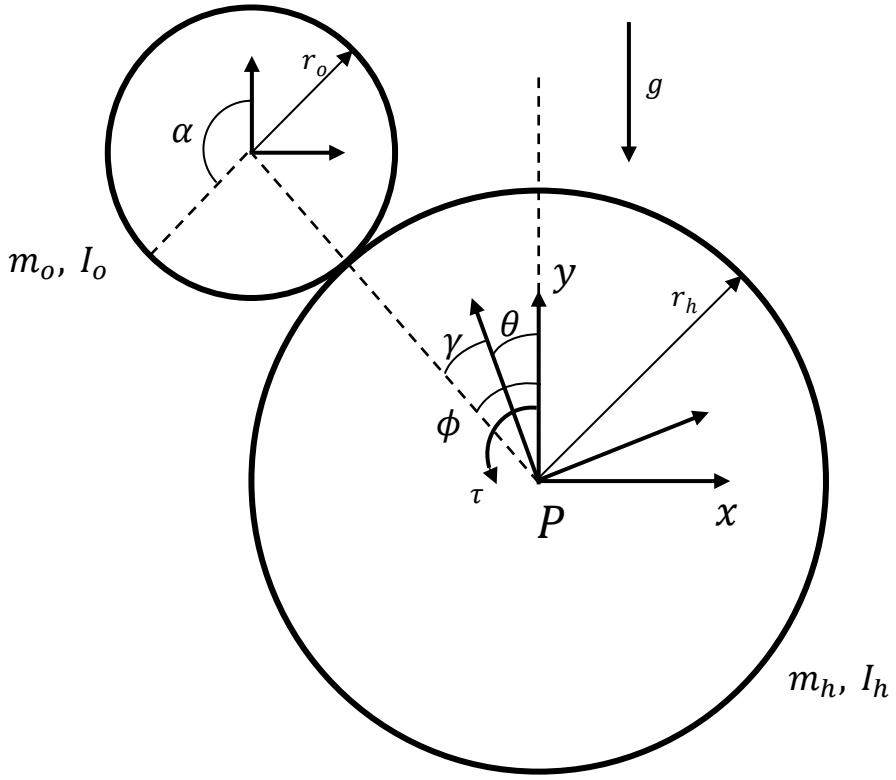


*Figure 2. Layout of ideal DoD system.*

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 $\theta$ about point P. The object has two associated degrees of freedom: the angle of rotation $\alpha$ about its centre and the angle of rotation $\phi$ about the point P. Assumption (A2) allows $\alpha$ to be described as a function of $\theta$ and $\phi$, reducing the system to two degrees of freedom.

There are two scenarios to consider when developing the kinematic model: the case where $\theta = 0$ and the case where $\phi = 0$.

In the first case, if $\theta = 0$ then

$$\alpha = \frac{r_h}{r_o}\gamma \text{ where } \phi = \gamma - \theta \tag{8}$$

$$\therefore \alpha = \frac{r_h}{r_o}\phi \tag{9}$$

Similarly, when $\phi = 0$

$$\alpha = -\frac{r_h}{r_o}\theta \tag{10}$$

This then gives the relationship:

$$\alpha = \frac{r_h}{r_o}(\theta - \phi) \text{ and } \dot{\alpha} = \frac{r_h}{r_o}(\dot{\theta} - \dot{\phi}) \tag{11}$$

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

$$\mathcal{T}_o^* = \frac{1}{2}m_o\dot{x}^2 + \frac{1}{2}m_o\dot{y}^2 + \frac{1}{2}J_o\dot{\alpha}^2 \tag{12}$$

$$\mathcal{T}_o^*(\dot{\theta}, \dot{\phi}) = \frac{1}{2}m_o\dot{\phi}^2(r_h + r_o)^2 + \frac{1}{2}J_o\left(\frac{r_h^2}{r_o^2}\dot{\phi}^2 - 2\frac{r_h^2}{r_o^2}\dot{\phi}\dot{\theta} + \frac{r_h^2}{r_o^2}\dot{\theta}^2\right) \tag{13}$$

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

$$\mathcal{T}_h^*(\dot{\theta}) = \frac{1}{2}J_h\dot{\theta}^2 \tag{14}$$

The total kinetic co-energy can then be factorised to find the mass matrix with the generalised coordinates $\mathbf{q} = \begin{bmatrix}\theta \\ \phi\end{bmatrix}$.

$$\mathcal{T}^*(\dot{\theta}, \dot{\phi}) = \frac{1}{2}\begin{bmatrix}\dot{\theta} \\ \dot{\phi}\end{bmatrix}\begin{bmatrix} J_h + J_o\frac{r_h^2}{r_o^2} & -J_o\frac{r_h^2}{r_o^2} \\ -J_o\frac{r_h^2}{r_o^2} & m_o(r_h + r_o)^2 + J_o\frac{r_h^2}{r_o^2}\end{bmatrix}[\dot{\theta} \quad \dot{\phi}] \tag{15}$$

$$\therefore \mathbf{M} = \begin{bmatrix} J_h + J_o\frac{r_h^2}{r_o^2} & -J_o\frac{r_h^2}{r_o^2} \\ -J_o\frac{r_h^2}{r_o^2} & m_o(r_h + r_o)^2 + J_o\frac{r_h^2}{r_o^2}\end{bmatrix} \tag{16}$$

The potential energy is given by

$$\mathcal{V}(\phi) = m_o g(r_h + r_o)\cos(\phi) \tag{17}$$

$$\therefore \ \mathbf{g}(\phi) = \begin{bmatrix} 0 \\ -m_o g(r_h + r_o)\sin(\phi) \end{bmatrix} \tag{18}$$

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

$$\boldsymbol{\tau} = \begin{bmatrix} \tau \\ 0 \end{bmatrix} \tag{19}$$

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

$$\begin{bmatrix} J_h + J_o\frac{r_h^2}{r_o^2} & -J_o\frac{r_h^2}{r_o^2} \\ -J_o\frac{r_h^2}{r_o^2} & m_o(r_h + r_o)^2 + J_o\frac{r_h^2}{r_o^2} \end{bmatrix}\begin{bmatrix} \ddot{\theta} \\ \ddot{\phi} \end{bmatrix} + \begin{bmatrix} 0 \\ -m_o g(r_h + r_o)\sin(\phi) \end{bmatrix} = \begin{bmatrix} \tau \\ 0 \end{bmatrix} \tag{20}$$

## 5.3 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.

$$\ddot{\mathbf{q}} = \mathbf{M}^{-1}(\boldsymbol{\tau} - \mathbf{g}(\mathbf{q})) \tag{21}$$

These states and derivatives are then used to calculate the next state. The layout of this model is shown in Figure 3 below.

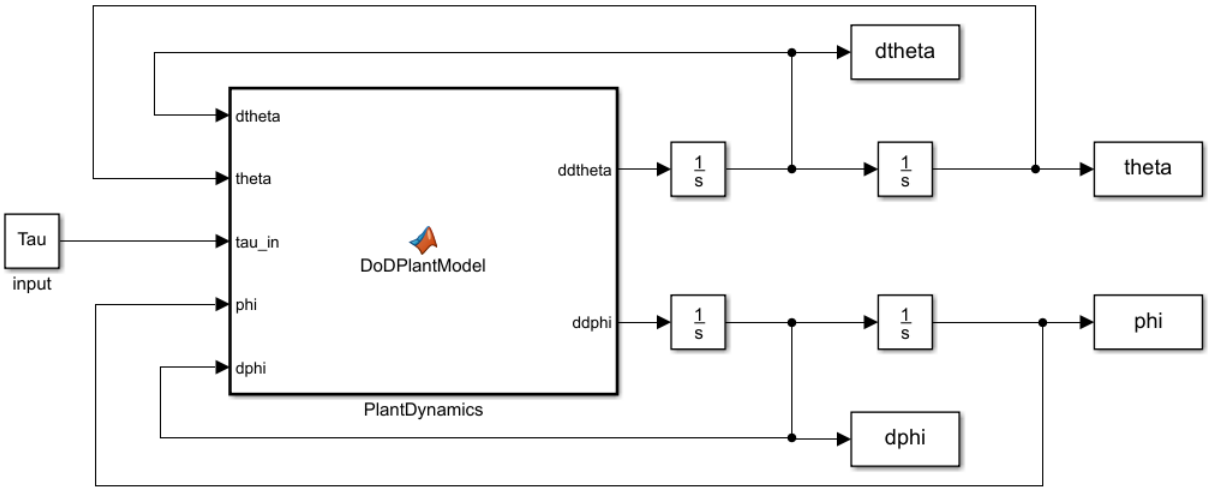


*Figure 3. Layout of Simulink model.*

The derived model can now be verified using this simulation. If the object is initially positioned 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 4 below shows the results of running the simulation for these conditions, verifying that the model is a valid representation of the DoD system.

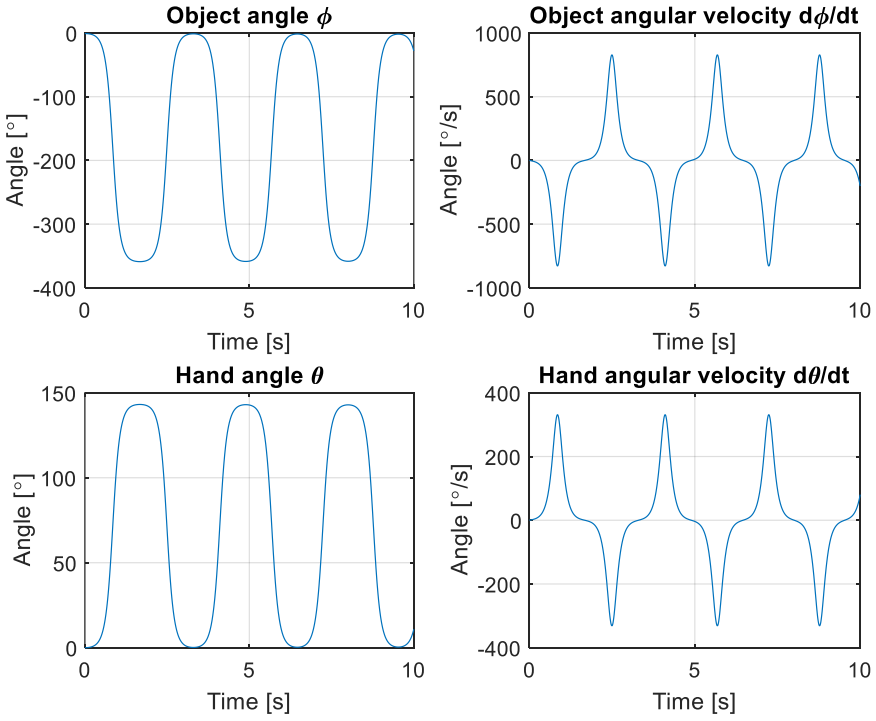


*Figure 4. Plots of hand and object angle and angular velocity.*

# 6 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 [5].

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.

## 6.1 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

$$\mathbf{x}_{k+1} = \mathbf{A}_d\mathbf{x}_k + \mathbf{B}_d\mathbf{u}_k \tag{22}$$

$$\mathbf{y}_k = \mathbf{C}x_k + \mathbf{D}u_k \tag{23}$$

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

$$\mathbf{u}_k = -\mathbf{K}\mathbf{x}_k \tag{24}$$

which minimises the cost function

$$\mathcal{V} = \sum_{k=0}^{\infty} \mathbf{x}_k^{\mathrm{T}}\mathbf{Q}_d\mathbf{x}_k + \mathbf{u}_k^{\mathrm{T}}\mathbf{R}_d\mathbf{u}_k \text{ where } \mathbf{Q}_d \geq 0, \ \mathbf{R}_d > 0 \tag{25}$$

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

$$\mathbf{K} = \left(\mathbf{R}_d + \mathbf{B}_d^{\mathrm{T}}\mathbf{S}\mathbf{B}_d\right)^{-1}\mathbf{B}_d^{\mathrm{T}}\mathbf{S}\mathbf{A}_d \tag{26}$$

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

$$\mathbf{0} = \mathbf{A}_d^{\mathrm{T}}\left(\mathbf{S} - \mathbf{S}\mathbf{B}_d\left(\mathbf{R}_d + \mathbf{B}_d^{\mathrm{T}}\mathbf{S}\mathbf{B}_d\right)^{-1}\mathbf{B}_d^{\mathrm{T}}\mathbf{S}\right)\mathbf{A}_d + \mathbf{Q}_d - \mathbf{S} \tag{27}$$

The closed-loop system is stable if $\mathbf{A}_d$ and $\mathbf{B}_d$ are stabilisable, and $\mathbf{A}_d$ and $\mathbf{Q}_d^{1/2}$ are detectable [6].

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 $\mathbf{q} = \begin{bmatrix}\theta\\ \phi\end{bmatrix}$. By choosing $\mathbf{v} \triangleq \dot{\mathbf{q}}$, (20) can be rearranged to

$$\begin{bmatrix}\dot{\mathbf{v}}\\ \dot{\mathbf{q}}\end{bmatrix} = \begin{bmatrix}-\mathbf{M}^{-1}\mathbf{D} & -\mathbf{M}^{-1}\mathbf{K}\\ \mathbf{I} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{v}\\ \mathbf{q}\end{bmatrix} + \begin{bmatrix}\mathbf{M}^{-1}\mathbf{e}\\ \mathbf{0}\end{bmatrix}\boldsymbol{\tau} \text{ where } \mathbf{e} = \begin{bmatrix}1\\ 0\end{bmatrix} \tag{28}$$

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

There is only one output to be regulated, the velocity of the hand $\dot{\theta}$, making the output regulation equations trivial.

$$y_r = \dot{\theta} = [1 \quad 0 \quad 0 \quad 0]\begin{bmatrix}\mathbf{v}\\ \mathbf{q}\end{bmatrix} \tag{29}$$

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

$$\mathbf{u}_k - \mathbf{u}_{ss} = -\mathbf{K}(\mathbf{x}_k - \mathbf{x}_{ss}) \tag{30}$$

As $k \to \infty$, then $\mathbf{x}_k \to \mathbf{x}_{ss}$ and $\mathbf{u}_k \to \mathbf{u}_{ss}$, the desired output is $\mathbf{r}$. Therefore, the following must be satisfied

$$\mathbf{x}_{ss} = \mathbf{A}_d\mathbf{x}_{ss} + \mathbf{B}_d\mathbf{u}_{ss} \tag{31}$$

$$\mathbf{r} = \mathbf{C}\mathbf{x}_{ss} + \mathbf{D}\mathbf{u}_{ss} \tag{32}$$

which can be rearranged into a matrix equation

$$\begin{bmatrix} \mathbf{A}_d - \mathbf{I} & \mathbf{B}_d \\ \mathbf{C} & \mathbf{D} \end{bmatrix} \begin{bmatrix} \mathbf{x}_{ss} \\ \mathbf{u}_{ss} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix} \mathbf{r} \tag{33}$$

The solution is linear in r, allowing $\mathbf{x}_{ss}$ to be set to $\mathbf{N}_x\mathbf{r}$ and $\mathbf{u}_{ss}$ to be set to $\mathbf{N}_u\mathbf{r}$. The equation can then be solved to find

$$\begin{bmatrix} \mathbf{N}_x \\ \mathbf{N}_u \end{bmatrix} = \begin{bmatrix} \mathbf{A}_d - \mathbf{I} & \mathbf{B}_d \\ \mathbf{C} & \mathbf{D} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix} \tag{34}$$

This gives the control law $\mathbf{u}_k$

$$\mathbf{u}_k = (\mathbf{N}_u + \mathbf{K}\mathbf{N}_x)\mathbf{r} - \mathbf{K}\mathbf{x}_k \tag{35}$$

The entire control law, consisting of the reference feedforward terms and the state feedback terms, can be implemented with the structure shown in Figure 5 [6].

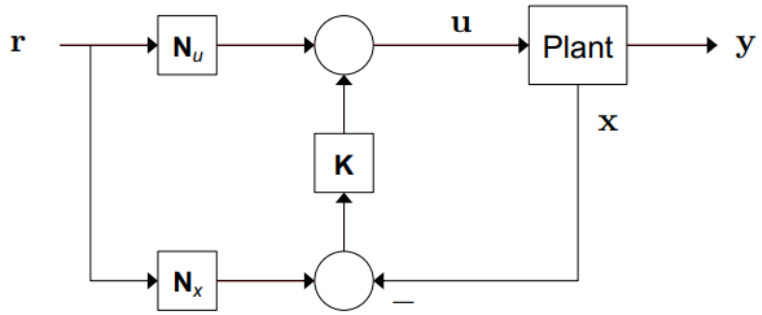


*Figure 5. Linear-Quadratic with Reference Feedforward control structure [6].*

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.

## 6.2 State estimation

As the name implies, state feedback control requires knowledge of the states. Measurement of the states is often unavailable or subject to a level of noise. In the case of the DoD system, angular position

measurements of the hand can be directly taken via an encoder on the motor drive shaft. An approximation of the hand velocity can be calculated using the difference in the position measurements between timesteps.

Critically, the position and velocity of the object is not easily measured. It is possible to attach an inertial measurement unit to the object, which would allow calculation of the angle and angular velocity. These measurements would need to be either communicated, via wired or wireless methods, which presents significant problems. A more elegant solution is the use of contactless measurement.

### 6.2.1 Computer Vision

Computer vision is the process of extracting information from still images or video, particularly the identification of features or objects. Using image processing techniques, a stream of images captured by a low-end webcam can be used to make relatively accurate measurements of the position of the object in the DoD system. The image capturing device often introduces distortion to the image which requires a correction process to negate the error in measurements.

The errors in measurements are caused by the camera lens distorting the light as it passes through, and the projection of the three-dimensional environment onto a two-dimensional plane. The nature of the DoD system reduces the impact of these errors. As the location of the camera relative to the plane containing the front of the two disks is constant, the distortion due to conversion between three-dimensional and two-dimensional space does not need to be recalculated each time a measurement is taken.

Calibration of the camera is a straightforward process with the aid of the MATLAB computer vision toolbox. A 'checkerboard' pattern shown in Figure 6 is photographed in several different positions in the three-dimensional space with the camera fixed in place.

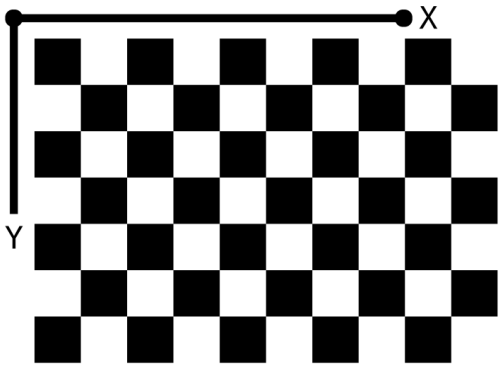


*Figure 6. Checkerboard pattern used for camera calibration.*

This pattern is used as it provides many well-defined cross-corners[2] that are easily identified by the corner detection algorithm. The detected corners in image coordinates are then used, in conjunction with the measured edge length of the squares, to solve for the intrinsic and extrinsic camera parameters in closed form using the method outlined in [7]. Initially the lens distortion is assumed to be zero. This closed form solution is then used to estimate all of the camera parameters simultaneously using the nonlinear least-squares minimisation method outlined in [8]. The closed form solution is used as the initial estimate for the camera parameters and the initial estimate of the distortion is zero [9].

Using these estimated camera parameters, an input image can be undistorted. A summary of the reprojection errors is also calculated during this process. In the absence of experiment apparatus to test the camera calibration process, a series of seven test images are used. Figure 7 (left) shows an original distorted image captured with the camera. The image includes both the checkerboard pattern and two regions of interest. Figure 7 (right) shows the undistorted image after the camera calibration process. The comparison shows the effects of radial lens distortion on the original image. If this distortion is not accounted for is it impossible to obtain accurate measurements of the locations of points of interest.


*Figure 7. The original distorted image (left) and the undistorted image after estimation of camera parameters (right).*

Figure 8 shows a plot magnitude of the reprojection errors. A reprojection error is a measure, in pixels, of how precise the estimation of the point's location is in the two-dimensional projection compared to its position in three-dimensional space.

[2] A cross-corner is the intersection of two white and two black squares.

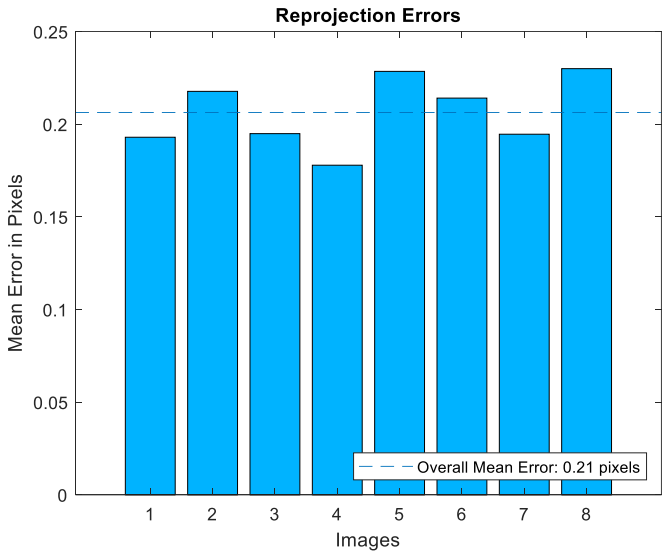


*Figure 8. Reprojection error for each image in the calibration set.*

The reprojection errors for each image are less than a pixel, which is generally considered suitable.

Using the calibrated camera, accurate feature recognition is possible. The DoD system can be configured to reduce the complexity of this process by specifying that points of interest have a consistent colour, and all other regions are consistently transparent or coloured differently. This specification allows for saturation thresholding to be used.

Images captured by the camera are initially in RGB (red, green, and blue) format. Each pixel in the image has three numbers associated with it which describe the red, green, and blue colour intensity, respectively. These values can be converted into HSV (hue, saturation, and value). Hue describes the wavelength of the colour of the pixel and value describes how light or dark the colour of a pixel is. Saturation is a measure of the intensity of the colour of the pixel. In an image that has clearly defined regions of colour, such as the image in Figure 7 and Figure 8, would ideally have three dominating levels of saturation: one for the dark areas, a second for the light areas, and a third for the blue circles. In practice, low quality or poorly lit images have far less consistent saturation values. The computer vision toolbox in MATLAB has a function for converting RGB images to HSV, named 'rgb2hsv', which uses the method derived in [10]. Figure 9 shows a HSV representation of the image in Figure 7. In this representation of the image the two blue circles can be clearly identified as having significantly different saturation levels to their surroundings. There is some overlap between the saturation levels of the black squares in the checkerboard pattern, most likely due to low camera quality and poor lighting.

The image can now be converted into grayscale using image thresholding. 'graythresh' is another MATLAB function which computes the normalised threshold level for an image, allowing it to be converted to binary. By extracting the saturation channel of the HSV image and thresholding with respect to this channel, the image can be converted into a binary representation based on saturation values, shown in Figure 10. The blue circles are now segmented from their surroundings. Figure 10 shows the image in its inverted state to reduce the amount of black ink used in printing this report.

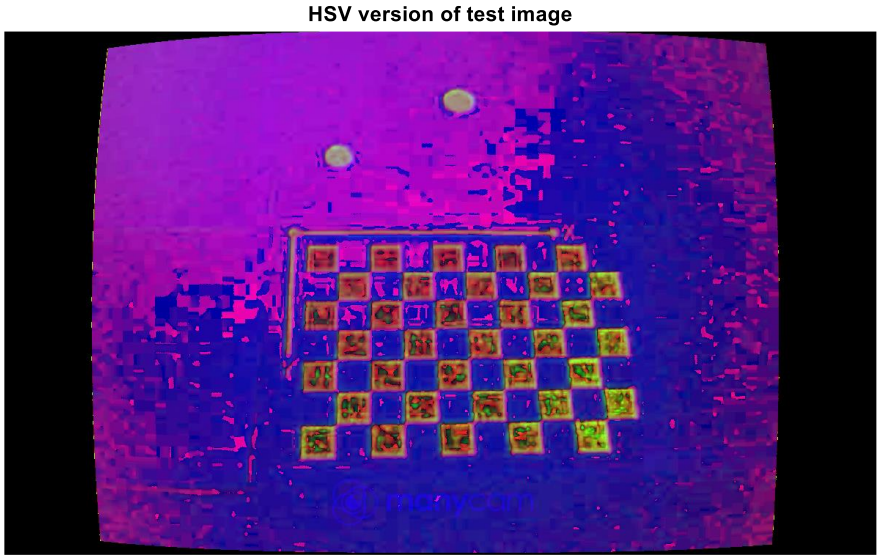


*Figure 9. HSV version of test image.*

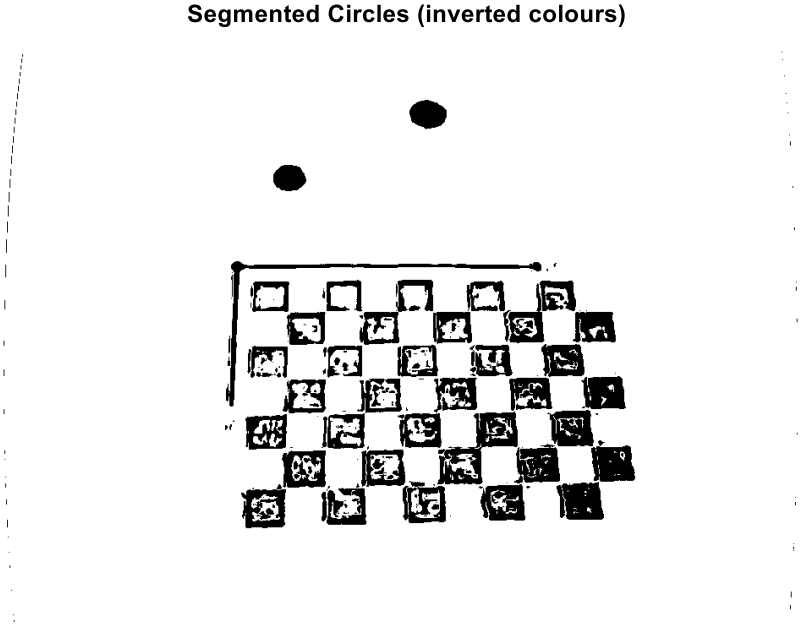


*Figure 10. Segmented circles (inverted colours).*

After the thresholding process each pixel has a value of either zero or one, representing whether the saturation of that pixel was below or above the mean saturation level of the HSV image. The blue circles are the only continuous region of consistent value, allowing the use of blob detection algorithms. The inbuilt blob detection algorithms in the MATLAB computer vision toolbox are able to compute statistics for regions of connected pixels in binary images, such as the one shown in Figure 10. The detection algorithm calculates the centroid of the detected blob and the bounding box. By setting a minimum and maximum size of the blob to be near the known size of the circles, and only searching in the region above the checkerboard pattern, the blob detection result is very accurate. The circles are detected by the algorithm and the bounding region reflects the shape of the circle. Figure 11 shows the result of the feature detection using the process outlined.

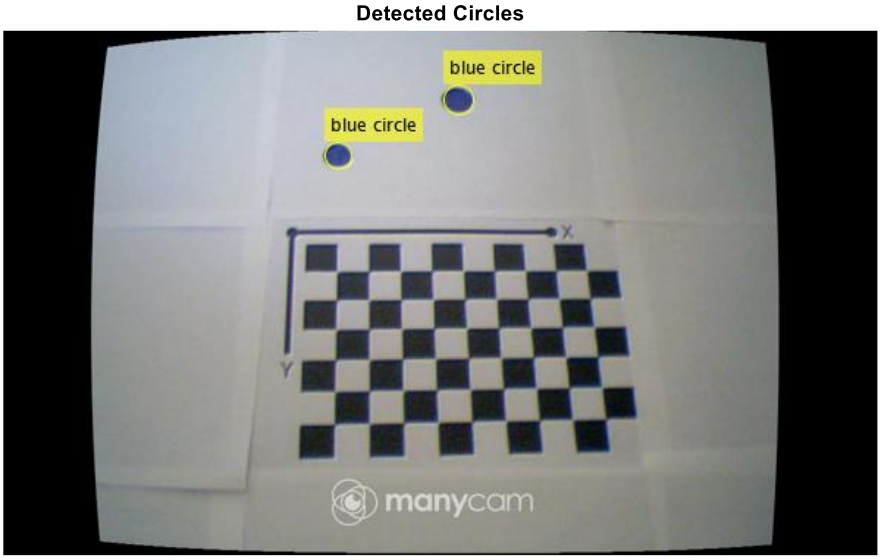


*Figure 11. Result of feature recognition using saturation thresholding.*

Using the statistics returned from the blob detection the size of the circles can be calculated. For the test image the measurement of diameter was accurate to within 0.5mm. The distance between the centroids can also be calculated using basic geometric relations, alongside the angle between the two circles.

In the final experiment apparatus, the camera will not move relative to the two disks during an experiment. Additionally, the object will only move in two dimensions at a fixed distance from the camera. Due to the combination of these factors the camera calibration parameters need only be computed once. The process of segmenting the points of interest from the surrounding environment is carried out only using matrix and binary operations, which are both efficient operations in the MATLAB environment. This enables the feature recognition process to be carried out for each frame

of a video stream in real time. The limiting factor on measurement frequency is the frame rate of the camera, which is 30 frames per second.

The measurements taken from the computer vision algorithm are subject to uncertainty from several areas. The large pixel size in comparison to object size introduces errors, as well as the estimation process of camera parameters. Additionally, if either of the points of interest are not correctly identified, erroneous measurement values can be generated which greatly impacts the performance of the controller.

### 6.2.2 The Kalman Filter

The state feedback with reference feedforward control structure shown in Figure 5 requires knowledge of the system states in order to calculate the required control law. In the DoD system only the position states, $\theta$ and $\phi$, can be directly measured with a rotary encoder and computer vision, respectively. The velocity states cannot be directly measured. Additionally, the measurements of the position states are subject to uncertainty. To ensure the controller performs well an accurate estimation of the states must be computed.

A Kalman Filter can be used for state estimation. The Kalman Filter is the closed form solution to the Bayes Filter when it is assumed the state-transition model and measurement model are multivariate normally distributed. A Bayes Filter relies on two assumptions: The state transition model is a Markov Chain and the measurement model is conditionally independent. These assumptions ensure that the probability density function (PDF) of the next state is only dependant on the previous state, and the measurement at a given time is only dependent on the state at that time. The Bayes Filter has the general form of Algorithm 1 [11].

Algorithm 1

Choose initial state PDF (prior)

$$p(x_0|y_0) = p(x_0) \tag{36}$$

for t = 1,2 .... T

Measurement update: update the prior with new information

$$p(x_t|y_{1:t}) = \frac{p(y_t|x_t)p(x_t|y_{1:t-1})}{\int p(y_t|x_t)p(x_t|y_{t-1})dx_t} \tag{37}$$

Prediction update: compute the state distribution at t+1

$$p(x_{t+1}|y_{1:t}) = \int p(x_{t+1}|x_t)p(x_t|y_{1:t})dx_t \tag{38}$$

end

The Bayes Filter has two implementation issues: the measurement update and prediction update both contain the product of PDF's and an integration [11]. The Kalman Filter does not have these issues due to the assumptions on the state and measurement PDF's. For a system with the form

$$\mathbf{x}_{t+1} = \mathbf{A}_t\mathbf{x}_t + \mathbf{B}_t\mathbf{u}_t + \mathbf{w}_t \tag{39}$$

$$\mathbf{y}_t = \mathbf{C}_t\mathbf{x}_t + \mathbf{D}_t\mathbf{u}_t + \mathbf{v}_t \tag{40}$$

where $\mathbf{w}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_t)$ and $\mathbf{v}_t = \mathcal{N}(\mathbf{0}, \mathbf{R}_t)$, then the Bayesian Filter has the closed form solution

$$p(\mathbf{x}_{t+1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{A}_t\mathbf{x}_t + \mathbf{B}_t\mathbf{u}_t, \mathbf{Q}_t) \tag{41}$$

$$p(\mathbf{y}_t|\mathbf{x}_t) = \mathcal{N}(\mathbf{C}_t\mathbf{x}_t + \mathbf{D}_t\mathbf{u}_t, \mathbf{R}_t) \tag{42}$$

$$p(\mathbf{x}_1) = \mathcal{N}(\boldsymbol{\mu}_{1|0}, \mathbf{P}_{1|0}) \tag{43}$$

This reduces the measurement and prediction step of the Bayes Filter to produce multivariate normal distributions, giving Algorithm 2 [11].

Algorithm 2

Choose initial state PDF (prior)

$$p(x_1) = \mathcal{N}(\mu_{1|0}, P_{1|0}) \tag{44}$$

for t = 1,2 .... T

Measurement update: update the prior with new information

$$p(x_t|y_{1:t}) = \frac{p(y_t|x_t)p(x_t|y_{1:t-1})}{\int p(y_t|x_t)p(x_t|y_{t-1})dx_t} = \mathcal{N}(\mu_{t|t}, P_{t|t}) \tag{45}$$

Prediction update: compute the state distribution at $\mathrm{t}+1$

$$p(x_{t+1}|y_{1:t}) = \int p(x_{t+1}|x_t)p(x_t|y_{1:t})dx_t = \mathcal{N}(\mu_{t+1|t}, P_{t+1|t}) \tag{46}$$

end

where

$$\mu_{t|t} = \mu_{t|t-1} + P_{t|t}C_t^{\mathrm{T}}\left(C_t P_{t|t-1} C_t^{\mathrm{T}} + R_t\right)^{-1}(y_t - C_t\mu_{t|t-1} - D_t u_t) \tag{47}$$

$$P_{t|t} = P_{t|t-1} - P_{t|t-1}C_t^{\mathrm{T}}\left(C_t P_{t|t-1} C_t^{\mathrm{T}} + R_t\right)^{-1} C_t P_{t|t-1} \tag{48}$$

$$\mu_{t+1|t} = A_t\mu_{t|t} + B_t u_t \tag{49}$$

$$P_{t+1|t} = A_t P_{t|t} A_t^T + Q_t \tag{50}$$

If the uncertainty on the states is assumed to be normally distributed, then the state space model derived in (22) and (23) is of the form required for a Kalman Filter.

Use of a Kalman Filter for state estimation ensures that the state estimates are suitable for control law calculations. The Kalman filter ensures that erroneous measurements that are extremely unlikely given the state transition model are correctly treated as noise.

There are effectively three tuning parameters associated with a Kalman Filter: the state transition model uncertainty, the measurement uncertainty, and the initial prediction density. These parameters are expressed as covariance matrices. The state transition model uncertainty captures the confidence

in the model. A model that does not accurately reflect the true system dynamics should have a large model uncertainty. For the DoD model shown in (28), there is very little uncertainty associated with the derivates of the position states, as the time derivative of position is velocity. There is a much more significant uncertainty associated with the velocity states. The initial prediction uncertainty describes the confidence in the initial state of the system, which is required to be experimentally determined. Measurement model uncertainty is a function of the camera. This is determined by taking a series of measurements of the positions of the disks while they are stationary. The measurement uncertainty is calculated by taking the square of the variance of the measurements from the mean.

Application of the Kalman Filter to the measurements obtained from the computer vision process generates a state estimation suitable for control.

# 7 Experiment Apparatus Design

While modelling and simulation are suitable for a proof of method, they do not reflect the complexities of a fully realised system. A real system will have a non-ideal motor that is subject to gearbox friction and non-linear torque curves; the disks may not be perfectly round, or have homogenous material properties leading to a centre of gravity that is not located at the geometric centre. The interaction between the two disks may be subject to slipping or skidding. These factors warrant the manufacture of an experiment apparatus to validate the control process. The design of the apparatus is based on the results of the simulation.

There are few constraints on the design of the system to preserve model accuracy. The primary constraints are that the disks be as round as possible, and that they are held in plane perpendicular to the ground. Most other design choices are driven by ease of construction, simplicity, and material availability. The disks are held between two clear enclosure plates. The front plate must be clear to allow the computer vision process to see the object. A frame suspends the enclosure plates and disks from the ground, provides stability against tipping, and provides mounting points for the hardware and camera. Torque is transferred from the motor to the hand via a keyed shaft. The shaft supports the load of the two disks and interacts with the enclosure plates through a bearing.

Enclosure plates are milled from 10mm polycarbonate sheet. Polycarbonate is a thermoplastic with good machinability and optical properties. The disks are also milled from this sheet. Perspex is an alternative material but is less flexible, and more prone to shattering, and demands a significantly higher financial investment.

20 x 20 x 2mm aluminium square section is used to construct the frame. This material provides adequate strength alongside good workability, with low weight. Sections are easily joined via welding. Bearings and shafts are off-the-shelf items appropriate for the size of the components. Figure 12 shows a general assembly of the apparatus.

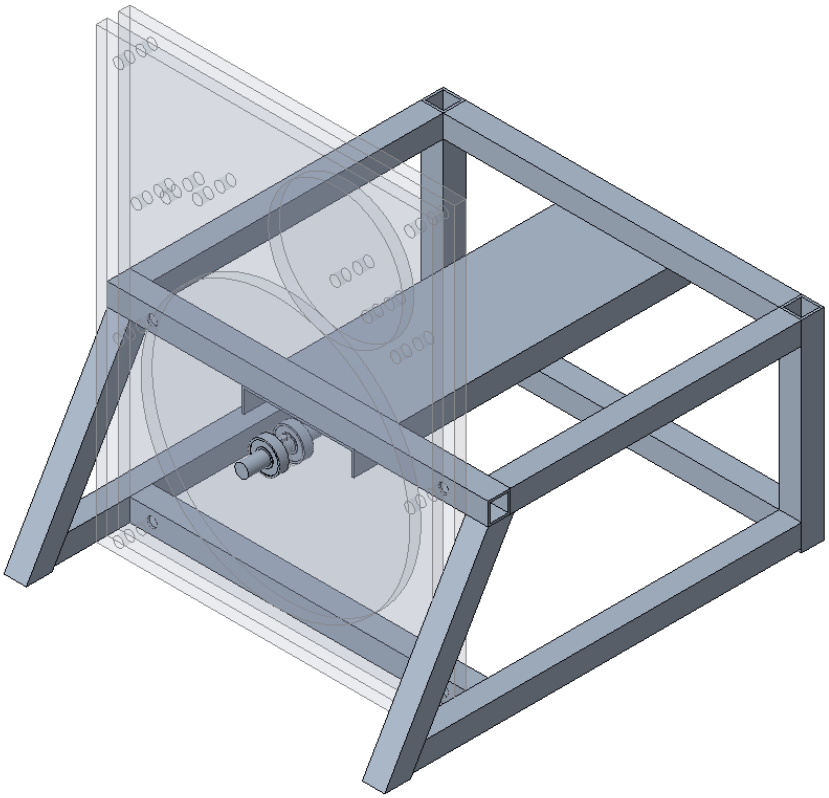

*Figure 12. Isometric view of experiment apparatus general assembly.*

Object and hand size determine the speed of the system dynamics and the required control torque. Geometrically smaller systems exhibit much faster system dynamics and demand higher frequency control and measurements. Larger systems do not have such high frequency dynamics but do require larger control forces due to greater inertia. Hobbyist electronic motors are often available in high-torque or high-speed options as these are inversely related. The model and simulation developed allow the torque and angular velocity demands of the system to be estimated. Figure 13 shows the angular velocity of the hand and object, alongside the demanded torque, subject to a control law. With an object diameter of 120mm and a hand diameter of 240mm the torque demanded for a 120RPM change are below the maximum output of a hobbyist DC motor such as the Pololu 4846.

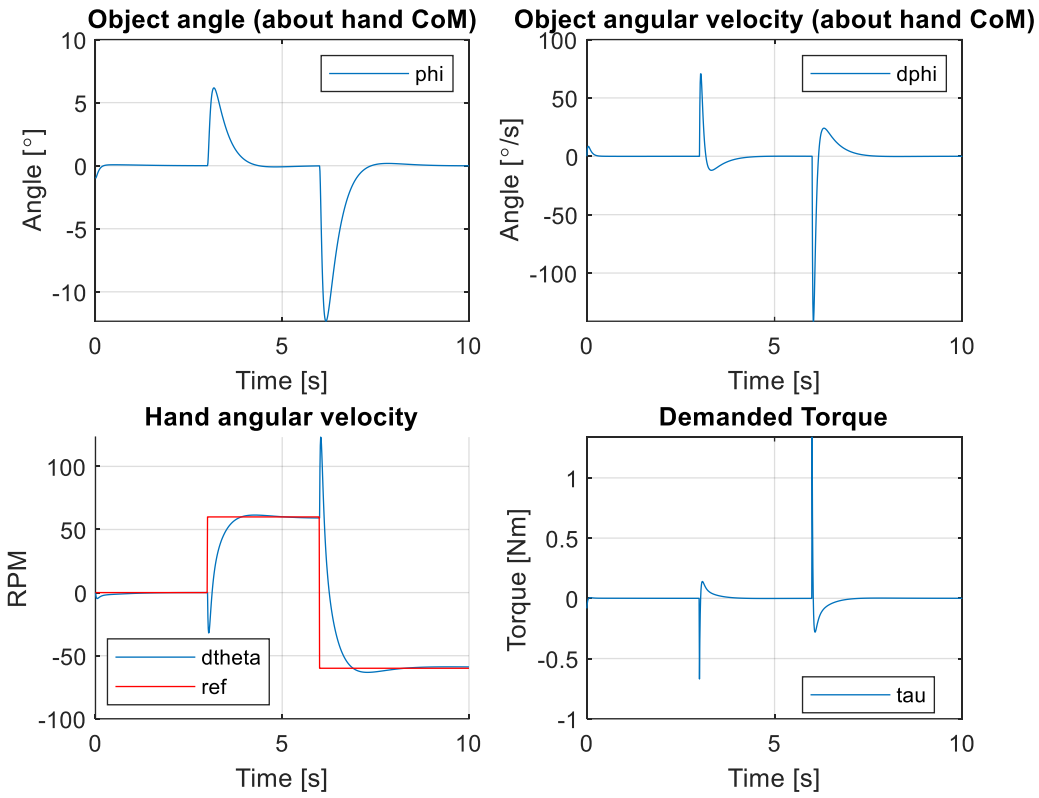


*Figure 13. Angle, angular velocity and demanded torque plots for applied control law.*

# 8 Blargh

Build:

- Add washers to reduce friction

Camera:

- Stickers for markers
- Add reference sticker
- Change exposure to be very short
- Change contrast and brightness for ideal HSV

SysID

- Start with linear model – Voltage input match velocity
- Nonlinear resistance – Voltage input match velocity
- Back to linear – omega and current input, match voltage
- Linear model to find inductance. V and omega input, match velocity
- Nonlinear model to find friction
    - Symmetric coulomb
    - Symmetric stribeck
    - Symmetric couloumb+stribeck
    - Asymmetric couloumb+stribeck
    - Asymmetric couloumb+stribeck+stiction
    - Smooth and non-smooth models
- Implement current control
    - Doesn't work due to stiction
- Remove current control loop, implement torque control
    - Works badly due to backlash in gearbox
- Implement KF and controller
    - Very chattery performance, backlash causing too many issues
    - Prediction is also poor due to backlash, leading to more issues

# 9 References


[1] A. Donaire, F. Ruggiero, L. R. Buonocore, V. Lippiello, and B. Siciliano, "Passivity-Based Control for a Rolling-Balancing System: The Nonprehensile Disk-on-Disk," *IEEE Transactions on Control Systems Technology,* vol. 25, no. 6, pp. 2135-2142, 2017, doi: 10.1109/TCST.2016.2637719.

[2] J. Ryu, F. Ruggiero, and K. M. Lynch, "Control of Nonprehensile Rolling Manipulation: Balancing a Disk on a Disk," *IEEE Transactions on Robotics,* vol. 29, no. 5, pp. 1152-1161, 2013, doi: 10.1109/TRO.2013.2262775.

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

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

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

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

[7] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence, Pattern Analysis and Machine Intelligence, IEEE Transactions on, IEEE Trans. Pattern Anal. Mach. Intell.,* Periodical vol. 22, no. 11, pp. 1330-1334, 11/01/ 2000, doi: 10.1109/34.888718.

[8] J. Heikkila and O. Silven, "A four-step camera calibration procedure with implicit image correction," ed, 1997, pp. 1106-1112.

[9] MathWorks. "estimateCameraParameters." MathWorks. (accessed 26/05, 2020).

[10] A. Smith, *Color Gamut Transform Pairs*. 1978, pp. 12-19.

[11] A. Wills, *Course Notes: Advanced Estimation*, The Universtiy of Newcastle, Australia., 2020.

# Appendix A – Time Log Graphic

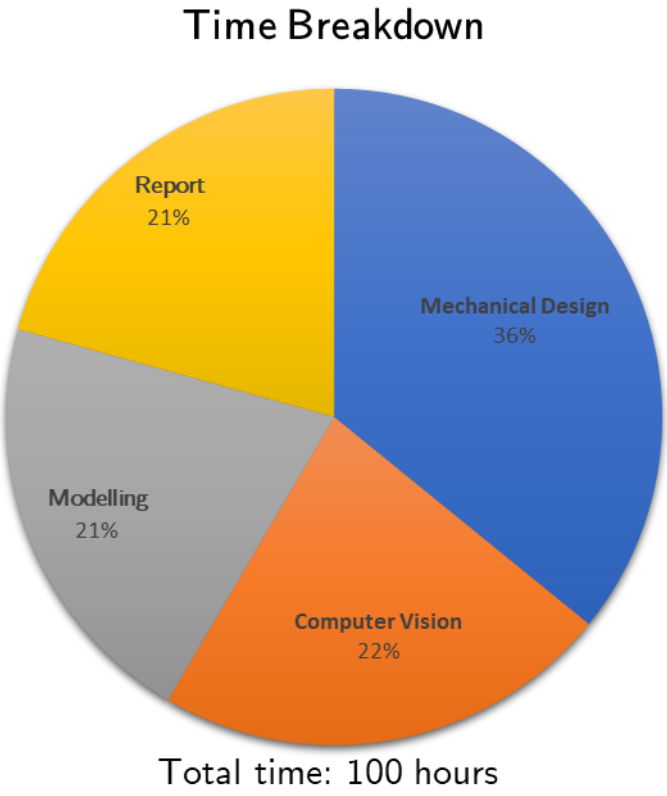


*Figure 14. Breakdown of time spent.*