

ABSTRACT

BERNSTEIN, AMANDA SUE. Modeling and Control: Applications to a Double Inverted Pendulum and Radio Frequency Interference. (Under the direction of Hien Tran).

In this work we consider methods for the modeling and control of physical systems. In particular, we examine a double inverted pendulum (DIP) system and the impact of radio frequency interference (RFI) on satellite operations (SATOPS) systems.

The DIP is a classic example of a nonlinear, multivariable, unstable system. For the stabilization problem, a controller is designed that stabilizes the pendulum system in the unstable equilibrium position. Physically, this is the position where the two pendulum rods are aligned in a vertical position. The mathematical model of the DIP is derived using Lagrange's energy method, and parameter estimation is performed to optimize the friction parameters. The resulting nonlinear system is linearized about its unstable equilibrium state, and a linear quadratic regulator (LQR) control and a power series based nonlinear control are each implemented in real-time in order to stabilize the DIP system. Since the real-time system only measures the three position states of the system, the three velocity states must also be estimated in order to apply the state feedback controls. Thus, three estimation methods are presented: a low pass derivative filter, a Kalman observer, and a nonlinear observer. Only the low pass derivative filter is successful in estimating the velocity states and is thus used to obtain the real-time results.

A series of analytic models to detect and predict RFI within a SATOPS system is presented which include the behavior of the carrier synchronization loops when evaluating the performance of the system. Models for the carrier acquisition time, the carrier tracking errors, and the bit error rate performance for a phase locked loop in the absence of RFI as well as in the presence of continuous wave or wideband RFI are presented. These models are then implemented in MATLAB and tested under actual operating conditions for purposes of validation and verification.

© Copyright 2018 by Amanda Sue Bernstein

All Rights Reserved

Modeling and Control: Applications to a Double Inverted Pendulum and Radio Frequency
Interference

by
Amanda Sue Bernstein

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Applied Mathematics

Raleigh, North Carolina

2018

APPROVED BY:

Stephen Campbell

Negash Medhin

Tien Khai Nguyen

Hien Tran
Chair of Advisory Committee

DEDICATION

To my family for all your support, and especially to Andrew.

BIOGRAPHY

Amanda (Coons) Bernstein was born and raised in Albion, NY, a small town halfway between Buffalo and Rochester. After graduating from high school, she continued her education at Clarkson University in Potsdam, NY, earning a B.S. in Mathematics and Mechanical Engineering in May 2012. She then continued to North Carolina State University for her graduate studies in Applied Mathematics, earning a M.S in Applied Mathematics in May 2015 and a Ph.D. with the successful completion of this dissertation.

ACKNOWLEDGEMENTS

I would like to thank my advisor, Dr. Hien Tran, for all his support and guidance as I worked on this research. Thank you also to our collaborator, Dr. Tien Nguyen, for his contributions to this research. I also thank my committee members, Dr. Stephen Campbell, Dr. Negash Medhin, and Dr. Khai Nguyen, for their time and feedback.

Thank you to the NCSU Mathematics Department and the CRSC for their financial support throughout my time here at NC State.

Finally, thank you to all my friends and family who have supported me, especially Luke, Cleveland, Steven, Mary, and Chris. And most importantly, thank you to Andrew, without whom I never would have made it.

TABLE OF CONTENTS

LIST OF TABLES	vii
LIST OF FIGURES	viii
Chapter 1 Introduction	1
1.1 The Double Inverted Pendulum	1
1.2 Radio Frequency Interference Modeling	4
1.3 Thesis Outline	5
Chapter 2 Modeling of an Inverted Double Pendulum System	6
2.1 Frame of Reference	6
2.2 Equations of Motion	7
2.2.1 Potential Energy	9
2.2.2 Kinetic Energy	9
2.2.3 Lagrange's Equations	11
2.2.4 Converting to Voltage Input	14
2.3 Model Calibration	15
Chapter 3 Stabilization Control of an Inverted Double Pendulum System	21
3.1 Overview of Existing Control Methods	21
3.2 Problem Statement	23
3.3 LQR Control	24
3.3.1 An Application Example	26
3.4 Nonlinear Control	28
3.4.1 Power Series Based Controller	29
3.4.2 An Application Example	30
3.5 Simulation Results	31
3.6 Experimental Results	33
3.6.1 Experimental Apparatus	33
3.6.2 Experimental Procedure	33
3.6.3 Results for the LQR Controller	35
3.6.4 Results for the Power Series Controller	38
Chapter 4 State Observation for an Inverted Double Pendulum System	43
4.1 Low Pass Derivative Filter	44
4.2 Kalman Observer	45
4.2.1 An Application Example	49
4.2.2 Application to the DIP System	53
4.3 Nonlinear Observer	53
4.3.1 An Application Example	57
4.3.2 Application to the DIP System	60
4.3.2.1 The Complex Step Method	60
4.3.2.2 Applying the Complex Step Method to the DIP	63

4.4	Summary of the Observer Used for the DIP System	64
Chapter 5	Modeling of Radio Frequency Interference for SATOPS Applications	65
5.1	Background	65
5.1.1	The RFI Detection Problem	67
5.1.2	The RFI Prediction Problem	67
5.1.3	Existing RFI Tools	69
5.2	RFI Analytical Models for Acquisition Time	69
5.2.1	Carrier Frequency Acquisition Mode	69
5.2.2	Carrier Phase Acquisition Mode	70
5.2.2.1	Sync Word Technique	71
5.2.2.2	Phase Sweeping Technique	71
5.2.3	PLL Acquisition in the Presence of CW RFI	72
5.2.4	PLL Acquisition in the Presence of WB RFI	73
5.2.5	Simulation Results for Verification and Validation	74
5.3	Analytical Models for RFI Detection	77
5.3.1	The PLL Lock Detector	77
5.3.2	PLL Tracking Jitter in the Absence of RFI	80
5.3.3	PLL Tracking Jitter in the Presence of CW RFI	81
5.3.4	PLL Tracking Jitter in the Presence of WB RFI	82
5.3.5	Simulation Results for Verification and Validation	82
Chapter 6	Conclusion	86
6.1	The Double Inverted Pendulum System	86
6.2	Radio Frequency Interference Analytic Models	87
REFERENCES	88
APPENDIX	92
Appendix A	DIP Model Parameters	93
A.1	Nomenclature	93
A.2	Parameter Values	96

LIST OF TABLES

Table 2.1	Optimized friction parameter values found using fminsearch.	18
Table 3.1	Maximum and average values for the states and control voltages using the LQR controller with various values of Q and R . (x_c [mm], α [degrees], θ [degrees], V_m [V]).	38
Table 3.2	Maximum and average values for the states and control voltages using the power series based controller with various values of Q and R . (x_c [mm], α [degrees], θ [degrees], V_m [V]).	41
Table 5.1	Parameters for the simulations of PLL acquisition performance: Carrier acquisition time.	74
Table 5.2	Parameters for the simulations of PLL acquisition performance: Tracking error and BER.	83
Table A.1	Nomenclature for the DIP system model.	93
Table A.2	Model parameters for the DIP system as provided by Quanser.	96

LIST OF FIGURES

Figure 1.1	Some applications of the double inverted pendulum: (a) a robot arm and (b) a gymnast.	2
Figure 1.2	The double inverted pendulum mounted on a cart.	3
Figure 1.3	A typical satellite control network (SCN).	4
Figure 2.1	Free body diagram of the DIP system mounted on a cart.	7
Figure 2.2	Electrical schematic of a standard DC motor.	14
Figure 2.3	Input voltage for use in model calibration.	16
Figure 2.4	Experimental and model simulation results using the parameter values provided by Quanser.	17
Figure 2.5	Experimental and model simulation results using the optimized friction parameter values.	19
Figure 2.6	Residuals between the experimental and model simulation results using the optimized friction parameter values.	20
Figure 3.1	LQR control applied to example problem: (a) state trajectories and (b) control effort.	28
Figure 3.2	Power series based control applied to example problem: (a) state trajectories and (b) control effort.	31
Figure 3.3	Simulation results for the state trajectories using the LQR controller with $Q = \text{diag}(30, 350, 100, 0, 0, 0)$ and $R = 0.1$	32
Figure 3.4	Control effort for the simulation using the LQR controller with $Q = \text{diag}(30, 350, 100, 0, 0, 0)$ and $R = 0.1$	33
Figure 3.5	A diagram of the experimental setup for the DIP system.	34
Figure 3.6	Simulink block diagram governing the real-time implementation of the DIP system.	34
Figure 3.7	Real-time experimental results for the state trajectories using the LQR controller with $Q = \text{diag}(30, 350, 100, 0, 0, 0)$ and $R = 0.1$	36
Figure 3.8	Control effort for the real-time experiment using the LQR controller with $Q = \text{diag}(30, 350, 100, 0, 0, 0)$ and $R = 0.1$	37
Figure 3.9	Real-time experimental results for the state trajectories using the power series based controller with $Q = \text{diag}(30, 350, 100, 0, 0, 0)$ and $R = 0.1$	39
Figure 3.10	Control effort for the real-time experiment using the power series based controller with $Q = \text{diag}(30, 350, 100, 0, 0, 0)$ and $R = 0.1$	40
Figure 4.1	Diagrammatic representation of the state observer problem.	43
Figure 4.2	The Kalman observer applied to the example problem using the power series based control: (a) x_1 and (b) x_2	50
Figure 4.3	Absolute error between the observer and system state values for the example problem using the Kalman observer.	51
Figure 4.4	The Kalman observer applied to the example problem including white noise: (a) x_1 and (b) x_2	52

Figure 4.5	Absolute error between the observer and system state values for the example problem using the Kalman observer including white noise.	52
Figure 4.6	Simulation results for the states of the observer and the linearized DIP system using the Kalman observer and the power series based controller. .	54
Figure 4.7	Control effort for the simulation using the Kalman observer and the power series based controller.	55
Figure 4.8	A one-link robot arm.	58
Figure 4.9	System and observer performance for the one-link robot arm.	61
Figure 5.1	Typical RFI scenarios involving friendly sources.	66
Figure 5.2	Interference bandwidth for evaluating potential RFI events and the corresponding CCI and ACI scenarios.	66
Figure 5.3	A general block diagram of a PLL.	68
Figure 5.4	Carrier acquisition times vs. loop SNR without RFI and with CW RFI at $\text{ISR} = -40$ dB.	75
Figure 5.5	Carrier acquisition times vs. loop SNR without RFI and with CW RFI at $\text{ISR} = -25$ dB.	76
Figure 5.6	Carrier acquisition times vs. loop SNR without RFI and with WB RFI at $\text{ISR} = -40$ dB.	77
Figure 5.7	Carrier acquisition times vs. loop SNR without RFI and with WB RFI at $\text{ISR} = -25$ dB.	78
Figure 5.8	A typical PLL lock detector used by SATOPS satellites.	79
Figure 5.9	PLL tracking jitters in the absence of RFI and the presence of CW and WB RFI signals with $\Delta f_{RFI} = 5$ Hz.	83
Figure 5.10	PLL tracking jitters in the absence of RFI and the presence of CW and WB RFI signals with $\Delta f_{RFI} = 10$ Hz.	84
Figure 5.11	BER in the absence of RFI and the presence of CW and WB RFI signals with $\Delta f_{RFI} = 5$ Hz. (b) shows detail on the BER in the presence of CW and WB RFI.	85
Figure 5.12	BER in the absence of RFI and the presence of CW and WB RFI signals with $\Delta f_{RFI} = 10$ Hz. (b) shows detail on the BER in the presence of CW and WB RFI.	85

Chapter 1

Introduction

Mankind has always sought to understand our universe and to control it for our own aims. As mathematicians, one of the ways we seek to understand is by creating models. Whether we are looking to comprehend the ups and downs of the stock market, the spread of pathogens through a population, or how a self-driving car operates, models can give us insight to these systems and allow us to predict what will happen in the future. Furthermore, by including control variables in our models, we can change the outputs to suit our desires. Models can take many forms, but most commonly we use systems of equations to represent them as well as to describe the ways we want to control them. In this thesis, we will examine the models of two different systems, a double inverted pendulum and radio frequency interference within a satellite operations system, as well as ways to control the double inverted pendulum system.

1.1 The Double Inverted Pendulum

An inverted pendulum refers to any pendulum that has its center of mass above its pivot point. A simple example would be a broomstick balanced on the open palm of your hand. Your palm is the pivot point for the pendulum and the mass of the broomstick is above that point. While this is an unstable position for the system, by moving your hand around, you can keep the broomstick balanced upright. Inverted pendulums are standard models of multivariable, nonlinear, unstable systems. Yet they are also controllable, and thus they are commonly used for pedagogy as well as for the introduction of intermediate and advanced control concepts. Inverted pendulums come in a variety of forms including classical inverted pendulum systems that are mounted on a base which moves in a linear manner and rotary inverted pendulum systems where the base doesn't move back and forth but rather applies a torque directly to the pendulum rod. Pendulum systems can also have differing numbers of pendulum rods attached to their base including single, double, triple, and quadruple inverted pendulum systems.

For this work, we will consider the double inverted pendulum (DIP) system mounted on a cart. This is an extension of the single inverted pendulum with an additional pendulum rod attached via a hinge. Imagine you break your broomstick into two pieces, reattach them using a hinge, and now balance that on your open palm. DIP systems can be used as models for a number of different applications, including self-stabilizing robots and robotic limbs [34, 36], human posture, and gymnast motion [38]. Currently, the DIP is used as a model for human standing where the pivot of the pendulum is the ankle joint and the hinge is the hip joint with the legs and the torso comprising the two pendulum rods [35, 39]. Some of these applications are shown in Figure 1.1.

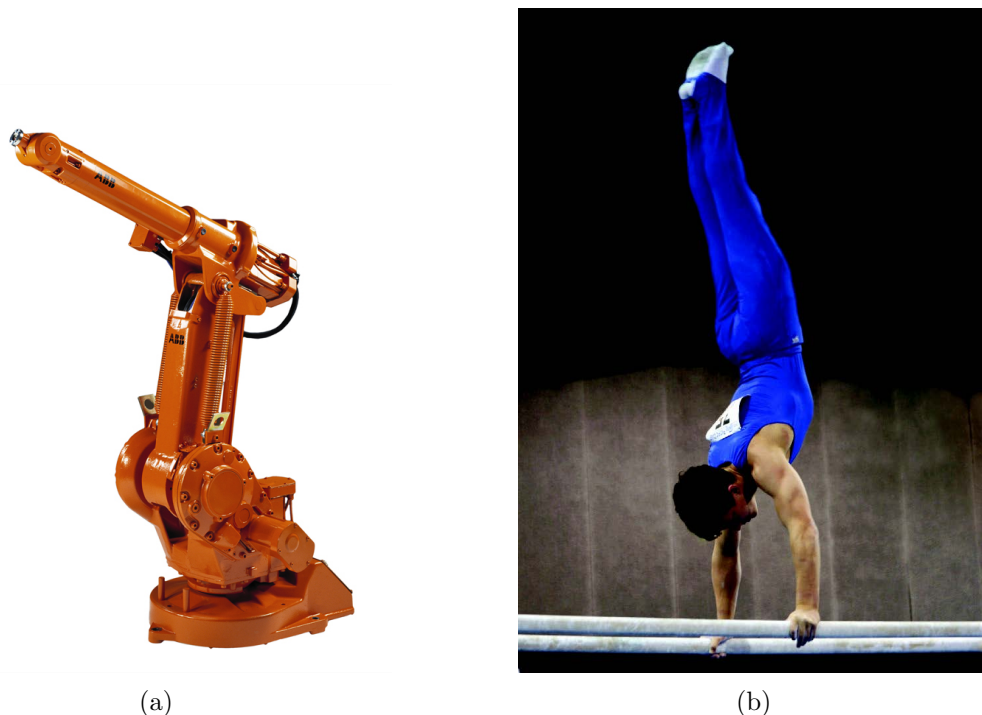


Figure 1.1: Some applications of the double inverted pendulum: (a) a robot arm and (b) a gymnast.

The double inverted pendulum is commonly used as a benchmark problem in nonlinear control theory. There are two types of control synthesis for an inverted pendulum system: swing-up and stabilization control. Both problems have the goal of balancing the pendulum in its unstable vertical position, but swing-up control starts the pendulum in the stable downward position while stabilization just refers to the balancing control necessary to maintain the unstable position. While many control methods have been applied to the DIP system, the vast majority use only numerical simulations to test the feasibility of their proposed controllers and do not

provide real-time experimental implementation on a physical system.

For our work, we use an apparatus of the DIP system which was provided by Quanser Consulting, Inc. (119 Spy Court, Markham, Ontario, L3R 5H6, Canada). This system is depicted in Figure 1.2. The DIP system consists of two aluminum rods, one 12 inches long and one 7

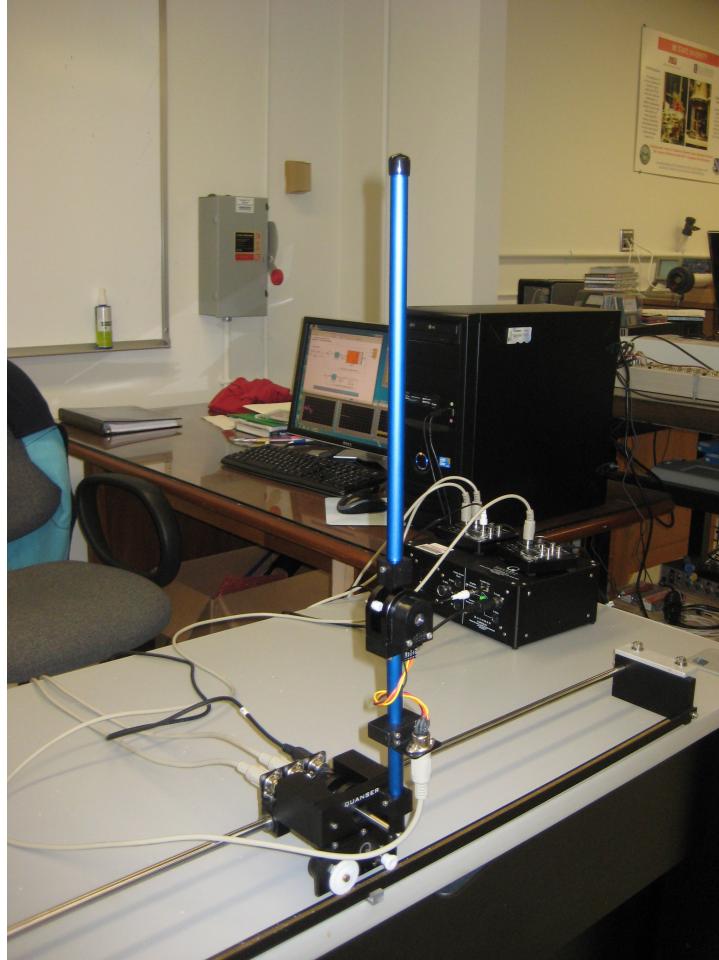


Figure 1.2: The double inverted pendulum mounted on a cart.

inches long. These rods are mounted on a linear servo base unit (IP02) consisting of a cart driven by a DC motor and two encoders. One encoder measures the position of the cart while the other measures the angular position of the short (lower) pendulum rod. The longer (upper) pendulum rod is attached via a hinge which also has another encoder to measure its angular position. Based on these measurements, we can compute a voltage to move the cart back and forth and balance the pendulums in the upright, vertical position. The control methods used

to compute this voltage will be one of the focuses of this work.

1.2 Radio Frequency Interference Modeling

Satellite operations (SATOPS) services are provided by a satellite control network (SCN) and include capabilities to provide tracking, telemetry, and command services [26]. Figure 1.3 shows an example of what a satellite control network may consist of. The ground tracking station

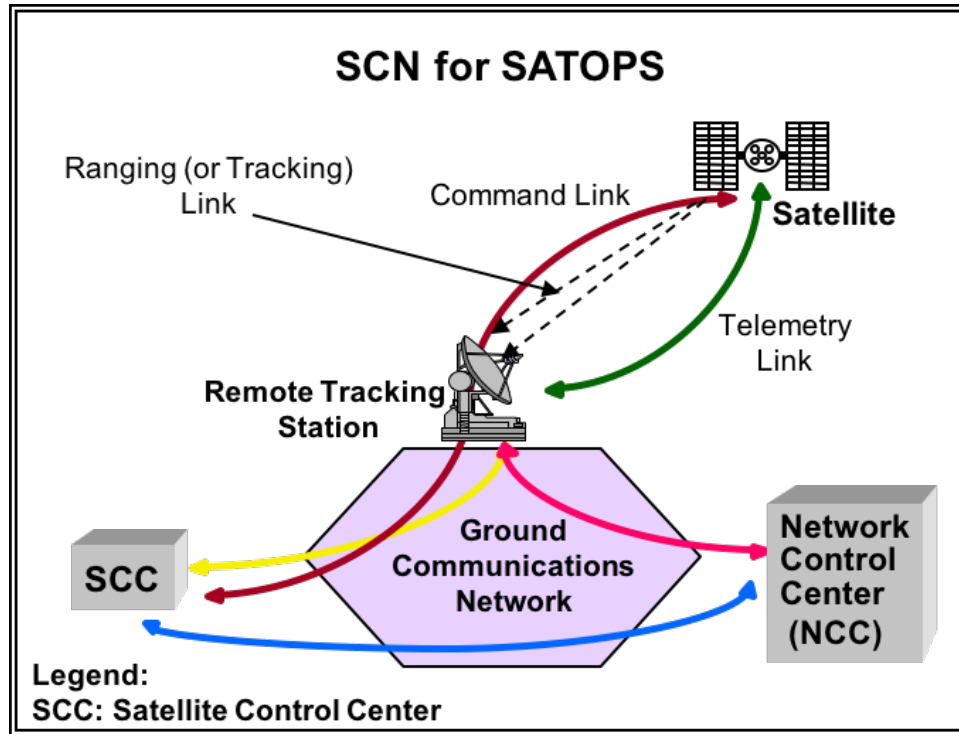


Figure 1.3: A typical satellite control network (SCN).

provides communication between the different components of the system, particularly between the satellites and the control centers, and is typically located in a remote location where there is a clear view of the sky. The satellite control center (SCC) provides support for the operation, control, and maintenance of the satellites and executes the telemetry, tracking, and command operations. The network control center (NCC) provides support for scheduling and configuring the satellite network as well as the operations of all the other system components. The ground communications network refers to the systems which support data transfers between the three on the ground components: the SCC, NCC, and the ground tracking stations.

SATOPS typically operate in L-band (1–2 GHz) or S-band (2–4 GHz) frequencies where there are many potential sources for radio frequency interference (RFI) [25, 28, 29]. RFI occurs when the victim receiver simultaneously collects power from its desired signal as well as other undesired interfering signals. Depending on the power of the interfering signals, this can degrade the system performance and could result in data blockages. These frequency bands are shared by many systems, ranging from the Global Positioning System (GPS) to wireless mobile communications and network systems such as WiFi. Therefore, it is important to be able to analyze the potential for RFI to occur, especially as more users want to operate in these bandwidths.

Many tools exist to perform this analysis, and these have been summarized in [26]. However, most of these tools do not consider the time factor of the interference when calculating the bit or carrier signal-to-noise ratio degradation. Those that do typically accomplish this by including the SATOPS schedules and the interference protection criteria (IPC) as recommended by the National Telecommunication and Information Administration (NTIA) and International Telecommunication Union (ITU) to identify RFI caused blockages, but they assume the RFI would increase the noise floors of the receiver without considering the effects of the RFI on the synchronization loops used by the receivers.

In this research, we consider new models to estimate the potential for RFI to occur including the evaluation of the interfering time factor applied at the level of the synchronization loops. We focus on SATOPS using USB waveforms which is of interest to the U.S. Air Force, and we provide verification of the RFI models using MATLAB.

1.3 Thesis Outline

The remainder of this thesis is structured as follows. In Chapter 2, we derive the full nonlinear model for the DIP system. We then verify the model performance and improve it by optimizing a selection of parameters. In Chapter 3, we discuss the stabilization of the DIP system in the unstable, vertical position. We accomplish this via two different control methods and compare the performance of the real-time system. In Chapter 4, we consider different observers for the DIP system to determine the velocity states which are not directly measured but which are necessary for computing the state feedback controllers. In Chapter 5, we consider models for the analysis of RFI and provide verification and validation. In Chapter 6, we summarize our conclusions and provide thoughts on future directions for this research.

Chapter 2

Modeling of an Inverted Double Pendulum System

In this chapter, we will derive the model for the double inverted pendulum mounted on a cart using Lagrange’s energy method. We will then verify the model performance relative to our experimental apparatus. The following model derivation is based on the single inverted pendulum model presented by Quanser in [32] and has been presented in [6]. This derivation follows the standard procedure to derive the model used in the literature as seen, for example, in [7, 9, 13, 15, 18, 33, 43], using Lagrange’s energy method. Our model differs from those presented in the literature because our state variables are measured relative to a different reference position. In the literature, the pendulum angles are both measured relative to the vertical position, whereas we measure the upper pendulum angle θ relative to the lower pendulum angle α . We formulate our model this way because that is how we measure the angles in our real-time system. This results in a more complicated model, but one which is easier to use for the real-time experimental implementation.

2.1 Frame of Reference

Figure 2.1 shows the free body diagram of the DIP system mounted on a cart. The system consists of two aluminum rods connected to each other by a hinge with the lower rod connected to the motorized cart. Quantities related to the lower rod are denoted by the number 1, and those related to the upper rod are denoted by the number 2. In our model, we consider the masses of the cart, M_c ; the rods, M_1 and M_2 ; and the hinge, M_h , as being concentrated at their center of gravities. The lengths of the pendulum rods are given by L_1 and L_2 , and the distances from each pendulum’s pivot point to its center of gravity is given by ℓ_1 and ℓ_2 . The corresponding nomenclature for this system is summarized in Appendix A.1.

The position of the cart x_c is zero at the center of its track; the angle α of the lower rod is zero when the rod is pointed perfectly upwards; and the angle θ of the upper rod is zero when it is perfectly aligned with the lower rod. We define the positive sense of the rotation to be counterclockwise and the displacement is positive towards the right when facing the cart.

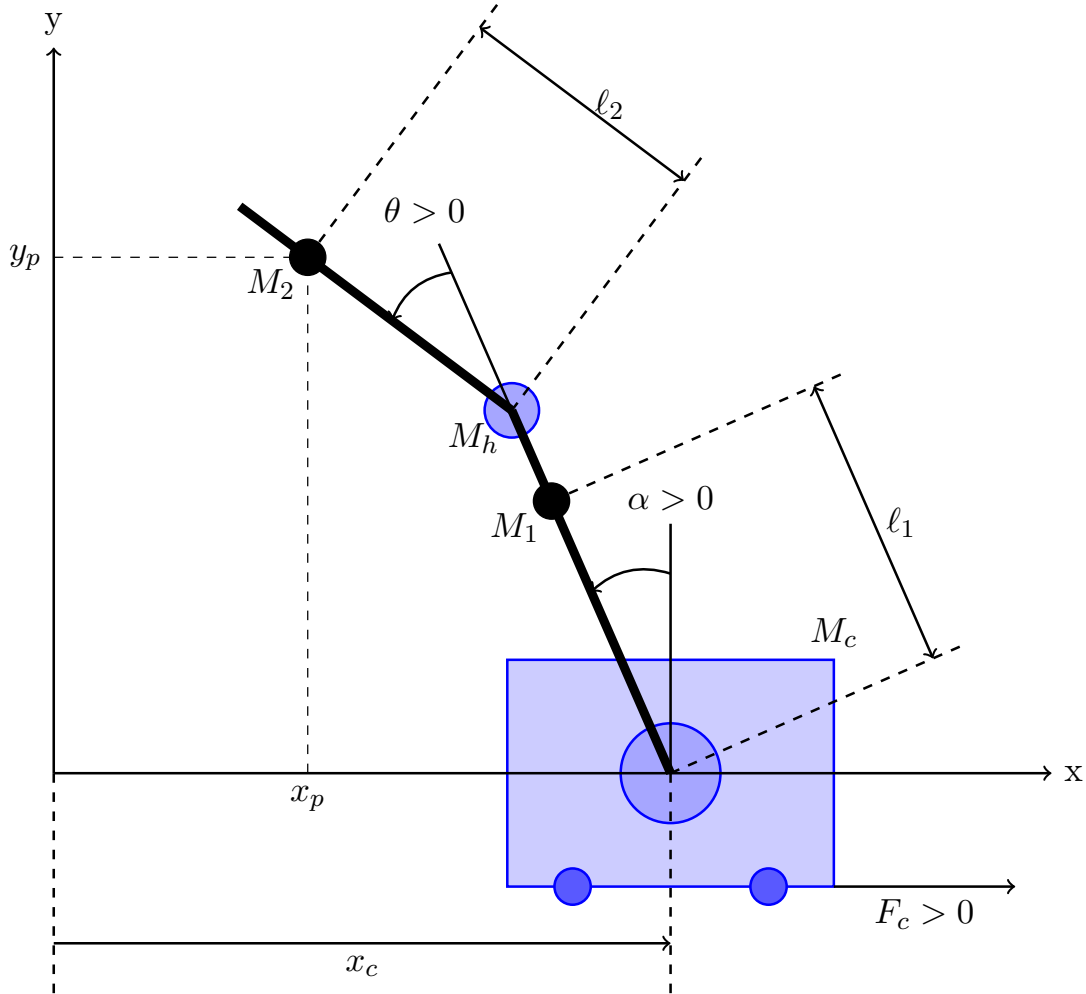


Figure 2.1: Free body diagram of the DIP system mounted on a cart.

2.2 Equations of Motion

We use Lagrange's energy method to derive the equations of motion for this system. This method is equivalent to using Newton's laws of motion for systems in classical mechanics but

has the benefit of working with generalized coordinate systems resulting in fewer and simpler equations. It is also frequently easier to apply for systems with many components since it considers energies instead of forces. For our system, the single input is the driving force F_c generated by the DC motor and acting on the cart via the motor pinion. The Lagrangian of the motion is computed from the calculation of the total potential and kinetic energies of the system.

In order to calculate the energy of the system, we first determine the absolute cartesian coordinates for the center of gravity of each pendulum rod and the hinge. For the lower rod, pendulum 1, the center of gravity is located at

$$\begin{pmatrix} x_1(t) \\ y_1(t) \end{pmatrix} = \begin{pmatrix} x_c(t) - \ell_1 \sin(\alpha(t)) \\ \ell_1 \cos(\alpha(t)) \end{pmatrix}, \quad (2.1)$$

and for the upper rod, pendulum 2, the center of gravity is located at

$$\begin{pmatrix} x_2(t) \\ y_2(t) \end{pmatrix} = \begin{pmatrix} x_c(t) - \ell_2 \sin(\alpha(t) + \theta(t)) - L_1 \sin(\alpha(t)) \\ \ell_2 \cos(\alpha(t) + \theta(t)) + L_1 \cos(\alpha(t)) \end{pmatrix}. \quad (2.2)$$

The center of gravity of the hinge is located at

$$\begin{pmatrix} x_h(t) \\ y_h(t) \end{pmatrix} = \begin{pmatrix} x_c(t) - L_1 \sin(\alpha(t)) \\ L_1 \cos(\alpha(t)) \end{pmatrix}. \quad (2.3)$$

We determine the linear velocity of each component by taking derivatives with respect to time of (2.1), (2.2), and (2.3) to get

$$\begin{pmatrix} x'_1(t) \\ y'_1(t) \end{pmatrix} = \begin{pmatrix} x'_c(t) - \ell_1 \alpha'(t) \cos(\alpha(t)) \\ -\ell_1 \alpha'(t) \sin(\alpha(t)) \end{pmatrix}, \quad (2.4)$$

$$\begin{pmatrix} x'_2(t) \\ y'_2(t) \end{pmatrix} = \begin{pmatrix} x'_c(t) - \ell_2 (\alpha'(t) + \theta'(t)) \cos(\alpha(t) + \theta(t)) - L_1 \alpha'(t) \cos(\alpha(t)) \\ -\ell_2 (\alpha'(t) + \theta'(t)) \sin(\alpha(t) + \theta(t)) - L_1 \alpha'(t) \sin(\alpha(t)) \end{pmatrix}, \quad (2.5)$$

$$\begin{pmatrix} x'_h(t) \\ y'_h(t) \end{pmatrix} = \begin{pmatrix} x'_c(t) - L_1 \alpha'(t) \cos(\alpha(t)) \\ -L_1 \alpha'(t) \sin(\alpha(t)) \end{pmatrix}. \quad (2.6)$$

2.2.1 Potential Energy

The total potential energy in a system, V_T , is the energy a system has due to work being or having been done to it. Typically potential energy is due to either vertical displacement (gravitational potential energy) or spring-related displacement (elastic potential energy). Here, we have no springs, and we assume all of the components of our system are rigid. Thus, there is no elastic potential energy, just the potential energy due to gravity. The cart is limited to horizontal motion and therefore has no gravitational potential energy. The potential energies of the pendulum rods and the hinge are given by

$$V_1(t) = M_1gy_1 = M_1g\ell_1 \cos(\alpha(t)), \quad (2.7)$$

$$V_2(t) = M_2gy_2 = M_2g[\ell_2 \cos(\alpha(t) + \theta(t)) + L_1 \cos(\alpha(t))], \quad (2.8)$$

$$V_h(t) = M_hgy_h = M_hgL_1 \cos(\alpha(t)). \quad (2.9)$$

Then the total potential energy of the system is the sum of each component's potential energy. Summing (2.7), (2.8), and (2.9) and rearranging, we obtain

$$V_T(t) = [M_1g\ell_1 + M_2gL_1 + M_hgL_1] \cos(\alpha(t)) + M_2g\ell_2 \cos(\alpha(t) + \theta(t)). \quad (2.10)$$

2.2.2 Kinetic Energy

The total kinetic energy of a system, T_T , is the amount of energy a system has due to motion. For the DIP system, the total kinetic energy is the sum of the translational and rotational energies of each component.

First, we consider the cart which has kinetic energy due to its linear motion along the track and due to the rotation of the DC motor. The translational kinetic energy of the cart is given by

$$T_{ct}(t) = \frac{1}{2}M_c(x'_c(t))^2. \quad (2.11)$$

The rotational kinetic energy of the cart's DC motor is given by

$$T_{cr}(t) = \frac{1}{2}J_m(\omega_c(t))^2 \quad (2.12)$$

where J_m is the rotational moment of inertia of the DC motor's output shaft. The angular velocity of the DC motor's output shaft ω_c is given by

$$\omega_c(t) = \frac{K_g}{r_{mp}}x'_c(t) \quad (2.13)$$

where K_g is the planetary gear box ratio and r_{mp} is the radius of the motor pinion. Then the

total kinetic energy of the cart is the sum of the translational and rotational kinetic energies,

$$T_c(t) = \frac{1}{2} \left[M_c + \frac{J_m K_g^2}{r_{mp}^2} \right] (x'_c(t))^2. \quad (2.14)$$

Next we consider the kinetic energy of each pendulum rod. For each, we assume the mass of the pendulum is concentrated at its center of gravity. Then, the translational kinetic energy of the lower pendulum rod is given by

$$\begin{aligned} T_{1t}(t) &= \frac{1}{2} M_1 [(x'_1(t))^2 + (y'_1(t))^2] \\ &= \frac{1}{2} M_1 [(x'_c(t))^2 - 2\ell_1 x'_c(t) \alpha'(t) \cos(\alpha(t)) + \ell_1^2 (\alpha'(t))^2], \end{aligned} \quad (2.15)$$

and the rotational kinetic energy is given by

$$T_{1r}(t) = \frac{1}{2} I_1 (\alpha'(t))^2. \quad (2.16)$$

I_1 is the moment of inertia of the lower pendulum rod at its center of gravity and is calculated by

$$I_1 = \frac{1}{12} M_1 L_1^2. \quad (2.17)$$

Thus, the total kinetic energy for the lower pendulum rod is given by

$$T_1(t) = \frac{1}{2} M_1 [(x'_c(t))^2 - 2\ell_1 x'_c(t) \alpha'(t) \cos(\alpha(t)) + \ell_1^2 (\alpha'(t))^2] + \frac{1}{2} I_1 (\alpha'(t))^2. \quad (2.18)$$

Similarly, the translational and rotational kinetic energies of the upper pendulum rod are given by

$$\begin{aligned} T_{2t}(t) &= \frac{1}{2} M_2 [(x'_2(t))^2 + (y'_2(t))^2] \\ &= \frac{1}{2} M_2 [(x'_c(t))^2 - 2\ell_2 x'_c(t) (\alpha'(t) + \theta'(t)) \cos(\alpha(t) + \theta(t)) \\ &\quad - 2L_1 x'_c(t) \alpha'(t) \cos(\alpha) + 2L_1 \ell_2 \alpha'(t) (\alpha'(t) + \theta'(t)) \cos(2\alpha + \theta) \\ &\quad + \ell_2^2 (\alpha'(t) + \theta'(t))^2 + L_1^2 (\alpha'(t))^2] \end{aligned} \quad (2.19)$$

$$T_{2r}(t) = \frac{1}{2} I_2 (\theta'(t))^2 \quad (2.20)$$

for a total kinetic energy of

$$\begin{aligned}
T_2(t) = & \frac{1}{2}M_2 [(x'_c(t))^2 - 2\ell_2 x'_c(t)(\alpha'(t) + \theta'(t)) \cos(\alpha(t) + \theta(t)) \\
& - 2L_1 x'_c(t)\alpha'(t) \cos(\alpha) + 2L_1 \ell_2 \alpha'(t)(\alpha'(t) + \theta'(t)) \cos(2\alpha + \theta) \\
& + \ell_2^2(\alpha'(t) + \theta'(t))^2 + L_1^2(\alpha'(t))^2] + \frac{1}{2}I_2(\theta'(t))^2.
\end{aligned} \tag{2.21}$$

Finally, the hinge has only translational kinetic energy since it cannot rotate about its own axis. Therefore the total kinetic energy of the hinge is given by

$$\begin{aligned}
T_h(t) = & \frac{1}{2}M_h [(x'_h(t))^2 + (y'_h(t))^2] \\
= & \frac{1}{2}M_h [(x'_c(t))^2 - 2L_1 x'_c(t)\alpha'(t) \cos(\alpha(t)) + L_1^2(\alpha'(t))^2].
\end{aligned} \tag{2.22}$$

Summing (2.14), (2.18), (2.21), and (2.22) we obtain the total kinetic energy of the system

$$\begin{aligned}
T_T(t) = & \frac{1}{2} \left[M_c + \frac{J_m K_g^2}{r_{mp}^2} + M_1 + M_2 + M_h \right] (x'_c(t))^2 \\
& - [M_1 \ell_1 + M_2 L_1 + M_h L_1] x'_c(t) \alpha'(t) \cos(\alpha(t)) \\
& + \frac{1}{2} [M_1 \ell_1^2 + I_1 + M_2 L_1^2 + M_h L_1^2] (\alpha'(t))^2 \\
& - M_2 \ell_2 x'_c(t) (\alpha'(t) + \theta'(t)) \cos(\alpha(t) + \theta(t)) \\
& + M_2 L_1 \ell_2 \alpha'(t) (\alpha'(t) + \theta'(t)) \cos(2\alpha(t) + \theta(t)) \\
& + \frac{1}{2} M_2 \ell_2^2 (\alpha'(t) + \theta'(t))^2 + \frac{1}{2} I_2 (\theta'(t))^2.
\end{aligned} \tag{2.23}$$

2.2.3 Lagrange's Equations

The Lagrangian, \mathcal{L} , of a system is given by the difference between the total kinetic and potential energies

$$\mathcal{L}(t) = T_T(t) - V_T(t). \tag{2.24}$$

Substituting (2.10) and (2.23) into (2.24) we obtain

$$\begin{aligned}
\mathcal{L}(t) = & \frac{1}{2} \left[M_c + \frac{J_m K_g^2}{r_{mp}^2} + M_1 + M_2 + M_h \right] (x'_c(t))^2 \\
& - [M_1 \ell_1 + M_2 L_1 + M_h L_1] x'_c(t) \alpha'(t) \cos(\alpha(t)) \\
& + \frac{1}{2} [M_1 \ell_1^2 + I_1 + M_2 L_1^2 + M_h L_1^2] (\alpha'(t))^2 \\
& - M_2 \ell_2 x'_c(t) (\alpha'(t) + \theta'(t)) \cos(\alpha(t) + \theta(t)) \\
& + M_2 L_1 \ell_2 \alpha'(t) (\alpha'(t) + \theta'(t)) \cos(2\alpha(t) + \theta(t)) \\
& + \frac{1}{2} M_2 \ell_2^2 (\alpha'(t) + \theta'(t))^2 + \frac{1}{2} I_2 (\theta'(t))^2 \\
& - g [M_1 \ell_1 + M_2 L_1 + M_h L_1] \cos(\alpha(t)) - M_2 g \ell_2 \cos(\alpha(t) + \theta(t)).
\end{aligned} \tag{2.25}$$

By definition, Lagrange's equations state

$$\frac{\partial}{\partial t} \left(\frac{\partial}{\partial x'_c} \mathcal{L} \right) - \frac{\partial}{\partial x_c} \mathcal{L} = Q_1 \tag{2.26}$$

$$\frac{\partial}{\partial t} \left(\frac{\partial}{\partial \alpha'} \mathcal{L} \right) - \frac{\partial}{\partial \alpha} \mathcal{L} = Q_2 \tag{2.27}$$

$$\frac{\partial}{\partial t} \left(\frac{\partial}{\partial \theta'} \mathcal{L} \right) - \frac{\partial}{\partial \theta} \mathcal{L} = Q_3 \tag{2.28}$$

where Q_1 , Q_2 , and Q_3 are the generalized forces applied to each of the generalized coordinates: x_c , α , and θ . When we compute the generalized forces we neglect the nonlinear Coulomb friction and the force due to the pendulum's action on the linear cart. Therefore the generalized forces just include our viscous damping frictional forces at the motor pinion and each pendulum axis and the input force and are given by

$$Q_1(t) = F_c(t) - B_c x'_c(t), \tag{2.29}$$

$$Q_2(t) = -B_1 \alpha'(t), \tag{2.30}$$

$$Q_3(t) = -B_2 \theta'(t). \tag{2.31}$$

We substitute (2.29), (2.30), and (2.31) into (2.26), (2.27), and (2.28), respectively, to obtain

$$\frac{\partial}{\partial t} \left(\frac{\partial}{\partial x'_c} \mathcal{L} \right) - \frac{\partial}{\partial x_c} \mathcal{L} = F_c(t) - B_c x'_c(t) \quad (2.32)$$

$$\frac{\partial}{\partial t} \left(\frac{\partial}{\partial \alpha'} \mathcal{L} \right) - \frac{\partial}{\partial \alpha} \mathcal{L} = -B_1 \alpha'(t) \quad (2.33)$$

$$\frac{\partial}{\partial t} \left(\frac{\partial}{\partial \theta'} \mathcal{L} \right) - \frac{\partial}{\partial \theta} \mathcal{L} = -B_2 \theta'(t). \quad (2.34)$$

Using (2.25), we take the derivatives indicated in (2.32) and rearrange to obtain

$$\begin{aligned} & \left[M_c + \frac{J_m K_g^2}{r_{mp}^2} + M_1 + M_2 + M_h \right] x''_c(t) - \left[(M_1 \ell_1 + M_2 L_1 + M_h L_1) \cos(\alpha(t)) \right. \\ & \quad \left. + M_2 \ell_2 \cos(\alpha(t) + \theta(t)) \right] x''_c(t) - M_2 \ell_2 \cos(\alpha(t) + \theta(t)) \theta''(t) + B_c x'_c(t) \\ & \quad \left[\left[(M_1 \ell_1 + M_2 L_1 + M_h L_1) \sin(\alpha(t)) + M_2 \ell_2 \sin(\alpha(t) + \theta(t)) \right] \alpha'(t) + \right. \\ & \quad \left. 2M_2 \ell_2 \sin(\alpha(t) + \theta(t)) \theta'(t) \right] \alpha'(t) + M_2 \ell_2 \sin(\alpha(t) + \theta(t)) (\theta'(t))^2 = F_c(t). \end{aligned} \quad (2.35)$$

Similarly, taking the derivatives indicated in (2.33) of (2.25) and rearranging yields

$$\begin{aligned} & - \left[(M_1 \ell_1 + M_2 L_1 + M_h L_1) \cos(\alpha(t)) + M_2 \ell_2 \cos(\alpha(t) + \theta(t)) \right] x''_c(t) \\ & + \left[M_1 \ell_1^2 + I_1 + M_2 L_1^2 + M_h L_1^2 + M_2 \ell_2^2 + 2M_2 L_1 \ell_2 \cos(2\alpha(t) + \theta(t)) \right] \alpha''(t) \\ & \quad + \left[M_2 L_1 \ell_2 \cos(2\alpha(t) + \theta(t)) + M_2 \ell_2^2 \right] \theta''(t) \\ & \quad + \left[B_1 - 2M_2 L_1 \ell_2 \sin(2\alpha(t) + \theta(t)) (\alpha'(t) + \theta'(t)) \right] \alpha'(t) \\ & \quad - M_2 L_1 \ell_2 \sin(2\alpha(t) + \theta(t)) (\theta'(t))^2 \\ & - g \left[M_1 \ell_1 + M_2 L_1 + M_h L_1 \right] \sin(\alpha(t)) - g M_2 \ell_2 \sin(\alpha(t) + \theta(t)) = 0. \end{aligned} \quad (2.36)$$

Finally, we take the derivatives indicated in (2.34) and rearrange to obtain

$$\begin{aligned} & -M_2 \ell_2 \cos(\alpha(t) + \theta(t)) x''_c(t) + \left[M_2 L_1 \ell_2 \cos(2\alpha(t) + \theta(t)) + M_2 \ell_2^2 \right] \alpha''(t) \\ & \quad + \left[M_2 \ell_2^2 + I_2 \right] \theta''(t) - M_2 L_1 \ell_2 \sin(2\alpha(t) + \theta(t)) (\alpha'(t))^2 + B_2 \theta'(t) \\ & \quad - g M_2 \ell_2 \sin(\alpha(t) + \theta(t)) = 0. \end{aligned} \quad (2.37)$$

Thus (2.35), (2.36), and (2.37) are the equations of motion for the system.

2.2.4 Converting to Voltage Input

In our real-time implementation of the DIP system, the input for the system is the cart's DC motor voltage, V_m , so we must convert the driving force F_c to V_m . The driving force of the cart is generated by the cart's DC motor and acts on the cart via a motor pinion. This force can be expressed as

$$F_c(t) = \frac{K_g}{r_{mp}} T_m(t) \quad (2.38)$$

where T_m is the motor torque.

A standard DC motor can be modeled using a simple circuit with an armature resistance R_m and inductance L_m . The electrical schematic of the equivalent armature circuit model is shown in Figure 2.2. Kirchhoff's Voltage Law states that the sum of the electrical potential

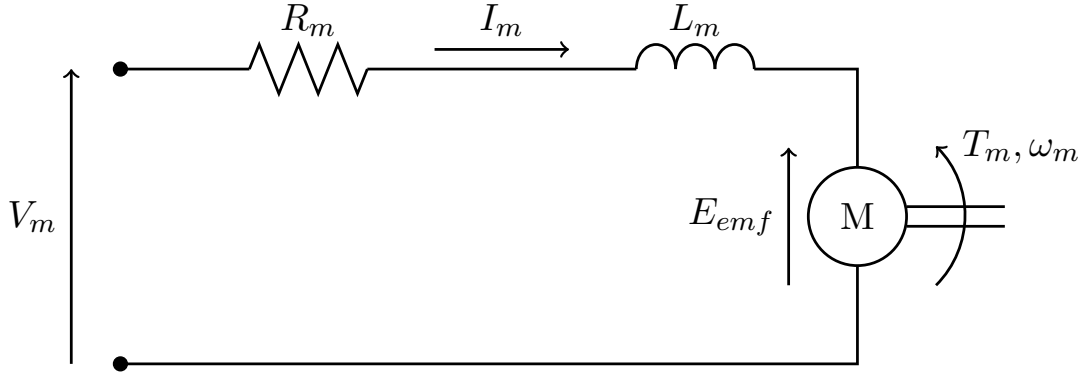


Figure 2.2: Electrical schematic of a standard DC motor.

differences around the closed loop of a circuit is zero. Applying this to the circuit shown, we obtain

$$V_m(t) - R_m I_m(t) - L_m \frac{d}{dt} I_m(t) - E_{emf}(t) = 0 \quad (2.39)$$

where I_m is the armature current and E_{emf} is the back-electromotive force voltage. Since $L_m \ll R_m$, we can neglect the motor inductance, and rearranging (2.39) we have

$$I_m(t) = \frac{V_m(t) - E_{emf}(t)}{R_m}. \quad (2.40)$$

The back-electromotive force voltage created by the motor is proportional to the angular velocity of the motor shaft,

$$E_{emf}(t) = K_m \omega_c(t), \quad (2.41)$$

so we rewrite (2.40) as

$$I_m(t) = \frac{V_m(t) - K_m \omega_c(t)}{R_m}. \quad (2.42)$$

The motor torque is proportional to the armature current, and assuming no electrical losses, can be expressed as

$$T_m(t) = K_t I_m(t). \quad (2.43)$$

Then, substituting (2.43) and (2.42) into (2.38), we obtain

$$F_c(t) = \frac{K_g K_t (V_m(t) - K_m \omega_c(t))}{R_m r_{mp}}. \quad (2.44)$$

Using (2.13) in (2.44) yields

$$F_c(t) = \frac{K_g K_t (r_{mp} V_m(t) - K_g K_m x'_c(t))}{R_m r_{mp}^2}. \quad (2.45)$$

This equation for the driving force of the cart can then be substituted into (2.35) to obtain the final equations of motion for the DIP system.

2.3 Model Calibration

To validate our nonlinear model for the DIP system, we apply a set voltage input and compare the measured performance of the real system to the simulation results. The parameter values used for this simulation are given in Appendix A.2. Our input voltage is a square wave with an amplitude of 1 V and a frequency of 0.2 Hz as shown in Figure 2.3. Figure 2.4 shows the real-time experimental results in blue (dashed) and the simulation results in red (solid) with this voltage input. We can see that the simulation results generally follow the same patterns as the experimental results; however, the magnitude of the simulation results is typically larger than the experimental results. This is particularly apparent for the cart position. In order to address the discrepancies between the magnitudes of the model and the experimental results, we decided to use parameter estimation techniques on the friction values used in the model. In this way, we hope to improve our model accuracy relative to the experimental results.

We perform our parameter estimation on the three viscous damping coefficients as seen at the motor pinion, B_c , and the upper and lower pendulum axes, B_1 and B_2 . The original values for these parameters were given by Quanser for our system as

$$B_c = 5.4 \text{ N.m.s/rad}, \quad B_1 = 0.0024 \text{ N.m.s/rad}, \quad B_2 = 0.0024 \text{ N.m.s/rad}.$$

To determine the optimal parameters for our system, we want to minimize the sum of the

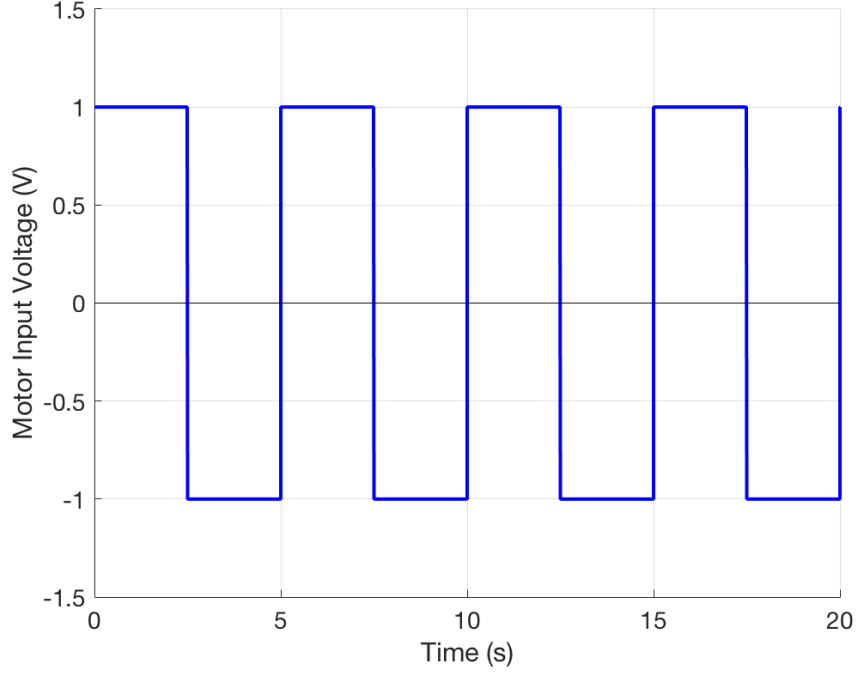


Figure 2.3: Input voltage for use in model calibration.

squared errors between the model and the experimental results. We write this optimization problem as

$$q^* = \arg \min_q \sum_{i=1}^N (y_i - X(t_i; q))^T (y_i - X(t_i; q)) \quad (2.46)$$

where $q = [B_c, B_1, B_2]$ is the set of parameters to optimized. We let X be the state vector solution to our nonlinear model with $X = [x_c(t), \alpha(t), \theta(t), x'_c(t), \alpha'(t), \theta'(t)]$. Then, N is the number of experimental data points, and y_i is the experimental data for a given time point t_i .

We use MATLAB's nonlinear optimization function `fminsearch` to solve (2.46). This function implements a Nelder-Mead algorithm to search for the minimizing arguments. This is a local search algorithm and does not guarantee a global minimum to the optimization problem. Therefore we use a number of initial guesses in order to determine the optimized parameters. Table 2.1 shows a number of those initial guesses along with the optimized parameters and the value of the minimization function, J , using those parameters. For all tested initial guesses, the resulting optimized values are very close as is the value of the minimization function. We also note that the viscous friction coefficients increased as we would expect from examining

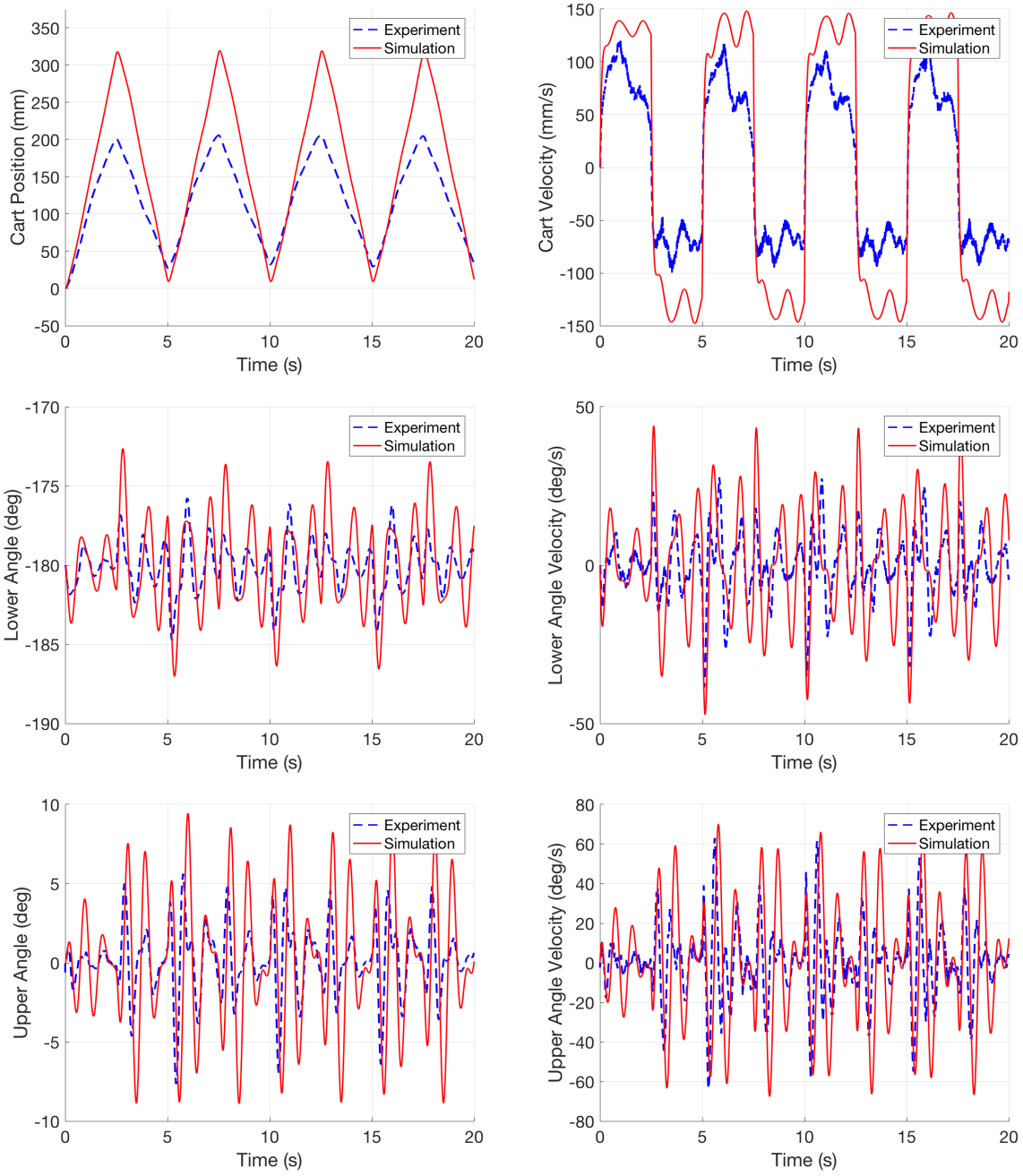


Figure 2.4: Experimental and model simulation results using the parameter values provided by Quanser.

Table 2.1: Optimized friction parameter values found using fminsearch.

B_c	B_1	B_2	B_c^*	B_1^*	B_2^*	J
5.4	0.0024	0.0024	11.1878	0.03473	0.006230	4.391e+03
4.3	0.02	0.02	11.1747	0.03475	0.006224	4.391e+03
13	0.02	0.02	11.1925	0.03472	0.006228	4.391e+03
13	0.08	0.02	11.1951	0.03475	0.006224	4.391e+03
13	0.08	0.08	11.1997	0.03476	0.006211	4.391e+03

Figure 2.4. Using the optimized values in the first row of the table,

$$B_c = 11.1878 \text{ N.m.s/rad}, \quad B_1 = 0.03473 \text{ N.m.s/rad}, \quad B_2 = 0.006230 \text{ N.m.s/rad},$$

we compare the new model values with the experimental values of each state. These results are shown in Figure 2.5. The values of the three position states from the model are much closer to the experimentally observed values. For the cart velocity, the overall magnitudes are much closer, though the model doesn't capture the smaller changes in velocity on the 2.5 second sub-intervals as closely as the previous model did. The two states for the angular velocities of the upper and lower pendulums are generally closer to the experimental results, but now the values of the experimental results are actually slightly larger than those of the model.

We also computed and graphed the residuals between the model and experimental values which are shown in Figure 2.6. We can clearly see that our residuals are not independent and identically distributed. This tells us that the errors we see are not merely due to random white noise in our experimental results, but rather there is something more systematic going on. Most likely this is related to assumptions we made in our model, particularly those related to forces we neglected. For example, our model doesn't include the force of friction between the cart's wheels and the track or exterior forces on the pendulums such as air resistance and the force of air currents present in the lab setup caused by the HVAC systems. Furthermore, we assumed that the motion of our pendulum was purely in a two-dimensional plane since our cart is constrained to travel along a track and the pendulum motion is also restricted to that plane. However, there could be small amounts of out-of-plane motion present since we are working in real three-dimensional space which may contribute to the errors in the model. Models which include these considerations could result in more accurate simulations, but we leave that for future work. The presented results will use the optimized friction values determined here.

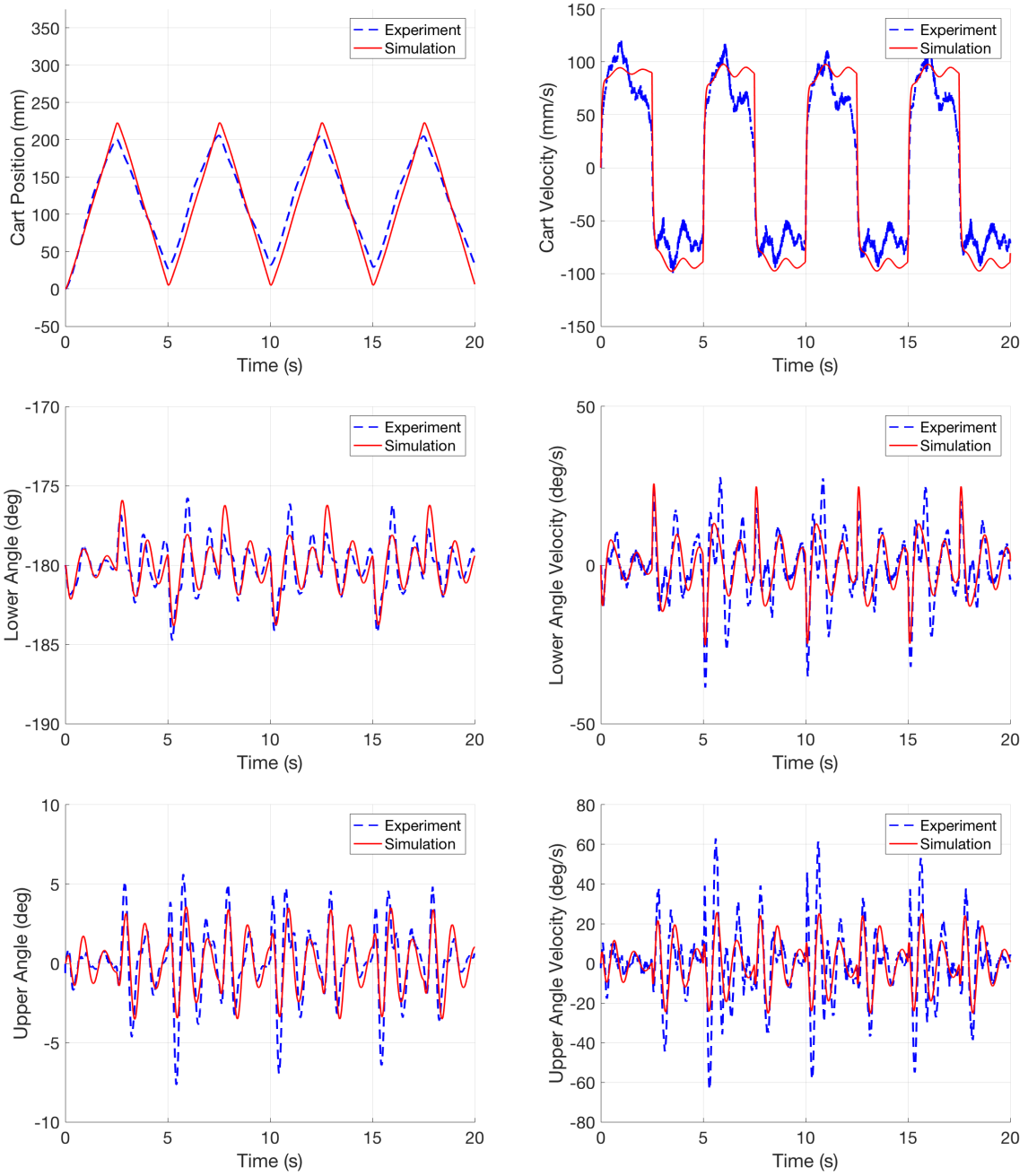


Figure 2.5: Experimental and model simulation results using the optimized friction parameter values.

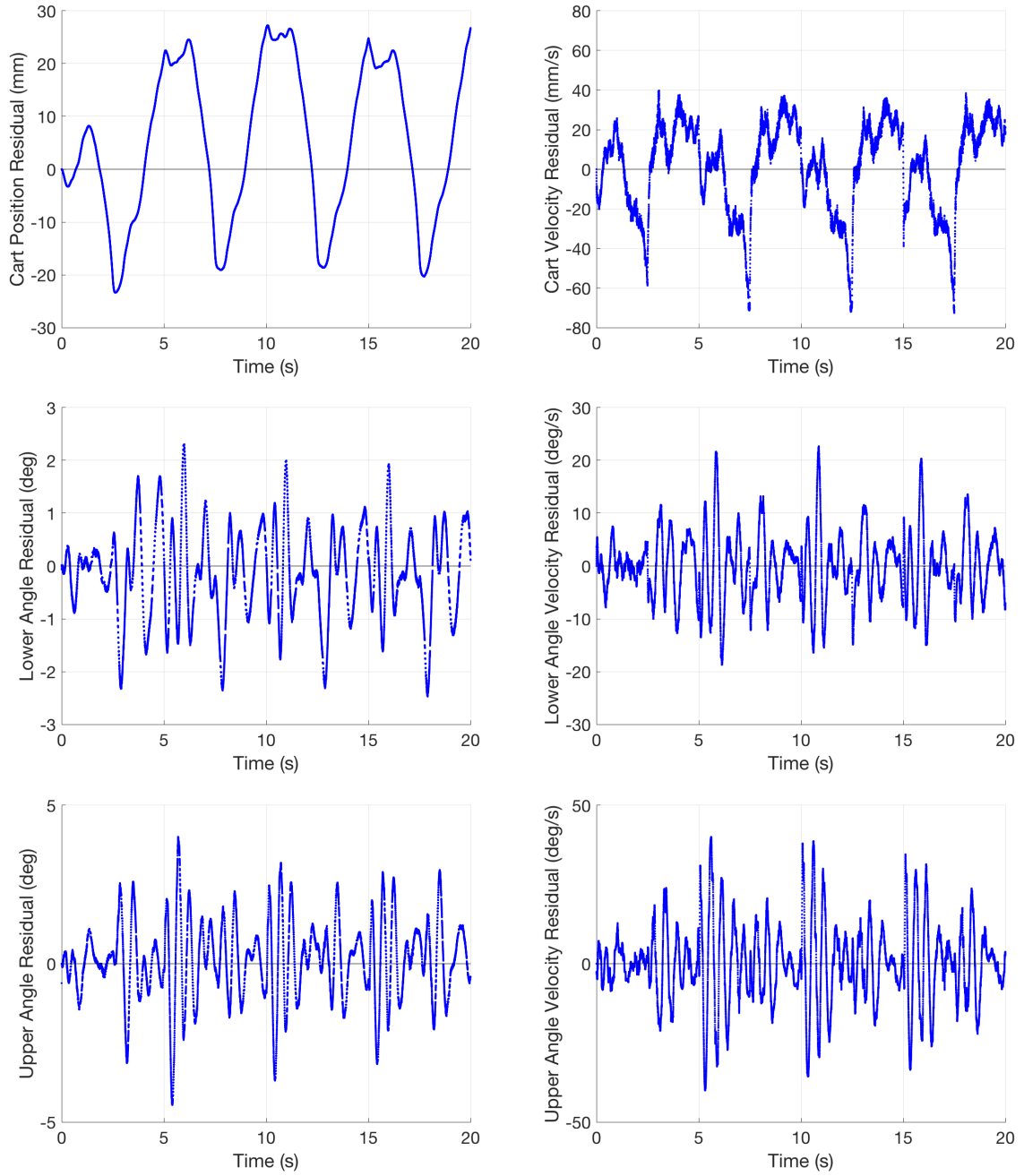


Figure 2.6: Residuals between the experimental and model simulation results using the optimized friction parameter values.

Chapter 3

Stabilization Control of an Inverted Double Pendulum System

In this chapter, we will discuss the stabilization of the double inverted pendulum system. We begin by discussing control methods currently used in the literature. Then we will derive a linear quadratic regulator controller and a power series based controller for the system. Both simulation and real-time experimental results will be presented for each controller.

3.1 Overview of Existing Control Methods

Control problems for the double inverted pendulum come in two varieties: swing-up and stabilization control. Swing-up control refers to beginning the pendulum at rest in its stable, downward hanging equilibrium state and then using a motor to bring the pendulum to its vertical, unstable equilibrium state and balancing it there. Stabilization control refers to just this second step, where the pendulum begins in the unstable equilibrium position and must be balanced there. Several control design approaches have been developed and applied to the DIP system for each of these cases. However, these methods are typically applied only in simulations and not on real-time experimental set ups. These simulations also frequently use simplified versions of the DIP model which ignore frictional effects.

We also note that the models seen in the literature are typically simpler than the one derived in Chapter 2 since the upper pendulum angle, θ , is measured from vertical rather than relative to the lower angle as we have done [7, 9, 13, 15, 18, 33, 43]. This reference point results in simpler model formulations but is not as easily applied to our real system since we can only measure θ relative to α .

The following is a summary of the control methods commonly seen in the literature.

Linear Quadratic Regulator (LQR) Control is a simple to implement control method

that is frequently used as a baseline for comparison with other methodologies [7, 9, 40]. However, since the DIP system is highly nonlinear and the LQR controller can only be applied to linear systems, this method requires that we first linearize about the unstable equilibrium $X = 0$. Therefore, this method can only be used near the equilibrium as a locally near-optimal stabilizing control as shown in [7]. This control is also commonly used to perform the stabilization after using a swing-up control once the pendulum is within a certain basin near the equilibrium [43].

State Dependent Riccati Equation (SDRE) Control is a method which uses a nonlinear form of the DIP model to develop a control that is dependent on the states. This method is sometimes considered to be a nonlinear extension of the LQR control. It has been shown to perform similarly to the LQR control when close to the equilibrium state and performs better at larger pendulum deflections [7]. However, this method can be very computationally expensive since it requires solving the state dependent Riccati equations at each time step.

Neural Networks allow for direct nonlinear optimization of a control. They are popular for controlling nonlinear systems because of their universal function approximation capabilities as discussed by Bogdanov in [7]. Due to the costs and challenges in training the neural networks, when used on their own, neural networks were unable to improve beyond the performance of the LQR control, only achieving stabilization in the same range. However, when combined with either the LQR or SDRE control, neural networks improved performance since they were able to generate “corrections” to the other controls, and in the case of the SDRE control, also achieved noticeable reductions in computational costs. Still, studies of this method have only been shown in simulation, often without limits on the magnitude of the control force.

Passivity Based Control has been used with partial feedback linearization, primarily for the swing-up of a DIP system as in [16, 43]. These are energy based methods where the stability is asymptotic to a manifold rather than a point, and therefore are typically used with LQR controls for stabilization. Again, only simulation results are typically presented, though [16] does include a detailed analysis of the motor performance in the construction of their control.

Feedforward/Feedback Control solves the swing-up control problem by solving an optimization problem where the unstable and stable equilibrium are boundary conditions and they seek to determine a path for the pendulum to follow for swing-up [13, 33]. These controllers then switch to a linear feedback control for the stabilization of the DIP. Experimental results have been presented in the literature for these controllers, but they are sensitive to the model parameters and therefore require optimizing the model parameters before applying them to the system [13].

Nonlinear Model Predictive Control also uses the solution of a two-point boundary value problem. However, this method computes the solution in real-time as a feedback control. This control has been shown to be effective in numerical simulations by [18].

Fuzzy Control methods attempt to stabilize the DIP system by constructing fuzzy rules that focus on each aspect of the system independently. They then use priority weighting to stabilize the complete system. This has been shown in simulation for a parallel double inverted pendulum, where two pendulums are mounted on a single cart but not connected to each other, but has not yet been applied to a DIP system of our type [41].

Sliding Mode Control is a robust control method that uses a discontinuous control signal. However, it can lead to undesirable performance issues such as chatter when combined with fast switching times. This control has been used to stabilize the DIP system in [15] and numerical simulations are presented.

3.2 Problem Statement

Our goal is to stabilize the DIP system about its unstable, vertical equilibrium with minimal cart and pendulum movement and control effort. In particular, we desire the pendulum angles to never exceed a 10° deflection from the vertical position, that is, $|\alpha(t)|, |\theta(t)| \leq 10^\circ$. To begin, we write the equations of motion for the DIP system in their state space representation as

$$\begin{aligned} X'(t) &= f(X(t)) + Bu(t) \\ X(0) &= X_0 \end{aligned} \tag{3.1}$$

with state vector $X(t) = [x_c(t), \alpha(t), \theta(t), x'_c(t), \alpha'(t), \theta'(t)]^T$ and control variable $u(t) = V_m(t)$. We also consider the cost functional

$$J(X_0, u) = \int_0^\infty (X^T Q X + R u^2) dt \tag{3.2}$$

where Q is a constant valued 6x6 symmetric and positive definite matrix, and R is a positive scalar.

Since we will be applying our controls to a real-time system, we have some additional constraints on our state variables. Our track length is finite, and therefore, for the safety of the system, we require that the cart does not run to the edge of the track. We assume the center of the track is at $x_c(t) = 0$, so we require $|x_c(t)| \leq 400$ mm. We also need to ensure that the power amplifier is not put into saturation by our applied control voltage, so we need to satisfy $|V_m(t)| \leq 10$ V.

3.3 LQR Control

First, we apply a standard linear quadratic regulator (LQR) controller to our system. Since this control is for linear systems, we must linearize the system about the zero equilibrium state. In practice, we accomplish this by using the Taylor function in Maple. Then we can write our system from (3.1) as

$$\begin{aligned} X'(t) &= AX(t) + Bu(t) \\ X(0) &= X_0. \end{aligned} \tag{3.3}$$

To determine the optimal control relative to the cost given in (3.2), we begin by considering the Hamiltonian function of this system given by

$$H(X, u, \lambda) = \frac{1}{2}(X^T QX + u^T Ru) + \lambda^T (AX + Bu) \tag{3.4}$$

where λ is a Lagrange multiplier referred to as the costate. We follow the derivation of [20] which uses the calculus of variations to show that the necessary conditions to obtain a minimum cost for this problem are

$$H_x = -\lambda' = QX + A^T \lambda, \tag{3.5}$$

$$H_\lambda = x' = AX + Bu. \tag{3.6}$$

For the unconstrained control problem, we would also have the stationarity condition given by

$$H_u = 0 = Ru + B^T \lambda. \tag{3.7}$$

Since in our problem, the input control is bounded, that is, $|u(t)| \leq u_{max}$, we use Pontryagin's minimum principle which requires that

$$H(x^*, u^*, \lambda^*, t) \leq H(x^*, u, \lambda^*, t) \tag{3.8}$$

for all u in the admissible space with $*$ denoting the optimal quantities. Using (3.4), we have

$$\frac{1}{2}(u^*)^T Ru^* + (\lambda^*)^T Bu^* \leq \frac{1}{2}u^T Ru + \lambda^T Bu \tag{3.9}$$

for all admissible u . Therefore, we want to select our input to minimize the quantity $\frac{1}{2}u^T Ru + \lambda^T Bu$. We add the term $\frac{1}{2}\lambda^T BR^{-1}B^T \lambda$ since it doesn't depend on u and alternately minimize

$$\frac{1}{2}(u + R^{-1}B^T \lambda)^T R(u + R^{-1}B^T \lambda). \tag{3.10}$$

Since $R > 0$, we know this is equivalent to minimizing

$$\frac{1}{2}(u + R^{-1}B^T\lambda)^T(u + R^{-1}B^T\lambda). \quad (3.11)$$

If the magnitude of $R^{-1}B^T\lambda(t)$ is smaller than u_{max} , then u^* is found by setting the derivative of (3.11) with respect to u equal to zero, that is,

$$u + R^{-1}B^T\lambda = 0. \quad (3.12)$$

Thus,

$$u = -R^{-1}B^T\lambda \quad (3.13)$$

if $|R^{-1}B^T\lambda(t)| \leq u_{max}$. Note that this is the same result obtained using the stationarity condition given in (3.7). However, if $|R^{-1}B^T\lambda(t)| \geq u_{max}$, then the result of minimizing u using (3.12) is an inadmissible value for u . Therefore, the best we can do is select u to make $u + R^{-1}B^T\lambda(t)$ as close to zero as possible. Thus,

$$u = \begin{cases} u_{max} & \text{if } R^{-1}B^T\lambda(t) < -u_{max}, \\ -u_{max} & \text{if } R^{-1}B^T\lambda(t) > u_{max}. \end{cases} \quad (3.14)$$

Combining (3.13) and (3.14), we find that the optimal control for the constrained input is

$$u = \begin{cases} u_{max} & \text{if } R^{-1}B^T\lambda(t) < -u_{max}, \\ -R^{-1}B^T\lambda & \text{if } |R^{-1}B^T\lambda(t)| \leq u_{max}, \\ -u_{max} & \text{if } R^{-1}B^T\lambda(t) > u_{max}. \end{cases} \quad (3.15)$$

Note, that if $|R^{-1}B^T\lambda(t)| \leq u_{max}$ for all t , then this is the same control as in the unconstrained control problem.

We now assume we are in the case where $|R^{-1}B^T\lambda(t)| \leq u_{max}$ and suppose that the costate λ is given by

$$\lambda = PX \quad (3.16)$$

for some matrix P . Substituting (3.16) into (3.5) and (3.15) for this case, we obtain

$$-\lambda' = QX + A^TPX, \quad (3.17)$$

$$u = -R^{-1}B^TPX. \quad (3.18)$$

We then substitute (3.18) into (3.6) to find

$$x' = AX - BR^{-1}B^T PX. \quad (3.19)$$

Then we differentiate (3.16) to obtain

$$\begin{aligned} \lambda' &= PX' \\ &= P(AX - BR^{-1}B^T PX) \\ &= PAX - PBR^{-1}B^T PX. \end{aligned} \quad (3.20)$$

Setting this equal to (3.17), we get

$$-QX - A^T PX = PAX - PBR^{-1}B^T PX. \quad (3.21)$$

This implies

$$PAX - PBR^{-1}B^T PX + QX + A^T PX = 0 \quad (3.22)$$

$$(PAX + A^T P - PBR^{-1}B^T P + Q)X = 0. \quad (3.23)$$

Since this is true for all X , we have

$$PA + A^T P - PBR^{-1}B^T P + Q = 0 \quad (3.24)$$

which is the well known matrix Riccati equation. Thus, the feedback control is given by

$$u = \begin{cases} u_{max} & \text{if } R^{-1}B^T PX < -u_{max}, \\ -R^{-1}B^T PX & \text{if } |R^{-1}B^T PX| \leq u_{max}, \\ -u_{max} & \text{if } R^{-1}B^T PX > u_{max}. \end{cases} \quad (3.25)$$

where P is the solution to the matrix Riccati equation given by (3.24).

3.3.1 An Application Example

As an exercise, we consider the following nonlinear system

$$x''(t) + \sin(x(t)) = u(t). \quad (3.26)$$

Our goal is to drive the states x and x' to zero using the control u . Let $x(t) = x_1(t)$ and $x'(t) = x_2(t)$, and $X(t) = [x_1(t), x_2(t)]^T$. Then, we can rewrite our system in its state space

representation as

$$X(t)' = f(X(t)) + Bu(t) \quad (3.27)$$

where

$$f(X(t)) = \begin{bmatrix} x_2 \\ -\sin(x_1) \end{bmatrix},$$
$$B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

In order to apply the LQR control, we begin by linearizing our system about the solution $X = [0, 0]^T$. Then our linearized system is

$$X'(t) = AX(t) + Bu \quad (3.28)$$

where

$$A = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix},$$
$$B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

We consider the cost functional

$$J(X_0, u) = \int_0^\infty (X^T Q X + Ru^2) dt \quad (3.29)$$

where

$$Q = \begin{bmatrix} 40 & 0 \\ 0 & 30 \end{bmatrix}$$
$$R = 10.$$

We implement the LQR control in MATLAB using the `lqr` function to compute the solution

P to the algebraic Riccati equation and the control gain,

$$K = R^{-1}B^TP. \quad (3.30)$$

Figure 3.1 shows the trajectories of the states of the system for an initial state of $X(0) = [2, 1]^T$ and the control effort. We can see that the LQR controller quickly drives the states to zero as desired.

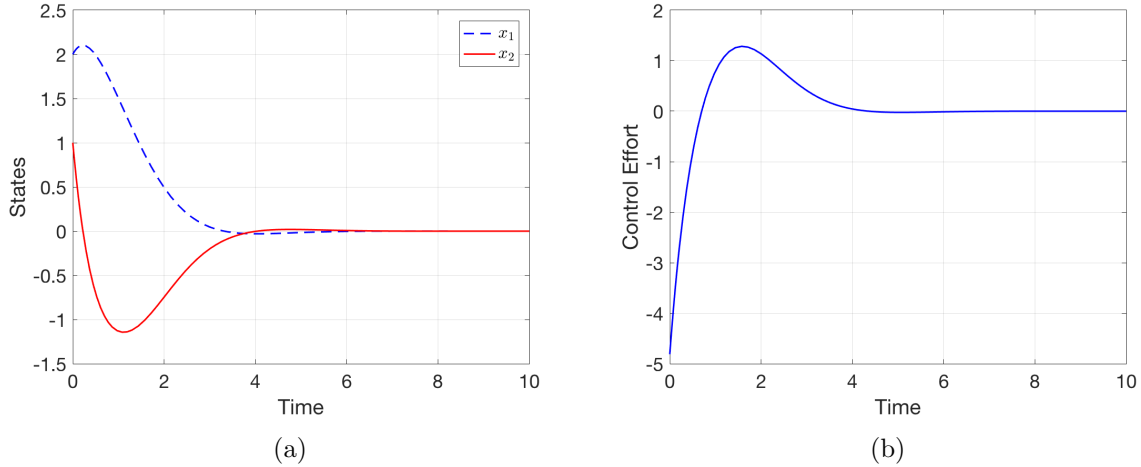


Figure 3.1: LQR control applied to example problem: (a) state trajectories and (b) control effort.

3.4 Nonlinear Control

We now consider the nonlinear system as given in (3.1) with cost (3.2). It is shown by [10] that the optimal nonlinear feedback control is of the form

$$u^*(X) = -\frac{1}{2}R^{-1}B^TV_X(X) \quad (3.31)$$

where the function V is the solution to the Hamilton-Jacobi-Bellman (HJB) equation given by

$$V_X^T(X)f(X) - \frac{1}{4}V_X^T(X)BR^{-1}B^TV_X(X) + X^TQX = 0. \quad (3.32)$$

3.4.1 Power Series Based Controller

Since the HJB equation is difficult to solve analytically, the main challenge to using the optimal control given in (3.31) is how to determine $V(X)$. Various efforts have been made to use a numerical approximation of the HJB equation to obtain a suboptimal control. We will follow Garrard and others [10, 11, 12] to numerically approximate the solution of the HJB equation from (3.32) using its power series representation:

$$V(X) = \sum_{n=0}^{\infty} V_n(X), \quad \text{where } V_n(X) = O(X^{n+2}). \quad (3.33)$$

We also rewrite the nonlinear function f from (3.1) as

$$f(X) = A_0 X + \sum_{n=2}^{\infty} f_n(X), \quad \text{where } f_n(X) = O(X^n). \quad (3.34)$$

We then substitute these expansions into the HJB equation to obtain

$$\begin{aligned} & \left(\sum_{n=0}^{\infty} (V_n)_X^T \right) \left(A_0 X + \sum_{n=2}^{\infty} f_n(X) \right) \\ & - \frac{1}{4} \left(\sum_{n=0}^{\infty} (V_n)_X^T \right) B R^{-1} B^T \left(\sum_{n=0}^{\infty} (V_n)_X \right) + X^T Q X = 0. \end{aligned} \quad (3.35)$$

Next, we separate out the terms according to powers of X to get a series of equations:

$$(V_0)_X^T A_0 X - \frac{1}{4} (V_0)_X^T B R^{-1} B^T (V_0)_X + X^T Q X = 0, \quad (3.36)$$

$$(V_1)_X^T A_0 X + (V_0)_X^T f_2(X) - \frac{1}{4} (V_1)_X^T B R^{-1} B^T (V_0)_X - \frac{1}{4} (V_0)_X^T B R^{-1} B^T (V_1)_X = 0, \quad (3.37)$$

$$(V_n)_X^T A_0 X + \sum_{k=0}^{n-1} ((V_k)_X^T f_{n+1-k}(X)) - \frac{1}{4} \sum_{k=0}^n ((V_k)_X^T B R^{-1} B^T (V_{n-k})_X) = 0, \quad (3.38)$$

where $n = 2, 3, 4, \dots$

Equation (3.36) can be solved by

$$V_0(X) = X^T P X \quad (3.39)$$

where P is the solution to the matrix Riccati equation (3.24). To solve (3.37), substitute in the equation

$$(V_0)_X = 2PX \quad (3.40)$$

to obtain

$$(V_1)_X^T A_0 X + 2X^T P f_2(X) - \frac{1}{2}(V_1)_X^T B R^{-1} B^T P X - \frac{1}{2} X^T P B R^{-1} B^T (V_1)_X = 0. \quad (3.41)$$

We rearrange terms to find

$$X^T (A_0^T (V_1)_X + 2P f_2(X) - P B R^{-1} B^T (V_1)_X) = 0. \quad (3.42)$$

Since this is true for all X , we can then solve for $(V_1)_X$ to obtain

$$(V_1)_X = -2(A_0^T - P B R^{-1} B^T)^{-1} P f_2(X). \quad (3.43)$$

We take the sum of (3.40) and (3.43) as an approximation for V_X . Then, substituting into (3.31), we obtain a quadratic feedback control law:

$$u^*(X) = -R^{-1} B^T \left[P X - (A_0^T - P B R^{-1} B^T)^{-1} P f_2(X) \right]. \quad (3.44)$$

In our DIP model $f_2(X) = 0$, so we consider (3.38) with $n = 2$, and our control is

$$u^*(X) = -R^{-1} B^T \left[P X - (A_0^T - P B R^{-1} B^T)^{-1} P f_3(X) \right]. \quad (3.45)$$

3.4.2 An Application Example

As an exercise, we again consider the example system from Section 3.3.1. To apply the power series based controller, we first take the necessary derivatives of f to find the power series representation

$$f(X(t)) = \begin{bmatrix} x_2 \\ -\sin(x_1) \end{bmatrix}, \quad (3.46)$$

$$= \begin{bmatrix} x_2 \\ -x_1 + \frac{1}{6}x_1^3 + \mathcal{O}(x_1^5) \end{bmatrix}. \quad (3.47)$$

From this we can see that the term $f_2(X) = 0$, so we use the control

$$u^*(X) = -R^{-1} B^T \left[P X - (A_0^T - P B R^{-1} B^T)^{-1} P f_3(X) \right]. \quad (3.48)$$

We implement this power series based control using MATLAB. The trajectories of the states and the control effort are shown in Figure 3.2 for an initial state of $X(0) = [2, 1]^T$. We note,

that in comparison to the LQR controller, when using the power series based control, the states reach zero slightly faster.

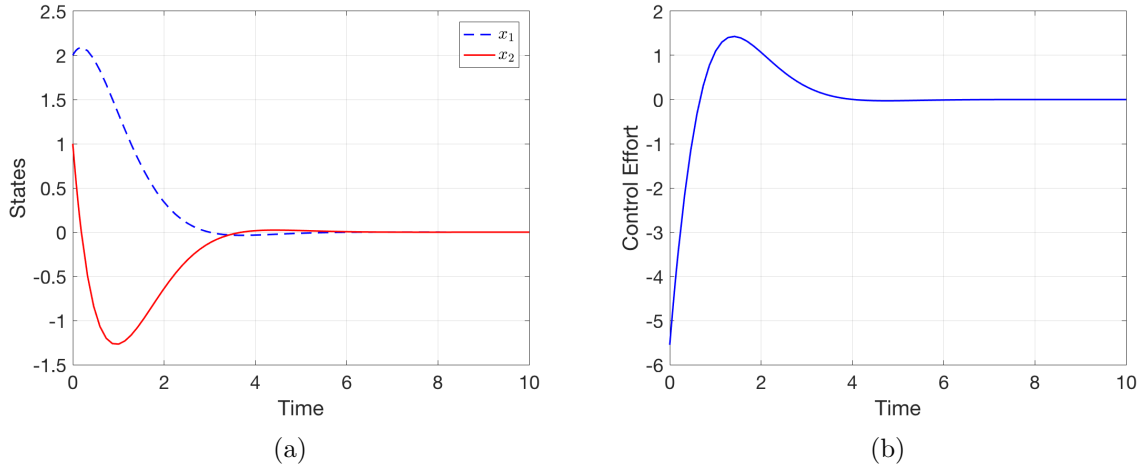


Figure 3.2: Power series based control applied to example problem: (a) state trajectories and (b) control effort.

3.5 Simulation Results

Prior to implementing either stabilization controller on the real-time DIP system, we check the performance in simulation using MATLAB Simulink. The initial condition for the simulations is to have the cart at rest at a position of 0 mm and both the upper and lower pendulum angles are deflected to 1 degree, i.e. $X = [0, 1, 1, 0, 0, 0]^T$. We require the cart's position to track a square wave with amplitude 50 mm and a frequency of 0.05 Hz.

Figure 3.3 shows the state responses and Figure 3.4 shows the corresponding control effort for a simulation run with $Q = \text{diag}(30, 350, 100, 0, 0, 0)$ and $R = 0.1$ using the LQR controller. From this simulation, we can see that all values of the states and the required control effort stay within the possible ranges for our physical apparatus. This check via the simulation is performed for each value of Q and R that we test in real-time.

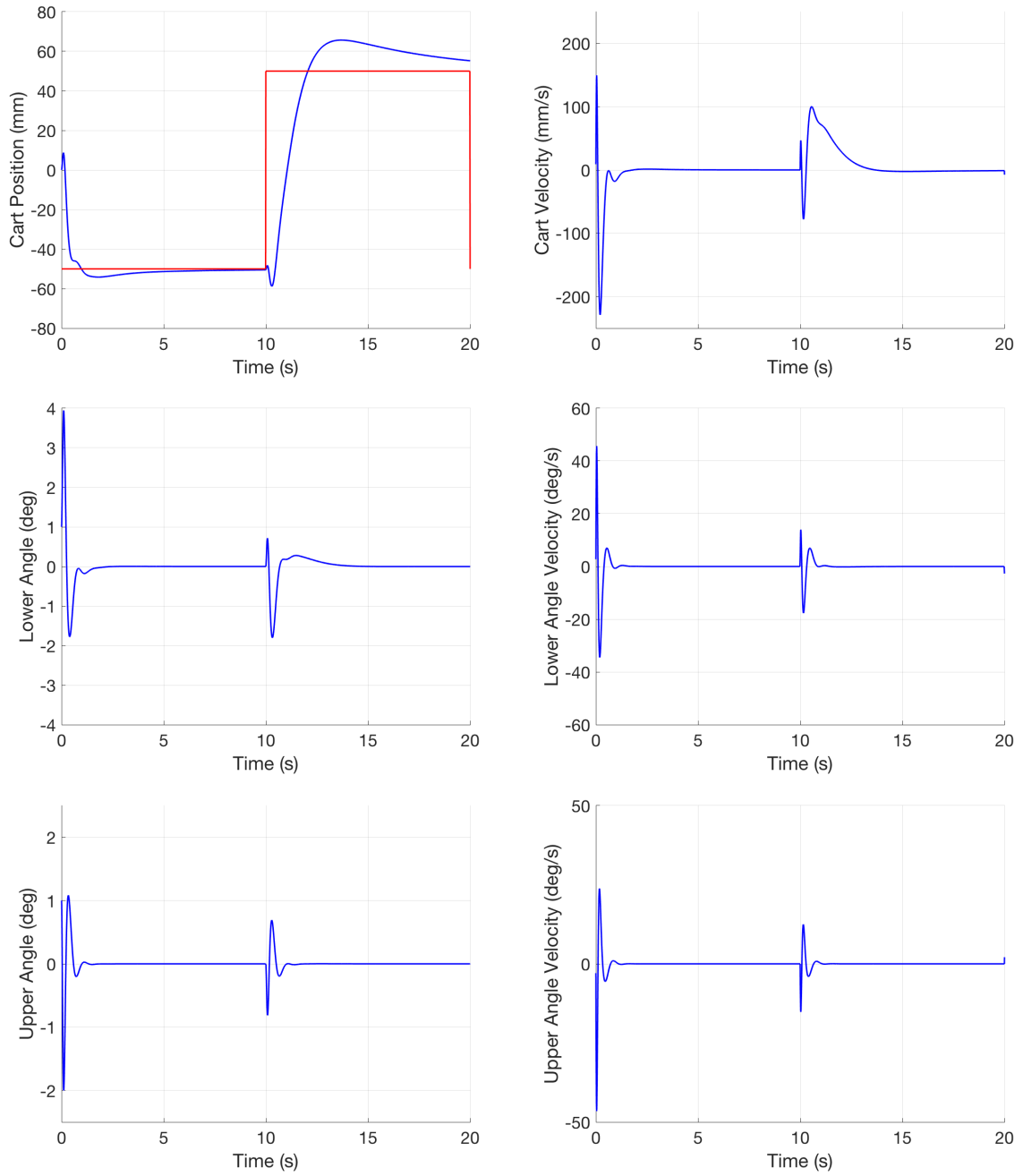


Figure 3.3: Simulation results for the state trajectories using the LQR controller with $Q = \text{diag}(30, 350, 100, 0, 0, 0)$ and $R = 0.1$.

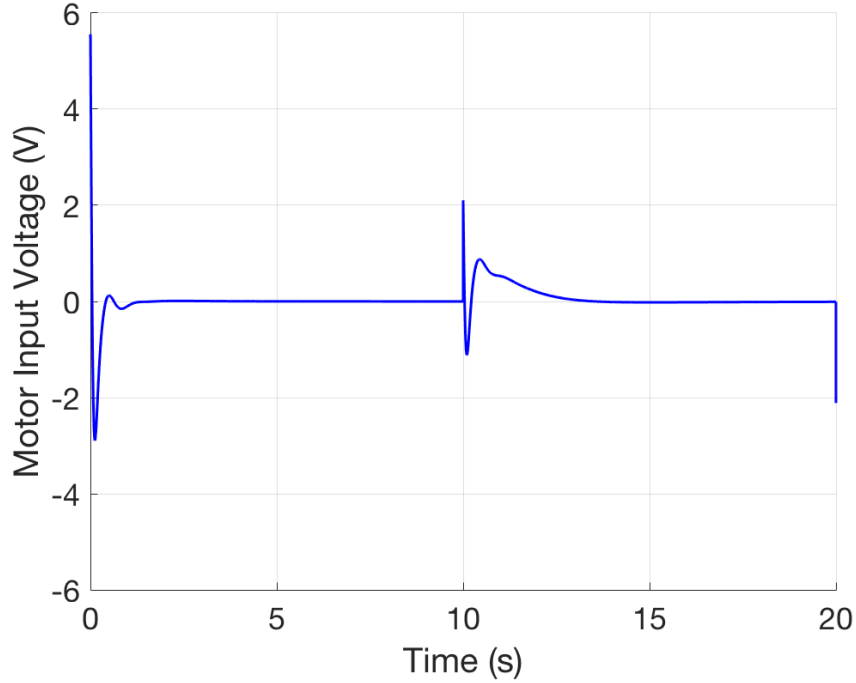


Figure 3.4: Control effort for the simulation using the LQR controller with $Q = \text{diag}(30, 350, 100, 0, 0, 0)$ and $R = 0.1$.

3.6 Experimental Results

3.6.1 Experimental Apparatus

As previously stated, our experimental apparatus for the DIP system was provided by Quanser Consulting, Inc. as shown in Figure 1.2 and described in Section 1.1. The two pendulum rods are mounted on an IP02 linear servo unit; voltage is applied to the cart via a VoltPAQ amplifier; and communication between the computer and the system is provided by two Q2-USB DAQ control boards. Figure 3.5 shows a diagrammatic representation of the experiment. Detailed technical specifications are found in [31]. The real-time experiments are run using a desktop computer running Windows 7 with a 3.20 GHz Intel Core i5 650 processor and 4 GB of RAM.

3.6.2 Experimental Procedure

To implement our control on the real-time system we use Quanser's Quarc software which is run through MATLAB Simulink. Figure 3.6 shows the Simulink block diagram which governs the real-time implementation. For the following experiments, we set the desired position of the system to be the zero state. The control block, shown in blue, is modified to implement the

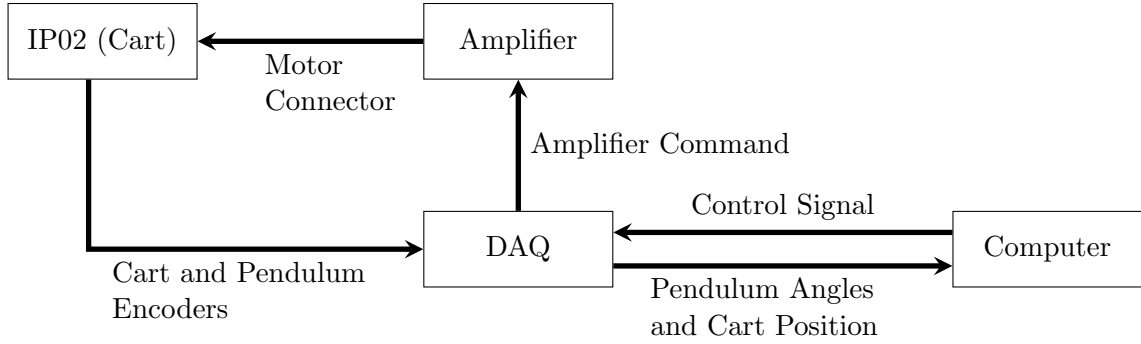


Figure 3.5: A diagram of the experimental setup for the DIP system.

different controls for each experiment. Real-time experiments are begun with the pendulum in the downward, stable configuration, completely at rest. The position of the system at that time is considered to be at $x_c(0) = 0$, $\alpha(0) = -180^\circ$, $\theta(0) = 0^\circ$. Once the real-time software is running, we manually move the pendulum to the desired vertical, unstable equilibrium position. When the pendulum is within 0.5 degrees of the desired α and θ position, that is, zero degrees, the feedback control turns on and will work to stabilize the pendulum.

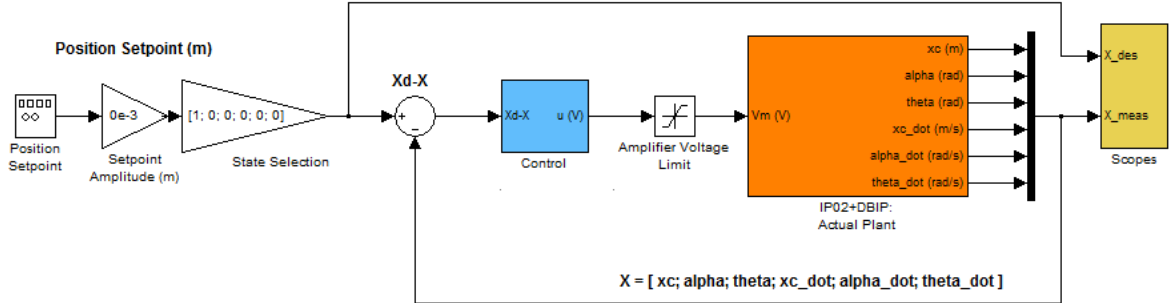


Figure 3.6: Simulink block diagram governing the real-time implementation of the DIP system.

We implement each controller for a variety of Q and R values to assess the performance of the controller. In order to improve our controller's performance, we adjust the values of Q and R according to the following guidelines:

1. If the cart motion is too large, increase Q_{11} and/or decrease Q_{22} or Q_{33} .

2. If the pendulum does not meet the specifications for the angle, increase Q_{22} or Q_{33} and/or decrease Q_{11} .
3. If the control voltage is too large or the cart vibrates excessively, increase R and/or decrease Q_{11} , Q_{22} , or Q_{33} .

3.6.3 Results for the LQR Controller

To implement the LQR controller, we compute the solution to the matrix Riccati equation given in (3.24) using the `lqr` command in MATLAB. We then use the Simulink diagram shown in Figure 3.6 to apply the control to the real time system.

Figure 3.7 shows sample experimental results using the LQR controller with weighting matrices $Q = \text{diag}(30, 350, 100, 0, 0, 0)$ and $R = 0.1$. When performing the experiment, we allow the system to run for some time between one and three minutes to ensure that the pendulum system will remain stabilized, and here we show the behavior on a typical 20 second subinterval. The behavior seen here repeats over the course of the entire experiment. We can see that the position of the cart remains within 100 mm of the center of the track. The lower pendulum angle stays within 8 degrees of its vertical position, and the upper pendulum angle remains within 4.5 degrees of alignment with the lower rod. Examining the input voltage used to control the system shown in Figure 3.8, we see that, for this sample of the real-time experiment, the motor voltage remains less than 9 V. While the voltage is staying within its saturation limits, we would prefer the system to stay closer to the center of the track and the pendulums to be closer to vertical. Keeping this in mind, we can then adjust the weighting matrices Q and R in pursuit of better system performance.

We adjusted the Q and R values used in the LQR controller and summarize the results for the state trajectories and the motor input voltage as shown in Table 3.1. We see that for some tested controllers, the amplifier enters voltage saturation at least once resulting in a maximum applied voltage of 10 V. However, we note that just because the maximum voltage was not at the saturation level, this does not always correspond to a lower average or cumulative motor voltage. For example, compare the second and third lines of the table. In the third tested case the amplifier enters saturation at least once, but the average and cumulative motor voltages are less than the second tested case where the amplifier did not enter saturation.

In the second tested case with $Q = \text{diag}(30, 350, 100, 0, 0, 0)$ and $R = 0.5$, the maximum control effort decreased by almost 3 V, but the average and cumulative motor voltages increased slightly. Similarly, the average values of the other state variables also increased. This is reasonable since, by increasing the weight on the control variable, the system will be unable to maintain the same values for the state variables, and the control punishes the high maximum motor voltage more strongly.

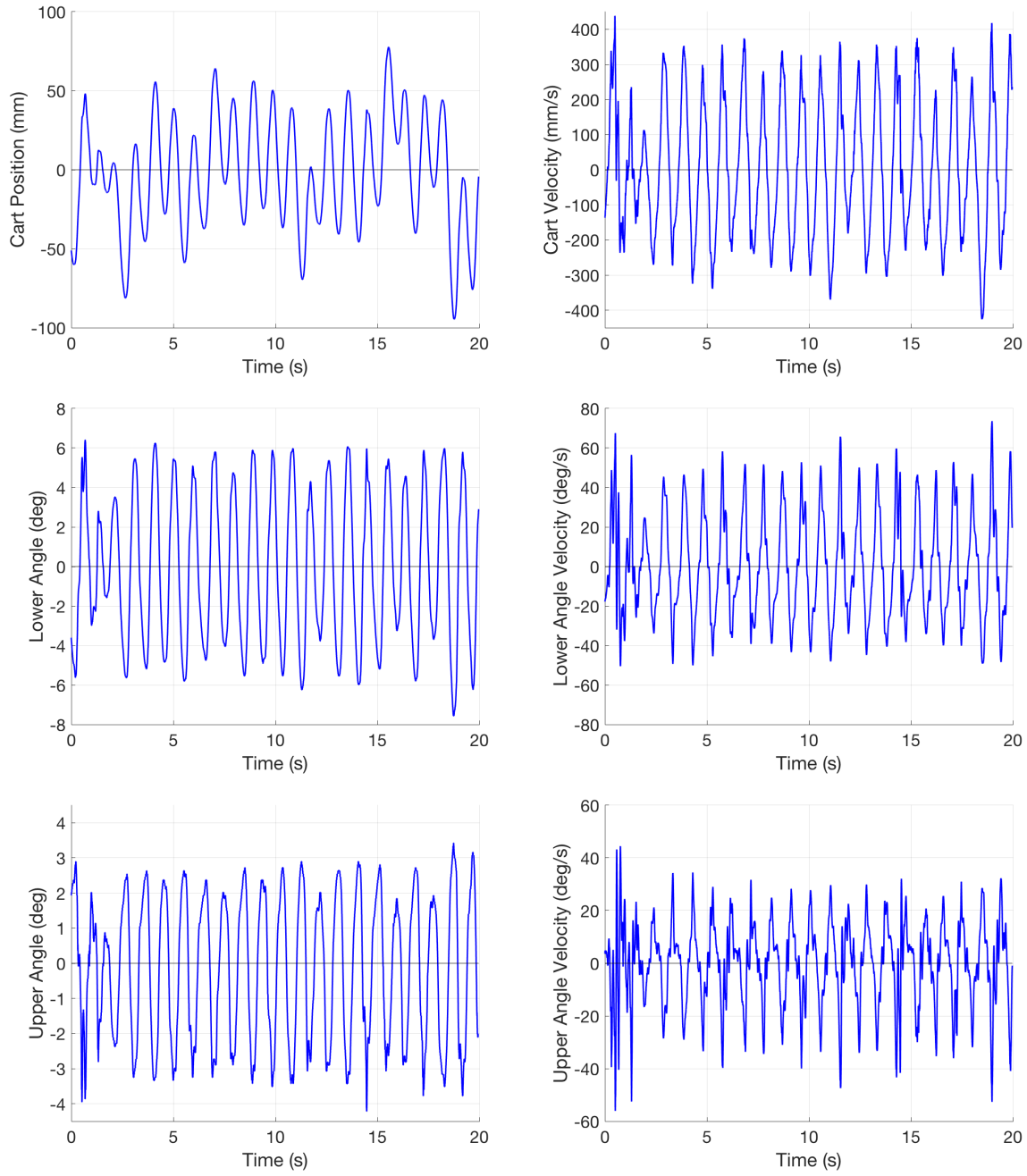


Figure 3.7: Real-time experimental results for the state trajectories using the LQR controller with $Q = \text{diag}(30, 350, 100, 0, 0, 0)$ and $R = 0.1$.

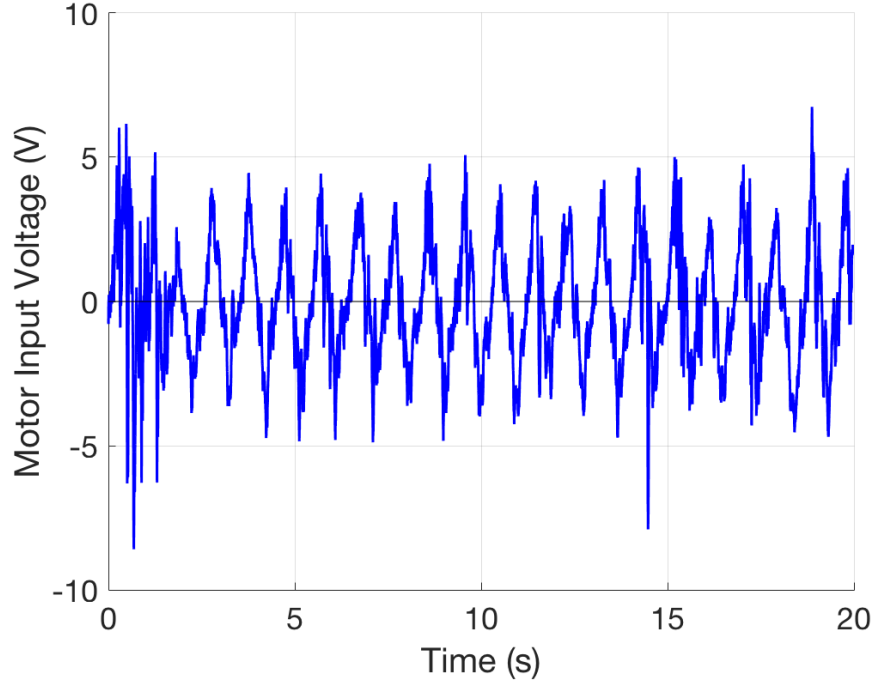


Figure 3.8: Control effort for the real-time experiment using the LQR controller with $Q = \text{diag}(30, 350, 100, 0, 0, 0)$ and $R = 0.1$.

We also attempt to keep the cart closer to the center of the track by increasing the value of the weight on the cart position, Q_{11} . The resulting control is shown in the third line of Table 3.1, and the average and maximum values for the cart position are smaller than previously seen, though not by a significant amount. The average angular positions are slightly larger than with the original control, but the control effort is similar on average.

The next tested case explored weighting the velocity states to see the effect on the system performance. When larger weights were applied to the velocity states, the pendulum failed to stabilize, so we restricted ourselves to small ones. With this control, the average and maximum values for the states and the voltages increased slightly in comparison to the previously tested values. However, this control was more likely to fail to stabilize the pendulum in general.

In the fifth tested case, we attempt to bring the cart position closer to the center of the track while also weighting the motor voltage at the higher rate. Unfortunately, the higher weighting on the cart position actually created larger oscillations in the cart motion as the control applied larger voltages in an attempt to bring it to center, overshoot the center, and used a larger voltage to quickly bring it back, again overshooting the target position. These control values resulted in the worst performance for the LQR control, amongst controls which succeeded in stabilizing

Table 3.1: Maximum and average values for the states and control voltages using the LQR controller with various values of Q and R . (x_c [mm], α [degrees], θ [degrees], V_m [V]).

Q	R	$\max x_c $ avg $ x_c $	$\max \alpha $ avg $ \alpha $	$\max \theta $ avg $ \theta $	$\max V_m $ avg $ V_m $	$\int_0^{20} V_m dt$
diag(30, 350, 100, 0, 0, 0)	0.1	94.5884	7.5586	4.2188	8.5906	33.2370
		29.4310	3.4183	1.8656	1.6646	
diag(30, 350, 100, 0, 0, 0)	0.5	86.3535	7.9102	4.4824	5.8618	41.6019
		34.1592	4.5009	2.3253	2.1029	
diag(100, 300, 100, 0, 0, 0)	0.1	92.0634	8.7891	5.0098	10	39.7388
		28.8757	4.1876	2.4168	1.9954	
diag(100, 300, 100, 1, 1, 1)	0.1	79.3924	7.9980	4.1309	10	43.5259
		31.4460	3.8657	1.9683	2.1865	
diag(200, 300, 100, 0, 0, 0)	0.5	133.3975	15.9961	8.6133	10	81.4041
		64.2152	9.6711	5.4195	4.1135	
diag(100, 300, 100, 0, 0, 0)	1.0	112.4916	14.1504	6.7676	8.7765	67.4756
		58.9119	8.1359	4.1689	3.3851	

the pendulum.

Lastly, we tried just increasing the weight on the control voltage to $R = 1$. This resulted in poorer performance of the state variables, and this was also a control which was more likely to fail to stabilize the pendulum at all.

3.6.4 Results for the Power Series Controller

We implement this power series based control for the same values of Q and R as were used with the LQR controller. Figure 3.9 shows the sample experimental results using the first control, $Q = \text{diag}(30, 350, 100, 0, 0, 0)$ and $R = 0.1$, and Figure 3.10 shows the control effort. Again, we run each experiment for between one and three minutes, and we show the behavior on a typical 20 second subinterval. The behavior shown continues throughout the experiment. While the control voltage V_m does reach a saturation level of 10 V a few times, it is mostly during the last 3 seconds of the plotted time. Examining the same time period for the state trajectory plots, we see that there is larger oscillatory motions for all states at that time. In particular,

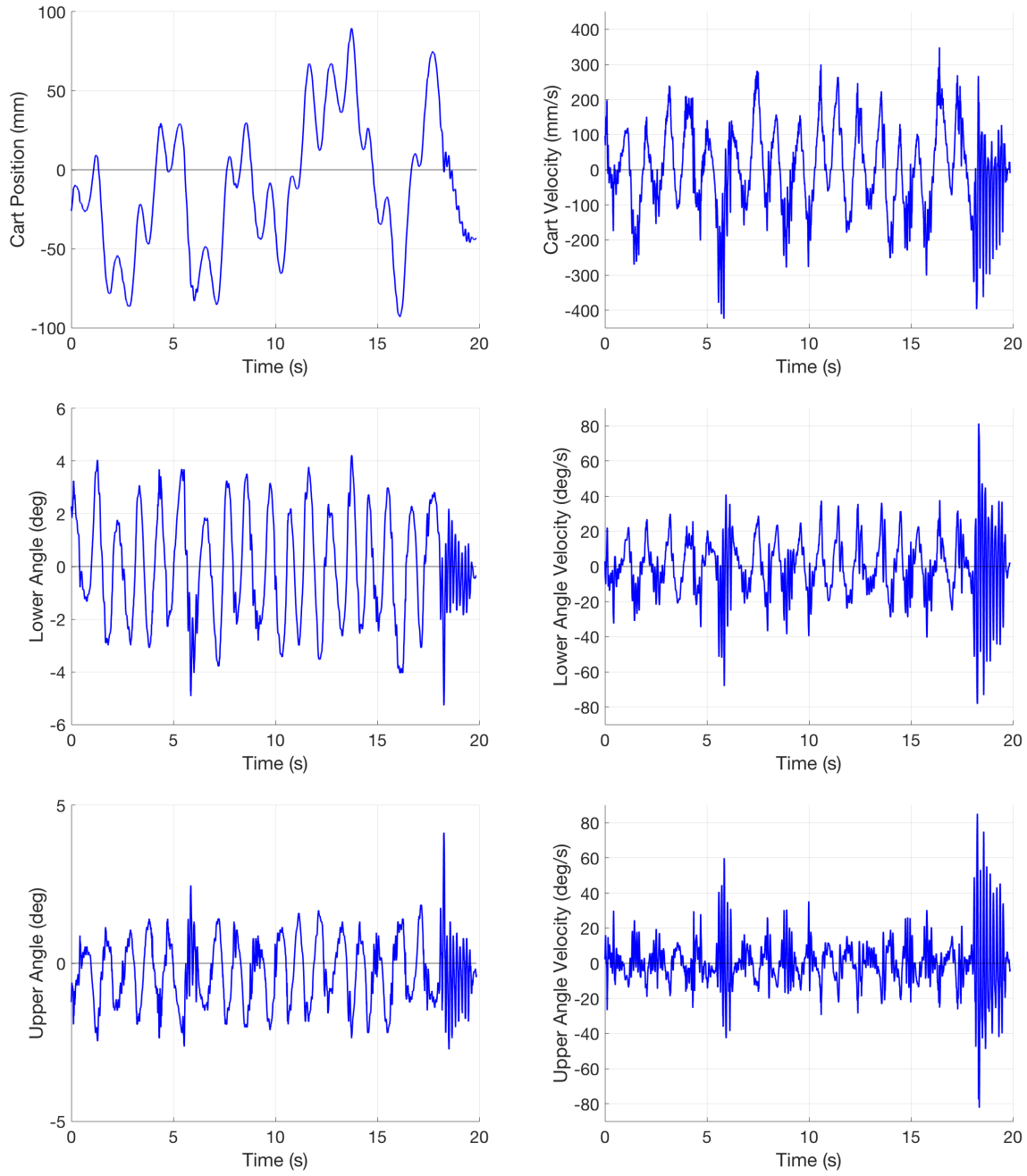


Figure 3.9: Real-time experimental results for the state trajectories using the power series based controller with $Q = \text{diag}(30, 350, 100, 0, 0, 0)$ and $R = 0.1$.

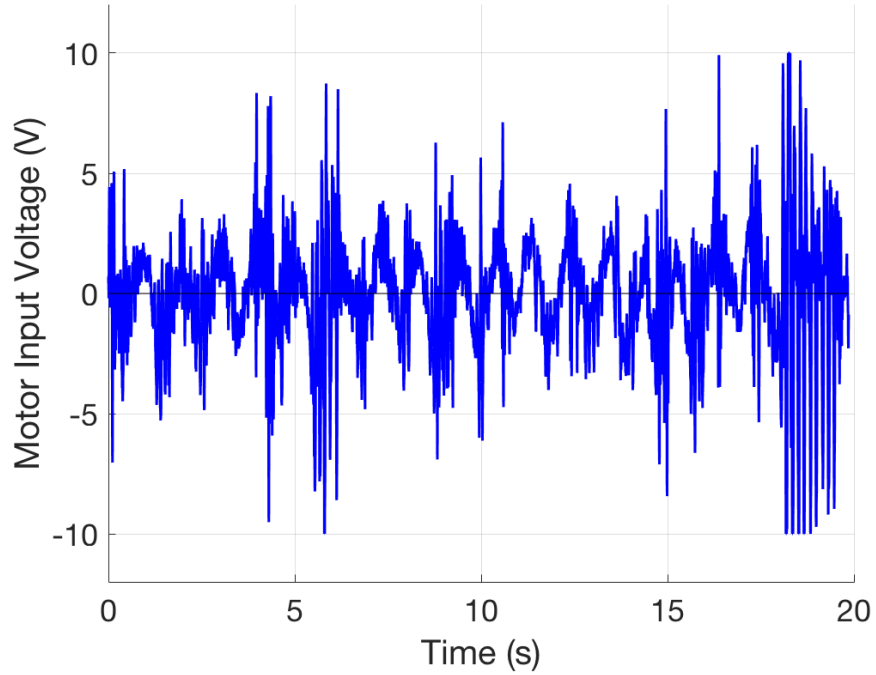


Figure 3.10: Control effort for the real-time experiment using the power series based controller with $Q = \text{diag}(30, 350, 100, 0, 0, 0)$ and $R = 0.1$.

looking at the plot of the cart position, we can see that the system had a short period of more jagged motion which is typically seen when the pendulum is very close to the zero state, as confirmed by the plots of the upper and lower pendulum angles. This behavior occurs regularly with the power series based controllers, but never continues for longer than five seconds before returning to the patterns shown in the rest of the figure. Otherwise, the motion is quite similar to that for the LQR controller with the same weighting matrices. The cart position remains within 100 mm of the center of the track, and the lower and upper angles remain within 6 and 5 degrees of their zero states, respectively. In fact, while the average position of the cart is farther from the center of the track than that obtained by the corresponding LQR controller, the average values for the upper and lower pendulum angles are significantly smaller. As seen in the first line of Table 3.2, the average angular states using the power series based controller are half the magnitude of the those using the corresponding LQR controller. Even so, this power series controller has average and cumulative motor input voltages only slightly higher than that applied by the LQR controller.

Table 3.2 also shows a summary of the state trajectories and motor input voltages for the other tested values of Q and R when using the power series based controller. The tested

weighting matrices were also likely to cause the power amplifier to go into voltage saturation at some point, but, in general, average and cumulative applied motor voltages were lower than those seen with the LQR controllers.

Table 3.2: Maximum and average values for the states and control voltages using the power series based controller with various values of Q and R . (x_c [mm], α [degrees], θ [degrees], V_m [V]).

Q	R	max $ x_c $	max $ \alpha $	max $ \theta $	max $ V_m $	$\int_0^{20} V_m dt$
		avg $ x_c $	avg $ \alpha $	avg $ \theta $	avg $ V_m $	
diag(30, 350, 100, 0, 0, 0)	0.1	93.0870	5.2734	4.1309	10	34.7876
		36.4901	1.7754	0.9047	1.7517	
diag(30, 350, 100, 0, 0, 0)	0.5	98.6832	7.6465	3.7793	6.6544	32.1967
		35.4940	3.1983	1.4241	1.6180	
diag(100, 300, 100, 0, 0, 0)	0.1	72.0446	6.5918	4.5703	10	37.2777
		26.3484	2.8903	1.4913	1.8682	
diag(100, 300, 100, 1, 1, 1)	0.1	62.4220	4.9219	3.2520	10	35.3103
		22.3453	2.0616	1.0954	1.7821	
diag(200, 300, 100, 0, 0, 0)	0.5	113.5835	8.7891	5.0098	6.3668	38.7577
		39.8354	4.4707	2.0734	1.9491	
diag(100, 300, 100, 0, 0, 0)	1.0	93.5648	7.8223	4.9219	10	44.6471
		40.0830	4.1154	1.7709	2.2386	

For the second test case where the weight R on the motor voltage was increased, we see that the applied voltage was less than that of case one, with $R = 0.1$, with the power series based controller. As expected, the values of the states were also higher on average, and thus further from the desired state. However, the state trajectory response was better than the corresponding LQR control for each of the pendulum angles and similar for the cart position.

Next we increased the weight on the cart position, Q_{11} , to bring the cart closer to the center of the track. Compared to the previous power series based controller, the cart position stayed much closer to the center of the track, decreasing by about 10 mm to an average distance of 26.3 mm. The angular position of the lower rod averaged a larger deflection from zero, which

makes sense since this control also has a slightly smaller Q_{22} weight of 300. The applied voltage for the control was close to the first case control which had the same value for R . Compared to the corresponding LQR controller, the state trajectories and applied motor voltage were smaller by all measures, showing better performance.

The fourth test case included nonzero weights for the velocity states. Again, including these weights resulted in controllers which did not always consistently stabilize the pendulum, so the value of the weights was kept low in comparison to those used for the position states. With the power series based controller, when stabilization was attained, the resulting performance was better than with many of the other tested controllers as shown in the table. This controller obtained the best results for the cart position, and was second only to the first test case for the performance of the pendulum rods and decreasing the control effort. This was also much better than the corresponding LQR controller. However, this was definitely the control that was most likely to fail to stabilize the pendulum at all.

For the fifth test case, we increased the weight on the cart position as well as the weight on the applied motor voltage. As was the case with the LQR control, the resulting power series based control had one of the worst overall performances in terms of both the state trajectories and the applied motor voltage, though it did perform better than the corresponding LQR control.

Lastly, we again tested the case with $Q = \text{diag}(100, 300, 100, 0, 0, 0)$ and $R = 1.0$ which places a high weight on the applied motor voltage. This case actually had the highest maximum, average, and cumulative applied motor voltage of the tested power series based controllers, as well as one of the worst performances for the state trajectories, attaining similar results to the fifth test case. The control did perform significantly better than the corresponding LQR controller. The average cart position was almost 20 mm closer to the center of the track; the lower pendulum angular position was 4 degrees closer to vertical, half the distance of the LQR controller; and the upper pendulum was over 2 degrees closer to its zero state, also half the distance of the LQR controller.

In general, we see that the power series based controller results in better stabilization performance of the DIP system. The angular positions in particular are closer to the desired zero state in both an absolute sense and on average. Also, the control effort via the motor voltage is also smaller in most cases in terms of average and cumulative measures, though it is more likely to go into saturation on occasion as well.

Chapter 4

State Observation for an Inverted Double Pendulum System

To this point, we have tacitly assumed that we have access to all of the state variables in our model in order to compute our desired feedback controls. However, in both this and many other applications we do not actually have the ability to measure every state variable. Therefore, in order to compute our feedback controls we must first construct or estimate the unmeasured state variables. This is known as the state estimator or state observer problem which is shown diagrammatically in Figure 4.1. The partial measurements of the system model are passed as inputs for a state estimator. This estimator outputs a full set of state variables, including the measured states and the constructed states, to use as input for the feedback control. The control is then used as an input for the system model.

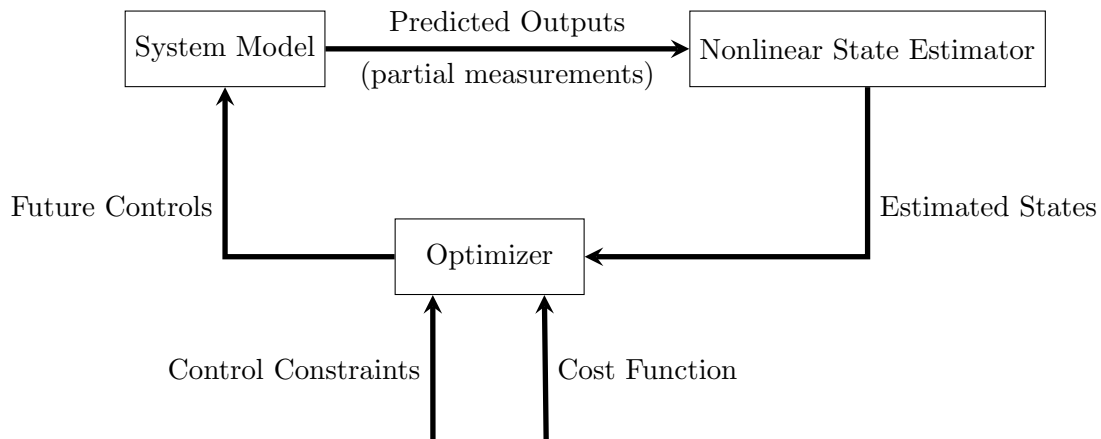


Figure 4.1: Diagrammatic representation of the state observer problem.

For the DIP system, our physical apparatus takes direct measurements of the three position state variables, x_c , α , and θ , using three encoders. The three velocity state variables, x'_c , α' , and θ' , are not measured directly, so they must be estimated. We will first discuss the current method implemented to estimate these variables and then we will examine two other proposed estimators for this system.

4.1 Low Pass Derivative Filter

A low pass filter is a filter that works to reduce noise from a signal by passing signals with a frequency lower than a certain cut-off frequency and attenuating, or reducing, the signal strength of higher frequency signals. Removing some signals in this way creates a smoother signal and improves the ability to see trends and increases the overall signal-to-noise ratio. The particular cut-off frequency used is a design decision for the filter. In a first order low pass filter, the signal amplitude is reduced by half each time the frequency doubles for frequencies above the cut-off. A second order filter attenuates high frequencies more quickly, reducing the signal amplitude to one-fourth its original value each time the frequency doubles.

The general equation for a second order low pass filter in the transfer function domain is given by

$$H(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (4.1)$$

where ω_n is the cut-off frequency and ζ is the damping ratio [30]. A derivative filter is then a combination of passing a signal through a low pass filter and taking a derivative, which in the transfer function domain is given by

$$H(s) = \frac{\omega_n^2 s}{s^2 + 2\zeta\omega_n s + \omega_n^2}. \quad (4.2)$$

We note that it is important to use a filter when taking a derivative of a signal since differentiating a signal amplifies the noise present in that signal.

As part of the Quanser provided software Quarc which communicates with the DIP system, a second order derivative filter is used when reading the data from the encoders to compute the velocity states. For the IP02 linear servo unit, the given values for design of the second order low pass filter are

$$\omega_{cf1} = 100\pi, \quad \zeta_1 = 0.9, \quad \omega_{cf2} = 20\pi, \quad \zeta_2 = 0.9,$$

where ω_{cf1} is the cut-off frequency and ζ_1 is the damping ratio for the encoder which measures the cart position and ω_{cf2} is the cut-off frequency and ζ_2 is the damping ratio for the encoders which measure the angular positions of the pendulum rods. The resulting filter allows us to

obtain the velocity states for the real-time experiments for use in our feedback controls as previously discussed in Chapter 3. However, we also explored other options to estimate these states.

4.2 Kalman Observer

The first observer we explored for the DIP system is the Kalman filter. This filter is constructed using linear systems, so we begin by considering a general linear system:

$$\begin{aligned}x'(t) &= Ax(t) + Bu(t) \\x(0) &= x_0.\end{aligned}\tag{4.3}$$

We also need to consider our output equation, that is, an equation returning what is being observed,

$$y(t) = Cx(t)\tag{4.4}$$

where $y \in \mathbb{R}^p$ and p is the number of observed states. In general, we can design a state observer \hat{x} so that it has the same dynamics as the original system which is given by

$$\hat{x}'(t) = A\hat{x}(t) + Bu(t) + G(y(t) - \hat{y}(t))\tag{4.5}$$

where

$$\hat{y}(t) = C\hat{x}(t).\tag{4.6}$$

The term $G(y(t) - \hat{y}(t))$ is the correction term. The observer gain G is designed to drive the state observer \hat{x} to the actual state x . To ensure that this happens, we define the error between the state observer and the actual state as

$$e(t) = \hat{x}(t) - x(t).\tag{4.7}$$

We differentiate this equation and use the state equation (4.3) and the observer equation (4.5) to obtain

$$\begin{aligned}e'(t) &= \hat{x}'(t) - x'(t) \\&= A\hat{x}(t) + Bu(t) + G(y(t) - \hat{y}(t)) - Ax(t) - Bu(t) \\&= A(\hat{x}(t) - x(t)) + G(Cx(t) - C\hat{x}(t)) \\&= (A - GC)e(t).\end{aligned}\tag{4.8}$$

If all of the eigenvalues of $(A - GC)$ have negative real parts, then the error e will approach zero, that is, the state observer \hat{x} will approach the actual state x . This is true even if there are large errors between the initial observer and the actual state. This is known as the Luenberger observer [4].

In the state feedback problem, we use the state observer for feedback, so

$$u(t) = K\hat{x}(t). \quad (4.9)$$

We substitute this into the state equation (4.3) to obtain

$$\begin{aligned} x'(t) &= Ax(t) + BK\hat{x}(t) \\ &= Ax(t) + BK(e(t) + x(t)) \\ &= (A + BK)x(t) + BKe(t). \end{aligned}$$

When we combine this with the error equation (4.8), we obtain the closed-loop system

$$\begin{pmatrix} x'(t) \\ e'(t) \end{pmatrix} = \begin{pmatrix} A + BK & BK \\ 0 & A - GC \end{pmatrix} \begin{pmatrix} x(t) \\ e(t) \end{pmatrix} \quad (4.10)$$

$$y(t) = \begin{pmatrix} C & 0 \end{pmatrix} \begin{pmatrix} x(t) \\ e(t) \end{pmatrix}. \quad (4.11)$$

We note that the system matrix for the closed-loop system is a block triangular matrix, and therefore the eigenvalues of the system are the eigenvalues of $A + BK$ and $A - GC$. Thus, the design of the state feedback and the state observer can be done independently. This is known as the separation principle [4].

We will now show that the state observer described above is the Kalman filter when white noise processes are added to the state and output equations following the presentation in [14]. We begin by examining the linear filtering problem given by

$$x'(t) = Ax(t) + Bu(t) + g(t)w(t) \quad (4.12)$$

$$y_k = Cx(t_k) + v_k \quad (4.13)$$

where measurements y are taken at discrete time points t_k and $w(t)$ and v_k are uncorrelated white noises with means zero and covariances \hat{Q} and \hat{R} , respectively. To solve this problem, the

time update equations are given by

$$\hat{x}' = A\hat{x}(t) + Bu(t) \quad (4.14)$$

$$P' = PA^T + AP + g\hat{Q}g^T \quad (4.15)$$

where the first equation is for the estimate and the second equation is for the error covariance. We solve these equations from $t = t_{k-1}$ to $t = t_k$ to obtain $\hat{x}_k^- = \hat{x}(t_k)$ and $P_k^- = P(t_k)$. Then, the measurement update equations are given by

$$K_k = P_k^- C^T (C P_k^- C^T + \hat{R})^{-1} \quad (4.16)$$

$$\hat{x}_k = \hat{x}_k^- + K_k [z_k - C\hat{x}_k^-] \quad (4.17)$$

$$P_k = [I - K_k C] P_k^- \quad (4.18)$$

where K_k is the Kalman gain. In the state observer problem, the measurement is continuous:

$$z(t) = Cx(t) + v(t). \quad (4.19)$$

If we assume that we use an integrating sensor, then the relationship between the discrete and the continuous measurement is

$$v_k = \int_{t_{k-1}}^{t_k} v(t) dt \quad (4.20)$$

and

$$\hat{R}_k = \frac{\hat{R}(t_k)}{h}, \quad (4.21)$$

where \hat{R}_k and $\hat{R}(t)$ are the covariances of the discrete and continuous white noise processes v_k and $v(t)$, and h is the sampling period. To derive the continuous-time Kalman filter equations, we first discretize the time update equations and then take the limit as the sampling period $h \rightarrow 0$. Let us denote $x_k = x(t_k)$, $x_{k+1} = x(t_k + h)$, and $P_k = P(t_k)$. The Euler's approximation for the time update equation (4.14) is given by

$$\frac{\hat{x}_{k+1}^- - \hat{x}_k}{h} = A\hat{x}_k + Bu_k \quad (4.22)$$

which can be rearranged as

$$\hat{x}_{k+1}^- = (I + hA)\hat{x}_k + hBu_k. \quad (4.23)$$

By taking the Euler's approximation for the second time update equation (4.15) and rearranging, we have

$$P_{k+1}^- = P_k + hP_k A^T + hAP_k + hg\hat{Q}g^T. \quad (4.24)$$

For the Kalman gain K_k , we combine (4.16) and (4.21) to get

$$K_k = P_k^- C^T \left(C P_k^- C^T + \frac{\hat{R}_k}{h} \right)^{-1} \quad (4.25)$$

$$K_k = h P_k^- C^T (h C P_k^- C^T + \hat{R}_k)^{-1}, \quad (4.26)$$

$$\frac{K_k}{h} = P_k^- C^T (h C P_k^- C^T + \hat{R}_k)^{-1} \quad (4.27)$$

We then take the limit as $h \rightarrow 0$ to obtain

$$\lim_{h \rightarrow 0} \frac{K_k}{h} = P_k^- C^T \hat{R}_k^{-1} \quad (4.28)$$

and we note that

$$\lim_{h \rightarrow 0} K_k = h P_k^- C^T \hat{R}_k^{-1} = 0. \quad (4.29)$$

Now considering the covariance equations, we substitute (4.18) into the discrete time covariance update equation (4.24) to find

$$P_{k+1}^- = [I - K_k C] P_k^- + h[I - K_k C] P_k^- A^T + hA[I - K_k C] P_k^- + hg\hat{Q}g^T. \quad (4.30)$$

We rearrange this as

$$\frac{P_{k+1}^- - P_k^-}{h} = \frac{-K_k C P_k^-}{h} + [I - K_k C] P_k^- A^T + A[I - K_k C] P_k^- + g\hat{Q}g^T. \quad (4.31)$$

We take the limit of both sides of this equation as $h \rightarrow 0$ to obtain

$$P'(t) = P A^T + A P - P C^T \hat{R}^{-1} C P + g\hat{Q}g^T \quad (4.32)$$

which we recognize as the matrix Riccati differential equation.

We now consider the state update equation (4.17) and shift the index k by $k + 1$ to obtain

$$\hat{x}_{k+1} = \hat{x}_{k+1}^- + K_{k+1}[z_{k+1} - C\hat{x}_{k+1}^-]. \quad (4.33)$$

We substitute the discrete state equation (4.23) for \hat{x}_{k+1}^- and then rearrange to find

$$\hat{x}_{k+1} = (I + hA)\hat{x}_k + hBu_k + K_{k+1}[z_{k+1} - C(I + hA)\hat{x}_k - hCBu_k] \quad (4.34)$$

$$\frac{\hat{x}_{k+1} - \hat{x}_k}{h} = A\hat{x}_k + Bu_k + \frac{K_{k+1}}{h}[z_{k+1} - C(I + hA)\hat{x}_k - hCBu_k]. \quad (4.35)$$

Taking the limit as $h \rightarrow 0$ of this equation yields

$$\hat{x}'(t) = A\hat{x} + Bu + PC^T\hat{R}^{-1}[z - C\hat{x}] \quad (4.36)$$

which is the state observer equation with observer gain $G = PC^T\hat{R}^{-1}$. In summary, equations (4.36) and (4.32) define the continuous-time Kalman estimator, also known as the Kalman-Bucy filter [14].

4.2.1 An Application Example

As an exercise, we again consider the system from Section 3.3.1. Recall that the nonlinear system was given as

$$x''(t) + \sin(x(t)) = u(t), \quad (4.37)$$

and we now consider the output equation

$$y = x. \quad (4.38)$$

We still want to drive the states x and x' to zero using the control u . However, since the only state we measure is x , we also want to construct an observer for the system to be used in the state feedback.

We again begin by linearizing our system about the solution $x = 0$, $x' = 0$. Let $x(t) = x_1(t)$ and $x'(t) = x_2(t)$, and $X(t) = [x_1(t), x_2(t)]^T$. Then,

$$X'(t) = AX(t) + Bu \quad (4.39)$$

$$y = CX(t) \quad (4.40)$$

where

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

$$B = \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$

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

We consider the same cost functional as previously given,

$$J(X_0, u) = \int_0^\infty (X^T Q X + R u^2) dt \quad (4.41)$$

with $Q = \text{diag}(400, 300)$ and $R = 10$, and will use the power series based controller from Section 3.4.1 given by

$$u^*(X) = -R^{-1} B^T \left[P X - (A_0^T - P B R^{-1} B^T)^{-1} P f_3(X) \right]. \quad (4.42)$$

The observer equations are given by

$$\hat{X}'(t) = A \hat{X} + B u + \hat{P} C^T \hat{R}^{-1} [y - C \hat{X}] \quad (4.43)$$

where \hat{P} is the solution to

$$\hat{P}'(t) = \hat{P} A^T + A \hat{P} - \hat{P} C^T \hat{R}^{-1} C \hat{P} + g \hat{Q} g^T. \quad (4.44)$$

We implement this system and observer using MATLAB. The resulting trajectories for the states and the control effort are shown in Figure 4.2 for initial states of the system and observer given by $X(0) = [2, 1]^T$ and $\hat{X}(0) = [1, 3]^T$, respectively. We can see that the observer quickly tracks the system and Figure 4.3 shows the absolute error between the observer and the system states.

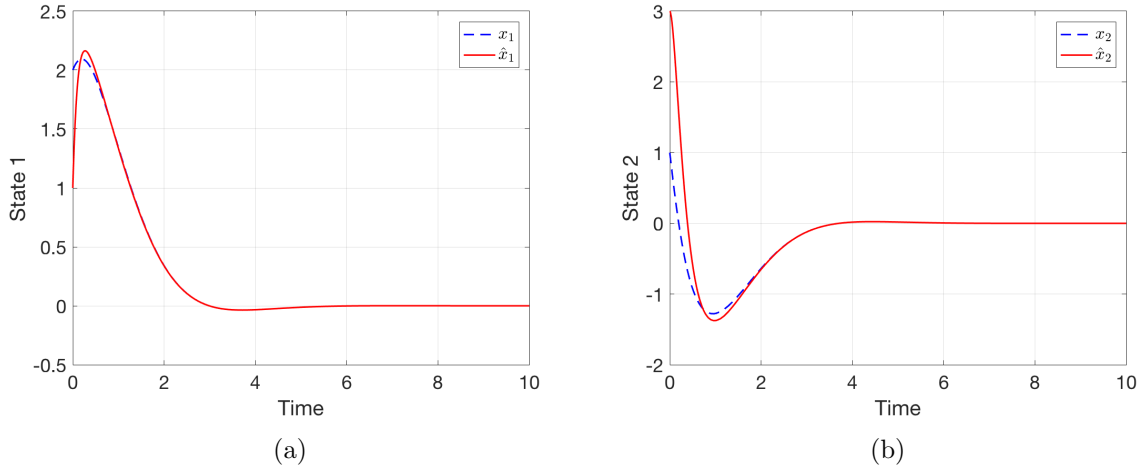


Figure 4.2: The Kalman observer applied to the example problem using the power series based control: (a) x_1 and (b) x_2 .

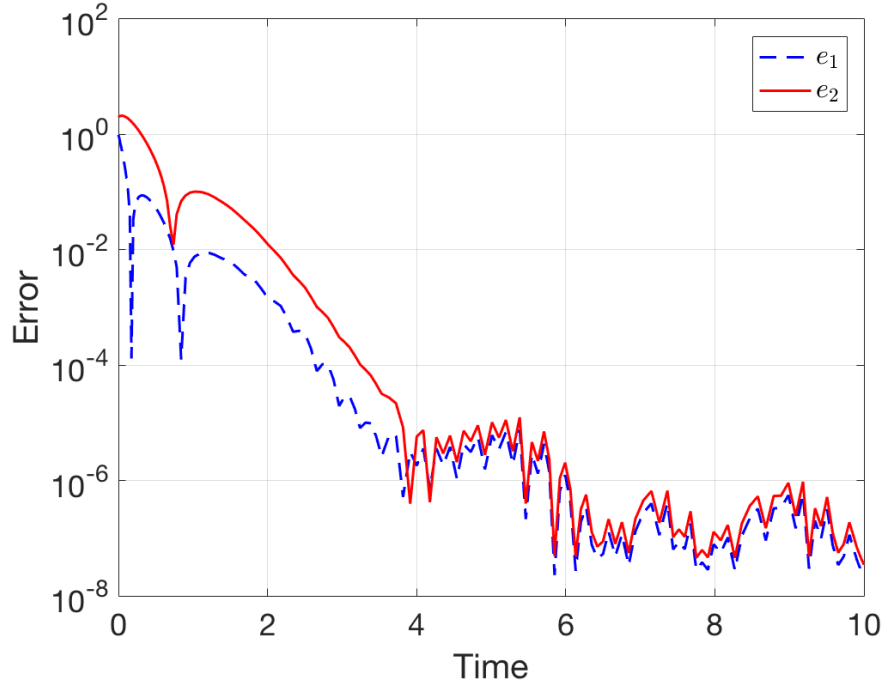


Figure 4.3: Absolute error between the observer and system state values for the example problem using the Kalman observer.

We also compute this example after adding white noise to the model and output equations,

$$x''(t) + \sin(x(t)) + w(t) = u(t), \quad (4.45)$$

$$y = x + v(t), \quad (4.46)$$

where w and v are gaussian white noises with mean zero and covariances \hat{Q} and \hat{R} . For the MATLAB simulation, we let $\hat{Q} = 1$ and $\hat{R} = 0.01$. The resulting trajectories for the state and observer variables are shown in Figure 4.4 and the absolute error is shown in Figure 4.5. We can see that the observer does track the states well, though the absolute error is not as small as before due to the presence of the white noise terms.

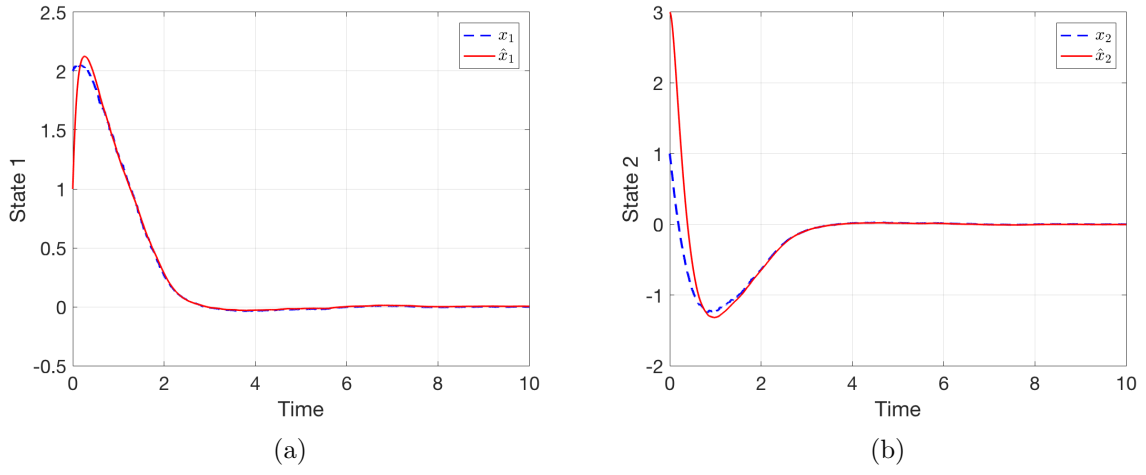


Figure 4.4: The Kalman observer applied to the example problem including white noise: (a) x_1 and (b) x_2 .

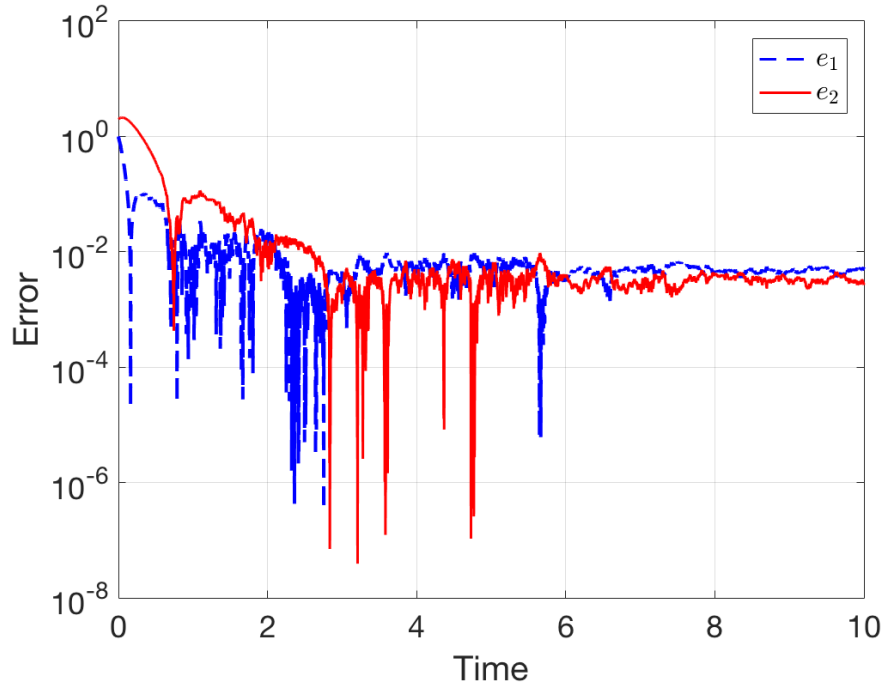


Figure 4.5: Absolute error between the observer and system state values for the example problem using the Kalman observer including white noise.

4.2.2 Application to the DIP System

For the DIP system, the output equation is

$$y = CX \quad (4.47)$$

where the matrix C is given by

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

Thus y returns the three measured position states of our system. For our MATLAB implementation, we compute the solution to the Riccati equation using the `lqr` function. We run the MATLAB simulation of the observer and the linearized DIP model with covariances $\hat{Q} = 10^{-7}$ and $\hat{R} = 10^{-6}$ for the white noises processes of the model and output equation, respectively. We use the power series based controller with weighting matrices $Q = \text{diag}(30, 350, 100, 0, 0, 0)$ and $R = 0.1$ for the cost function. The resulting state trajectories for the system and the observer are shown in Figure 4.6, and the control effort is shown in Figure 4.7.

We can see that the observer quickly and accurately tracks the states of the system for the simulation. However, we found that when we attempted to apply this observer to the real-time experiment, we were unable to stabilize the pendulum at all. Various values of the covariances for the white noise processes were tried for the real-time experiment, and in an effort to better predict what values may improve the system performance, simulations were run using the full nonlinear DIP model presented in Chapter 2. In these simulations, we were also unable to have the observer track the state variables. Therefore, we decided to apply a different observer made for nonlinear systems to the DIP system.

4.3 Nonlinear Observer

We next looked at an observer that requires no linearization of our model. This observer is an extension of the Luenberger observer for use with nonlinear continuous time systems and was proposed by [8].

We begin by considering a single-input-single-output nonlinear system given by

$$x'(t) = f(x(t)) + g(x(t))u(t) \quad (4.48)$$

$$y(t) = h(x(t)) \quad (4.49)$$

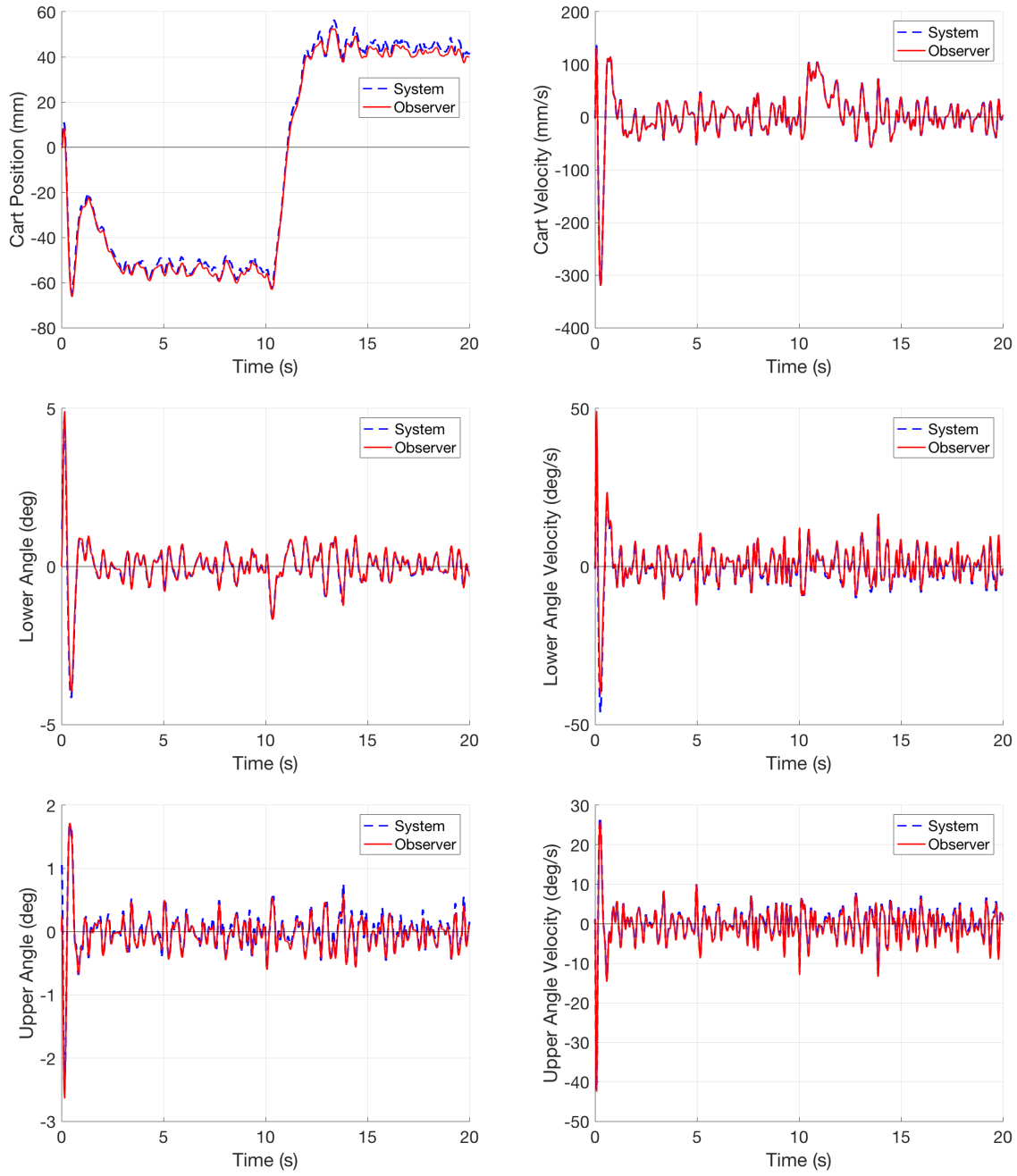


Figure 4.6: Simulation results for the states of the observer and the linearized DIP system using the Kalman observer and the power series based controller.

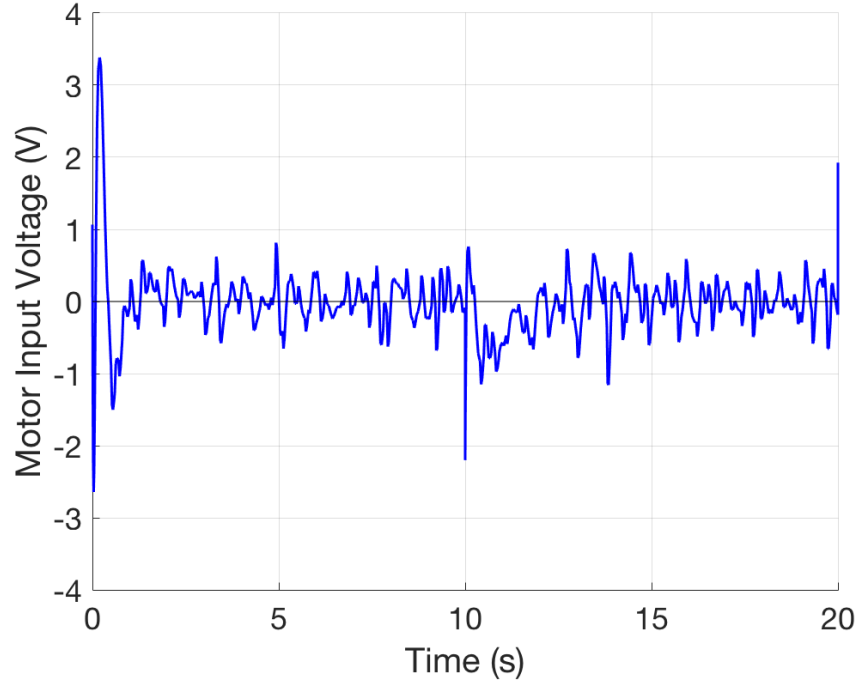


Figure 4.7: Control effort for the simulation using the Kalman observer and the power series based controller.

where $x(t) \in \mathbb{R}^n$ is the state vector, h is a \mathbb{C}^∞ real-valued function, f and g are \mathbb{C}^∞ real-valued vector fields, and u is a uniformly bounded input function.

Following [17], we define the derivative of a function h along a vector field f as

$$L_f h(x) = \langle \nabla h(x), f(x) \rangle = \sum_{i=1}^n \frac{\partial h(x)}{\partial x_i} f_i(x). \quad (4.50)$$

To compute the k -times derivative of h along f , we write $L_f^k h(x)$ and compute it using the recursion

$$L_f^k h(x) = L_f(L_f^{k-1} h(x))(x) \quad (4.51)$$

$$L_f^0 h(x) = h(x). \quad (4.52)$$

We also write the observability matrix $\mathcal{Q}(x)$ of (4.48) and (4.49) as

$$\mathcal{Q}(x) = \frac{d}{dx} \begin{bmatrix} h(x) \\ L_f h(x) \\ \vdots \\ L_f^{n-1} h(x) \end{bmatrix} = \frac{d\phi(x)}{dx}. \quad (4.53)$$

Note that, in the linear case where $h(x) = Cx$ and $f(x) = Ax$, $\mathcal{Q}(x)$ reduces to the usual observability matrix given by

$$\mathcal{Q}(x) = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix} \quad (4.54)$$

since

$$\begin{aligned} \frac{d}{dx} h(x) &= C \\ \frac{d}{dx} L_f h(x) &= \frac{dh}{dx} f(x) = CA \\ \frac{d}{dx} L_f^2 h(x) &= \frac{dL_f h}{dx} f(x) = CA^2 \\ \frac{d}{dx} L_f^{n-1} h(x) &= \frac{dL_f^{n-2} h}{dx} f(x) = CA^{n-1}. \end{aligned}$$

Lastly, we say that the system of (4.48) and (4.49) has relative degree r at a point x^0 if

1. $L_g L_f^k h(x) = 0$ for all x in a neighborhood of x^0 and all $k < r - 1$
2. $L_g L_f^{r-1} h(x^0) \neq 0$

as defined by [17].

We can now compute an observer for (4.48) and (4.49) which is given in the following convergence theorem by [8].

Theorem 1 *Let $\mathcal{Q}(x)$ be the observability matrix associated with the pair $(f(x), h(x))$ of functions in (4.48) and (4.49). Assume $u(t)$ is a real-valued, uniformly bounded function for all $t \geq 0$, and the triple $(f(x), g(x), h(x))$ has relative degree n . If $\mathcal{Q}(x)$ has full rank for all $x \in \mathbb{R}^n$*

and

$$\begin{aligned} \sup_{\|u\| \leq M} & \|L_f^n h(\phi^{-1}(\zeta_1)) + u L_g L_f^{n-1} h(\phi^{-1}(\zeta_1)) - L_f^n h(\phi^{-1}(\zeta_2)) \\ & - u L_g L_f^{n-1} h(\phi^{-1}(\zeta_2))\| \leq \gamma_g \|\zeta_1 - \zeta_2\|^\delta \end{aligned}$$

with $\delta \in (0, 1]$, for all $\zeta_1, \zeta_2 \in \mathbb{R}^n$ and γ_g is the Holder constant, then there exists a finite gain vector $K \in \mathbb{R}^n$ such that the solution of the following system of equations

$$\begin{aligned} \hat{x}'(t) &= f(\hat{x}(t)) + g(\hat{x}(t))u(t) + \mathcal{Q}(\hat{x}(t))^{-1}K[y(t) - h(\hat{x}(t))] \\ \hat{x}(0) &= \bar{x} \end{aligned} \tag{4.55}$$

has the following properties:

1. for $\delta \in (0, 1)$

$$\lim_{t \rightarrow \infty} \|\hat{x}(t) - x(t)\| \leq \epsilon$$

for any $\epsilon > 0$, for all $\hat{x}(0) \in \mathbb{R}^n$, and

2. for $\delta = 1$

$$\lim_{t \rightarrow \infty} \|\hat{x}(t) - x(t)\| = 0$$

for all $\hat{x}(0) \in \mathbb{R}^n$.

[8] then shows that the gain matrix K can be computed simply by first choosing λ_i such that $\text{Real}\{\lambda_i\} < -\gamma$ for all $i = 1, \dots, n$. Then compute the polynomial

$$(\lambda - \lambda_1)(\lambda - \lambda_2) \dots (\lambda - \lambda_n) = \lambda^n + K_1 \lambda^{n-1} + K_2 \lambda^{n-2} + \dots + K_n \tag{4.56}$$

and take $K = [K_1, K_2, \dots, K_n]^T$.

4.3.1 An Application Example

We will now go through a simple example taken from [17] to show how to construct this observer. Consider a one-link robot arm whose rotary motion is controlled by an elastically coupled actuator. This elasticity can be modeled by inserting a torsional spring at each joint. An illustration of such a model is shown in Figure 4.8. We describe this system mathematically using two second order differential equations: one for the mechanical balance of the actuator shaft and one for the mechanical balance of the link. Let q_1 and q_2 be the angular positions of

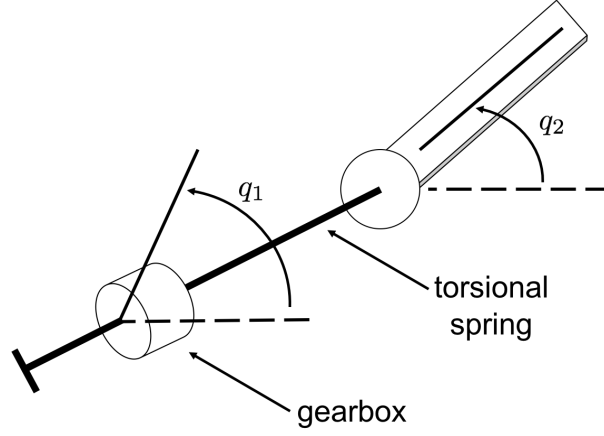


Figure 4.8: A one-link robot arm.

the actuator shaft and link, respectively, so the equations are

$$J_1 q_1'' + F_1 q_1' + \frac{H}{N} \left(q_2 - \frac{q_1}{N} \right) = T \quad (4.57)$$

$$J_2 q_2'' + F_2 q_2' + H \left(q_2 - \frac{q_1}{N} \right) + mgd \cos(q_2) = 0. \quad (4.58)$$

The parameters J_1 and F_1 represent the inertia and viscous friction constants, H is the elasticity constant, N is the transmission gear ratio, T is the torque produced at the actuator axis, m is the mass of the link, g is gravity, and d is the position of the center of gravity of the link.

We let the state vector for our system be $x = [q_1, q_2, q_1', q_2']^T$ and the input be $u(t) = T$ and rewrite the system as

$$x'(t) = f(x(t)) + g(x(t))u(t)$$

$$x'(t) = \begin{bmatrix} x_3(t) \\ x_4(t) \\ -\frac{H}{J_1 N^2} x_1(t) + \frac{H}{J_1 N} x_2(t) - \frac{F_1}{J_1} x_3(t) \\ \frac{H}{J_2 N} x_1(t) - \frac{H}{J_2} x_2(t) - \frac{mgd}{J_2} \cos(x_2(t)) - \frac{F_2}{J_2} x_4(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{J_1} \\ 0 \end{bmatrix} u(t) \quad (4.59)$$

For the output of the system, we choose the angular position of the link q_2 , that is,

$$y(x(t)) = h(x(t)) = x_2(t). \quad (4.60)$$

We calculate the quantities

$$\begin{aligned}
L_f h(x) &= \sum_{i=1}^4 \frac{\partial h(x)}{\partial x_i} f_i(x) = f_2(x) = x_4 \\
L_f^2 h(x) &= \sum_{i=1}^4 \frac{\partial L_f h(x)}{\partial x_i} f_i(x) = f_4(x) \\
L_f^3 h(x) &= \sum_{i=1}^4 \frac{\partial L_f^2 h(x)}{\partial x_i} f_i(x) = \frac{\partial f_4(x)}{\partial x_1} f_1(x) + \frac{\partial f_4(x)}{\partial x_2} f_2(x) + \frac{\partial f_4(x)}{\partial x_4} f_4(x) \\
&= \frac{H}{J_2 N} x_3 + \left(-\frac{H}{J_2} + \frac{mgd}{J_2} \sin(x_2) \right) x_4 - \frac{F_2}{J_2} f_4(x).
\end{aligned}$$

We note that, since $f_4(x)$ does not depend on x_3 ,

$$\begin{aligned}
L_g h(x) &= L_g L_f h(x) = L_g L_f^2 h(x) = 0 \\
L_g L_f^3 h(x) &= \frac{\partial L_f^3 h(x)}{\partial x_3} g_3(x) = \frac{H}{J_1 J_2 N}.
\end{aligned}$$

Therefore, the system has a relative degree $n = 4$ at each point x^0 in the state space, a necessary condition to use this observer. We then compute the observability matrix \mathcal{Q} as

$$\begin{aligned}
\mathcal{Q}(x) &= \frac{d}{dx} \begin{bmatrix} h(x) \\ L_f h(x) \\ L_f^2 h(x) \\ L_f^3 h(x) \end{bmatrix} \\
&= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ \frac{H}{J_2 N} & \frac{mgd}{J_2} \sin(x_2) - \frac{H}{J_2} & 0 & \frac{-F_2}{J_2} \\ \frac{-F_2 H}{J_2^2 N} & \frac{mgd}{J_2} \cos(x_2) x_4 + \frac{F_2 H}{J_2^2} - \frac{F_2 mgd}{J_2^2} \sin(x_2) & \frac{H}{J_2 N} & \frac{-H^2}{J_2} + \frac{mgd}{J_2} \sin(x_2) + \frac{F_2^2}{J_2^2} \end{bmatrix}.
\end{aligned} \tag{4.61}$$

We can compute $\mathcal{Q}(x)^{-1}$ using Maple, and then the observer equations from (4.55) are the

following:

$$\hat{x}_1' = \hat{x}_3 + \left(\frac{N(H - mgd \sin(\hat{x}_2))}{H} K_1 + \frac{F_2 N}{H} K_2 + \frac{J_2 N}{H} K_3 \right) (y - \hat{x}_2) \quad (4.62)$$

$$\hat{x}_2' = \hat{x}_4 + K_1 (y - \hat{x}_2) \quad (4.63)$$

$$\begin{aligned} \hat{x}_3' = & -\frac{H}{J_1 N^2} \hat{x}_1 + \frac{H}{J_1 N} \hat{x}_2 - \frac{F_1}{J_1} \hat{x}_3 + \frac{1}{J_1} u \\ & + \left(\frac{-N mgd \cos(\hat{x}_2) \hat{x}_4}{H} K_1 + \frac{N(H - mgd \sin(\hat{x}_2))}{H} K_2 + \frac{F_2 N}{H} K_3 + \frac{J_2 N}{H} K_4 \right) (y - \hat{x}_2) \end{aligned} \quad (4.64)$$

$$\hat{x}_4' = \frac{H}{J_2 N} \hat{x}_1 - \frac{H}{J_2} \hat{x}_2 - \frac{mgd}{J_2} \cos(\hat{x}_2) - \frac{F_2}{J_2} \hat{x}_4 + K_2 (y - \hat{x}_2). \quad (4.65)$$

We implement this system and observer in MATLAB with an input of $u(t) = 2 + \sin(t)$. All constants are set equal to one for simplicity, and to compute K we let $\lambda_1 = -2$, $\lambda_2 = -2.1$, $\lambda_3 = -2.2$, $\lambda_4 = -2.3$. We initialize the system to $x(0) = [2, 1, -1.25, -1]^T$ and the observer to $\hat{x}(0) = [3, 2, -2, 0]^T$. The resulting system and observer are shown in Figure 4.9. We can see that the observer rapidly approaches the system and accurately predicts the state variables.

4.3.2 Application to the DIP System

To apply this observer to the DIP system, we began by computing the derivatives $L_f h, L_f^2 h, \dots, L_f^5 h$ analytically using Maple. We then used Maple to compute the matrix \mathcal{Q} and then imported that equation to MATLAB for use in our numerical solver. Unfortunately, due to the highly complicated nonlinear expressions in our system of equations and thus the very long expressions in \mathcal{Q} , attempting to evaluate the observer equations crashed the MATLAB program. Note, this was initially tried on a MacBook Pro laptop with a 2.4 GHz Intel Core i5 processor and 16 GB 1600 MHz DDR3 of RAM.

Knowing that we ultimately want to run this observer for the real-time experiments, we next tried to compute the x derivative necessary for \mathcal{Q} in MATLAB using the complex step method. We can then import the comparatively simpler analytic expressions for $L_f h, L_f^2 h, \dots, L_f^5 h$ into MATLAB and calculate \mathcal{Q} there.

4.3.2.1 The Complex Step Method

The complex step method is a method of approximating partial derivatives using complex functions. The method was first originated in the work of Lyness and Moler [22] and Lyness [21]. It was then rediscovered by Squire and Trapp [37] and has become popular in aerodynamic optimizations [2, 3, 23] due to its high degree of accuracy, robustness, and surprisingly simple

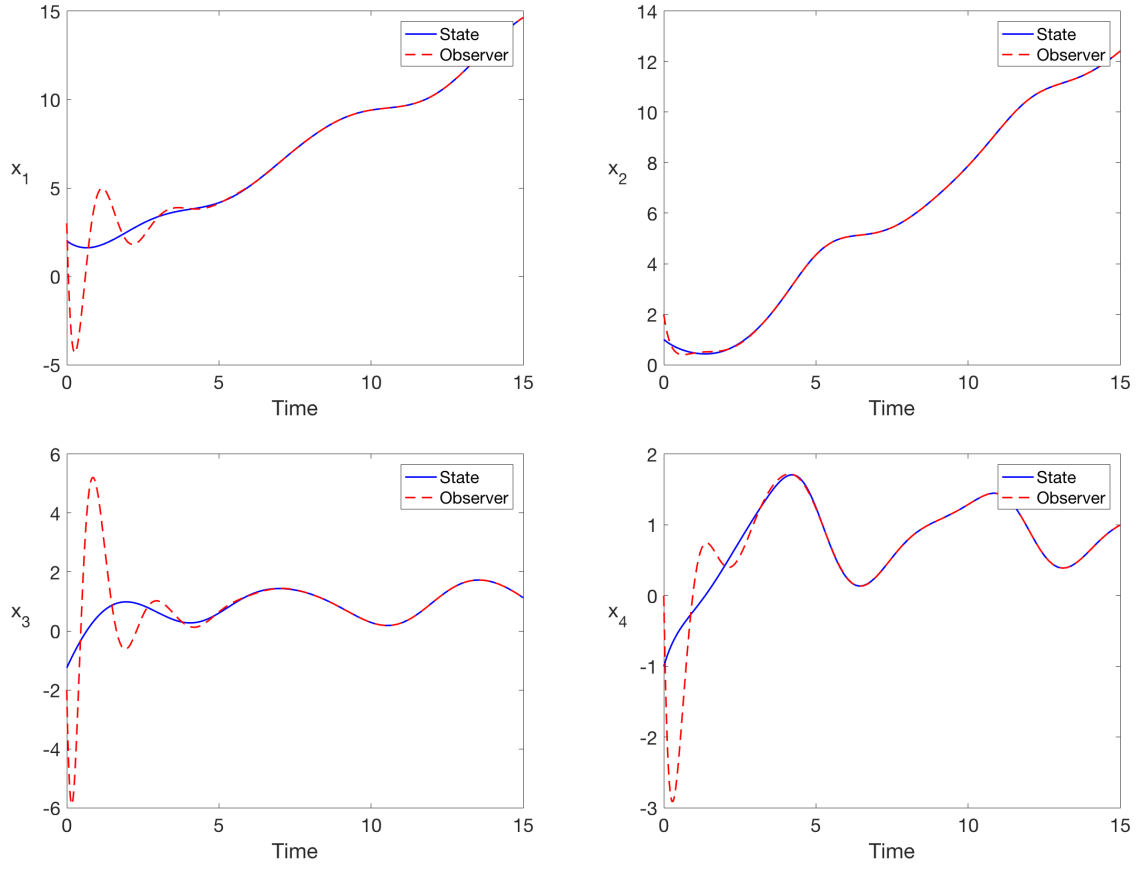


Figure 4.9: System and observer performance for the one-link robot arm.

implementation. Martins et al. compared the performance to automatic differentiation as well as finite difference methods, and we will follow the procedure they presented [23].

Consider a complex number $z = x + iy$ and a function

$$f(z) = u(x + iy) + iv(x + iy) \quad (4.66)$$

where f is an analytic complex function with real part u and imaginary part v . Since f is analytic, it satisfies the Cauchy-Riemann equation and thus

$$u_x(x, y) = v_y(x, y) \quad (4.67)$$

$$u_y(x, y) = -v_x(x, y). \quad (4.68)$$

Using the definition of the derivative in (4.67), we have

$$u_x(x, y) = \lim_{h \rightarrow 0} \frac{v(x + i(y + h)) - v(x + iy)}{h}. \quad (4.69)$$

Since we want to take the derivative of a real function, we now restrict the function f to the real axis such that

$$y = 0 \quad (4.70)$$

$$v(x, 0) = \text{Im}(f(x)) = 0 \quad (4.71)$$

$$f(x, 0) = u(x, 0). \quad (4.72)$$

Then we have

$$\frac{df}{dx} = u_x(x, 0) \quad (4.73)$$

$$= \lim_{h \rightarrow 0} \frac{v(x + ih) - v(x)}{h} \quad (4.74)$$

$$= \lim_{h \rightarrow 0} \frac{v(x + ih)}{h}. \quad (4.75)$$

Thus, by taking a small h , the first derivative of f can be approximated by

$$\frac{df}{dx} \approx \frac{\text{Im}(f(x + ih))}{h}. \quad (4.76)$$

This is called the complex step derivative approximation. It has clear advantages over the usual finite difference approximation since it requires just one function evaluation and has no subtraction and therefore no subtraction cancellation errors.

To determine the error of this method, we consider the Taylor series expansion of f given by

$$f(x + ih) = f(x) + ihf'(x) - \frac{h^2}{2!}f''(x) - \frac{ih^3}{3!}f'''(x) + \dots \quad (4.77)$$

Taking the imaginary parts of both sides and dividing by h we obtain

$$f'(x) = \frac{\text{Im}(f(x + ih))}{h} + \frac{h^2}{3!}f'''(x) + \dots \quad (4.78)$$

Therefore, we have $\mathcal{O}(h^2)$ error, and since there is no subtractive error, we can reduce h to a very small value in order to achieve high accuracy for the derivative. It was shown by Banks et al. that h could be reduced to a value of $h_{crit} = 10^{-320}$ on machines with a machine accuracy of 10^{-324} [5].

We also note that this approximation can also be extended to vector valued functions. For

a function $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ and step size h , the complex step approximation of the Jacobian is given by

$$J \approx \frac{1}{h} \begin{bmatrix} f_1(\mathbf{x} + ih\mathbf{e}_1) & \cdots & f_1(\mathbf{x} + ih\mathbf{e}_j) & \cdots & f_1(\mathbf{x} + ih\mathbf{e}_n) \\ \vdots & & \vdots & & \vdots \\ f_p(\mathbf{x} + ih\mathbf{e}_1) & \cdots & f_p(\mathbf{x} + ih\mathbf{e}_j) & \cdots & f_p(\mathbf{x} + ih\mathbf{e}_n) \\ \vdots & & \vdots & & \vdots \\ f_m(\mathbf{x} + ih\mathbf{e}_1) & \cdots & f_m(\mathbf{x} + ih\mathbf{e}_j) & \cdots & f_m(\mathbf{x} + ih\mathbf{e}_n) \end{bmatrix} \quad (4.79)$$

where f_p is the p -th component of the function \mathbf{f} and \mathbf{e}_j is the j -th unit vector in \mathbb{R}^n [19].

4.3.2.2 Applying the Complex Step Method to the DIP

We implemented the complex step method in MATLAB by evaluating a function f at an imaginary step $x + ih$, taking the imaginary part of the result, and dividing by h . To compute \mathcal{Q} as given in (4.53), we use the complex step approximation of the Jacobian from (4.79) using the analytic formulations of the derivatives $L_f h, L_f^2 h, \dots, L_f^5 h$ computed in Maple. Using these comparatively simpler expressions with the approximate derivatives allowed us to successfully compute \mathcal{Q} for given values of the state variables of x . However, individual evaluations of \mathcal{Q} took approximately 30 seconds to complete, much too slow to be of use for real-time implementation. We also noted at this time that for some typical values of the states, \mathcal{Q} was very ill-conditioned, having a condition number on the order of 10^6 . This is concerning since our observer formulation requires taking the inverse of \mathcal{Q} and ill-conditioned matrices have large errors in inverse calculations.

In an effort to find a faster way to compute \mathcal{Q} , we also considered applying the complex step method to compute the derivatives $L_f h, L_f^2 h, \dots, L_f^5 h$. We note that to compute these derivatives using the complex step method and then compute \mathcal{Q} will require a second derivative. Consider the second row of \mathcal{Q} :

$$\begin{aligned} \frac{d}{dx}(L_f h(x)) &= \frac{d}{dx} \langle \nabla h(x), f(x) \rangle \\ &= \frac{d}{dx} \sum_{i=1}^6 \left(\frac{\partial h(x)}{\partial x_i} f_i(x) \right). \end{aligned}$$

Therefore, we need to consider how to compute the second derivative using the complex step method. Once again we take the Taylor series of the analytic function f :

$$f(x + ih) = f(x) + ihf'(x) - \frac{h^2}{2!}f''(x) - \frac{ih^3}{3!}f'''(x) + \frac{h^4}{4!}f^{(4)}(x) + \dots \quad (4.80)$$

Now to obtain an expression for the second derivative of f we take the real parts of both sides of the equation and solve for $f''(x)$ to obtain

$$\frac{h^2}{2!}f''(x) = \frac{2}{h^2}(f(x) - \operatorname{Re}(f(x + ih))) + \frac{2h^2}{4!}f^{(4)}(x) + \dots \quad (4.81)$$

Therefore, with $\mathcal{O}(h^2)$ error, the complex step approximation for the second derivative is

$$f''(x) \approx \frac{2}{h^2}(f(x) - \operatorname{Re}(f(x + ih))). \quad (4.82)$$

This approximation now contains a difference, so we again need to be concerned about having error due to subtractive cancellation. Knowing that for our case the matrix \mathcal{Q} is ill-conditioned for some values of the states, introducing errors in calculating the derivatives is highly undesirable.

4.4 Summary of the Observer Used for the DIP System

As we have seen in this chapter, after exploring two other options to construct an observer for the DIP system, we have not yet found a method that works better than the Quanser provided low pass derivative filter for the real-time implementation. We have been unable to find suitable parameters for the Kalman observer in order to implement it on the real-time system, and the nonlinear observer has not been computationally practical to implement. Therefore, all presented real-time results in this work have been obtained using the low pass derivative filter from Section 4.1. Those results can be seen in Chapter 3. For our future work we are interested in continuing to explore adjustments that could be made to the Kalman and nonlinear observers presented here for use with the DIP system as well as other state estimators.

Chapter 5

Modeling of Radio Frequency Interference for SATOPS Applications

In this chapter, we will describe the radio frequency interference (RFI) prediction models which are then implemented for verification in MATLAB. This work was conducted with collaborators at The Catholic University of America, Intelligent Fusion Technology, Inc., and the Air Force Research Laboratory and has been presented in [26] and [27].

5.1 Background

Radio frequency interference sources are classified into two categories: friendly and unfriendly. Friendly sources are unintentional interferers whose data transmissions are within the field of view of the victim's SATCOM links. Figure 5.1 depicts how this sort of RFI may occur for both the uplink and downlink cases. Depending on the transmitting carrier frequencies, these sources can cause co-channel interference (CCI) or adjacent channel interference (ACI). See Figure 5.2 for a graphical representation of how the power and frequency of different signals interact to create interference. Common friendly sources of RFI include personal communication systems such as 3G, 4G, WiFi, and WiMAX. Unfriendly sources of RFI are caused by intentional jamming signals. These could be classical jammers which use high powered signals intended to shut down an entire USB SATOPS system operation or significantly reduce such a system's efficiency; or these could be smart jammers which monitor USB SATOPS data links and use weak powered signals to degrade the quality of signal for more specific channels in use. The models presented here are capable of detecting and predicting RFI signals from both friendly and unfriendly sources.

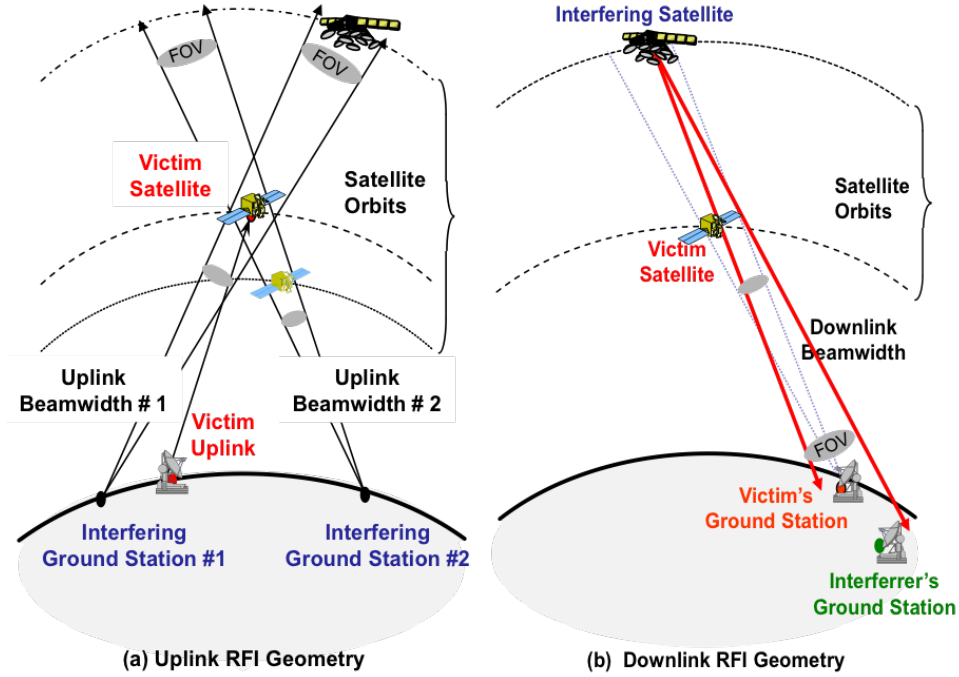


Figure 5.1: Typical RFI scenarios involving friendly sources.

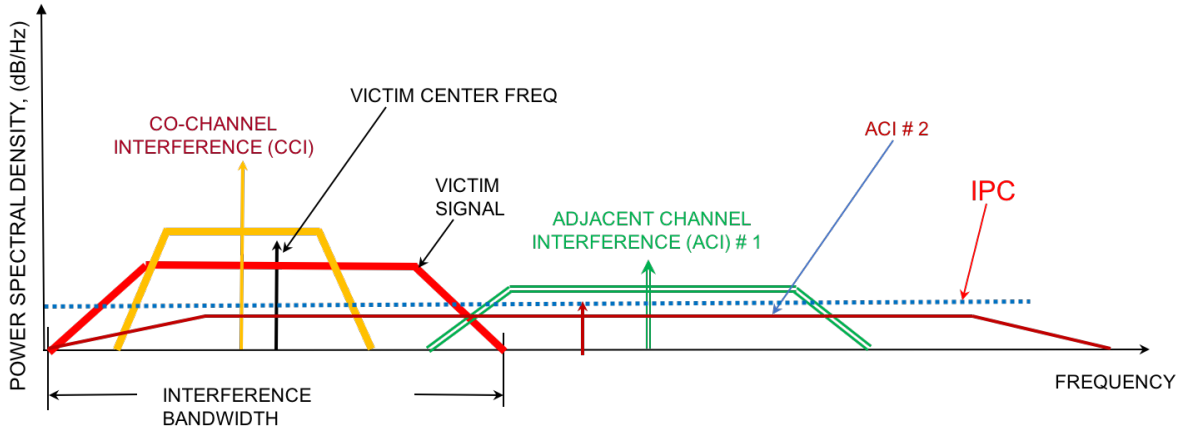


Figure 5.2: Interference bandwidth for evaluating potential RFI events and the corresponding CCI and ACI scenarios.

In this work we also consider RFI caused by two types of interfering signals. Wideband (WB) interference refers to signals whose energy is spread over a wide band of frequencies that interfere with the system. Continuous wave (CW) interference refers to signals composed of an electromagnetic wave with constant amplitude and frequency which generally causes less interference since it is active on a particular frequency band.

5.1.1 The RFI Detection Problem

For our RFI detection models, we assume that the interfering link is operating and that the following four considerations are accounted for.

First is the geometric consideration. Simply put, the victim receiver must be within the field of view of an interferer's transmitting antenna, and, simultaneously, the interferer must be within the field of view of the victim receiver's antenna. This necessary geometry is easily seen in Figure 5.1.

Next is the frequency consideration. For RFI to occur, any interfering link must have at least part of its signal power spectrum within the victim receiver's interference bandwidth, as seen in Figure 5.2. This interference bandwidth is based on the power spectral density of the victim signal. The frequency separation between the victim and interfering receivers and the spectral spreading of the interfering signal determines the contribution to the interference power, and the spectral spreading of the interfering signal is dependent on its power spectral density. This is based on its waveform which includes the data rate, coding, and modulation scheme.

Then we have the interfering power consideration. In order to declare that a potential RFI event is to occur, the RFI effects from all interfering links, both friendly and unfriendly, must exceed the interference protection criteria (IPC). The IPC is usually determined from the victim's receiver operating threshold or given by national or international regulatory bodies, such as the NTIA, FCC, or ITU.

Lastly is the interfering time consideration. The time duration (or time factor) of the effects from all the interfering links must be sufficient to cause an unacceptable level of degradation in the victim receiver performance in order to declare an RFI event has been detected. This time factor should be included in the receiver's synchronization loops in order to ensure that the detected event is really caused by an RFI source.

The RFI detection models shown here incorporate the above mentioned considerations for the interference links.

5.1.2 The RFI Prediction Problem

For the prediction of an RFI event, the models must incorporate the same considerations as listed for the detection problem. The power of the interfering links must be strong enough to exceed the IPC and must be long enough to cause the synchronization loops to go out-of-lock. The operator may not declare an RFI event if the loops can reacquire and track the desired signal quickly. Furthermore, the following four considerations must also be accounted for in the prediction problem.

First we have the carrier tracking performance consideration. In order for an RFI event to occur, the interference signals must exceed the carrier threshold power level and remain within

the carrier tracking loop bandwidth for a certain amount of time before the carrier loop drops lock. Also, the carrier loop must be unable to require the carrier signal within a particular time window.

Next we have the sub-carrier tracking performance consideration. This is the same as the previous carrier tracking consideration but applied to the sub-carrier instead.

Next is the symbol timing synchronization performance consideration, which is again similar to the previous considerations now applied to the symbol signal-to-noise ratio (SNR) and the symbol synchronization loop.

Finally, we have the hop/de-hop/spread/de-spread consideration. This applies to victim receivers using waveforms that employ the spread spectrum or frequency hop technique. With this technique, the signal will rapidly hop between multiple sub-frequencies of a particular frequency band in a predetermined order. This helps protect the signal from interference at a specific frequency since the signal only uses a specific frequency for a short amount of time and is very common for both military and civilian use. Taking this into account, for an RFI event to occur the interfering signals must stay within the spreading or hop bandwidth long enough to cause the de-spread or de-hop synchronizer to drop lock.

The synchronization loops are implemented as phase locked loops (PLL). In general, a PLL is a feedback control system which generates an output signal whose phase is related to that of the input signal. Therefore they can be used to synchronize signals by locking on to the phase of the signal we want to track. PLLs contain a phase detector, a voltage controlled oscillator (VCO), and a loop filter linked via feedback. A block diagram of a basic PLL is shown in Figure 5.3. The periodic reference signal is compared to the output of the VCO using the phase

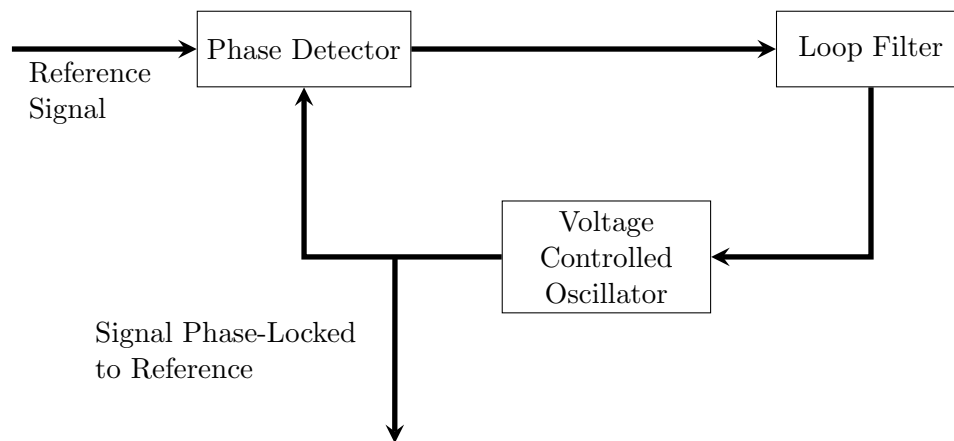


Figure 5.3: A general block diagram of a PLL.

detector. This output is then low pass filtered through the loop filter and used to drive the VCO. By doing this, the internal oscillator in the loop will lock onto the reference sinusoid so that the frequency and phase differences go to zero. Therefore the internal sinusoid is a filtered version of the reference one [1]. The PLL used in this research has additional features to remove and reduce the loop noise and smooth the tracking phase error in order to create improved carrier tracking performance, namely, an additional low-pass filter and phase smoother.

For the synchronization loops, we evaluate their performance in terms of the tracking jitter. This is the difference between the true signal and the tracked signal returned by the synchronization loop. We know that as RFI power increases, the tracking jitter grows, so the RFI prediction models must take this into account.

5.1.3 Existing RFI Tools

Many RFI analysis tools are available to analyze the impacts of RFI on digital communication systems. However, most of these do not consider the time factor when computing the bit or carrier signal-to-noise ratio (SNR) degradation. Rather they just assume that the RFI power increases the receiver noise floor. A limited number of tools do consider the time factor by incorporating SATOPS schedules with the IPC to identify data blockages due to RFI, but they still use the assumption that RFI power increases the receiver noise floor. None of the surveyed tools consider the effects of RFI on synchronization loops as explained in Section 5.1.2. For a list of existing RFI tools and their capabilities see [26].

5.2 RFI Analytical Models for Acquisition Time

The first set of models we examined concern carrier acquisition performance. This refers to the ability of a synchronization loop, the PLL, to reacquire a signal after an RFI event and is broken into two modes: carrier frequency acquisition and carrier phase acquisition. We will first consider the modes without RFI and then in the presence of CW or WB RFI in order to compute the carrier acquisition time.

5.2.1 Carrier Frequency Acquisition Mode

The most commonly used technique for the frequency acquisition mode is the frequency sweep technique. First, we select the frequency sweep rate, $\Delta\omega$ based on the estimated received carrier signal-to-noise power ratio, or carrier SNR. This is derived based on simulations and actual operational data. Next, the uplink command carrier frequency is swept using the selected sweep rate during the carrier acquisition. Then, once the command signal carrier frequency and the

receiver PLL VCO frequency coincide, the PLL will lock onto the carrier signal. This assumes that the selected sweep rate is not too large.

For the frequency sweep technique, the carrier frequency acquisition time, T_{Freq} is determined via simulation results using operational conditions from practical SATOPS missions and given by

$$T_{Freq} = \frac{\Delta\omega_{Un}}{\Delta\dot{\omega}} = \frac{2\pi f_{Un}}{\Delta\dot{\omega}} \quad (5.1)$$

where f_{Un} is the frequency uncertainty due to factors such as the Doppler effect or VCO drift. We also have that the sweep rate, given in units of Hz/sec, is

$$\Delta\dot{\omega} = \begin{cases} \omega_N^2 \left[1 - \left(\frac{1}{\text{LSNR}_{Acq} - 2} \right)^{1/2} \right], & \text{for } 3 < \text{LSNR}_{Acq} < 4.75 \\ \omega_N^2 \left[1 - \left(\frac{2}{\text{LSNR}_{Acq} - 4} \right)^{1/2} \right], & \text{for } 6 < \text{LSNR}_{Acq} < 9.5 \\ \frac{\omega_N^2}{2}, & \text{for } \text{LSNR}_{Acq} \geq 9.5 \end{cases} \quad (5.2)$$

where LSNR is the loop signal-to-noise ratio (SNR) and the natural frequency of the loop ω_N is

$$\omega_N^2 = \frac{2B_{LAcq}}{\zeta} \left[1 + \frac{1}{\zeta^2} \right], \quad (5.3)$$

where $\zeta = 0.707$ is the loop damping factor and B_{LAcq} is the acquisition loop bandwidth. During acquisition, the loop SNR is defined as

$$\text{LSNR}_{Acq} = \frac{J_0^2(m)P_T}{N_0 B_{LAcq}} \quad (5.4)$$

where the ratio P_T/N_0 is the total transmitted power-to-noise ratio, J_0 is a Bessel function of the first kind, and m is the command modulation index.

5.2.2 Carrier Phase Acquisition Mode

The most commonly used techniques for the carrier phase acquisition mode are carrier phase sweep and synchronization (or sync) word. For the carrier phase acquisition process, we first determine T_0 which is the time required for the PLL to estimate the initial phase θ_0 and T_e which is the time required for the PLL to reach the steady state phase error θ_e . Then the carrier phase acquisition time, T_{PAcq} , is the sum

$$T_{PAcq} = T_0 + T_e. \quad (5.5)$$

Using the sync word or phase sweep techniques to estimate the initial phase significantly improves the carrier phase acquisition time.

5.2.2.1 Sync Word Technique

A typical carrier phase acquisition technique is to use sync word, also called alternating sequence or acquisition sequence, to estimate the initial phase of the signal. We assume that the Doppler frequency and timing are known and that the command data is formatted as a non-return-to-zero (NRZ) rectangular pulse, that is,

$$p(t) = \begin{cases} 1 & \text{if } 0 \leq t \leq T_b, \\ 0 & \text{otherwise} \end{cases} \quad (5.6)$$

where T_b is the bit time duration. We also assume that the sync word has length L , is known, and is transmitted using direct modulation with the word sequence. The modulated carrier signal is received at the SATOPS USB demodulator and processed by a maximum likelihood phase estimator. As given in [24], the performance of the phase estimator is expressed in terms of the variance of the estimated carrier phase, θ_c ,

$$\sigma_{\hat{\theta}_c}^2 = \text{var}(\hat{\theta}_c - \theta_c) = \frac{1}{2L} \frac{N_0}{(2J_1^2(m)P_T T_b)} = \frac{1}{2L} \frac{1}{\text{BitSNR}_{Acq}} \quad (5.7)$$

where BitSNR_{Acq} is the bit signal-to-noise ratio. Note that this equation as a function of the bit SNR is only valid for acquisition mode, and can be used to determine the length L of the sync word using a given bit SNR and a specified variance of the estimated carrier phase. Then, the carrier phase acquisition time, $T_{PhaseSync}$, is given by

$$T_{PhaseSync} = \frac{1}{2B_{LAcq}} + LT_b. \quad (5.8)$$

The length of the sync word must be chosen so that the variance of the carrier phase estimator is small and the time it takes the PLL to reach the steady state phase error is about $1/2B_{LAcq}$. This ensures the probability of synchronization failure at 10^{-8} .

5.2.2.2 Phase Sweeping Technique

The carrier phase acquisition time using the phase sweeping technique is based on simulation results from various sources and found to be

$$T_{PhaseSwp} = \begin{cases} \frac{1}{2B_{LAcq}} + \frac{12.2}{4B_{LAcq}} & \text{for } \text{LSNR}_{Acq} = 10 \text{ dB}, \\ \frac{1}{2B_{LAcq}} + \frac{8}{4B_{LAcq}} & \text{for } \text{LSNR}_{Acq} = 14 \text{ dB}. \end{cases} \quad (5.9)$$

The total carrier acquisition time is then calculated as the sum

$$T_{Acq} = T_{Freq} + T_{Phase} \quad (5.10)$$

where T_{Freq} is given in (5.1) and T_{Phase} is given in (5.8) or (5.9), depending on the technique used by the SATOPS USB receiver.

5.2.3 PLL Acquisition in the Presence of CW RFI

We now consider the impacts of CW RFI on the PLL acquisition performance. The CW RFI signal is defined as

$$I(t) = \sqrt{2P_I} \cos(2\pi(f_c + \Delta f_c)t + \theta_I) \quad (5.11)$$

where P_I is the RFI power, Δf_c is the RFI frequency offset from the desired carrier frequency f_c , and θ_I is the phase of the RFI signal.

For CW RFI, the effects of the RFI depend on the RFI carrier frequency. Case 1 is out-of-band interference which occurs when the RFI carrier frequency is outside the carrier acquisition loop bandwidth, B_{LAcq} . Case 2 is in-band interference and happens when the RFI carrier frequency is within the carrier acquisition loop bandwidth.

In the presence of CW RFI, the loop SNR is given by

$$\text{LSNR}_{CW} = \frac{\text{LSNR}_{Acq}}{\Delta_{AcqCW}} \quad (5.12)$$

where Δ_{AcqCW} is a signal degradation factor determined by

$$\Delta_{AcqCW} = \begin{cases} 1 & \text{for Case 1,} \\ 1 + \text{INR} & \text{for Case 2.} \end{cases} \quad (5.13)$$

INR is the interference power-to-noise ratio given by

$$\text{INR} = \frac{P_I}{N_0}. \quad (5.14)$$

For Case 1, $\Delta f_c > B_{LAcq}$, and from (5.12), the loop SNR in the presence of CW RFI is the same as the acquisition loop SNR. Thus, the carrier acquisition time T_{Acq} remains the same as the noninterference case presented previously.

For Case 2, $\Delta f_c < B_{LAcq}$, and the carrier acquisition time will be longer due to the degradation of the loop SNR by a factor of Δ_{AcqCW} . Therefore, in the calculations of Sections 5.2.1 and 5.2.2, we substitute the value of LSNR_{CW} for LSNR_{Acq} .

We note that in the presence of CW RFI, the performance of the maximum likelihood phase

estimator given in (5.7) becomes

$$\sigma_{\hat{\theta}_c, CW}^2 = \frac{1}{2L_{CW}} \frac{\Delta_{AcqCW}}{\text{BitSNR}_{Acq}}. \quad (5.15)$$

In order to compensate for the presence of CW RFI, we can increase the length of the sync word, L_{CW} , or the bit SNR.

5.2.4 PLL Acquisition in the Presence of WB RFI

Next we consider the impacts of WB RFI on the PLL acquisition performance. The WB RFI signal is defined as

$$I(t) = \sqrt{2P_I} d_I(t) \cos(2\pi(f_c + \Delta f_c)t + \theta_I) \quad (5.16)$$

$$d(t) = \sum_{n=-\infty}^{\infty} I_n p(t - nT_I - \tau) \quad (5.17)$$

where P_I is the RFI power, T_I is the RFI bit duration, Δf_c is the frequency off set from the desired carrier frequency f_c , and θ_I is the phase of the RFI signal. The equation $d(t)$ characterizes the type of RFI where I_n is an RFI data sequence, $p(t)$ is an RFI unit amplitude rectangular pulse of duration T_I , and τ is the time asynchronism of the RFI with respect to the desired signal. For WB RFI, I_n is an independent and identically distributed sequence taking on an equiprobable value of ± 1 , and $T_I \geq T_b$, the bit duration of the desired signal.

As in the case for CW RFI, the loop SNR in the presence of WB RFI has the form

$$\text{LSNR}_{WB} = \frac{\text{LSNR}_{Acq}}{\Delta_{AcqWB}}. \quad (5.18)$$

For WB RFI, the signal degradation factor is given by

$$\Delta_{AcqWB} = [1 + \text{INR} \cdot S_{RFI-WB}(B_{DL}, f_{Un})] \quad (5.19)$$

where

$$S_{RFI-WB}(f, f_{Un}) = \frac{1}{2} [S_I(f - f_{Un}) + S_I(f + f_{Un})] \quad (5.20)$$

$$S_I(f) = T_I \int_{-B_{Acq-DL}}^{B_{Acq-DL}} \left[\frac{\sin(\pi f T_I)}{\pi f T_I} \right]^2 df. \quad (5.21)$$

To obtain the equations for the carrier acquisition time in the presence of WB RFI, we substitute LSNR_{WB} in for LSNR_{Acq} in the equations of Sections 5.2.1 and 5.2.2. Again, we note that in the presence of WB RFI, the performance of the maximum likelihood phase estimator given

in (5.7) becomes

$$\sigma_{\hat{\theta}_c, WB}^2 = \frac{1}{2L_{WB}} \frac{\Delta_{AcqWB}}{\text{BitSNR}_{Acq}}. \quad (5.22)$$

To compensate for the WB RFI, we can increase the length of the sync word or the bit SNR, just as in the case with CW RFI.

5.2.5 Simulation Results for Verification and Validation

The analytic models for the total carrier acquisition time described in this section were implemented in MATLAB for purposes of verification and validation. Multiple operating scenarios were tested to ensure that the models were accurate for the needs of a typical USB SATOPS Command system. These MATLAB programs were implemented on an Apple MacBook Pro laptop.

For the assessment of the impact of CW and WB RFI on PLL acquisition performance, we use the parameters shown in Table 5.1. Note that, in the simulations, the interference power-

Table 5.1: Parameters for the simulations of PLL acquisition performance: Carrier acquisition time.

Symbol	Description	Value
f_c	Signal carrier frequency	2075 MHz
m	Command modulation index	1.1 rad
σ	Threshold carrier jitter	$\pi/2$ rad
R_b	Command bit rate	2 Kbps
R_I	Interference data bandwidth	1 Hz

to-noise ratio, INR, is calculated as the sum of the interference-to-signal power ratio, ISR, and the received signal-to-noise ratio, SNR, that is

$$\text{INR} = \text{ISR} + \text{SNR}. \quad (5.23)$$

Figure 5.4 shows the PLL acquisition performance in the presence of CW RFI at $\text{ISR} = -40$ dB. The plot in Figure 5.4a shows the loop SNR as a function of the received SNR. Figure 5.4b shows the carrier frequency acquisition time as a function of the loop SNR both with and without CW RFI. We can see that there is only a small difference between the time acquisition

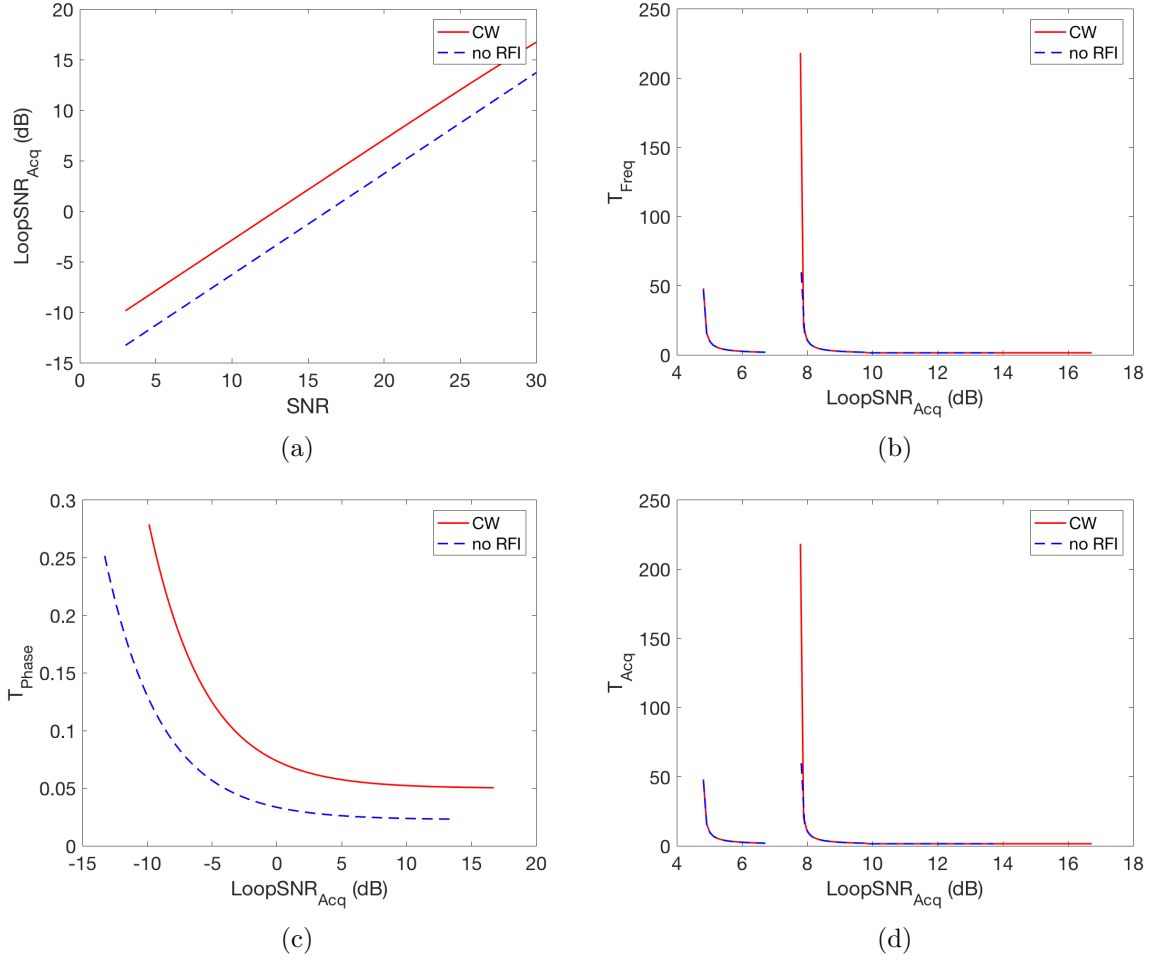


Figure 5.4: Carrier acquisition times vs. loop SNR without RFI and with CW RFI at $\text{ISR} = -40$ dB.

for the two signals with this level of interference signal power. The plot in Figure 5.4c shows the carrier phase acquisition time as a function of loop SNR. The acquisition time in the presence of CW RFI is larger than that with no RFI as we expect. Lastly, Figure 5.4d shows the total carrier acquisition time which is the sum of the results of Figures 5.4b and 5.4c.

We then consider the case for CW RFI at $\text{ISR} = -25$ dB. Figure 5.5 shows the corresponding series of plots. We can see that even with this larger RFI source, the results are similar to those from the first case. The carrier acquisition time is slightly larger for this case as we expect, but it is not a large change.

We next consider each of these cases for signals in the presence of WB RFI. First, we consider the case with $\text{ISR} = -40$ dB. The resulting acquisition times are shown in Figure 5.6. We notice that while the carrier frequency acquisition times with and without WB RFI are

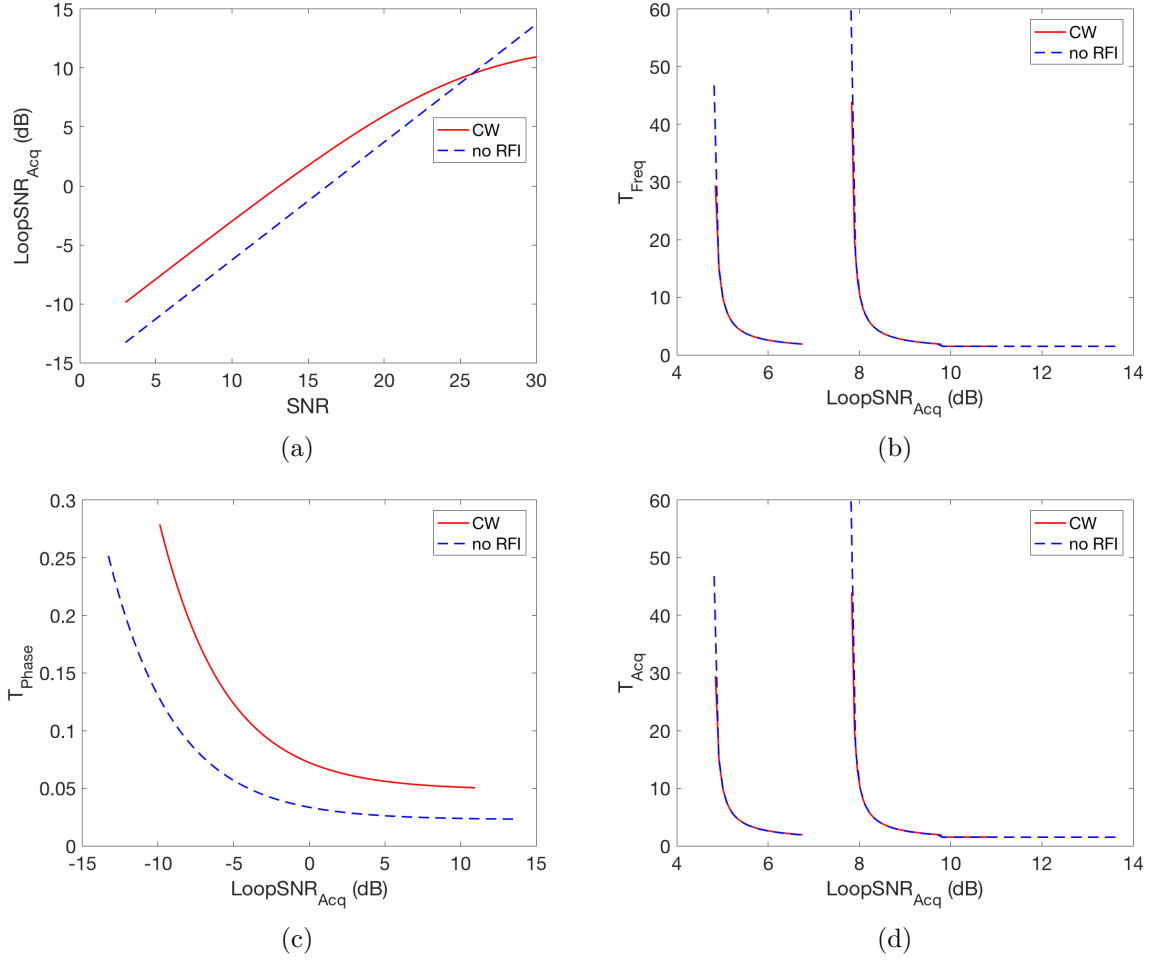


Figure 5.5: Carrier acquisition times vs. loop SNR without RFI and with CW RFI at $\text{ISR} = -25$ dB.

close, the carrier phase acquisition time shown in Figure 5.6c is over 0.5 seconds longer in the presence of WB RFI than not. This is also larger than what we saw in the case of CW RFI which is what we would expect.

Lastly, we present the carrier acquisition times for a signal in the presence of WB RFI at $\text{ISR} = -25$ dB. The results are shown in Figure 5.7. We see that the general behavior of the acquisition time is the same as before, but the values are slightly larger, which is reasonable for the higher value of the ISR.

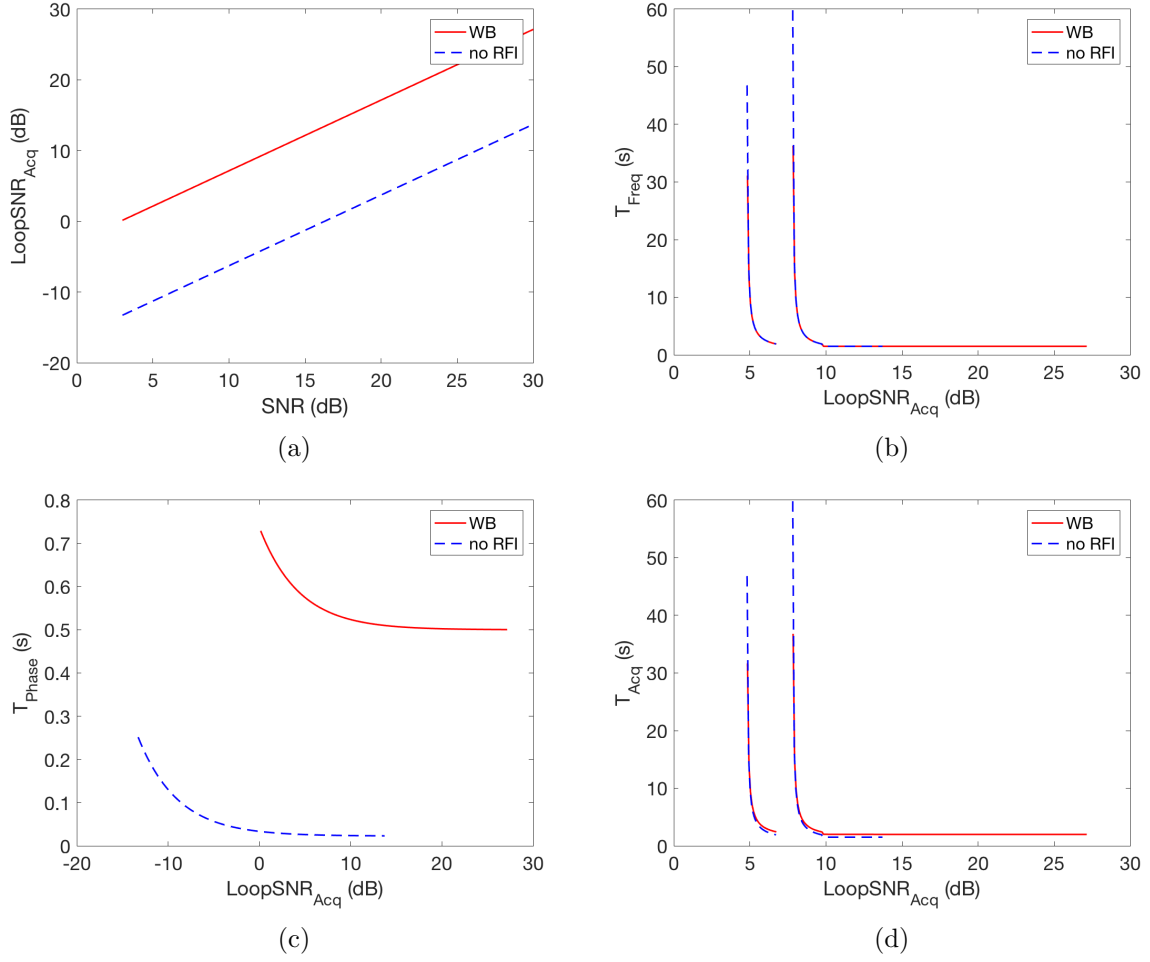


Figure 5.6: Carrier acquisition times vs. loop SNR without RFI and with WB RFI at $\text{ISR} = -40 \text{ dB}$.

5.3 Analytical Models for RFI Detection

We now consider models for RFI detection which will also calculate the tracking jitter errors and the bit error rate (BER) due to the presence of RFI.

5.3.1 The PLL Lock Detector

Figure 5.8 shows the way that a PLL lock detector is integrated with a PLL in order to acquire and track the carrier signal. The PLL starts in the acquisition mode in order to acquire the carrier signal component. The phase lock loop monitors the carrier phase error over an observation time, T_L . If the phase error is less than a threshold value $\sigma_{T_{hcar}}$, then the PLL will switch to tracking mode. The PLL tracking loop bandwidth is then set according to the product

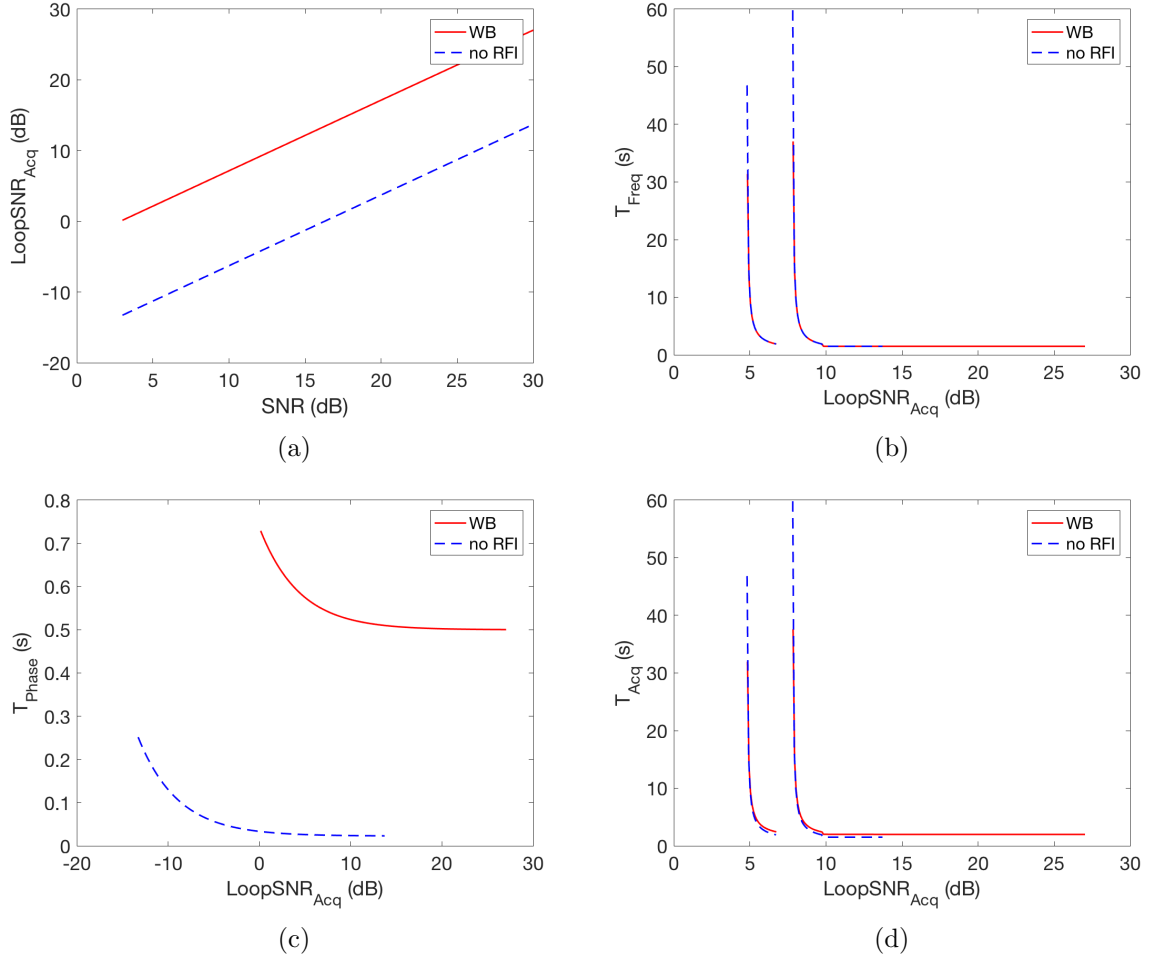


Figure 5.7: Carrier acquisition times vs. loop SNR without RFI and with WB RFI at $\text{ISR} = -25$ dB.

$B_L T_b \leq 0.01$ where B_L is the loop bandwidth and T_b is the command data bit duration.

The value of the threshold σ_{Thcar} is selected in order to have a probability of lock greater than 90%. It is chosen by the relationship

$$\sigma_{Thcar} = \left[\frac{1}{\sqrt{\text{LSNR}_{Eff}}} + \sigma_{margin} \right] \quad (5.24)$$

where LSNR_{Eff} is the effective loop SNR and σ_{margin} is the phase margin error. The effective loop SNR is the loop SNR in the presence of a noisy carrier reference for a specified bit error rate (BER) performance. The phase margin error is specified by the communications designer in order to take into account phase jitters caused by factors such as VCO drift and residual Doppler, and in practice it is 5 to 10% of the value of $1/\sqrt{\text{LSNR}_{Eff}}$.

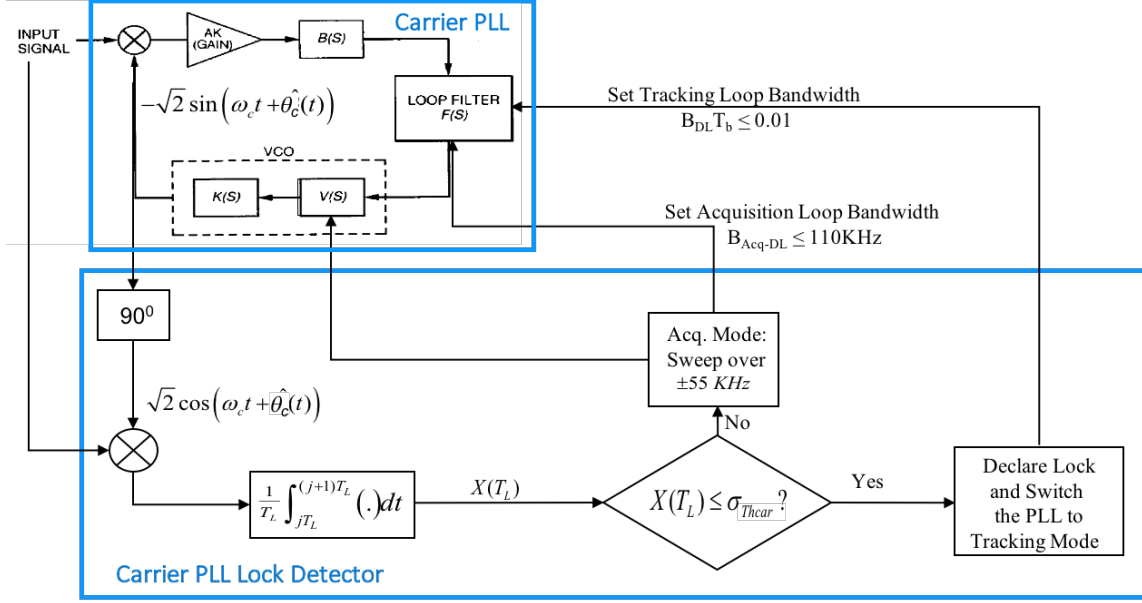


Figure 5.8: A typical PLL lock detector used by SATOPS satellites.

We use the effective loop SNR that is evaluated in the absence of RFI in order to select the threshold value σ_{Thear} . It is assumed that the PLL tracking error follows the Tikhonov, or circular normal, distribution and that the carrier tracking phase error is small. Following [42], we can approximate the uncoded BER as

$$\text{BER}_{RL} = \frac{1}{2} \text{erfc}(\sqrt{\text{BSNR}_0}) + \frac{1}{2} \sqrt{\frac{1}{\pi} \text{BSNR}_0} \sigma_\theta^2 e^{-\text{BSNR}_0} \quad (5.25)$$

where the bit SNR is given by

$$\text{BSNR}_0 = 2J_1^2(m) \frac{P_T T_b}{N_0}. \quad (5.26)$$

As before, we define the carrier tracking loop SNR as

$$\text{LSNR}_0 = J_0^2(m) \frac{P_T}{N_0 B_L} \quad (5.27)$$

and the carrier tracking jitter

$$\sigma_\theta^2 = \frac{1}{\text{LSNR}_0}. \quad (5.28)$$

Then we can solve (5.27) for the ratio P_T/N_0 and substitute into (5.25) and (5.26) to rewrite

the uncoded BER as

$$\text{BER}_{RL} = \frac{1}{2} \text{erfc}(\sqrt{\alpha B_L T_b \text{LSNR}_0}) + \frac{1}{2} \frac{\sqrt{\frac{1}{\pi} \alpha B_L T_b \text{LSNR}_0}}{\text{LSNR}_0} e^{-\alpha B_L T_b \text{LSNR}_0} \quad (5.29)$$

where

$$\alpha = 2 \frac{J_1^2(m)}{J_0^2(m)}. \quad (5.30)$$

The first term of (5.29) represents the ideal BER performance with perfect carrier tracking, and the second term gives the BER degradation due to imperfect carrier tracking due to the presence of gaussian white noise. This imperfect carrier tracking is the tracking jitter, σ_θ^2 .

We can now see that the effective loop SNR is defined from the first term of (5.29) as

$$\text{LSNR}_{Eff} = \alpha B_L T_b \text{LSNR}_0. \quad (5.31)$$

Therefore, we can rewrite (5.29) as

$$\text{BER}_{RL} = \frac{1}{2} \text{erfc}(\sqrt{\text{LSNR}_{Eff}}) + \frac{\alpha B_L T_b}{2} \frac{\sqrt{\frac{1}{\pi} \text{LSNR}_{Eff}}}{\text{LSNR}_{Eff}} e^{-\text{LSNR}_{Eff}}. \quad (5.32)$$

The effective loop SNR needs to be chosen so that the uncoded BER meets a specified threshold BER value, typically around 10^{-9} .

5.3.2 PLL Tracking Jitter in the Absence of RFI

The performance of the PLL is characterized by the variance of the tracking phase error, $\sigma_{\theta_c}^2$, also known as the tracking jitter. In the absence of RFI, this is given by

$$\sigma_{\theta_c}^2 = \frac{\Delta}{\text{LSNR}} \quad (5.33)$$

where LSNR is the loop SNR given by

$$\text{LSNR} = \frac{P_C}{N_0 B_L}, \quad P_C = J_0^2(m) P_T. \quad (5.34)$$

Note that P_C/N_0 is the carrier SNR. Then we define the factor Δ as

$$\Delta = \left[1 + \frac{P_C}{N_0} S_{CD}(1, 0) \right] \quad (5.35)$$

$$S_{CD}(k, f) = \sum_{k=1, k=\text{odd}}^{\infty} J_k^2(m) [S_d(f - kf_{SC}) + S_d(f + kf_{SC})] \quad (5.36)$$

$$S_d(f) = T_b \int_{-B_L}^{B_L} \left[\frac{\sin(\pi f T_b)}{\pi f T_b} \right]^2 df \quad (5.37)$$

$$T_b = \frac{1}{R_b}. \quad (5.38)$$

When $f_{SC} \geq 4R_b$, $\sigma_{\theta_c}^2$ can be approximated as

$$\sigma_{\theta_c}^2 \approx \frac{1}{\text{LSNR}}, \quad (5.39)$$

and we will assume this case for the remainder of the chapter. In practice, the loop SNR is set at around 11 dB or more to ensure reliable tracking performance. We now consider how the tracking jitter is changed by the presence of RFI.

5.3.3 PLL Tracking Jitter in the Presence of CW RFI

Recall from Section 5.2.3 that the CW RFI signal is given as

$$I(t) = \sqrt{2P_I} \cos(2\pi(f_c + \Delta f_c)t + \theta_I) \quad (5.40)$$

where P_I is the RFI power, θ_I is the RFI phase, and Δf_c is the RFI frequency offset from the desired carrier frequency, f_c . The variance of the tracking phase error in the presence of CW RFI is given by

$$\sigma_{\theta_c, CW}^2 = \frac{\Delta_{CW}}{\text{LSNR}}. \quad (5.41)$$

As we saw before, the value of Δ is dependent on whether or not the RFI carrier frequency is within the carrier loop bandwidth, B_L , and is given by

$$\Delta_{CW} = \begin{cases} 1 & \text{if } \Delta f_c > B_L \\ 1 + \text{INR} & \text{if } \Delta f_c < B_L \end{cases} \quad (5.42)$$

where INR is the interference-to-noise ratio given by P_I/N_0 . We also note that when the INR is larger than the carrier SNR, the PLL will drop lock on the command signal and lock onto the CW RFI signal instead.

In the presence of CW RFI, the effective loop SNR is calculated as

$$\text{LSNR}_{Eff,CW} = \frac{\text{LSNR}}{\Delta_{CW}}. \quad (5.43)$$

Therefore, to calculate the SATOPS BER performance in the presence of CW RFI and gaussian white noise, we can use this effective loop SNR in (5.32).

5.3.4 PLL Tracking Jitter in the Presence of WB RFI

Recall from Section 5.2.4 that the WB RFI signal is given by

$$I(t) = \sqrt{2P_I} d_I \cos(2\pi(f_c + \Delta f_c)t + \theta_I) \quad (5.44)$$

$$d(t) = \sum_{n=-\infty}^{\infty} I_n p(t - nT_I - \tau). \quad (5.45)$$

The variance of the tracking phase error in the presence of WB RFI is given by

$$\sigma_{\theta_c, WB} = \frac{\Delta_{WB}}{\text{LSNR}}. \quad (5.46)$$

Then the degradation factor Δ_{WB} is given by

$$\Delta_{WB} = 1 + \text{INR} \cdot S_{WB}(B_L, T_I, \Delta f_c) \quad (5.47)$$

where INR is the ratio P_I/N_0 and

$$S_{WB}(f, T_I, \Delta f_c) = S_{RFI}(f - \Delta f_c) + S_{RFI}(f + \Delta f_c) \quad (5.48)$$

$$S_{RFI}(f) = T_I \int_{-B_L}^{B_L} \left[\frac{\sin(\pi f T_I)}{\pi f T_I} \right]^2 df. \quad (5.49)$$

We can again compute the effective loop SNR in the presence of WB RFI as

$$\text{LSNR}_{Eff,WB} = \frac{\text{LSNR}}{\Delta_{WB}}. \quad (5.50)$$

Then, the SATOPS BER performance in the presence of WB RFI and gaussian white noise can be computed by using this effective loop SNR in (5.32).

5.3.5 Simulation Results for Verification and Validation

We implement the above models in MATLAB for purposes of verification and validation. The parameters used to obtain these results are shown in Table 5.2.

Table 5.2: Parameters for the simulations of PLL acquisition performance: Tracking error and BER.

Symbol	Description	Value
m	Command modulation index	1.1 rad
R_b	Command bit rate	2 Kbps
ISR	Interference power-to-signal ratio	-10 dB
R_I	Interference data bandwidth	10 bps
B_L	Loop bandwidth	10 Hz
σ_{margin}	Phase error margin	2 deg

Figure 5.9 shows the PLL tracking jitter in the absence of RFI as well as in the presence of either CW or WB RFI as a function of the loop SNR with $\Delta f_{RFI} = 5$ Hz. The tracking error in the presence of RFI is larger for all values of the loop SNR as we would expect. We also see that the tracking performance of the PLL is somewhat worse in the presence of CW RFI as

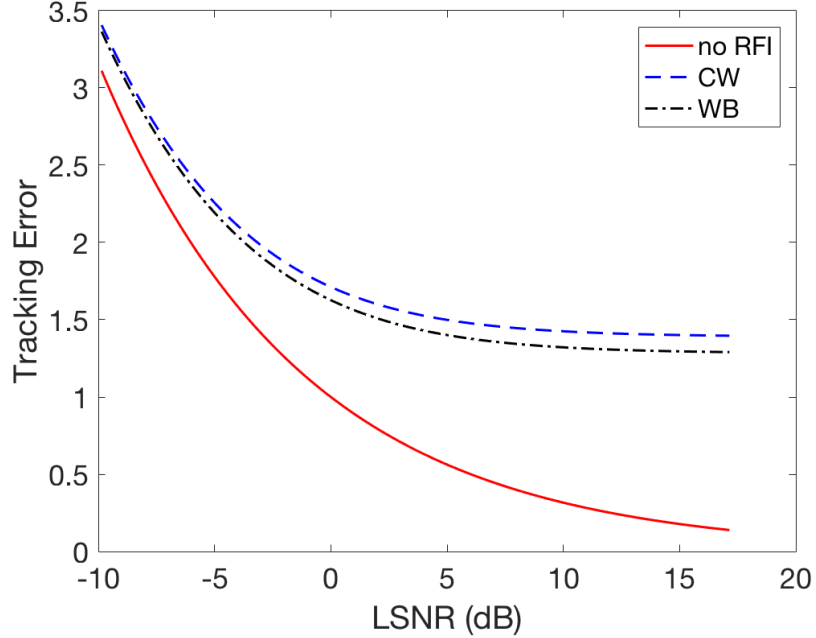


Figure 5.9: PLL tracking jitters in the absence of RFI and the presence of CW and WB RFI signals with $\Delta f_{RFI} = 5$ Hz.

compared to WB RFI.

We also show the PLL tracking jitter in each RFI case with $\Delta f_{RFI} = 10$ Hz in Figure 5.10. The tracking jitter is still worse for the case of RFI than in the case with no RFI, and the presence of CW RFI has a greater negative impact than that of WB RFI for the same operating conditions. However, the performance in the presence of WB RFI is better than the previous case with $\Delta f_{RFI} = 5$ Hz. This is also expected since the RFI signal frequency is further away from the center carrier frequency.

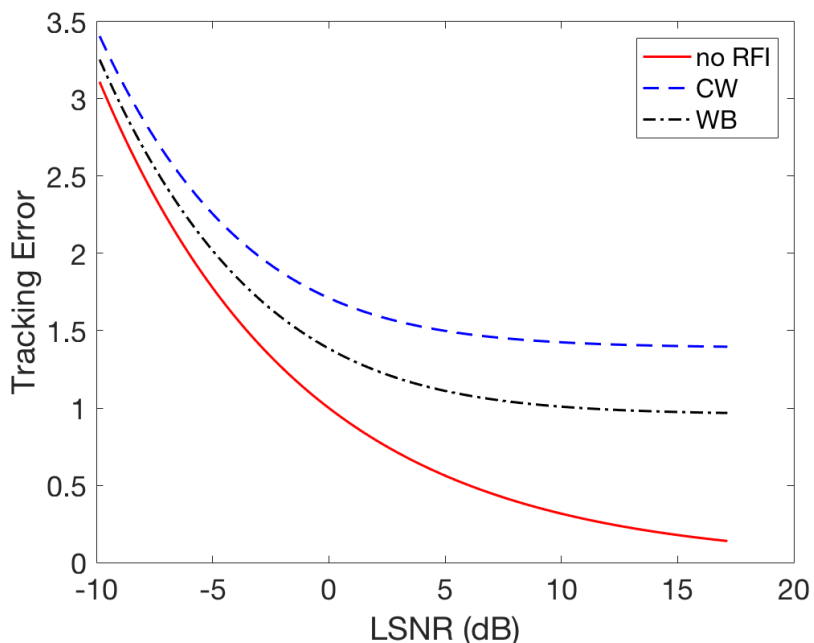


Figure 5.10: PLL tracking jitters in the absence of RFI and the presence of CW and WB RFI signals with $\Delta f_{RFI} = 10$ Hz.

We next consider the BER performance due to gaussian white noise and the RFI signal as functions of the loop SNR as described in (5.32). Figure 5.11 shows the BER performance with and without either CW or WB RFI with $\Delta f_{RFI} = 5$ Hz. From Figure 5.11a we can see that the BER decreases with increased loop SNR and that the performance in the presence of RFI is much worse, particularly for higher values of loop SNR. Figure 5.11b shows just the BER performance in the presence of each RFI type for detail. Again, the BER performance under CW RFI is worse than the performance with WB RFI under the same operating conditions.

Figure 5.12 shows the BER performance without and in the presence of each type of RFI with $\Delta f_{RFI} = 10$ Hz. As we saw with the corresponding plots of the tracking jitters, the BER

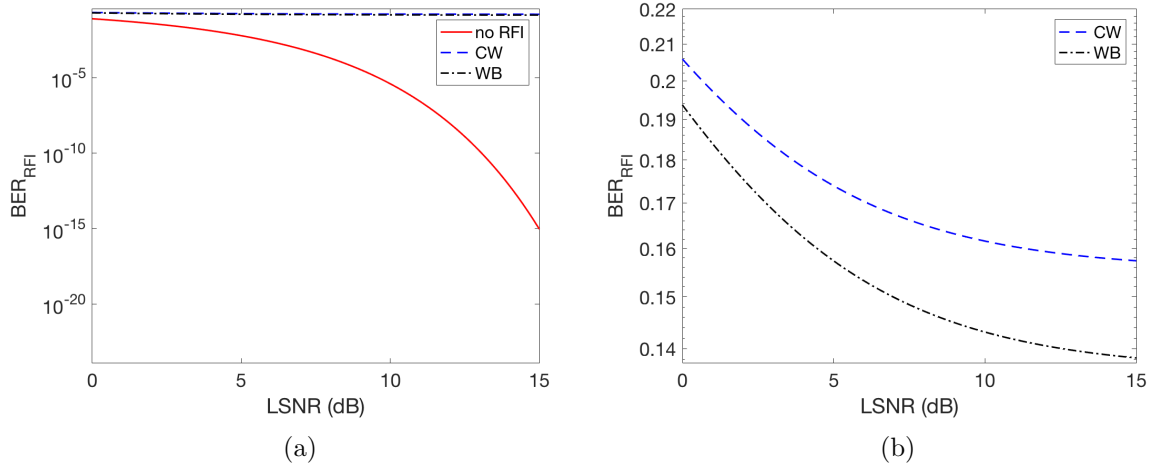


Figure 5.11: BER in the absence of RFI and the presence of CW and WB RFI signals with $\Delta f_{RFI} = 5$ Hz. (b) shows detail on the BER in the presence of CW and WB RFI.

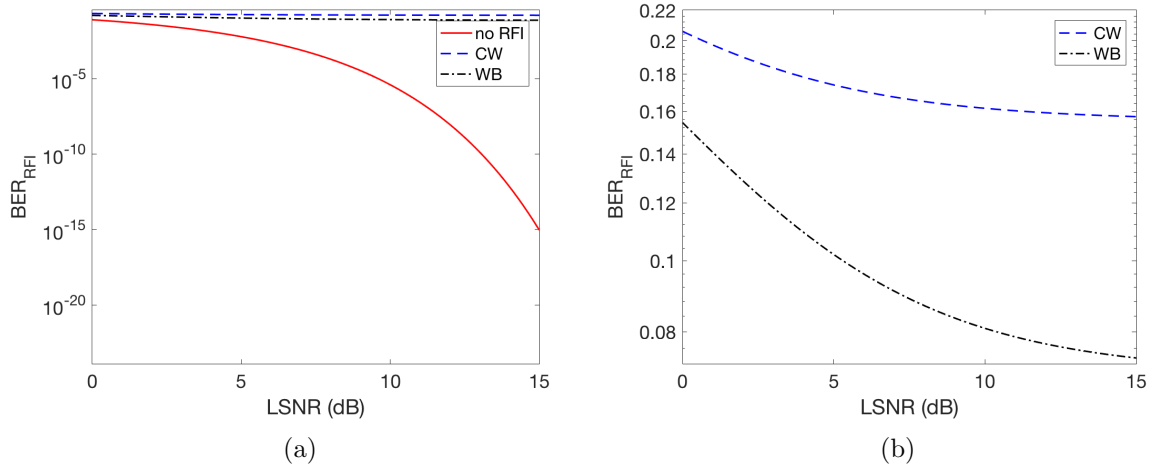


Figure 5.12: BER in the absence of RFI and the presence of CW and WB RFI signals with $\Delta f_{RFI} = 10$ Hz. (b) shows detail on the BER in the presence of CW and WB RFI.

performance of the PLL in the presence of RFI is worse than without, and the performance with CW RFI is worse than that with WB RFI under the same conditions. Comparing Figure 5.12b with Figure 5.11b, we see that the BER performance in the presence of WB RFI is better with $\Delta f_{RFI} = 10$ Hz than with $\Delta f_{RFI} = 5$ Hz, as we expect since the center frequency of the signal is farther from the RFI signal.

Chapter 6

Conclusion

6.1 The Double Inverted Pendulum System

The stabilization of a double inverted pendulum is a classic nonlinear control problem. Several control design approaches have been applied to the stabilization of the DIP, but almost all of them have only presented numerical simulations of the system and control performance and have not provided real-time experimental implementation on a physical system.

In Chapter 2, we modeled the inverted double pendulum and verified the model against our physical apparatus. We also performed parameter estimation to improve the performance of our model relative to the experimental results.

In Chapter 3, we considered two methods for the stabilization of the DIP system in its unstable, vertical position: LQR control and a power series based control. Each control was applied to our physical system for real-time experiments, taking into account viscous damping and the physical limitations of our system due to a finite track length and the voltage saturation limits on the power amplifier, and were successful in stabilizing the DIP. In general, the power series control used a smaller control effort measured via the average or cumulative voltage applied. However, the LQR control was less likely to have the voltage reach the saturation level at 10 V, though the voltage did go into saturation when using both control methods during tests. The power series based controller also had generally better performance as shown by lower average and maximum deviations from the unstable equilibrium state.

In Chapter 4, we examined possible observers to use with the physical DIP system since the velocity states are not directly measured. This has not been considered in the literature, primarily due to the fact that typically only numerical simulations are performed and full state feedback is assumed. We discussed the current method of attaining the velocity states for use in the feedback control by using a low pass derivative filter. Then we considered a Kalman observer, though we were unable to determine parameter values for the Kalman observer which

would result in the stabilization of the pendulum. We next examined a nonlinear observer as described in [8]. This observer has been proposed as an easy to implement observer for nonlinear systems. Constructing this observer required computing a series of derivatives of our system which were too complex to include analytically. Instead we implemented the complex step method to approximate the necessary derivatives. However, this still took too long to compute to be practical for real-time implementation. Therefore all presented results, as shown in Chapter 3, use the low pass derivative filter to estimate the velocity states.

In the future, we would be interested in designing and applying other controls to the DIP system for comparison with those presented here such as the state dependent Riccati equation. We would also like to continue modifying the controls already presented in order to obtain better stabilization results. Moreover, we would like to incorporate other effects in the model such as the static friction between the cart wheels and the track. Additional work will also be done on the observer portion of this problem in order to design an observer which can accurately estimate the velocity states for use in the state feedback control.

6.2 Radio Frequency Interference Analytic Models

Chapter 5 considered a series of analytic models to detect and predict RFI within SATOPS systems. These models include the behavior of the carrier synchronization loops when calculating bit or carrier signal-to-noise ratio degradation due to interference. This allows them to include the time factor involved with the synchronization loops which is rarely included in the currently available RFI analysis tools. Models for the carrier acquisition time, the tracking errors, and the bit error rate performance of a PLL in the absence of RFI as well as in the presence of CW or WB RFI were presented. These models were then implemented in MATLAB and tested using actual operating conditions in order to verify and validate the models. The models were also implemented into an RFI analysis tool produced by our collaborators at Intelligent Fusion Technology, Inc. These models focused on carrier tracking performance. In the future, we would like to implement similar models for subcarrier systems.

REFERENCES

- [1] Daniel Abramovitch. Phase-locked loops: A control centric tutorial. In *American Control Conference, 2002. Proceedings of the 2002*, volume 1, pages 1–15. IEEE, 2002.
- [2] W. Kyle Anderson and Eric J. Nielsen. Sensitivity analysis for navier– stokes equations on unstructured meshes using complex variables. *AIAA Journal*, 39, 2001.
- [3] W. Kyle Anderson, Eric J. Nielsen, and D.L. Whitfield. Multidisciplinary sensitivity derivatives using complex variables. *MSSU-COE-ERC98-08*, July 1998.
- [4] Panos J. Antsaklis and Anthony N. Michel. *Linear Systems*. McGraw-Hill New York, 1997.
- [5] H.T. Banks, Kidist Bekele-Maxwell, Lorena Bociu, Marcella Noorman, and Kristen Tillman. The complex-step method for sensitivity analysis of non-smooth problems arising in biology. *Eurasian Journal of Mathematical and Computer Applications*, 3:16–68, 2015.
- [6] Amanda Bernstein and Hien T. Tran. Real-time implementation of a lqr-based controller for the stabilization of a double inverted pendulum. In *Proceedings of The International MultiConference of Engineers and Computer Scientists 2017*, pages 245–250, 2017.
- [7] A. Bogdanov. Optimal control of double inverted pendulum on a cart. *Oregon Health and Science University, Tech. Rep. CSE-04-006, OGI School of Science and Engineering, Beaverton, OR*, 2004.
- [8] G. Ciccarella, M. Dalla Mora, and A. Germani. A luenberger-like observer for nonlinear systems. *International Journal of Control*, 57(3):537–556, 1993.
- [9] Mustafa Demirci. Design of feedback controllers for linear system with applications to control of a double-inverted pendulum. *International Journal of Computational Cognition*, 2(1):65–84, 2004.
- [10] W. L. Garrard. Suboptimal feedback control for nonlinear systems. *Automatica*, 8:219–221, 1972.
- [11] W. L. Garrard, D. F. Enns, and S. A. Snells. Nonlinear feedback control of highly maneuverable aircraft. *International Journal of Control*, 56:799–812, 1992.
- [12] W. L. Garrard and J. M. Jordan. Design of nonlinear automatic flight control systems. *Automatica*, 13:497–505, 1977.
- [13] Knut Graichen, Michael Treuer, and Michael Zeitz. Swing-up of the double pendulum on a cart by feedforward and feedback control with experimental validation. *Automatica*, 43(1):63–71, 2007.
- [14] Mohinder S. Grewal and Angus P. Andrews. *Kalman Filtering: Theory and Practice Using MATLAB*. John Wiley & Sons Inc., 2nd edition, 2001.

- [15] Tomohiro Henmi, Mingcong Deng, and Akira Inoue. Swing-up control of a serial double inverted pendulum. In *American Control Conference, 2004. Proceedings of the 2004*, volume 5, pages 3992–3997. IEEE, 2004.
- [16] Chin-I Huang and Li-Chen Fu. Passivity based control of the double inverted pendulum driven by a linear induction motor. In *Control Applications, 2003. CCA 2003. Proceedings of 2003 IEEE Conference on*, volume 2, pages 797–802. IEEE, 2003.
- [17] Alberto Isidori. *Nonlinear Control Systems: An Introduction*. Springer-Verlag, 2nd edition, 1989.
- [18] Pathompong Jaiwat and Toshiyuki Ohtsuka. Real-time swing-up of double inverted pendulum by nonlinear model predictive control. In *5th International Symposium on Advanced Control of Industrial Processes*, pages 290–295, 2014.
- [19] Kok-Lam Lai and John L. Crassidis. Generalizations of the complex-step derivative approximation. *AIAA Paper 2006-6348*, 2006.
- [20] Frank L. Lewis, Draguna L. Vrabie, and Vassilis L. Syrmos. *Optimal Control*. John Wiley & Sons Inc., 3rd edition, 2012.
- [21] J. N. Lyness. Numerical algorithms based on the theory of complex variables. In *Proc. ACM 22nd Nat. Conf.*, volume 4, pages 124–134, 1967.
- [22] J. N. Lyness and C. B. Moler. Numerical differentiation of analytic functions. *SIAM J. Numer. Anal.*, 4:202–210, 1967.
- [23] Joaquim R. R. A. Martins, Ilan M. Kroo, and Juan J. Alonso. An automated method for sensitivity analysis using complex variables. *AIAA Paper 2000-0689*, 2000.
- [24] U. Mengali and A. D’Andrea. *Synchronization Techniques for Digital Receivers*. Plenum Press, 1997.
- [25] Tien M. Nguyen, Hien T. Tran, Ashok Mathur, Jack Kreng, Donald Konold, and Gilbert Takahashi. Assessment of the reduction in transmitted power for future SGLS and USB services. *Aerospace Report No. TOR-2001 (1570)-3*, 2001.
- [26] Tien M. Nguyen, Hien T. Tran, Zhonghai Wang, Amanda Coons, Charles C. Nguyen, Steven A. Lane, Khanh D. Pham, Genshe Chen, and Gang Wang. RFI modeling and prediction approach for SATOPS applications. In *Proc. SPIE 9469, Sensors and Systems for Space Applications VIII*, page 94690D. International Society for Optics and Photonics, 2015.
- [27] Tien M. Nguyen, Hien T. Tran, Zhonghai Wang, Amanda Coons, Charles C. Nguyen, Steven A. Lane, Khanh D. Pham, Genshe Chen, and Gang Wang. RFI modeling and prediction approach for SATOP applications: RFI prediction models. In *Proc. SPIE 9838, Sensors and Systems for Space Applications IX*, page 98380I. International Society for Optics and Photonics, 2016.

- [28] Tien M. Nguyen, Charles C. Wang, and James Yoh. Bandwidth-efficient waveform study, part i: Comparison of SGLS waveform to USB and other waveforms. *Aerospace TOR-2000 (1795)-1*, 1999.
- [29] Tien M. Nguyen, Charles C. Wang, James Yoh, and Bil Deng. Assessment of S-band utilization for future satellite operations. *Aerospace Report No. ATR-99 (1795)-1*, 1999.
- [30] Sophocles J. Orfanidis. *Introduction to Signal Processing*. Pearson Education, Inc., 1996.
- [31] Quanser Consulting, Inc. *Linear Motion Servo Plants: IP01 or IP02 - IP01 and IP02 User Manual*, 5.0 edition.
- [32] Quanser Consulting, Inc. *Linear Motion Servo Plants: IP01 or IP02 - Linear Experiment #5: LQR Control - Student Handout*, 4.1 edition.
- [33] J. Rubi, A. Rubio, and A. Avello. Swing-up control problem for a self-erecting double inverted pendulum. *IEEE Proceedings-Control Theory and Applications*, 149(2):169–175, 2002.
- [34] Fuminori Saito, Toshio Fukuda, and Fumihito Arai. Swing and locomotion control for two-link brachiation robot. In *Robotics and Automation, 1993. Proceedings., 1993 IEEE International Conference on*, pages 719–724, 1993.
- [35] Shun Sasagawa, Masahiro Shinya, and Kimitaka Nakazawa. Interjoint dynamic interaction during constrained human quiet standing examined by induced acceleration analysis. *Journal of Neurophysiology*, 111(2):313–322, 2014.
- [36] Mark W. Spong. The swing up control problem for the acrobot. *IEEE control systems*, 15(1):49–55, 1995.
- [37] W. Squire and G. Trapp. Using complex variables to estimate derivatives of real functions. *SIAM Review*, 10(1):100–112, March 1998.
- [38] S. Takashima. Control of gymnast on a high bar. In *Intelligent Robots and Systems' 91. Intelligence for Mechanical Systems, Proceedings IROS'91. IEEE/RSJ International Workshop on*, pages 1424–1429. IEEE, 1991.
- [39] D. A. Winter. Human balance and posture control during standing and walking. *Gait and Posture*, 3(4):193–214, 1995.
- [40] Sandeep Kumar Yadav, Sachin Sharma, and Narinder Singh. Optimal control of double inverted pendulum using lqr controller. *International Journal of Advanced Research in Computer Science and Software Engineering*, 2(2), 2012.
- [41] Jianqiang Yi, Naoyoshi Yubazaki, and Kaoru Hirota. A new fuzzy controller for stabilization of parallel-type double inverted pendulum system. *Fuzzy Sets and Systems*, 126(1):105–119, 2002.
- [42] Joseph H. Yuen, editor. *Deep Space Telecommunications Systems Engineering*. NASA/JPL, California Institute of Technology, JPL Publication 82-76, 1983.

- [43] Wei Zhong and Helmut Rock. Energy and passivity based control of the double inverted pendulum on a cart. In *Control Applications, 2001.(CCA'01). Proceedings of the 2001 IEEE International Conference on*, pages 896–901. IEEE, 2001.

APPENDIX

Appendix A

DIP Model Parameters

A.1 Nomenclature

Table A.1: Nomenclature for the DIP system model.

Symbol	Description
α	Angle of Lower Pendulum Rod Relative to Upright Position
α'	Lower Pendulum Angular Velocity
α''	Lower Pendulum Angular Acceleration
B_c	Viscous Damping Coefficient as seen at the Motor Pinion
B_1	Viscous Damping Coefficient as seen at the Lower Pendulum Axis
B_2	Viscous Damping Coefficient as seen at the Upper Pendulum Axis
F_c	Cart Driving Force Generated by the DC Motor
g	Gravitational Constant
I_1	Moment of Inertia of the Lower Pendulum Rod at its Center of Gravity
I_2	Moment of Inertia of the Upper Pendulum Rod at its Center of Gravity
J_m	Rotational Moment of Inertia of the DC Motor's Output Shaft

Table A.1 – (continued)

Symbol	Description
K_g	Planetary Gearbox Gear Ratio
ℓ_1	Length of Lower Pendulum from Pivot to Center of Gravity
ℓ_2	Length of Upper Pendulum from Hinge to Center of Gravity
L_1	Total Length of Lower Pendulum
\mathcal{L}	Lagrangian
M_c	Cart Mass
M_h	Hinge Mass
M_1	Lower Pendulum Mass
M_2	Upper Pendulum Mass
r_{mp}	Motor Pinion Radius
T_{ct}	Translational Kinetic Energy of the Cart
T_{cr}	Rotational Kinetic Energy Due to the Cart's DC Motor
T_c	Total Kinetic Energy of the Cart
T_{1t}	Translational Kinetic Energy of the Lower Pendulum
T_{1r}	Rotational Kinetic Energy of the Lower Pendulum
T_1	Total Kinetic Energy of the Lower Pendulum
T_{2t}	Translational Kinetic Energy of the Upper Pendulum
T_{2r}	Rotational Kinetic Energy of the Upper Pendulum
T_2	Total Kinetic Energy of the Upper Pendulum
T_h	Total Kinetic Energy of the Hinge
T_T	Total Kinetic Energy of the System

Table A.1 – (continued)

Symbol	Description
V_1	Potential Energy of the Lower Pendulum
V_2	Potential Energy of the Upper Pendulum
V_h	Potential Energy of the Hinge
V_T	Total Potential Energy of the System
x_c	Cart Linear Position
x'_c	Cart Velocity
x''_c	Cart Acceleration
x_h	Absolute x -coordinate of the Hinge's Center of Gravity
x_1	Absolute x -coordinate of the Lower Pendulum Rod's Center of Gravity
x_2	Absolute x -coordinate of the Upper Pendulum Rod's Center of Gravity
x'_h	x -component of the Velocity of the Hinge's Center of Gravity
x'_1	x -component of the Velocity of the Lower Pendulum Rod's Center of Gravity
x'_2	x -component of the Velocity of the Upper Pendulum Rod's Center of Gravity
y_h	Absolute y -coordinate of the Hinge's Center of Gravity
y_1	Absolute y -coordinate of the Lower Pendulum Rod's Center of Gravity
y_2	Absolute y -coordinate of the Upper Pendulum Rod's Center of Gravity
y'_h	y -component of the Velocity of the Hinge's Center of Gravity
y'_1	y -component of the Velocity of the Lower Pendulum Rod's Center of Gravity
y'_2	y -component of the Velocity of the Upper Pendulum Rod's Center of Gravity
θ	Angle of Upper Pendulum Rod Relative to Lower Pendulum Rod's Position
θ'	Upper Pendulum Angular Velocity

Table A.1 – (continued)

Symbol	Description
θ''	Upper Pendulum Angular Acceleration
ω_c	Motor Shaft Angular Velocity

A.2 Parameter Values

Table A.2: Model parameters for the DIP system as provided by Quanser.

Symbol	Description	Value
B_c	Viscous Damping Coefficient as seen at the Motor Pinion	5.4 N.m.s/rad
B_1	Viscous Damping Coefficient as seen at the Lower Pendulum Axis	0.0024 N.m.s/rad
B_2	Viscous Damping Coefficient as seen at the Upper Pendulum Axis	0.0024 N.m.s/rad
g	Gravitational Constant	9.81 m/s ²
I_1	Moment of Inertia of the Lower Pendulum Rod at its Center of Gravity	2.6347E-004 kg.m ²
I_2	Moment of Inertia of the Upper Pendulum Rod at its Center of Gravity	1.1987E-003 kg.m ²
J_m	Rotational Moment of Inertia of the DC Motor's Output Shaft	3.9E-007 kg.m ²
K_g	Planetary Gearbox Gear Ratio	3.71
K_m	Back-ElectroMotive-Force Constant	0.00767 V.s/rad
K_t	Motor Torque Constant	0.00767 N.m/A

Table A.2 – (*continued*)

Symbol	Description	Value
ℓ_1	Length of Lower Pendulum from Pivot to Center of Gravity	0.1143 m
ℓ_2	Length of Upper Pendulum from Hinge to Center of Gravity	0.1778 m
L_1	Total Length of Lower Pendulum	0.2096 m
L_2	Total Length of Upper Pendulum	0.3365 m
M_c	Cart Mass	0.57 kg
M_h	Hinge Mass	0.170 kg
M_w	Extra Weight Mass	0.37 kg
M_1	Lower Pendulum Mass	0.072 kg
M_2	Upper Pendulum Mass	0.127 kg
R_m	Motor Armature Resistance	2.6 Ω
r_{mp}	Motor Pinion Radius	6.35E-003 m