Neural Network based robust control stabilization of Spherical Inverted Pendulum with Dual-Axis Reaction Wheels

by

Yerzhan Rzagaliyev

Submitted to the Department of Robotics and Mechatronics
in partial fulfillment of the requirements for the degree of

Master of Science in Robotics

at the

NAZARBAYEV UNIVERSITY

Apr 2020

Author .....
Department of Robotics and Mechatronics
Apr 29, 2020

Certified by .....
Huseyin Atakan Varol
Chair and Professor
Thesis Supervisor

Accepted by .....
Vassilios D. Tourassis
Dean, School of Engineering and Digital Sciences

Neural Network based robust control stabilization of Spherical Inverted Pendulum with Dual-Axis Reaction Wheels

by

Yerzhan Rzagaliyev

Submitted to the Department of Robotics and Mechatronics
on Apr 29, 2020, in partial fulfillment of the
requirements for the degree of
Master of Science in Robotics

Abstract

As the new paradigm shift happened in the industrialization, introducing "Industry 4.0", many new applications in robotics are emerged. One of them safe human robot interaction field. Such an approach can be executed by utilizing variable impedance actuators, which imitates human and animal bodies, guaranteeing safety during the interaction. But on the other hand, these actuators have some limitations due to the damping elements in their construction. These elements behave as filters and may decrease the accuracy and positioning of the end effector, and may cause vibrations. Therefore, to avoid such unfavorable conditions, some stabilization elements should be integrated with the actuators. The thesis work will introduce a two-axis reaction wheel based inverted pendulum, which can deal with the nuisance by liquidating vibrations and adjust accuracy. In this work the utilization of Neural Networks is implemented to stabilize the pendulum. Moreover, it will consider the robust control Neural Network for the case of uncertainties and external disturbances. By generating the simulations and conducting real world experiments, the work will demonstrate the advantages of Neural Network employment.

Thesis Supervisor: Huseyin Atakan Varol
Title: Chair and Professor

Acknowledgments

Thanks to my thesis supervisor, Professor Huseyin Atakan Varol, who suggested this topic and his guidance.

Great thanks to Daulet Baimukashev who helped me during the whole thesis work, with such things like conducting experiments, training neural network models and calculating system parameters.

Also I would like to thank all ARMS lab members, including former members, especially, Bexultan Rakhim, for sharing their suggestions and thoughts.

Contents

List of Figures

Chapter 1

Introduction

Industry has economical advantages that enhance mechanical and automatization processes. As industrialization began, many technological advances caused 'industrial revolutions' [2]. Starting with the first industrial revolution which involved mechanization by using water and steam, followed by the second industrial revolution involved utilization of electricity for promotion of mass production. Then the third industrial revolution which resulted in the replacement of mechanical and electrical production machines by programmable devices, so-called Programmable Logic Controllers (PLCs) [2], [3]. As the factories became digitalized, Internet and future-oriented technologies are combined to one field called 'smart' objects, such novelties, as a result cause paradigm shift in industrialization. By looking ahead the future, the production process is controlled by its own way. According to this scenario, it seems that these manufactures can maintain mass production [4] [5] [6]. This expectation engendered new 'industrial revolution' called 'Industry 4.0' [2] .

Industry 4.0 refers to several main aspects in this revolution, such as computer-enabled mass production, where the product should be adapted for particular and individual needs [7]. Furthermore, these production can be flexible and automatically adapt for immediate change of the requirements [7] [8]. Therewith, the products can track and have self-awareness, and also they can communicate with other products. This revolution has enhanced human-robot interaction paradigm, so that human and robot can coexist and operate in the same plant [9] [10]. In the virtue of Internet of

Things (IoT) communication of smart technologies, production process is significantly optimized [7].

As the one of the main aims of Industry 4.0 is human-robot interaction, the term defines that human and machines can cooperate. However, in the present factories that utilize industry machines humans are isolated from robots by safety walls. Therefore, in the aspect of the safety of robotic manipulators for human should be the main focus [11]. In order to define this safety, scientists created standards to calculate the injury severity, such as Head injury coefficient(HIC) where it is calculated by the equation [12]

$$\begin{aligned} HIC_{36} &= max_{\Delta t}[\Delta t(\frac{1}{\Delta t}\int_{t_1}^{t_2} a^*(t)\,dt)^{2.5}] \\ \Delta t_{max} &= maxt_2 - t_1 < 36ms \end{aligned} \tag{1.1}$$

As one of the outstanding examples of Industry 4.0, Variable Impedance Actuators have been created as a new paradigm shift. Since they provide safe and efficacious physical human-robot interaction [13].

Variable impedance actuated robots are consisted of stiffness and damping elements between their joints and links. On the other hand, these impedance elements may worsen robots motion control performance, due to their nature, because they act like mechanical filters, as it leads to decrease the bandwidth of the robot. According to the researchers, by integrating variable impedance robots [14] with reaction wheels to their structure [15] can increase the motion control performance of variable impedance robots. Therefore, variable impedance actuators integration with reaction wheel can also solve human robot interaction problem.

It has firstly been introduced by [16] and, from pedagogical point of view, as one of simplest nonlinear systems that can be used to illustrate advanced control designs based on recently developed geometric methods.

According to [17] from a mechanical point of view, the Reaction Wheel Pendulum can be considered as a simple pendulum with a rotating wheel, at the end. The wheel is attached to the shaft of motor actuator and, in order to control the motion of the system, it has been utilized the coupling torque between the wheel and pendulum.

The Reaction Wheel Pendulum may be thought of as a simple pendulum in parallel with a torque-controlled inertia.

Due to its high accuracy and its independence for fuel, it is used in the space industry, but recently it has gained interest in the robotics field as it can be utilized for the stabilization of inverted pendulums.

In order to control this kind of system will require motion planning of reaction wheels for determined time horizon, for some seconds, in future, so that it can stabilize the system on a final rest configuration. Therefore, it will be sufficient to use optimal control solution, especially solving by using optimal control problem(OCP) over an infinite time horizon [1]. For the moment researchers implemented DNN for Reaction Wheel Inverted Pendulum which can be accepted as current state of the art in the area [18] . And it has much faster stabilization time comparing with OCP. However, NNs, due to their very nature, without considering system parameter change, may overfit, since they were trained on some specific system parameter. If we consider the case when one of the system parameters fails or changes due to environment, these NNs may also fail. Thus, our controller should be robust to this kind of system changes. As a consequence, Robust control is needed to avoid stabilization errors caused by environment and for faster response time.

This thesis will describe robust control for stabilization of dual axis spherical inverted pendulum utilizing Neural Networks. The thesis sections into five main chapters. The first chapter provides general overview what is reaction wheel inverted pendulum and why it is advantegeous.The chapter two presents the physical setup that will be used to conduct experiment and by detail will describe about the components of the reaction wheel pendulum. The chapter three provides information about system dynamics and kinematics of the pendulum. The chapter four presents stabilization control algorithms for the pendulum. At the end the thesis work will be concluded with work done during the experiment and give some future work information.

Chapter 2

Physical setup

The physical setup is needed in order to check simulated trajectories and algorithms in the real world. In this section it will be described about the components of the physical setup and will give detailed description about them. Overall, the setup is divided into two main parts: the lower part and upper part. The lower part is made of two separate plastic boards connected with a universal joint at the middle with two encoders attached on it. In order to emulate the variable impedance, eight springs are utilized and are parallel connected on four corners of two plastic boards. Then the upper part is attached to the lower part with an aluminium rod. The upper part is made of reaction wheels established on top of rod and mass holder which is established at the middle of rod, so that it will be around the center of mass of the whole pendulum. All plastic parts of the setup are 3D printed.

2.1 The Lower Part components

The figures on 2-1 and 2-2 show the assembly design of the lower part. Compared with previously used design, the present design has been decently modernized. Due to tension of springs, the plastic boards start to bend. Therefore, new plates have been designed patterns that will withstand such tensions. The universal joint is made of two small orthogonally connected steel shafts with bearings attached to the protruded sections of the plastic board. On these shafts the encoders are embedded in order to

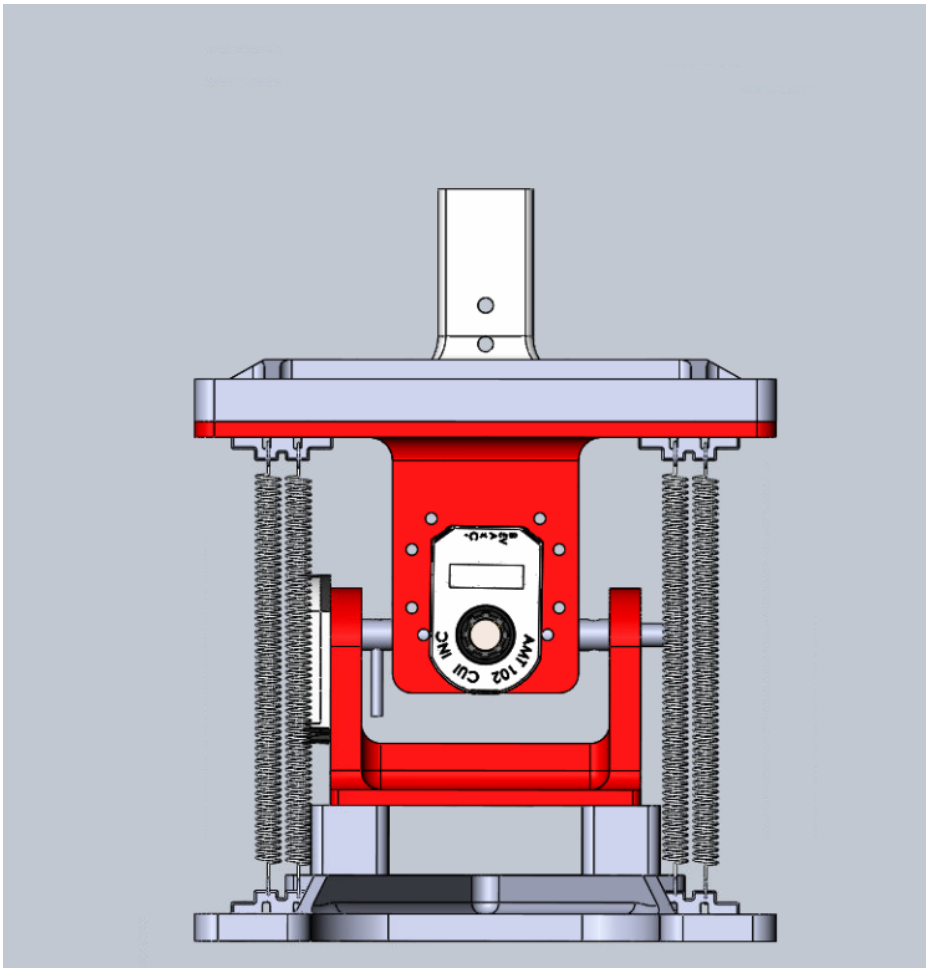

Figure 2-1: The design of lower part from front view.

measure the angular displacement of the pendulum. The setup springs are placed by pair parallel connected on the corners and they are on strained condition. Since this setup will be integrated with VIAs, these springs and bearings on the universal joint will mimic variable impedance behaviour.

2.2 The Upper Part components

In figure 2-3 it can be seen in the design of the upper part. As it shows it consists of the reaction wheels, plastic cases, and mass holders. The reaction wheel is made of steel and laser cut. It has a mass of 160 grams, a radius of 8 cm and its thickness is 3 mm. Moreover, it has an optimized design such that it will have increased the moment of inertia of the reaction wheel. Then these two reaction wheels are attached

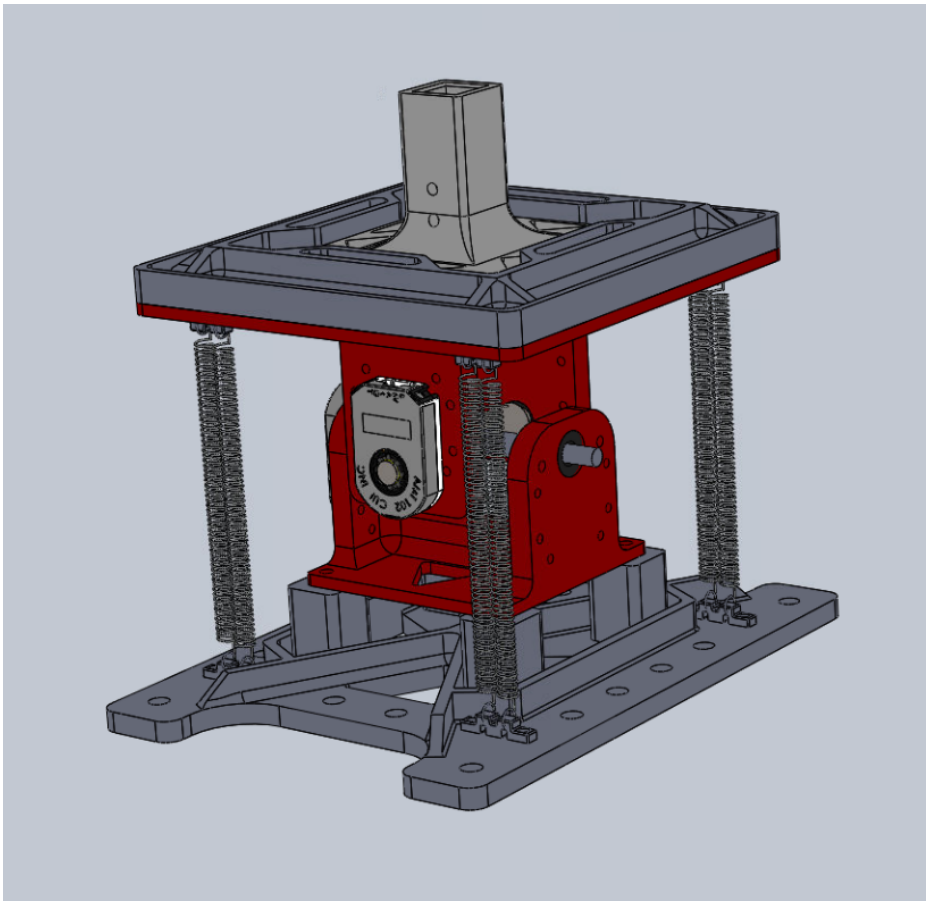

Figure 2-2: The design of lower part from isometric view.

to the motors, which are fixed by the motor holder in order to be able to attach to the setup. Afterwards the wheels are covered with the plastic cases. The plastic case of the reaction wheel is rapid prototyped, and made such that it will consider safety and eliminate vibrations during the wheel rotation. Then the encoders are integrated to the tip of the motors to be able to measure angular displacements and velocities. Another main component of the upper part is the mass holder(2-4 and 2-5). This holder is needed to test robust control control stabilization of the pendulum. In experiments it will be attached up to 20 circular weights with mass of 50 grams each, and with dimensions of 9 mm thickness and 15 mm in radius.

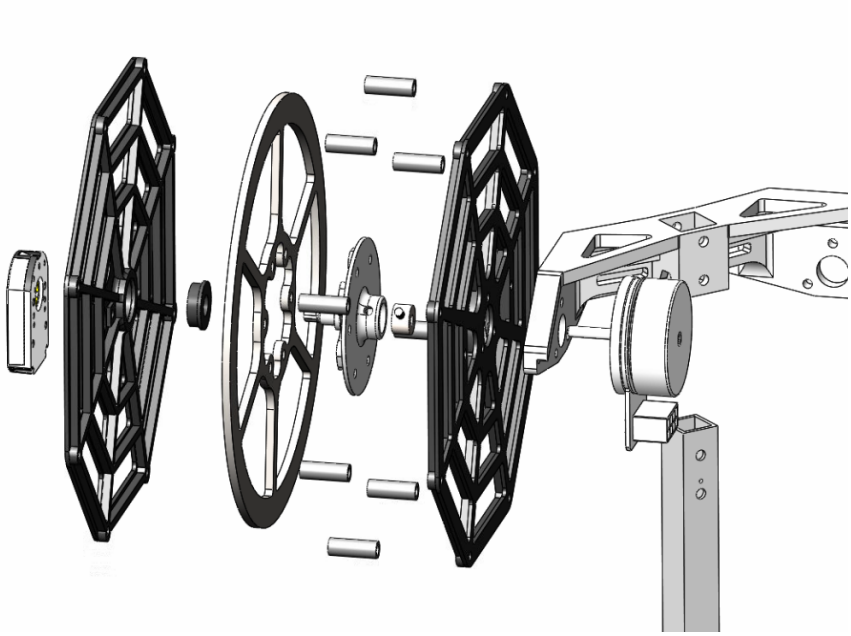

Figure 2-3: Exploded view of Reaction wheel.

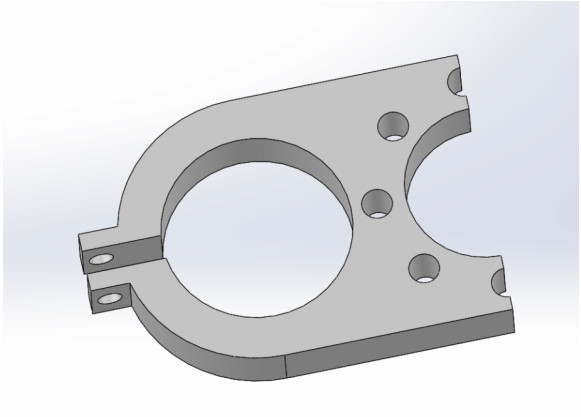

Figure 2-4: The design of the mass holder.

2.3 Hardware components

Two EC 45 Maxon Motors are utilized to actuate the reaction wheels. These motors are brushless DC motors, and at the nominal voltage 48 V they have values with the nominal current (max. continuous current) 0.936 A and with the nominal torque (maximum continuous torque) 134 mNm, the stall current is 6.97 A and stall torque is 915 mNm. The motors have the maximum efficiency of 85 %. They also have the characteristics of the torque constant with value 131 nNm/A and the speed constant 72 rpm/V. Mentioning about the mechanical characteristics, the motors weigh 140

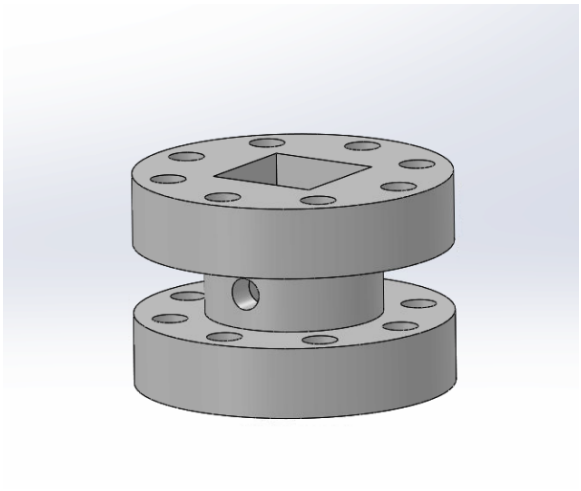

Figure 2-5: The design of the attachment for mass holder.

g, have a diameter of 42.8 mm and the rotor inertia of 181 gcm^2. The motors are controlled with Digital Electronic Commutated (DEC) controllers in the current mode and the currents measured with Hall effect sensors inside the motors.

In order to measure the angular velocity of the pendulum two VEX Yaw Rate Gyroscope Sensor V1.0 gyro sensors were used. The sensors will measure angular velocities of the link around x and y axis and are connected to NI DAQ data acquisition cards to send analog signals.

For the whole setup it is utilized four encoders: two of them are mounted on the upper part in order to measure the angular position of the flywheels and the other two measure the angular position of the link on the lower part of the setup. For this purpose, the modular incremental encoders were used from the AMT10 series (specifically AMT102 type). It has low power consumption, also featured with a modular package design and a light weight (20 g). Then the data acquired from encoders are sent to the NI DAQ data acquisition card.

For reading the signals from encoders and gyroscopes, and transmit control outputs to the DEC, it is utilized by two National Instruments Series Data Acquisitions (DAQ) PCIe-6259. They can be connected to the PC, in order to make high resolution detection. These data acquisition cards measure the signal in 16 bit resolution. Then they are connected to the computer with Intel Core i7-4790 32 GB RAM to

make computational tasks there.

Chapter 3

System dynamics

In this chapter it is discussed about the system dynamics of the reaction wheel pendulum. In order to derive the dynamics equations Lagrange principle of stationary action is implemented. First, these equations are obtained numerically and then using symbolic toolbox in MATLAB, the dynamics equations run in simulation.

3.1 Kinematics

As can be shown from 3-1, the angle θ_1 corresponds to the rotation of the pendulum around y-axis with respect to the frame (x_0, y_0, z_0), and the angle θ_2 corresponds to the rotation of the pendulum around x-axis with respect to the frame (x_1, y_1, z_1). For this system the vector of generalized coordinates is taken as $q = [\theta_1, \theta_2, \phi_1, \phi_2]^T$. Therefore, the rotation matrices with mentioned angles are:

$$ {}^{0}_{1}R = \begin{bmatrix} cos(\theta_1) & 0 & sin(\theta_1) \\ 0 & 1 & 0 \\ -sin(\theta_1) & 0 & cos(\theta_1) \end{bmatrix} \tag{3.1} $$

$$ {}^{1}_{2}R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\theta_2) & sin(\theta_2) \\ 0 & -sin(\theta_2) & cos(\theta_2) \end{bmatrix} \tag{3.2} $$

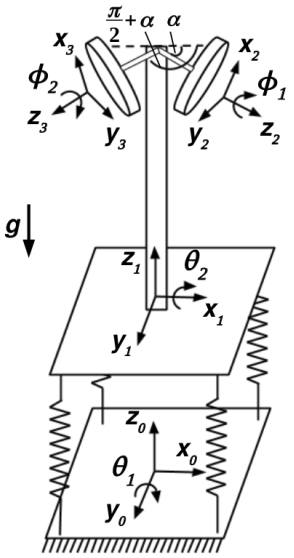

Figure 3-1: Block diagram of the dual-axis reaction wheel inverted pendulum [1]

So, by taking into account 3.1 and 3.2 the whole transformation matrix will be ${}^0_2R = {}^0_1R^1_2R$. The reaction wheels have fixed angle α between each other around z-axis and they will have rotation matrices are denoted as R_{ω_1} and R_{ω_2}. Accordingly, the angular velocities of the pendulum and reaction wheels are expressed as

$$\begin{aligned} \hat{\omega}_p &=^0_2 R({}^1_2R^{-1}[0\ \dot{\theta}_1\ 0]^T + [\dot{\theta}_2\ 0\ 0]^T) \\ \hat{\omega}_{\omega_1} &=^0_2 RR_{\omega_1}[\dot{\phi}_1\ 0\ 0]^T \\ \hat{\omega}_{\omega_2} &=^0_2 RR_{\omega_2}[\dot{\phi}_2\ 0\ 0]^T \end{aligned} \tag{3.3}$$

3.2 Lagrangian Function

Taking into account Lagrangian principle of stationary action equation $L = T - V$, we can derive dynamics of mechanical body, where T is kinetic energy and V is potential energy of the pendulum. By considering pendulum and reaction wheels, we get general equation for kinetic energy as follows

$$T = \frac{1}{2}\hat{\omega}_p^T I_1 \hat{\omega}_p + \frac{1}{2}\hat{\omega}_{\omega_1}^T I_{\omega_1}\hat{\omega}_{\omega_1} + \frac{1}{2}\hat{\omega}_{\omega_2}^T I_{\omega_2}\hat{\omega}_{\omega_2} \tag{3.4}$$

where I_1 is inertia matrix of the whole pendulum around rotation point, I_{ω_1} and I_{ω_2} are inertia matrices around their rotation points. In the same way, we can find the

potential energy of the system, by taking into consideration, gravitational potential energy of the entire pendulum and elastic potential energy generated by springs as

$$V = \overline{g}^{T0}{}_{2}R(m_p l_p + m_{\omega_1} l_{\omega_1} + m_{\omega_2} l_{\omega_2}) + \frac{1}{2}(k_1(\theta_1 - \theta_{1_i})^2 + k_2(\theta_2 - \theta_{2_i})^2) \tag{3.5}$$

where $\overline{g}$ is gravitational acceleration vector ([0, 0, -9.81]), m_p is mass of the entire pendulum, l_p is the distance of the center of mass of the pendulum, $k1$ and $k2$ are rotational spring constants, which are should be derived by using the geometry of the platform and linear spring constants.

By substituting our Lagrangian function into Euler-Lagrange equation which is expressed as

$$\frac{d}{dt}\frac{\partial}{\partial \dot{q}}L - \frac{\partial}{\partial q}L = \tau \tag{3.6}$$

then we get equation

$$\frac{d}{dt}\frac{\partial}{\partial \dot{q}}T - \frac{\partial}{\partial q}T + \frac{\partial}{\partial q}U = \tau \tag{3.7}$$

Afterwards we can express 3.7 in the form of dynamics equation as follows

$$M(q)\ddot{q} + C(q, \dot{q}) + G(q) = \tau \tag{3.8}$$

where $M(q)$ is mass matrix, $C(q, \dot{q})$ is Coriolis and centrifugal force matrix, $G(q)$ is gravity force matrix, τ is external torque applied to the pendulum.

In order to control stabilization of the pendulum, we need to move motors and provide them some current inputs, motor torques can be derived as $\tau_1 = Ki_1$ and $\tau_2 = Ki_2$, where K is the motor constant, i_1 and i_2 are the currents applied to the motors. Since, we consider not ideal case for the rotation of the reaction wheels, therefore, they will be affected by damping forces $b_{\omega_1}\dot{\phi}_1$ and $b_{\omega_2}\dot{\phi}_2$. So, we have an equation

$$\tau = \begin{bmatrix} cos(\theta_2)cos(\alpha)(-K(i_{\omega_1} + i_{\omega_2}) + b_{\omega_1}\dot{\phi}_1 + b_{\omega_2}\dot{\phi}_2) - b_1\theta_1 \\ sin(\alpha)(-K(i_{\omega_1} - i_{\omega_2}) + b_{\omega_1}\dot{\phi}_1 - b_{\omega_2}\dot{\phi}_2) - b_2\theta_1 \\ Ki_{\omega_1} - b_{\omega_1}\dot{\phi}_1 \\ Ki_{\omega_2} - b_{\omega_1}\dot{\phi}_1 \end{bmatrix} \tag{3.9}$$

where α is, as mentioned before, the angle between the reaction wheels around z-axis, b_1 and b_2 are the rotational friction coefficients of the joints of the pendulum.

Since $M(q)$ is square and invertible matrix, the equation 3.8, can be written as $\ddot{q} = M(q)^{-1}(\tau - C(q,\dot{q}) - G(q))$. Then the state-space representation of our pendulum will be

$$\dot{x} = f(x,u) = \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ (M(q)^{-1}(\tau - C(q,\dot{q}) - G(q)))^T \end{bmatrix} \tag{3.10}$$

Chapter 4

Physical System identification and Controller implementations of the pendulum

In this chapter we will talk about physical parameters of the system and its control implemented to stabilize. At the beginning it shows the physical system parameters as moments of inertia, mass, spring constants and friction coefficients that is necessary to find. Then, to check whether these parameters are correct, it will be verified by simulation and experiments. After that, different Neural Network (NN) based control algorithms will be proposed for stabilization of the system. In order to stabilize it is implemented nominal NN and robust NN. Also they will be compared with different system characteristics that identified and consider the parameter uncertainty cases, both in simulations and real world.

4.1 Physical System identification

In order to satisfy to the system dynamics as shown in previous chapter, we are needed to identify physical characteristics of the system. First of all, we need to measure mass of the system. The masses of the system are measured in real-world and put to Solidworks in similar manner as in real-world into the simulation. By

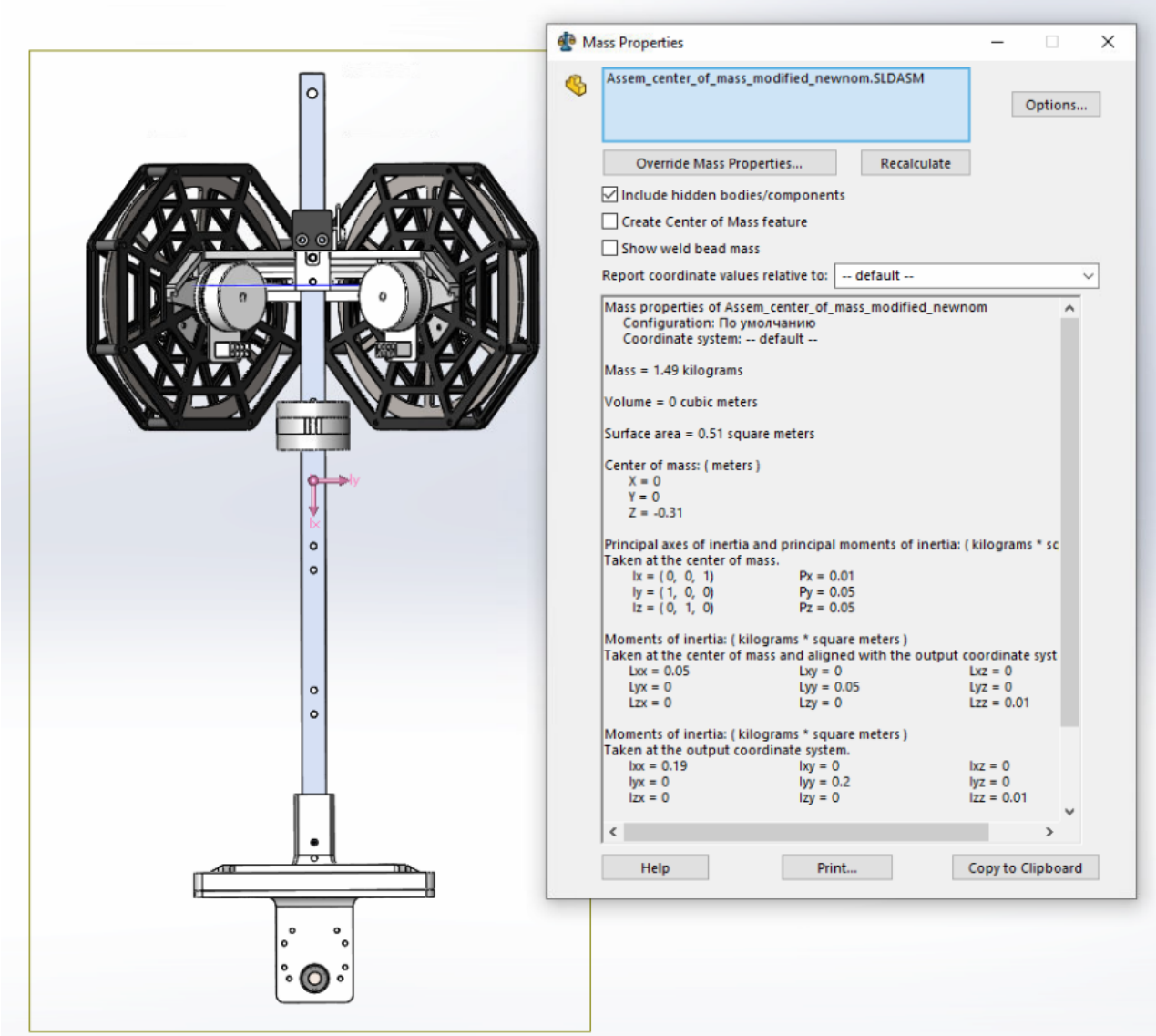

Figure 4-1: Mechanical properties of the Reaction wheel pendulum.

applying it, we get characteristics as shown in Figure 4-1.

This figure is done using Mass Properties add-on in Solidworks. As can be seen from Figure 4-1 the mass of the system is 1.49 kilograms and moments of inertia for in three dimensions are around 0.19 kg·m^2 along x-axis, 0.19 kg·m^2 along y-axis and 0.01 kg·m^2 along z-axis, the center of mass of the system is 0.31 meters on z-axis. Then we find the spring constants and friction coefficients for our system, so that it will be considered as damping elements of the pendulum and stabilize it on zero

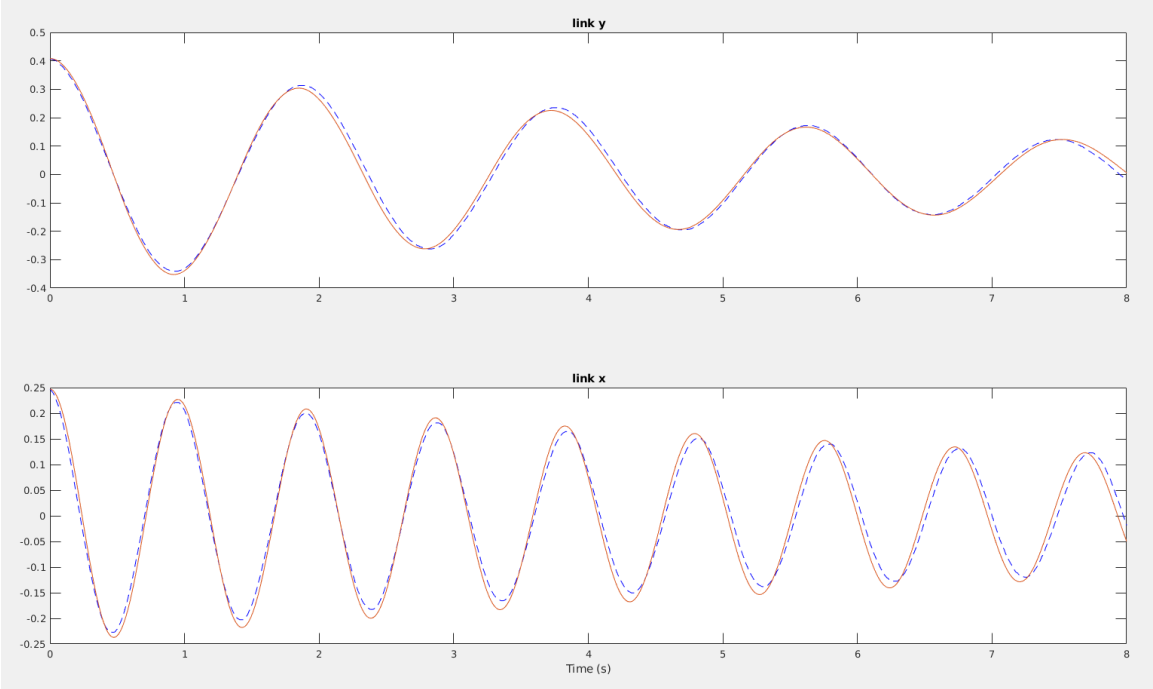

Figure 4-2: The simulation result of rotational spring constant and pendulum friction coefficient. Real world result (blue dashed line) vs simulated result (red line)

position. Since, our springs are linear and have linear spring constant, for the system we need to find rotational spring constants. They are found first theoretically and then verified experimentally, they more or less equal each other, comparing with real-world simulation we get 12.5 N·m/rad and 6.9 N·m/rad. By the similar manner the friction coefficients for the pendulum are found, and they are 0.09 N·m·s/rad and 0.12 N·m·s/rad. The expected value of the simulation with real world is shown in Figure 4-2.

For finding friction coefficient for flywheels b|$omega$1 and b|$omega$2. By applying a constant torque to the reaction wheels for a small time, it will start rotating by some velocity, and it can be derived by the equation like

$$I_n \frac{d}{dt}\omega(t) + b_{\omega n}\omega(t) = 0, \omega(0) = \omega_0 \tag{4.1}$$

and solving the differential equation $\omega(t)$ will be

$$\omega(t) = \omega_0 e^{-\frac{b_{\omega n}}{I_n}t} \tag{4.2}$$

Parameter	Symbol	Value	Unit
Mass of the whole pendulum	m_p	1.49	kg
Center of mass along z-axis	l_p	0.31	m
Rotational spring constant around x-axis	k_1	12.5	$N{\cdot}m/rad$
Rotational spring constant around y-axis	k_2	6.9	$N{\cdot}m/rad$
Friction coefficient on x-axis of the pendulum	b_1	0.09	$N{\cdot}m{\cdot}s/rad$
Friction coefficient on y-axis of the pendulum	b_2	0.12	$N{\cdot}m{\cdot}s/rad$
Friction coefficient of the reaction wheel	$b_{\omega 1}$	1.11e-4	$N{\cdot}m{\cdot}s/rad$
Friction coefficient of the reaction wheel	$b_{\omega 2}$	0.94e-4	$N{\cdot}m{\cdot}s/rad$
Motor torque constant	K	0.130	$N{\cdot}m/A$
Moment of inertia of the pendulum	I_p	0.19	$kg{\cdot}m^2$
Moments of inertia of the reaction wheels	$I_1 = I_2$	0.753e-4	$kg{\cdot}m^2$

Table 4.1: Nominal system parameters

by using this equation and curve fitting we get the angular velocities b_{ω_n} of both reaction wheels. So, from the mentioned parameter values, we can make table of the nominal system parameters as shown in Table 4.1

4.2 Controller Implementations

In this section it will be explained about the different control approaches utilized in order to stabilize the pendulum. In this experiment two different NN-based control algorithms have been implemented: nominal NN trained with nominal parameters and robust NN trained with nominal parameter and parameter uncertainties evaluated with Optimal Control Problem (OCP). First, these algorithms are going to be compared between each other in nominal cases. Then, they will be compared for the cases when parameters have changed in simulation and real world experiment.

4.2.1 Optimal Control Problem Formulation

In order to control the reaction wheel system it is necessary to design an algorithm for motion planning with some time horizon in the future. Let's consider the initial condition x(0) when time t = 0, and for some time t ∈ [0, T], will have x(t) in interval x[0, T]. In order to control the system, we also consider some input u(t) and it will

control x(t) within some time T. This input term is used for system stabilization and can be referred to as u[0,T]. So, for some T we have the cost function of OCP which is

$$J_{OCP}(x_{[0,T]}, u_{[0,T]}) = \int_0^T l(x(t), u(t))dt \tag{4.3}$$

where $l(x(t), u(t)) = x(t)^T Q x(t) + u(t)^T R u(t)$, and $Q \in \mathbb{R}^{6\times6}$, $R \in \mathbb{R}^{2\times2}$ are the symmetric, positive definite matrices [1]. Then the OCP can be written in standard form

$$\begin{aligned} &\min_{x_{[0,T]}, u_{[0,T]}} J_{OCP}(x_{[0,T]}, u_{[0,T]}) \\ &\qquad subject\ to\ x(0) = x_0 \\ x(t) = x(0) + &\int_0^T l(x(t), u(t))dt \\ &\qquad x(t) \in X, t \in [0, T] \\ &\qquad u(t) \in U, t \in [0, T] \end{aligned} \tag{4.4}$$

constrained in

$$X = x(t)_{min} \leq x(t) \leq x(t)_{max} \tag{4.5}$$

and

$$U = u(t)_{min} \leq u(t) \leq u(t)_{max} \tag{4.6}$$

Then the cost function weights for OCP as shown in Equation 4.3 can be determined as

$$Q = \begin{bmatrix} 50000 & 0 & 0 & 0 & 0 & 0 \\ 0 & 50000 & 0 & 0 & 0 & 0 \\ 0 & 0 & 500 & 0 & 0 & 0 \\ 0 & 0 & 0 & 100 & 0 & 0 \\ 0 & 0 & 0 & 0 & 10^{-2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 10^{-2} \end{bmatrix} \tag{4.7}$$

$$R = \begin{bmatrix} 10^{-5} & 0 \\ 0 & 10^{-5} \end{bmatrix} \tag{4.8}$$

At the end, the OCP formulation will have the time horizon with 4 s. Also with x_{min} and x_{max} which are the limitation of reaction wheel angular velocities at -300 and 300 rad/s, and with u_{min} and u_{max} which are current limitations at -3 and 3 A. In order to make discretization of the dynamics of the system, for OCP solution 4th-order Runge-Kutta integrator is used with discretization interval 10 ms. The initial conditions for the OCP are chosen randomly and they include themselves angular positions (θ_1 and θ_2) of the pendulum, angular velocities ($\dot{\theta_1}$ and $\dot{\theta_2}$) of the pendulum and angular velocities ($\dot{\phi_1}$ and $\dot{\phi_2}$) of the reaction wheels. For instance, angular positions are in range of θ_1, $\theta_2 \in$ [-$\pi/4$; $\pi/4$] rad, angular positions are in range of $\dot{\theta_1}$, $\dot{\theta_2} \in$ [-5; 5] rad/s, and angular positions are in range of $\dot{\phi_1}$, $\dot{\phi_2} \in$ [-300; 300] rad/s.

The OCP trajectories are run in multiple PCs with different system characteristics. For OCP trajectory generation, it is used MATMPC toolbox [19]. For nominal parameters, 8000 trajectories were generated, and, since, OCP can not find solutions for all cases there will be some failed or not converged trajectories, so these trajectories are filtered with some threshold. After filtering, we got around 5300 clean data. For changed parameters, 15000 trajectories were generated, after filtering got around 8000 trajectories. As parameter uncertainty examples, the mass, the spring constants and the pendulum friction coefficients are chosen. Since, we consider these parameters as uncertainties, they vary for some percentage, for the mass it was chosen between 0 and 30 %, considered that this mass will be attached on the center of mass of the pendulum, for the spring constants it was chosen between -20 and 0 %, and for friction coefficients it's chosen -20 and 20 %. The mentioned clean trajectories for nominal and changed parameters are used for NN training. For testing the data for nominal and changed parameters it was taken 500 trajectories for each of the instances.

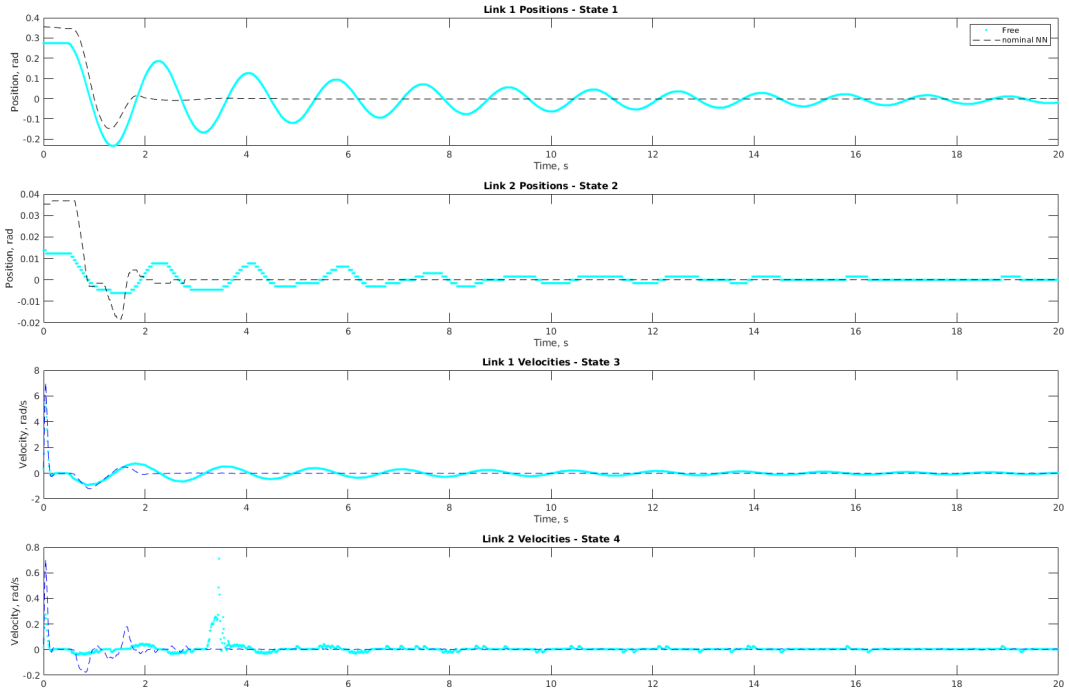

Figure 4-3: Comparison of NN and free response around y-axis(θ_1).

4.2.2 Nominal Neural Network implementation for Nominal System Parameters

As discussed before we use MATMPC generated OCP trajectories for training NNs. For training the nominal NN we used deep learning method based on fully connected feedforward neural network (FNN) with different amount of layers and neurons. The architecture of the nominal NN consists of three layers and 128 neurons in each layer with ReLU activation function. As an optimization function it is utilized RMSprop with learning rate 0.0008. For training the nominal NN, 4800 nominal parameter trajectories are used. The model is trained on DGX-2 for two hours for 100 epochs. Afterwards, we run nominal NN in order to check whether it stabilizes. As can be seen from 4-3 and 4-4, it stabilizes faster than free response started from similar position.

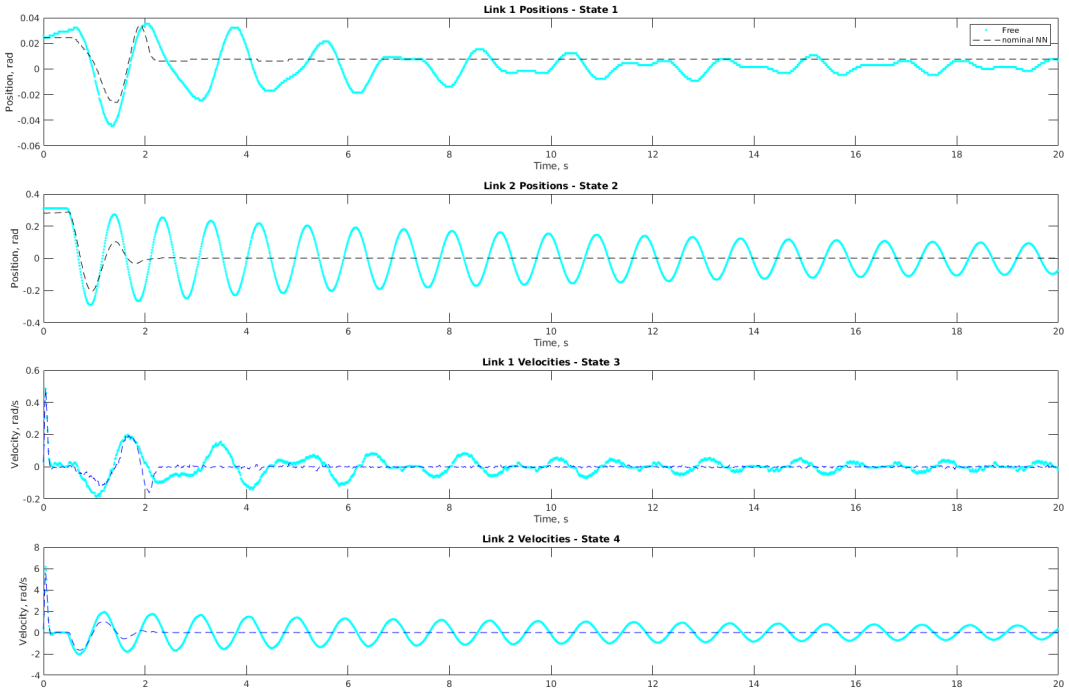

Figure 4-4: Comparison of NN and free response around x-axis(θ_2).

4.2.3 Robust Neural Network implementation for Parameter Uncertainty system

The robust control deals with the designing of the feedback controllers for dynamical systems where it might occur some disturbances and uncertainties in the system [20], meaning that it will assure that it can show efficiency at worst case scenario. By using linear matrix inequalities (LMI), robust control task can be solved in frequency and time domains [21]. However, using linear controllers for such problem significantly confines the performance [22]. Therefore, utilizing nonlinear algorithms can enhance the performance overall. For training the robust NN, Recurrent Neural Network (RNN) based algorithm was implemented. The robust NN consists of two hidden layers and with hidden size of 128 of Long Short-Term Memory (LSTM), which is the artificial RNN-like model widely used in deep learning [23], and two hidden layers of FNN with the activation function ReLU with 128 hidden neurons in each with dropout function of 0.25 (25%). RMSprop with learning rate 0.0008 is used as the optimizer. For training data, it is chosen 2000 nominal parameter and 7000 changed parameter

	Nominal parameters	Changed Parameters
FNN based nominal NN	2	56
	2	68

Table 4.2: The failure rate of FNN based nominal NN for 100 trajectories.

	Nominal parameters	Changed Parameters
RNN based nominal NN	3	56
	12	70

Table 4.3: The failure rate of RNN based nominal NN for 100 trajectories.

trajectories. Then the robust NN model is trained on DGX-2 server approximately for 2 hours.

One may be interested why is it chosen FNN as the nominal NN, not RNN. For that purpose and not to be biased in our assumptions, two networks are trained with nominal parameters and compared. As can be seen from 4.2 and 4.3, both networks behave similar. Moreover, RNN based nominal NN, in some cases, demonstrates slightly worse behaviour.

4.3 Simulations and Experimental Results

4.3.1 Simulations

This section provides the comparison of the simulations of two mentioned NNs. After training of both models, they were tested on 500 nominal and changed parameter trajectories. According to the Table 4.4, the robust NN behaves better than the nominal NN, for nominal parameter trajectories both networks show approximately similar results. However, as the system parameters are differed, the nominal NN considerably fails and can solve at lower than 50 % cases. But the robust NN can deal with such uncertainties.

Then on Figure 4-5, it is shown histogram of the simulation costs of both NNs. The costs for both of them are more or less similar, and the mean of the nominal NN is around 1.17, for the robust NN is around 1.28, the standard deviation for both

		Random initial conditions
Nominal NN	Nominal Parameters	478 (95.6 %)
	Changed Parameters	233 (46.6 %)
Robust NN	Nominal Parameters	482 (96.4 %)
	Changed Parameters	420 (84 %)

Table 4.4: The simulation results of success rate Nominal NN and Robust NN for 500 random trajectories

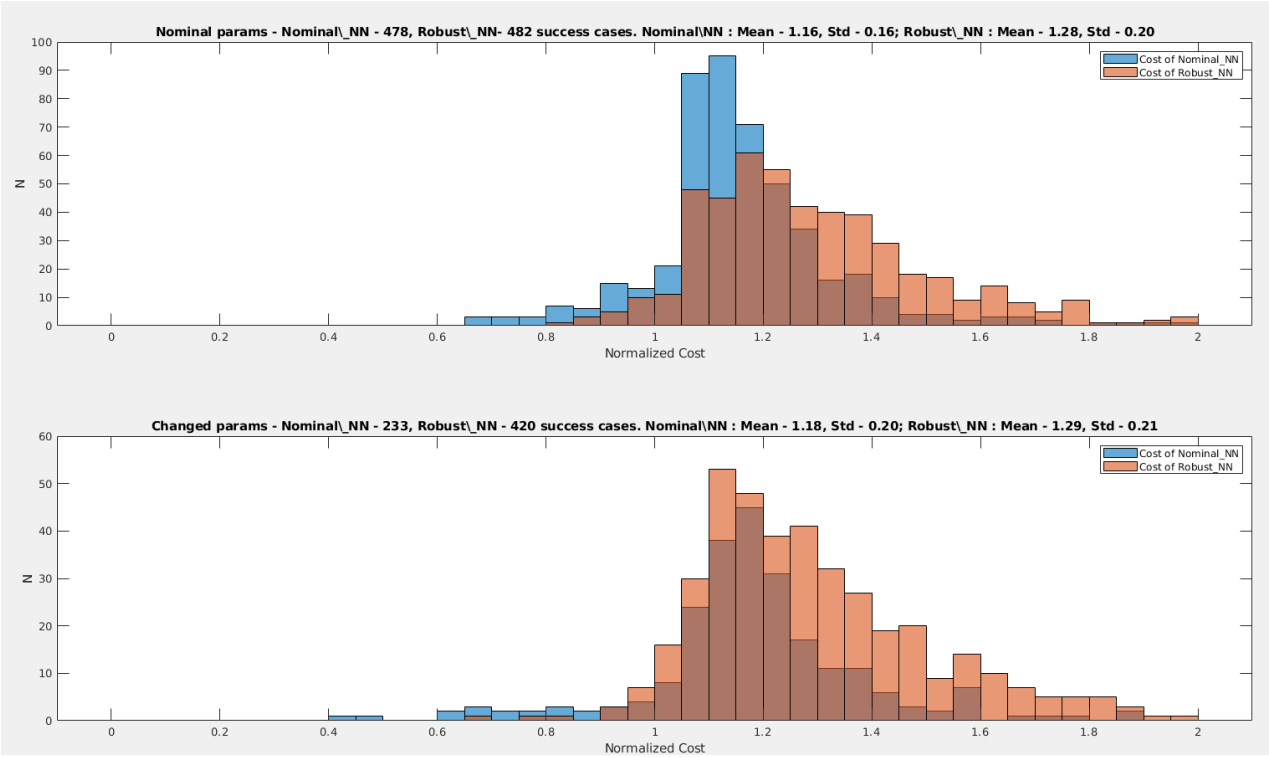

Figure 4-5: The simulation results histogram of costs of nominal NN and robust NN.

NNs is approximately 0.2.

From the Figures from 4-6 till 4-9 it can be seen successful and failed points around initial positions at x and y axes. The figures demonstrate that the robust NN have more successful points.

4.3.2 Experiments

In order to support the simulation results, we conducted the experiments by attaching the different masses to our setup, namely 100 grams, 200 grams, 300 grams. In the

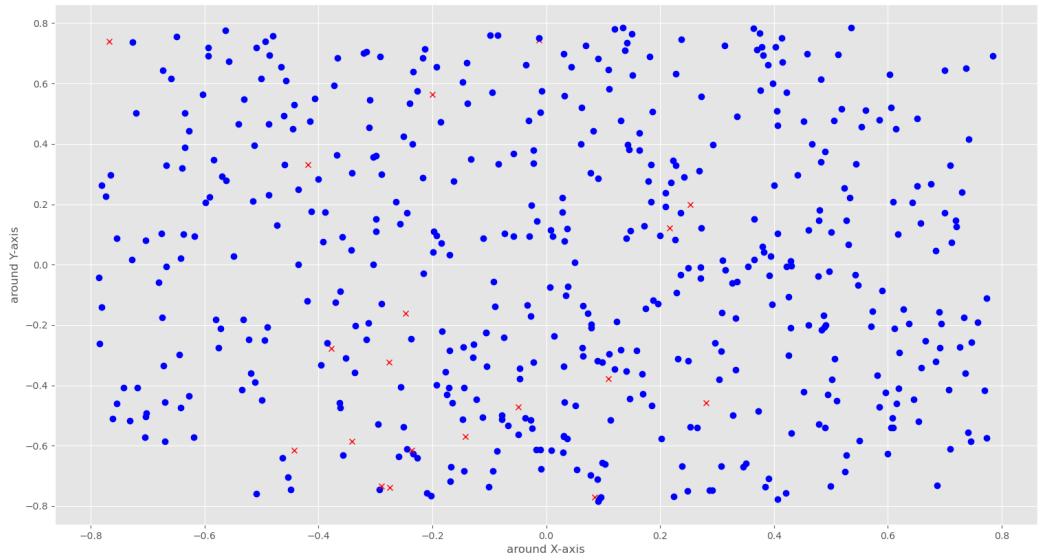

Figure 4-6: The simulation results of nominal NN results for the nominal system parameter (blue successful, red failed).

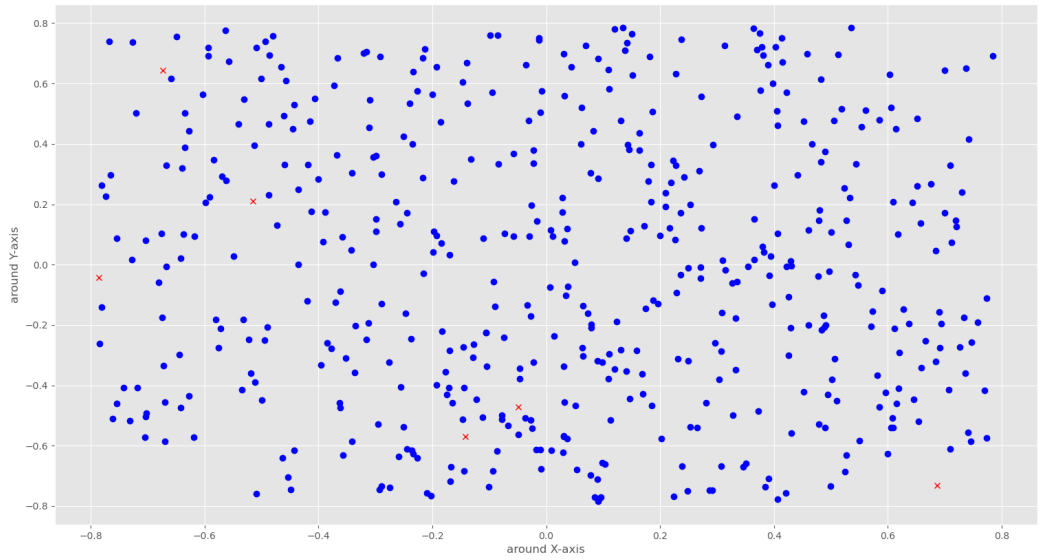

Figure 4-7: The simulation results of robust NN results for the nominal system parameter (blue successful, red failed).

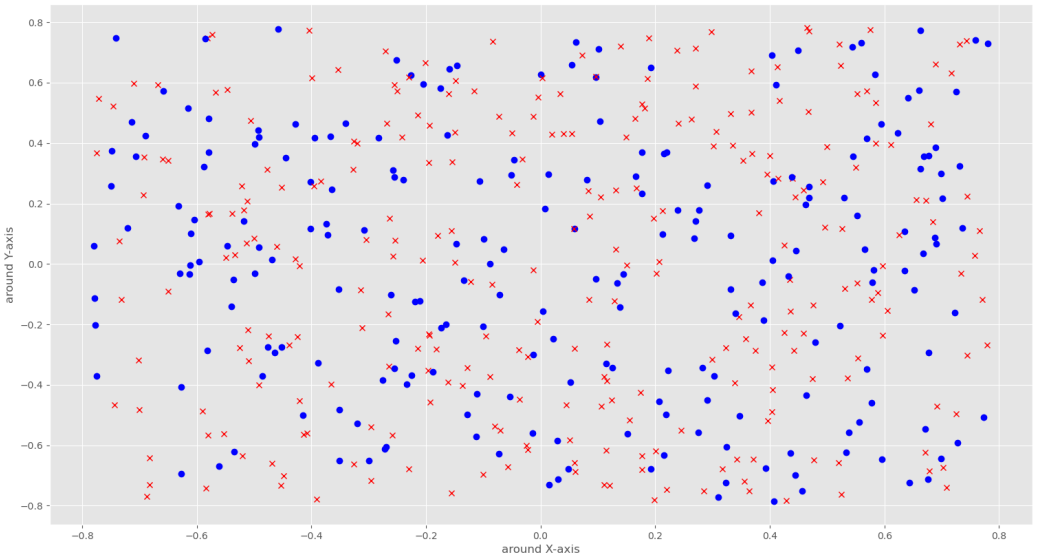

Figure 4-8: The simulation results of nominal NN results for the changed system parameter (blue successful, red failed).

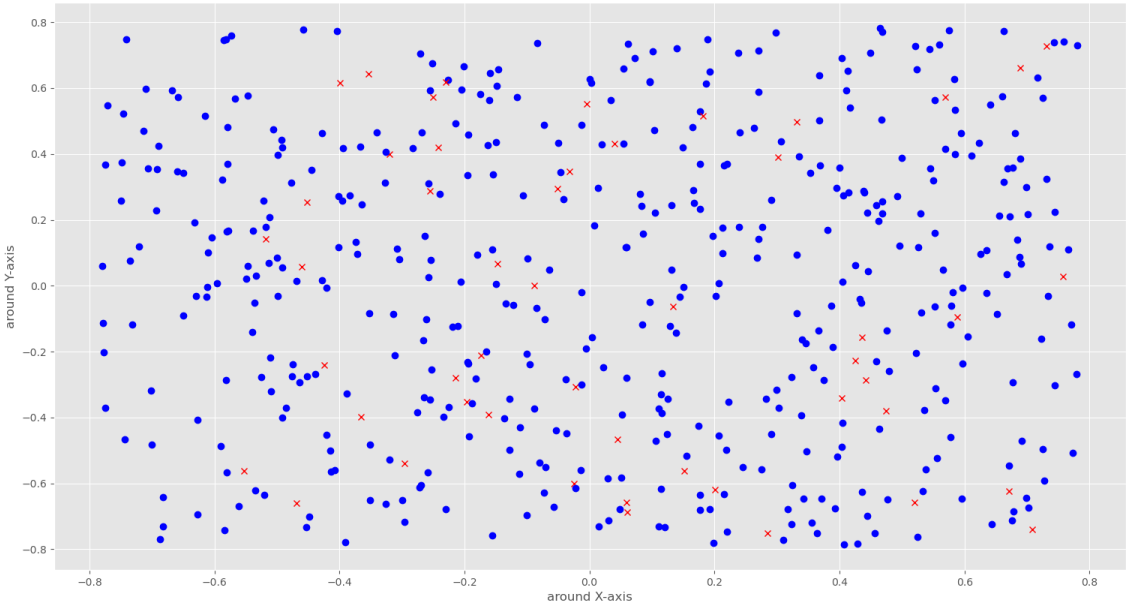

Figure 4-9: The simulation results of robust NN results for the changed system parameter (blue successful, red failed).

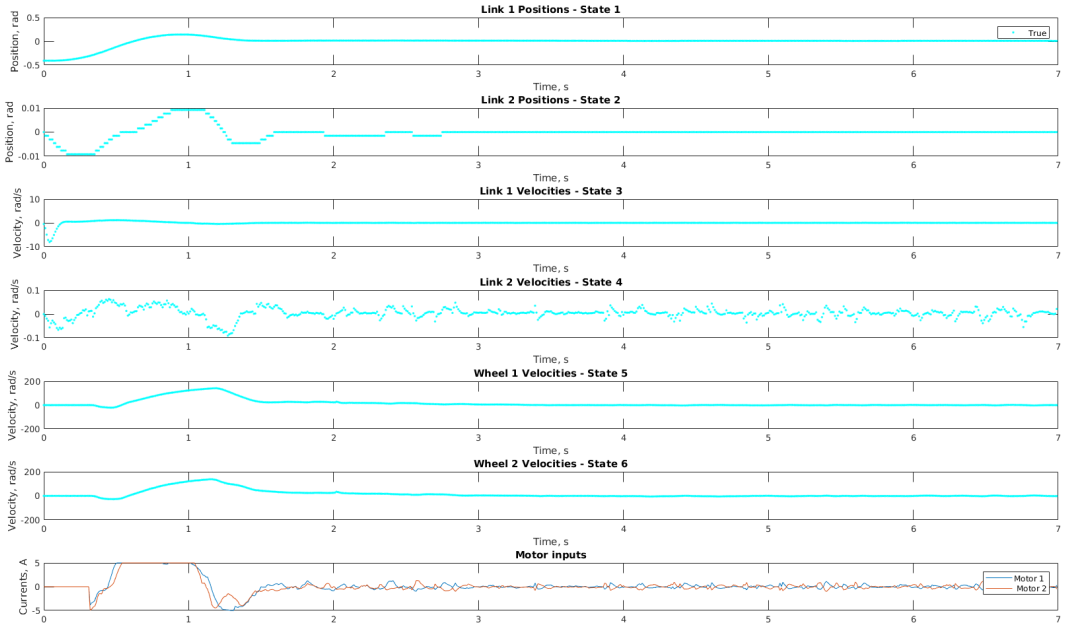

Figure 4-10: The experimental results of the trajectories of the nominal NN with no mass attachments.

	Mass (grams)			
	0	100	200	300
Nominal NN	10	6	4	0
Robust NN	10	9	5	0

Table 4.5: The experimental results of the success rates of Nominal NN and Robust NN for 10 different trajectories.

experiment, the setup is launched from ten different random initial positions, and is investigated stabilization behaviour. The stabilization plot of the NNs are shown in figures from 4-10 till 4-17. As can be seen from the plots, when the mass attached to the setup increases, the systems starts to converge on the angle deviated from the zero and considered as failing experiment. The experimental results for successful cases for both NNs are shown on the Table4.5. Then it is provided the average cost for the nominal NN and robust NN on the Table 4.6.

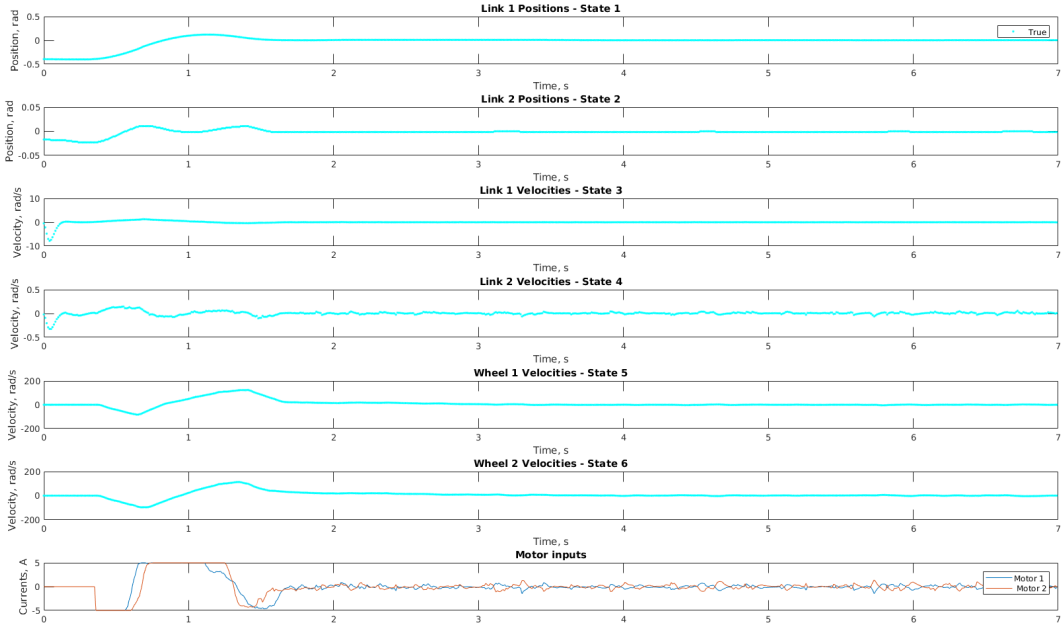

Figure 4-11: The experimental results of the trajectories of the nominal NN with 100 grams attached on pendulum.

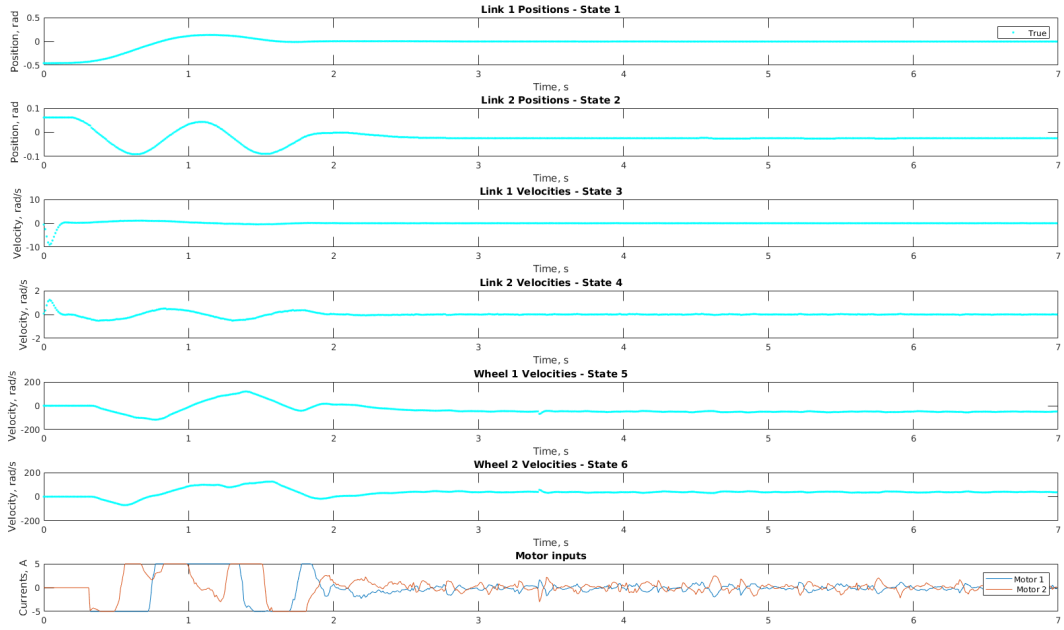

Figure 4-12: The experimental results of the trajectories of the nominal NN with 200 grams attached on pendulum.

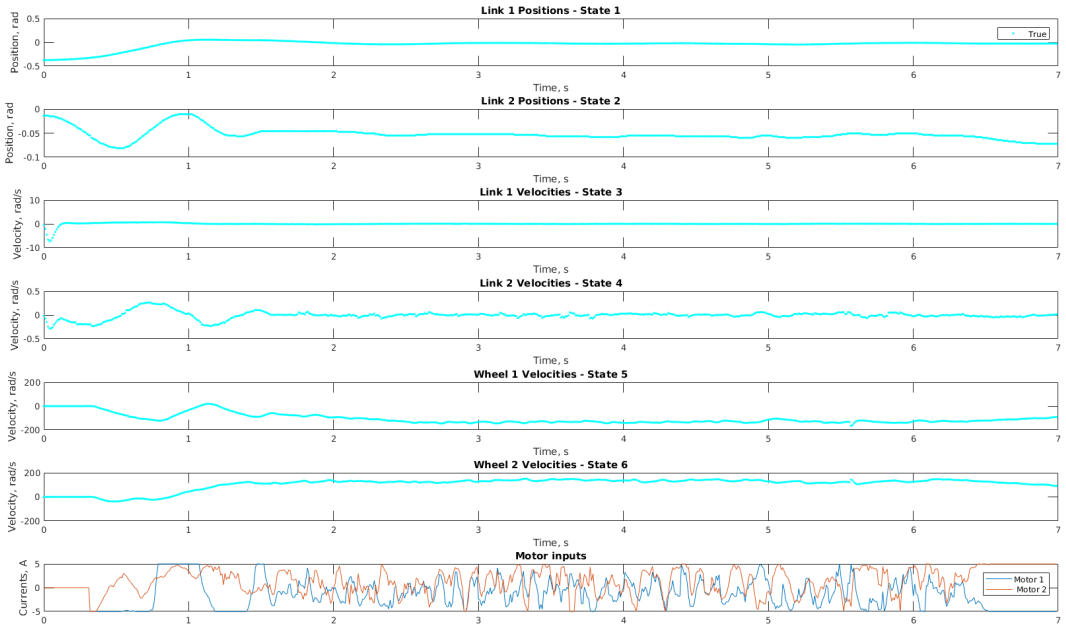

Figure 4-13: The experimental results of the trajectories of the nominal NN with 300 grams attached on pendulum.

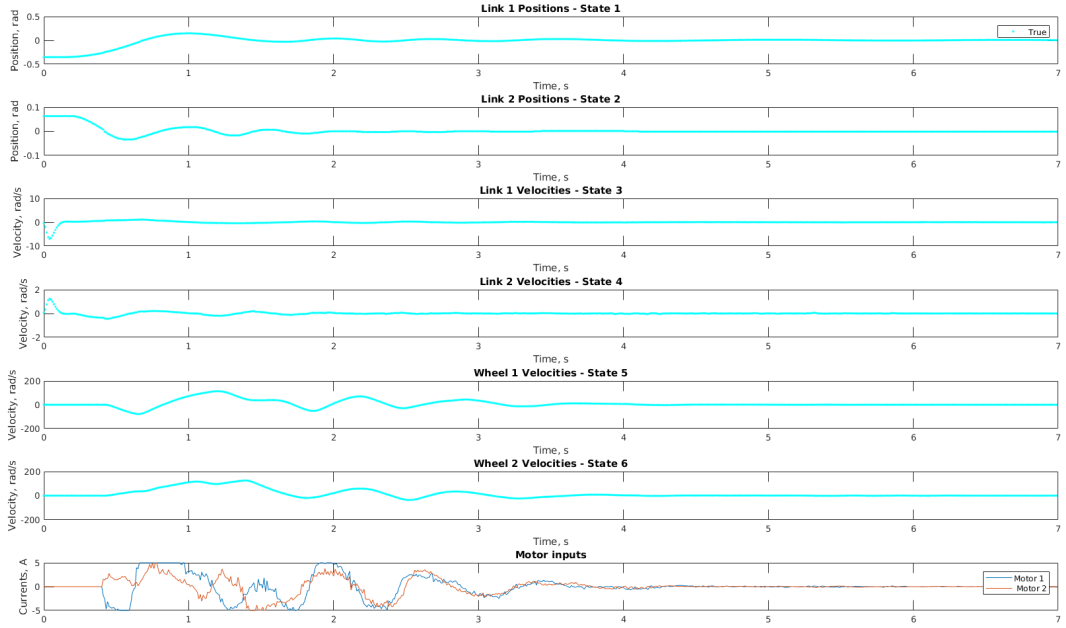

Figure 4-14: The experimental results of the trajectories of the robust NN with no mass attachments.

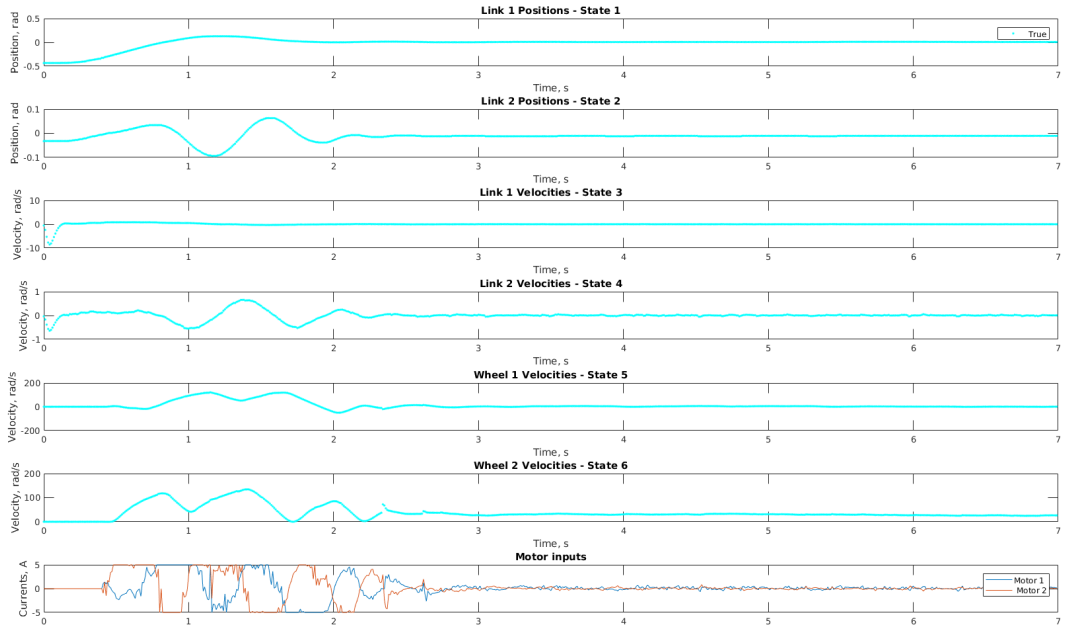

Figure 4-15: The experimental results of the trajectories of the robust NN with 100 grams attached on pendulum.

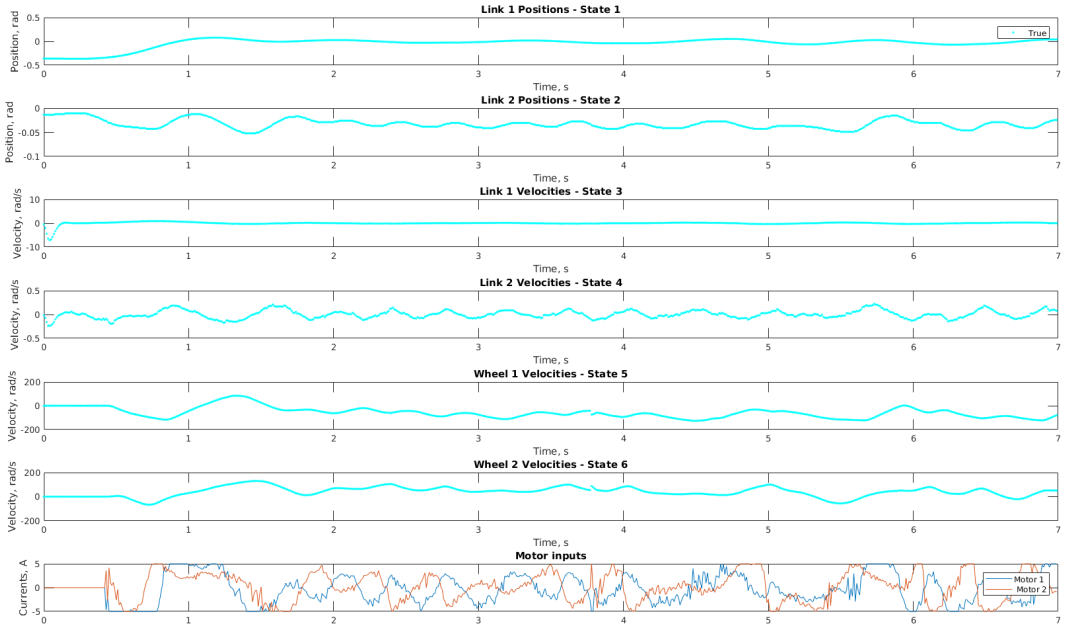

Figure 4-16: The experimental results of the trajectories of the robust NN with 200 grams attached on pendulum.

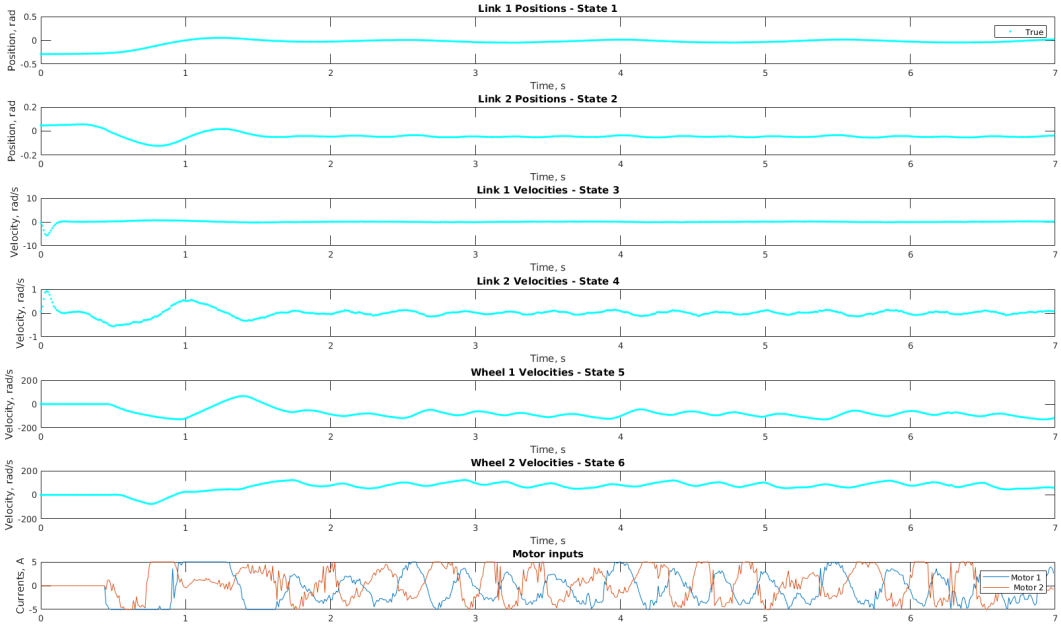

Figure 4-17: The experimental results of the trajectories of the robust NN with 300 grams attached on pendulum.

	Mass (grams)			
	0	100	200	300
Nominal NN (x1e-4)	21.65	26.27	28.64	29.46
Robust NN (x1e-4)	21.99	22.83	26.31	27.32

Table 4.6: The experimental results of average cost of Nominal NN and Robust NN for 10 different trajectories.

Chapter 5

Conclusion

In this thesis work, it has represented stabilization of the dual-axis reaction wheel inverted pendulum. At the beginning, it is given the brief information why the usage of this inverted pendulum is advantageous. Afterwards, the physical setup, which will be used as a real world experiment, was introduced by detail. In addition, its system dynamics and identification were described. Due to the recent prevalence of Neural Networks in the control theory field, their utilization for this kind of stabilization task is more than suitable. This work presented the comparison of NNs with OCP algorithms. Furthermore, it is also compared the behavior of the system due to some parameter changes. It introduced two different NNs for checking the robust control of the system, one for nominal case and another for parameter uncertainty situation. Since the systems and the mathematical models can not be ideal, and by time they lose their previous properties, it is necessary to consider the system operation taking into account some margin. For this purpose, the robust control NN was employed. As the results demonstrate, this parameter uncertainty consideration might be the case for application. In most of the cases with parameter change nominal NN fails, while robust NN can find the solution. In the future, it is regarded that reaction wheels will be combined with VSA, and in further implementation of this reaction wheel inverted pendulum as an integrated part of VSAs, these algorithms should find the place to be.

Bibliography

[1] Daulet Baimukashev, Nazerke Sandibay, Bexultan Rakhim, Huseyin Atakan Varol, and Matteo Rubagotti. Deep learning-based approximate optimal control of a reaction-wheel-actuated spherical inverted pendulum. In *2020 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pages 1322–1328. IEEE, 2020.

[2] Heiner Lasi, Peter Fettke, Hans-Georg Kemper, Thomas Feld, and Michael Hoffmann. Industry 4.0. *Business & Information systems engineering*, 6(4):239–242, 2014.

[3] Mehmet Baygin, Hasan Yetis, Mehmet Karakose, and Erhan Akin. An effect analysis of industry 4.0 to higher education. In *2016 15th international conference on information technology based higher education and training (ITHET)*, pages 1–4. IEEE, 2016.

[4] Mario Hermann, Tobias Pentek, and Boris Otto. Design principles for industrie 4.0 scenarios. In *2016 49th Hawaii international conference on system sciences (HICSS)*, pages 3928–3937. IEEE, 2016.

[5] Yang Lu. Industry 4.0: A survey on technologies, applications and open research issues. *Journal of industrial information integration*, 6:1–10, 2017.

[6] Ray Y Zhong, Xun Xu, Eberhard Klotz, and Stephen T Newman. Intelligent manufacturing in the context of industry 4.0: a review. *Engineering*, 3(5):616–630, 2017.

[7] Jorge Posada, Carlos Toro, Iñigo Barandiaran, David Oyarzun, Didier Stricker, Raffaele De Amicis, Eduardo B Pinto, Peter Eisert, Jürgen Döllner, and Ivan Vallarino. Visual computing as a key enabling technology for industrie 4.0 and industrial internet. *IEEE computer graphics and applications*, 35(2):26–40, 2015.

[8] Ana C Pereira and Fernando Romero. A review of the meanings and the implications of the industry 4.0 concept. *Procedia Manufacturing*, 13:1206–1214, 2017.

[9] Morteza Ghobakhloo. The future of manufacturing industry: a strategic roadmap toward industry 4.0. *Journal of Manufacturing Technology Management*, 2018.

[10] Sachin S Kamble, Angappa Gunasekaran, and Shradha A Gawankar. Sustainable industry 4.0 framework: A systematic literature review identifying the current trends and future perspectives. *Process Safety and Environmental Protection*, 117:408–425, 2018.

[11] Antonio Bicchi, Giovanni Tonietti, Michele Bavaro, and Marco Piccigallo. Variable stiffness actuators for fast and safe motion control. In *Robotics research. The eleventh international symposium*, pages 527–536. Springer, 2005.

[12] Sami Haddadin. *Towards safe robots: approaching Asimov's 1st law*, volume 90. Springer, 2013.

[13] Bram Vanderborght, Alin Albu-Schäffer, Antonio Bicchi, Etienne Burdet, Darwin G Caldwell, Raffaella Carloni, MG Catalano, Oliver Eiberger, Werner Friedl, Ganesh Ganesh, et al. Variable impedance actuators: A review. *Robotics and autonomous systems*, 61(12):1601–1614, 2013.

[14] Altay Zhakatayev, Matteo Rubagotti, and Huseyin Atakan Varol. Closed-loop control of variable stiffness actuated robots via nonlinear model predictive control. *IEEE Access*, 3:235–248, 2015.

[15] Almaskhan Baimyshev, Altay Zhakatayev, and Huseyin Atakan Varol. Augmenting variable stiffness actuation using reaction wheels. *IEEE Access*, 4:4618–4628, 2016.

[16] Mark W Spong, Peter Corke, and Rogelio Lozano. Nonlinear control of the reaction wheel pendulum. *Automatica*, 37(11):1845–1851, 2001.

[17] Daniel J Block, Karl J Åström, and Mark W Spong. The reaction wheel pendulum. *Synthesis Lectures on Control and mechatronics*, 1(1):1–105, 2007.

[18] Richard Vinter. *Optimal control*. Springer Science & Business Media, 2010.

[19] Yutao Chen, Mattia Bruschetta, Enrico Picotti, and Alessandro Beghi. Matmpc-a matlab based toolbox for real-time nonlinear model predictive control. In *2019 18th European Control Conference (ECC)*, pages 3365–3370. IEEE, 2019.

[20] Kemin Zhou and John Comstock Doyle. *Essentials of robust control*, volume 104. Prentice hall Upper Saddle River, NJ, 1998.

[21] Stephen Boyd, Laurent El Ghaoui, Eric Feron, and Venkataramanan Balakrishnan. *Linear matrix inequalities in system and control theory*. SIAM, 1994.

[22] Priya L Donti, Melrose Roderick, Mahyar Fazlyab, and J Zico Kolter. Enforcing robust control guarantees within neural network policies. *arXiv preprint arXiv:2011.08105*, 2020.

[23] Felix A Gers, Jürgen Schmidhuber, and Fred Cummins. Learning to forget: Continual prediction with lstm. 1999.