Design and Control of a Self-balancing Bicycle Using an Electric Linear Actuator

by

Ziad Fawaz

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science in Engineering
(Electrical Engineering)
in the University of Michigan-Dearborn
2019

Master's Thesis Committee:

Associate Professor Sridhar Lakshmanan, Chair
Lecturer Michael Putty
Lecturer Paul Muench

Acknowledgements

This research starting as a class project, I firstly would like to thank my instructor, Dr. Paul Muench, for presenting us with this project-centered learning opportunity, my advisor, Dr. Sridhar Lakshmanan, for his academic guidance, and to the Electrical and Computer Engineering (ECE) Chair Paul Richardson for granting us the opportunity to conduct this research. A special thanks to Dr. Alireza Mohammadi and Dr. Michael Putty for their contribution and advice. I also would like to thank my colleagues, Ronald Grant Smith and Bethanie King, for their teamwork and leadership.

Table of Contents

List of Figures

Figure

Abstract

In this thesis, we utilize a high speed electric linear actuator to provide a bicycle with automatic steering capabilities that enable self-balancing when set in motion. Equations of motion of a bicycle are derived and a second-order linear system is used to model and simulate the bicycle. A PID controller design governs the steering actuation mechanism from lean angle measurements collected from a MEMS 6-axis gyroscope and accelerometer. Experimental tests on the feedback system are presented. These experimental results are improved upon by using a gain scheduling controller scheme. Additional results are presented from experiments conducted on the semi-autonomous bicycle at various speeds.

Chapter 1: Introduction

1.1 Problem Statement

The initial requirement set for this thesis was to implement a feedback controller on a bicycle using an electric linear actuator to achieve self-balancing for at least thirty feet. Because a bicycle's dynamical system is underactuated with unstable non-minimum phase roll dynamics, automatic control of a bicycle can be a real challenge. Nonetheless, leveraging certain intrinsic properties of a bicycle in motion can simplify the control approach. One such property is a bicycle's tendency to steer into its lean to maintain balance when set in motion uncontrolled. Integrating elements of control system design with this basic phenomenon of bicycle stability at non-zero speeds makes self-balancing an achievable task.

1.2 Overview of Existing Self-Stabilization Strategies

This section summarizes several approaches of bicycle stability by various researchers, including physical system designs, plant models, and underlying control strategies. The most widely used techniques to achieve self-stability include steering control, gyros, a moving mass, or a combination of the aforementioned [1].

1.2.1 Moving Masses and Gyros

Several systems have been implemented to stabilize single-track vehicles using spinning gyros and moving masses. These methods were most effective in stabilizing at either stationary or low

speeds. In moving mass stabilization, lean torque is the primary controlled state, either via an inverted pendulum or a laterally moving mass [1]. Another application of moving mass control is by Yamakita et al., where two separate controllers, a non-linear stabilizing controller and a linearized input-output controller, are utilized for stabilization [1]. In their simulations, successful results for stabilization and orientation tracking are shown. However, the use of only lean torque requires much larger feedback gains for stabilization compared to steering control, making stabilization more difficult for moving mass systems. To improve performance, an H_∞ controller is added to the feedback loop of their control algorithm [1]. As a result, this technique can minimize the closed loop impact of a perturbation, improving stabilization. An example where gyroscopic stabilization is thoroughly applied is by Yetkin et al. [2], where control moment gyroscope stabilization employs the reactive procession torque of a high speed flywheel about an axis that will balance the vehicle. They implement their system using a proportional-integral-derivative (PID) and sliding-mode controller (SMC) and evaluate the performance characteristics of their system. Their experiment results validate that CMG stabilization of a single-axis gimbal flywheel can be used to actively control inherently unstable bodies [2].

1.2.2 Steering Control

In automatic steering applications, steer torque and angle are generally the controlled variables [1]. The first robotic motorcycle with automatic steering capabilities was reported in the literature by Ruijs and Pacejka [3], where steer torque control was used to control the steer angle of the motorcycle. In order to provide their motorcycle with self-stabilization capabilities, Ruijs and Pacejka employed a remote link to set the roll angle. Later, Saguchi et al. [4] employed steer torque control for their robotic single-track machines in order to achieve path tracking under constant path curvatures. Another example is by Andreo, Cerone, et al. [5], where a setup using a

torque-driven servomotor provides a torque on the steering axis. Their system considers three degrees of freedom, namely the speed, lean angle, and steering angle. They rely on the Whipple model to design and analyze their system, and utilize a linear-parameter-varying (LPV) control scheme formulated in terms of a linear matrix inequality feasibility problem [5], with the speed *v(t)* being the time-varying parameter. Experimental tests on their system confirm the effective ability to balance a bicycle at low speeds and in spite of external disturbances [5]. Andreo et al. and Michini and Torrez [6] have recently equipped standard city bicycles with proper sensors and actuators in order to achieve automatic self-balancing. In the work by Michini and Torrez [6], a chain drive connects the bicycle's front fork to the steering motor. The use of electric linear actuators in robotics applications, such as actuating joints of anthropomorphic robotic hands, achieving high speed locomotion in quadrupedal robots, and providing required joint torques during walking in powered prostheses is gradually finding more scientific ground [7]. Electric linear actuators, with features including good controllability and high force/torque densities, can tackle control and performance drawbacks in a variety of robotic systems and substitute solutions that rely on heavy machinery.

Chapter 2: Derivation of the Bicycle Dynamical Model

Figure 1 displays a single-track, moving bicycle with forward velocity U and lateral velocity V, lean angle θ being the controlled variable, and the yaw rate ω being the second angular rotation variable. The vehicle's center of mass (CoM) is at a distance h from the ground plane, with vehicle mass m and moment of inertia I_1. The constants a and b are the distances from the front axle and rear axle to the vehicle's CoM, respectively.

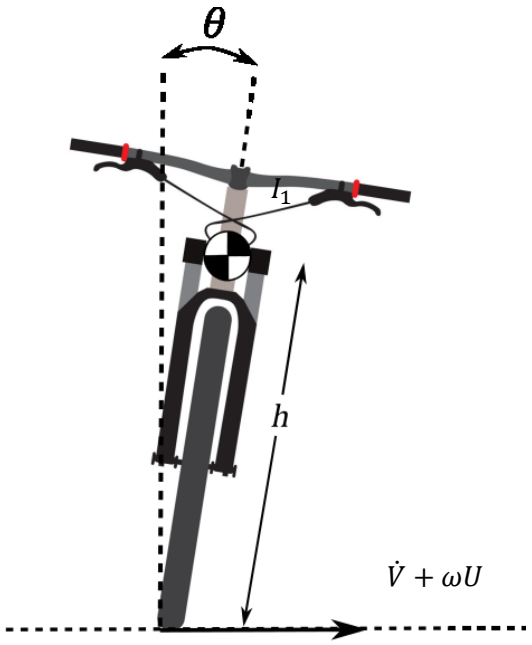

Figure 1. Front-view of a bicycle in motion

2.1 Derivation of the Equation of Motion

Using the Euler-Lagrange equation, the equation of motion for a moving bicycle is derived. Assuming zero slip angles, the tire lateral forces, that are perpendicular to the tire velocities, can be neglected. The expression for the potential energy is just *mg* times the height *h* in the gravity field [8], given by

$$V = mgh\cos\theta. \quad (1)$$

The kinetic energy expression is the sum of the translational and rotational motion energy components:

$$T = \frac{1}{2}mv^2 + \frac{1}{2}I_1\dot{\theta}^2. \quad (2)$$

Figure 2 displays the forward and lateral velocity components of the projection of the bicycle CoM.

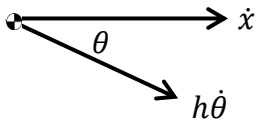

Figure 2. Forward and lateral velocity components

The translational kinetic energy is found by evaluating the velocity's x- and y-components:

$$\frac{1}{2}mv^2 = \frac{1}{2}m\left(v_x{}^2 + v_y{}^2\right) = \frac{1}{2}m\left[\left(\dot{x} + h\dot{\theta}\cos\theta\right)^2 + \left(h\dot{\theta}\sin\theta\right)^2\right]$$

$$= \frac{1}{2}m\left[\dot{x}^2 + 2h\dot{x}\dot{\theta}\cos\theta + h^2\dot{\theta}^2\cos^2\theta + h^2\dot{\theta}^2\sin^2\theta\right]$$

$$= \frac{1}{2}m(\dot{x}^2 + 2h\dot{x}\dot{\theta}\cos\theta + h^2\dot{\theta}^2). \quad (3)$$

The Lagrangian equation is then expressed as:

$$L = T - V = \frac{1}{2}m\left(\dot{x}^2 + 2h\dot{x}\dot{\theta}\cos\theta + h^2\dot{\theta}^2\right) + \frac{1}{2}I_1\dot{\theta}^2 - mgh\cos\theta. \qquad (4)$$

Partial derivatives with respect to lean angle and lean rate are expressed:

$$\frac{\partial L}{\partial \theta} = -mh\dot{x}\dot{\theta}\sin\theta + mgh\sin\theta \qquad (5)$$

$$\frac{\partial L}{\partial \dot{\theta}} = -mh\dot{x}\cos\theta + mh^2\dot{\theta} + I_1\dot{\theta} \qquad (6)$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}}\right) = mh\left(\ddot{x}\cos\theta - \dot{x}\dot{\theta}\sin\theta\right) + mh^2\ddot{\theta} + I_1\ddot{\theta}. \qquad (7)$$

Substituting Equations (5)-(7) in Euler-Lagrange,

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}}\right) - \frac{\partial L}{\partial \theta} = 0 \qquad (8)$$

$$mh\ddot{x}\cos\theta - mh\dot{x}\dot{\theta}\sin\theta + mh^2\ddot{\theta} + I_1\ddot{\theta} + mh\dot{x}\dot{\theta}\sin\theta - mgh\sin\theta = 0 \qquad (9)$$

$$(I_1 + mh^2)\ddot{\theta} - mgh\sin\theta = -mh\ddot{x}\cos\theta. \quad (10)$$

The lateral acceleration $\ddot{x}$ is the sum of the lateral speed derivative and the forward speed times the yaw rate ω [note that V in Eq. (11) differs from the potential energy expression in Eq. (1)]:

$$\ddot{x} = \dot{V} + \omega U. \qquad (11)$$

Expressions for the turn radius R, lateral velocity V, and yaw rate ω are derived in terms of the steering angle δ using basic geometric considerations [9]:

$$R = \frac{a+b}{\delta}, \qquad (12)$$

$$V \cong U\frac{b\delta}{a+b} \qquad (13)$$

$$\omega = U\frac{\delta}{a+b}. \qquad (14)$$

Using the above expressions in Eq. (10), we obtain the final form of the equation of motion:

$$(I_1 + mh^2)\ddot{\theta} - mgh\sin\theta = -\frac{mh\cos\theta}{a+b}\left(bU\dot{\delta} + U^2\delta\right). \qquad (15)$$

Small angle approximations can be considered with minimal lean angles, i.e. upright orientation:

$$(I_1 + mh^2)\ddot{\theta} - mgh\theta = -\frac{mh}{a+b}\left(bU\dot{\delta} + U^2\delta\right), \qquad (16)$$

where I_1 is the bicycle's moment of inertia, m is the bicycle's mass, h the height of the bicycle CoM from the ground, θ the lean angle, δ the steering angle, a and b the distances from the front and rear axle to the CoM, respectively, and U the bicycle's forward speed.

2.2 The Second-order Transfer Function

By dividing Eq. (16) by mgh, introducing time constants τ_1, τ_2, and a gain K, the equation appears in the form

$$\tau_1\ddot{\theta} - \theta = -K\left(\tau_2\dot{\delta} + \delta\right) \qquad (17)$$

where

$$\tau_1 = \frac{I_1+mh^2}{mgh}, \qquad (18)$$

$$\tau_2 = \frac{b}{U}, \qquad (19)$$

$$K = \frac{U^2}{g(a+b)}. \qquad (20)$$

Defining the time constants τ_1 and τ_2, and defining the gain factor *K* puts the transfer function in Eq. (21) in standard form, and reveals the parameters affecting the system poles and zeros. More precisely, that the poles of the system rely on τ_1, a function of the bicycle's physical parameters, and that the zeros rely on both the bicycle's physical parameters and the bicycle's speed.

Taking the Laplace transform of Eq. (16), the steering angle to lean angle transfer function can be found in the standard way:

$$\frac{\theta(s)}{\delta(s)} = \frac{-K(\tau_2 s+1)}{\tau_1 s^2-1}. \qquad (21)$$

With the transfer function shown in Eq. (21), it can be stated that by introducing lean angle feedback to the system with a controller in the open-loop, the steering angle can be accordingly actuated in a controlled manner to achieve stability and self-balancing.

Chapter 3: Overview of the Self-Balancing System

Hardware selection and configuration on the bicycle are crucial for the functionality and performance of the self-balancing system. In this section, we present the physical design, the sensors and actuators on the self-balancing system, as well as the electronic control system unit.

3.1 Physical Design

Prior to mounting any components on the bicycle, we took into account considerations pertaining to balance and stability of the bicycle. In order to maintain uniform mass distribution, the two pivot points of the linear actuator are mounted on the right-hand side of the bicycle's seat tube and the left-hand side of the bicycle's handlebar, as Figure 3 depicts. This configuration helps keep steady balance, evenly distributing the linear actuator's weight on both sides. The mounting brackets are placed in their respective positions. The easily removable locks simplify switching the modes of operation for the user. The accelerometer is placed on a level surface and centered across the vertical plane of the bicycle in order to accurately measure lean angle. For testing purposes, training wheels are installed in such a way that they do not touch the ground when the bicycle is at a lean angle range of approximately -15º to +15º. A battery rack installed above the rear wheel locks in place a 48 V battery and provides ample surface area for a control box that encloses the bicycle system's electronics.

The battery's heavy weight of 5.7 kg needs to be positioned along the center of the bicycle's vertical plane for symmetrical mass distribution. Figure 3 shows a detailed description of the overall system, including a top-view illustration, while Table 1 lists measurements for the system components including mass and respective heights from the ground as well as inertial calculations for the overall system.

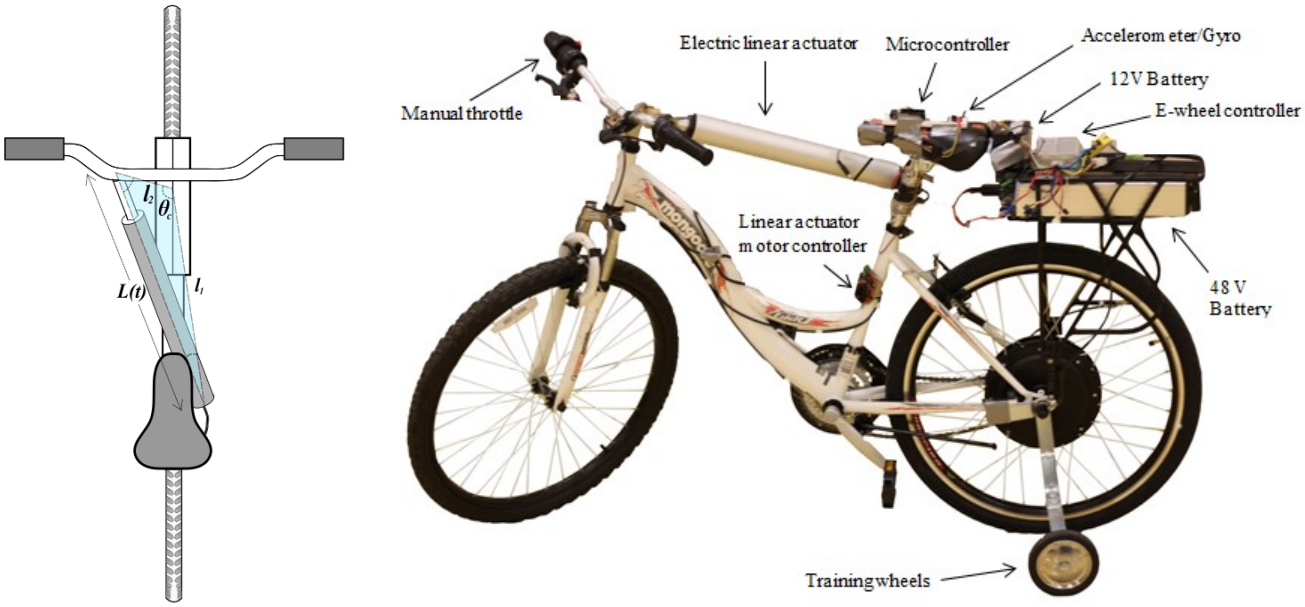

Figure 3. Physical structure description: (a) the linear actuator placement topology, (b) the self-balancing bicycle and its components

Table 1. Bicycle inertial calculations and measurements

Bicycle		Steering Electronics		Battery		Powered Wheel	
Mass	17.25 kg	Mass	3.45 kg	Mass	5.7 kg	Mass	7.5 kg
Height	0.038 m	Height	0.745 m	Height	0.77 m	Height	0.33 m
Width	0.99 m	Inertia	1.914 kg·m^2	Inertia	3.38 kg·m^2	Inertia	0.817 kg·m^2
CG Height	0.648 m				Total Mass	33.9 kg	
Inertia	8.47 kg·m^2				Total Inertia	14.581 kg·m^2	

3.2 Actuation

The first of two actuators used in the system is the MPC LAD-HS10 high-speed linear actuator and is used to control the front wheel steering of the bicycle. For this application, high-speed actuation is a preferred setting to allow for even the slightest rapid changes in steering angle to maintain stability. With a maximum stroke length of 10 inches, the linear actuator can accommodate for the desired steering angle range. The second principal actuator of the semi-autonomous bicycle is a 26-inch electric wheel that replaces the rear wheel of the bicycle. The installed e-wheel, which is taken from an e-bike kit, has its own control box that converts the 48V DC voltage to a 3-phase AC voltage. Turning the installed throttle on the handlebar controls the speed of the e-wheel. The motor actuator codes are developed using recommendations provided by the supplier of the motor driver. Several tests were performed to find the relationship between the duty cycle of the input PWM signals and the actuation speed. The microprocessor regulates the speed and direction of the linear actuator via a PWM signal transmitted to the motor controller unit. A 12V DC battery is regulated by the motor controller in order to power the electric linear actuator accordingly.

3.3 Electronics and Control System Unit

The first of two essential sensors used in this system is the InvenSense MPU6050 6-axis gyroscope and accelerometer. This motion processing unit can calculate the yaw, pitch, and roll rate of a moving object, as well as the object's acceleration in the x, y, and z axes. The MPU6050 was used to measure the roll angle and roll rate of the bicycle. Based on good reviews for quality and vast support, the MPU6050 was deemed the best sensor for this project. The accelerometer uses I^2C protocol to communicate with the microcontroller. First attempts to use

the MPU6050 chip resulted in inconsistent angle readings, leading to considerable time spent on calibrating the sensor.

To measure speed, a hall-effect sensor mounted on the chain stay of the rear wheel sends an interrupt to the microcontroller when any of the four magnets on the wheel cross the sensor's path. The hall-effect sensor thus enables travel distance calculation. By incorporating a timer, the bicycle's speed can be measured. Figure 4 shows a block diagram of the electronic control system.

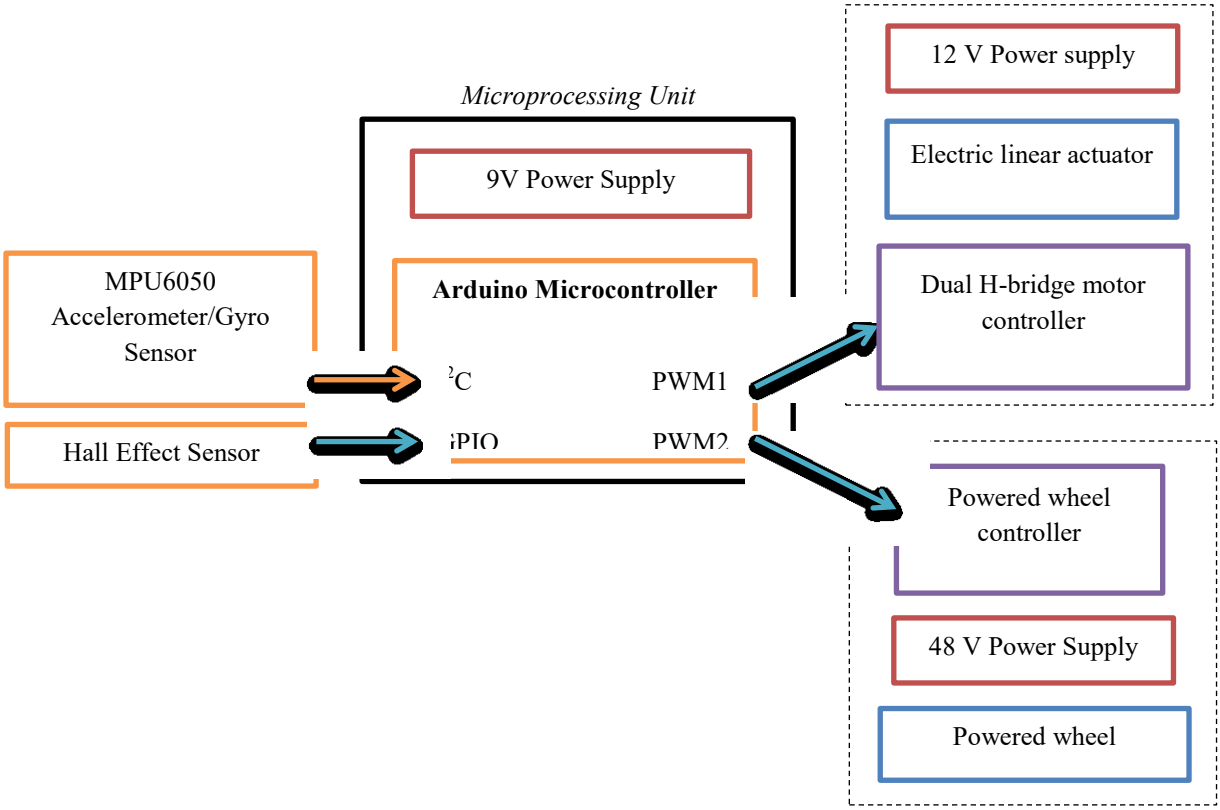

Figure 4. Block diagram of the electronic control system of the semi-autonomous bicycle

Chapter 4: Control and Experimental Results

In this section, the results of the experiments conducted on the self-balancing system are displayed, including open-loop tests, experiments with a single PID controller, and improvements to the experimental results with an upgraded, gain scheduling controller.

4.1 Open-loop Tests

4.1.1 Steering Actuation Model

The electric linear actuator drives the steering angle of the bicycle. A PWM-controlled motor controller regulates the speed and direction of extension of the linear actuator. Open-loop tests are conducted in order to map the driving PWM signal to the linear actuator extension speed. Using the obtained mapping and the law of cosines, the rate of change of the steering angle can be expressed as a function of the duty cycle of the applied PWM signal. Figure 5 depicts the functional relationship between the duty cycle of the input PWM signal and the extension speed of the linear actuator. The resulted profile shows that maximum achievable extension speed for the linear actuator is equal to 6.2 cm/s. Figure 6 depicts the time profile of the steering angle response of the semi-autonomous bicycle.

Using the law of cosines and from Figure 3a, the geometric relationship between the linear actuator length, i.e., $L(t)$, and the reference steering angle $\theta_c(t)$, is given by

$$L^2(t) = {l_1}^2 + {l_2}^2 - 2l_1 l_2 \cos(\theta_c(t)), \qquad (22)$$

where the two fixed-length segments of the triangle are given by l_1 and l_2, respectively. In addition, $\theta_c(t)$ and $L(t)$ denote the angle between the lengths l_1 and l_2, and the actuator length (plus the length of the mounting brackets) as a function of time t, respectively. Using the principle of virtual work, the relationship between actuating force and steering torque is given by

$$\tau(t) = F(t)\frac{dL}{d\theta_c} = F(t)\frac{l_1 l_2 \sin(\theta_c(t))}{\sqrt{{l_1}^2+{l_2}^2-2l_1 l_2 \cos(\theta_c(t))}}, \qquad (23)$$

where $F(t)$ is the force generated by the electric linear actuator and $\tau(t)$ is the generated steering torque. Furthermore, using Eq. (4), the relationship between the rate of change of the reference steering angle, i.e., $\dot{\theta}_c(t)$, and the linear actuator extension speed, i.e., $\dot{L}(t)$, is given by

$$\dot{\theta}_c(t) = \frac{d\theta_c}{dt}\dot{L}(t) = \frac{\sqrt{{l_1}^2+{l_2}^2-2l_1 l_2 \cos(\theta_c(t))}}{l_1 l_2 \sin(\theta_c(t))}\dot{L}(t) \qquad (24)$$

A realistic linearized dynamic model of the lean angle should take into account the non-minimum phase nature of the bicycle steering. In particular, the non-minimum phase bicycle lean dynamics should reflect the intuitive fact that steering becomes ineffective for balancing the bicycle at rest or at very low speeds. Once the bicycle reaches a certain speed, the automatic steering system can be used for making the bicycle follow a desired path.

The angles $\theta, \delta,$ and θ_c denote the lean angle, steering angle, and the reference steering angle, respectively. The lean angle dynamics are actuated through the steering angle dynamics. The

steering angle, in turn, is actuated through the electric linear actuator. Therefore, we first derive the dynamics relating the steering angle to the linear actuator length. In order to so, we note that the steering angle speed is equal to the rate of change of the reference steering angle, i.e.,

$$\dot{\theta}_c(t) = \dot{\delta}(t)$$

Therefore, using the law of cosines, the relationship

$$\dot{\delta}(t) = \frac{\sqrt{{l_1}^2 + {l_2}^2 - 2l_1 l_2 \cos(\theta_c(t))}}{l_1 l_2 \sin(\theta_c(t))} \dot{L}(t) \qquad (6)$$

holds between the linear actuator speed, i.e., $\dot{L}(t)$, and the front-wheel steering angle speed $\dot{\delta}(t)$.

4.1.2 Wheel speed actuation model

The installed electric wheel provides the bicycle with self-propulsion capabilities under the appropriate control system. A PWM-controlled motor controller regulates the desired speed. Open-loop tests are conducted in order to map the duty cycle of the driving PWM signal to the desired wheel speed. PWM and speed data are collected and curve-fitted in terms of the duty cycle to obtain the function shown below. Figure 8 shows the functional relationship between the duty cycle of the input PWM signal and the wheel speed.

$$wheel\ speed = 0.441(PWM\ duty\ cycle) - 27.248.$$

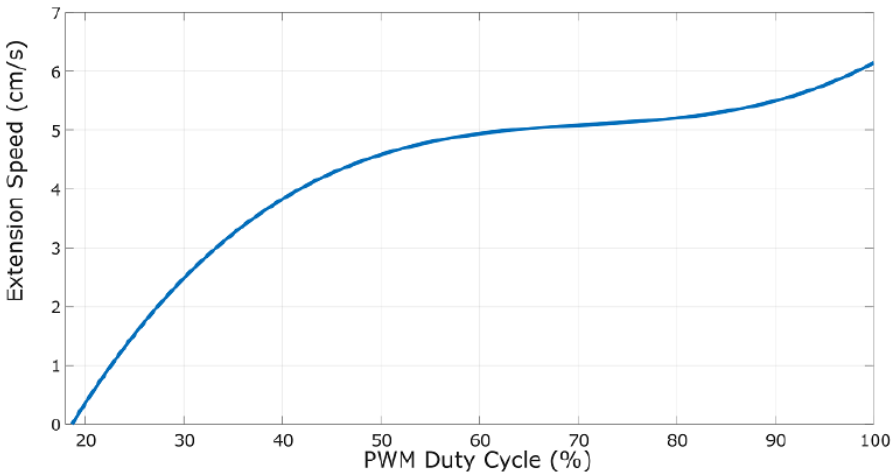

Figure 5. The relationship between the duty cycle of the input PWM signal and the linear actuator extension speed

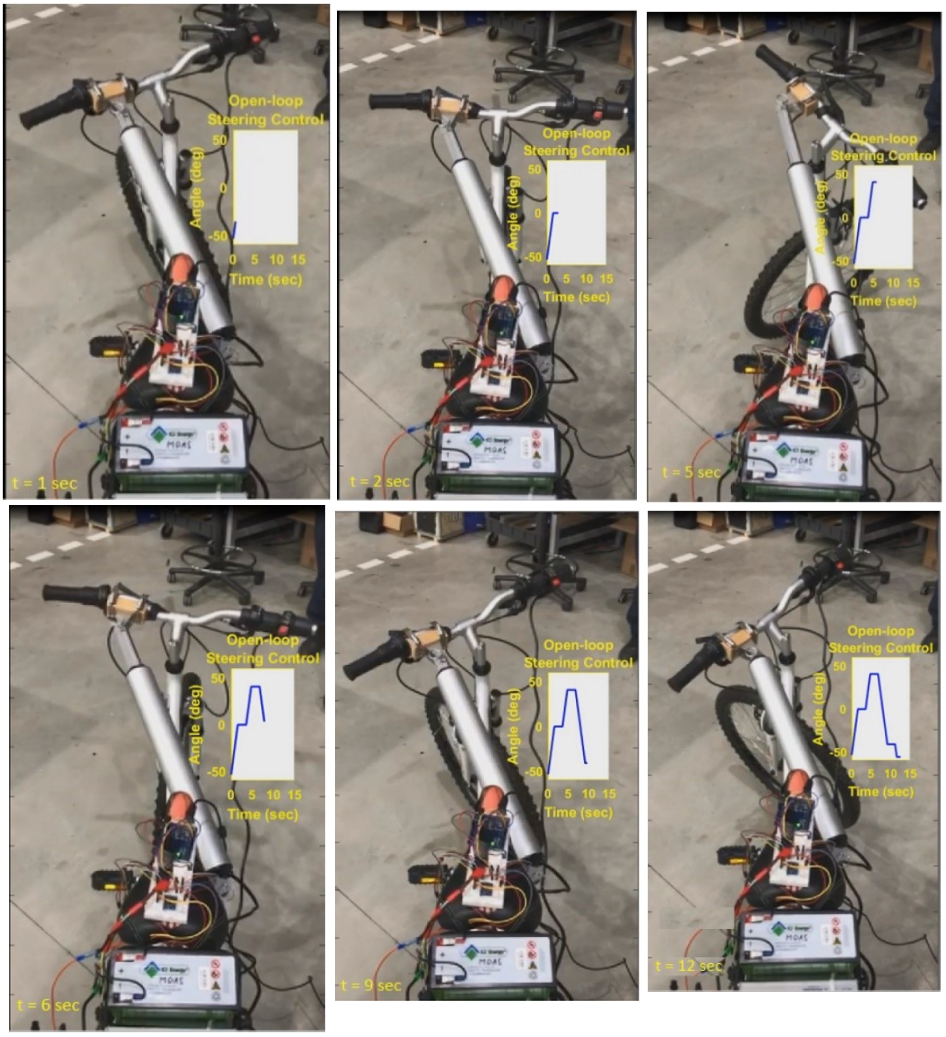

Figure 6. Automatic steering experiments

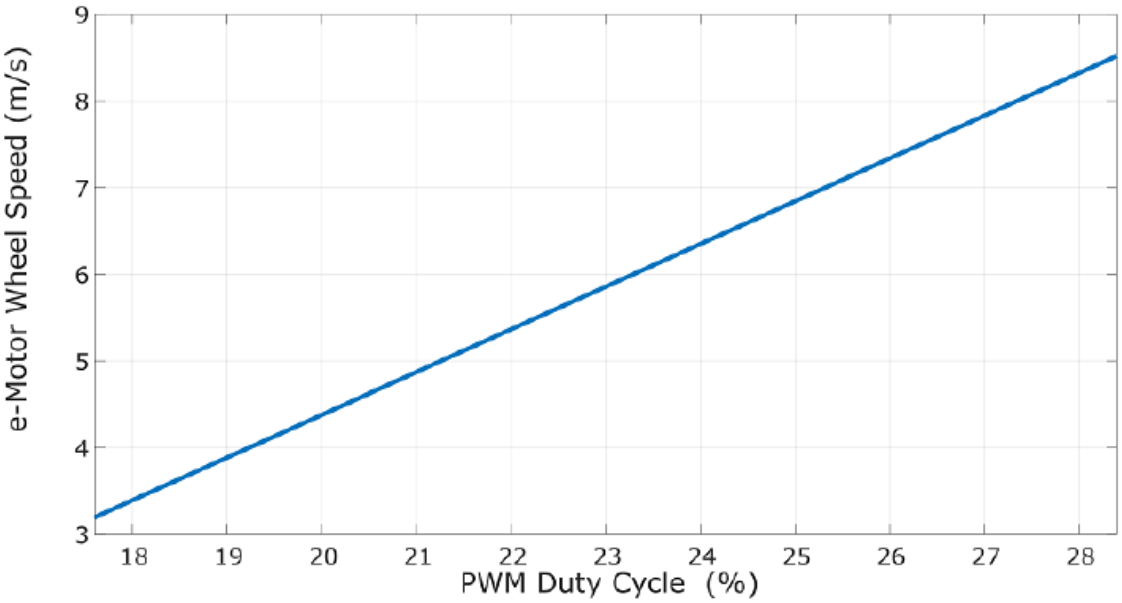

Figure 7. Wheel speed response vs. PWM duty cycle

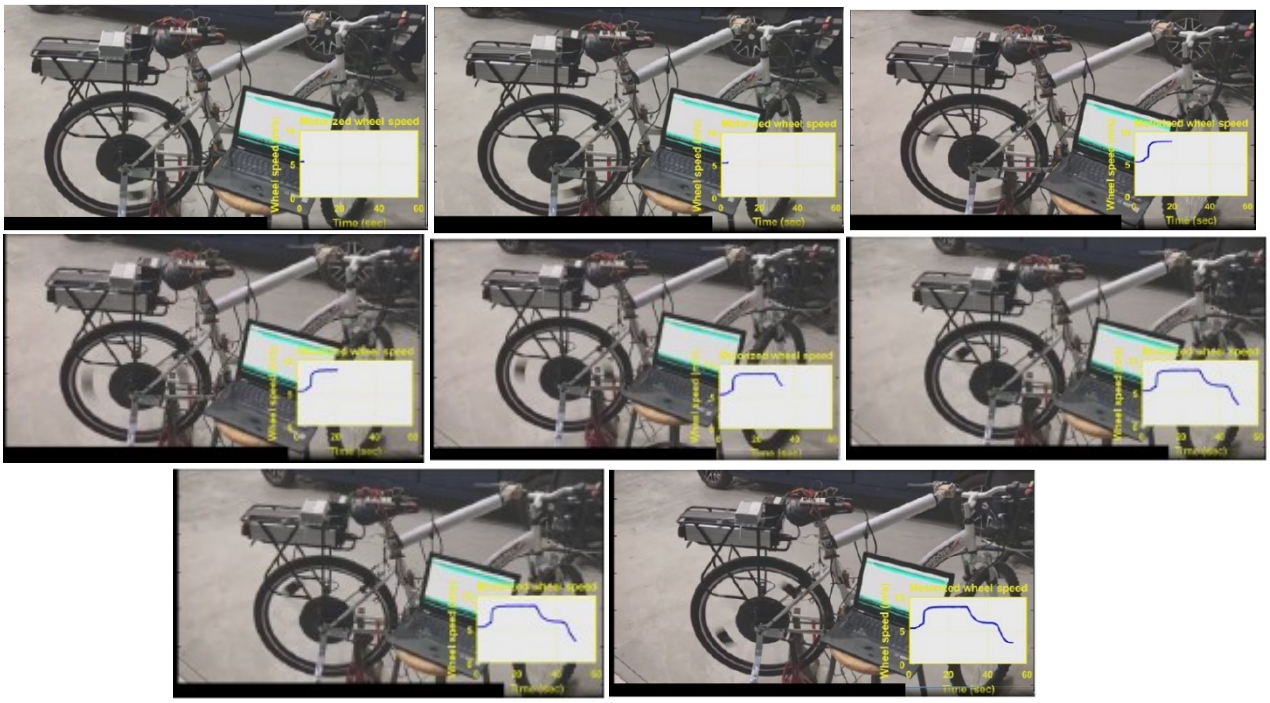

Figure 8. Wheel speed time profile

4.2 PID Control

4.2.1 PID control loop

The PID control loop for the controlled system is shown in Figure 9.

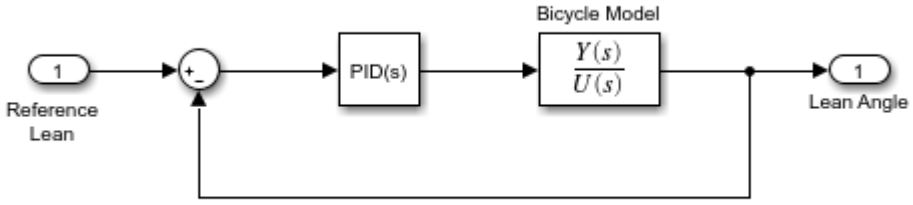

Figure 9. Closed-loop system block diagram

The first experiments using this control loop in the code were successful for PID constants

$$k_p = 10, k_i = 0.01, k_d = 13.3,$$

which were experimentally tuned. The error signal feeds into the PID block, and the output of the PID block is the PWM signal that will control the electric linear actuator's extension speed and direction. Using an Arduino Uno, the PWM signal ranges between 0 and 255. The PWM signal is clipped if that range is exceeded. The sign of the PWM signal determines the direction of actuation, i.e., a negative PWM actuates the motor backward to steer left, and a positive PWM actuates the motor forward to steer to the right.

4.2.2 Gain Scheduling Control and Experimental Results

In this section, we present a gain scheduling control scheme implemented on the bicycle to achieve self-balancing with the aim of being able to balance at different speeds. Since we are using an electric linear actuator for automatic steering, we need to take into account the nonlinear

dynamics given by Eq. (6), which depend on the reference steering angle in a nonlinear manner. While we could have used linear matrix inequalities (LMIs) for designing LPV-based control strategies as in the work by Andreo et al. [6], we decided to use gain-scheduled PID controllers for the preliminary implementation on the bicycle. In the gain scheduling control framework, the design plants consist of a collection of linearizations about equilibrium points indexed by a group of measurable variables, or scheduling variables. In our experiments, lean angle is the scheduling variable that sets the PID controller gains to achieve bicycle self-balancing according to the following conditions:

$$\dot{\delta}_{ref} = k_{P1} e_\theta + k_{D1} \dot{e}_\theta + \int k_{I1} e_\theta(\tau) d\tau\,, \quad for\ |\theta(t)| < 2°,$$

and

$$\dot{\delta}_{ref} = k_{P2} e_\theta + k_{D2} \dot{e}_\theta + \int k_{I2} e_\theta(\tau) d\tau\,, \quad for\ |\theta(t)| > 2°,$$

where $e_\theta = \theta_d - \theta$ is the lean angle tracking error. Setting different PID constants for lean angles less than ±2º in magnitude allows varying the steering dynamics when the bicycle is close to upright orientation. This is helpful when the electric linear actuator is too sensitive to changes in lean angle, and less sensitivity could reduce oscillations. Nonetheless, this sensitivity is core to the system performance. In order to achieve self-balancing at the upright position, θ_d is set to zero. In our experiments, the bicycle is initially manually pushed until reaching a sufficient speed level to continue the path of travel balancing itself. Figure 10 depicts snapshots of the experiments with the average peak forward speed equal to 3 m/s, while Figure 11 depicts snapshots with average peak forward speed at 4 m/s. The lean angle and speed time profiles in both experiments associated with faster and slower peak forward speeds are depicted in Figures 12 and 13, respectively.

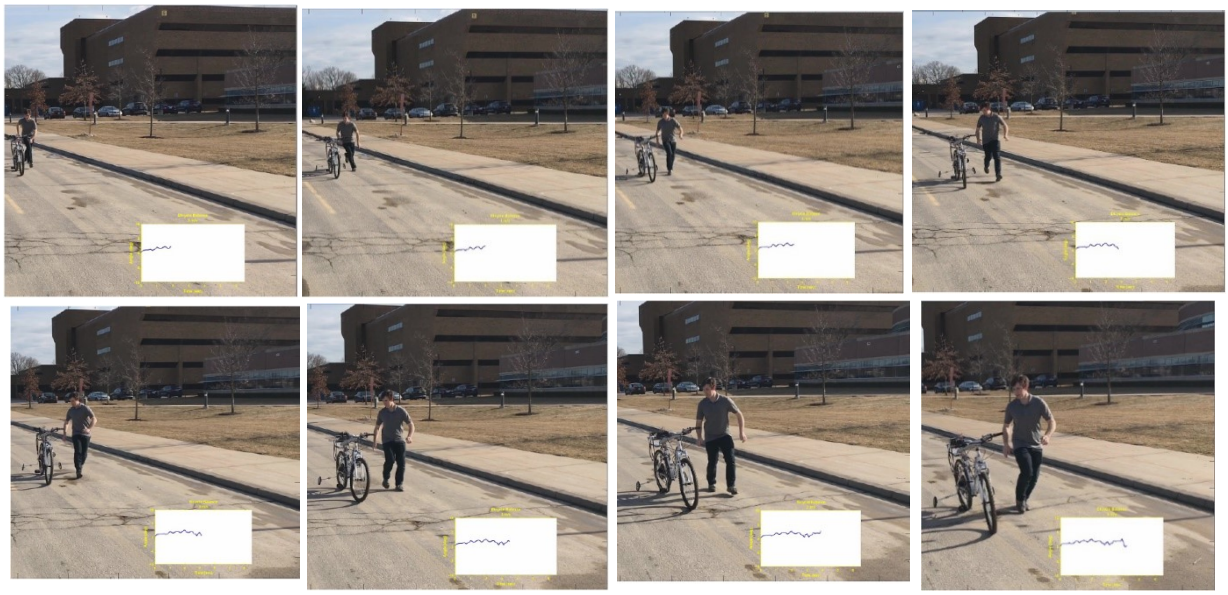

Figure 10. Snapshots of the self-balancing experiment with peak forward speed equal to 3 m/s

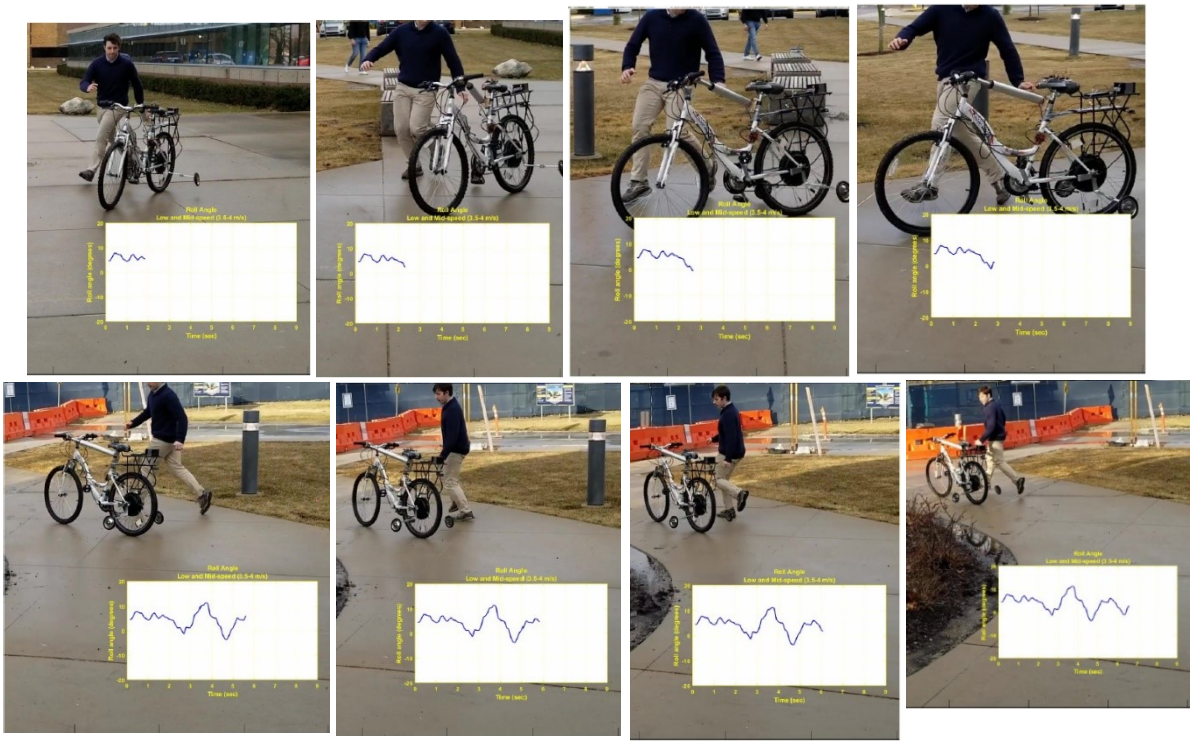

Figure 11. Snapshots of the self-balancing experiment with peak forward speed equal to 4 m/s

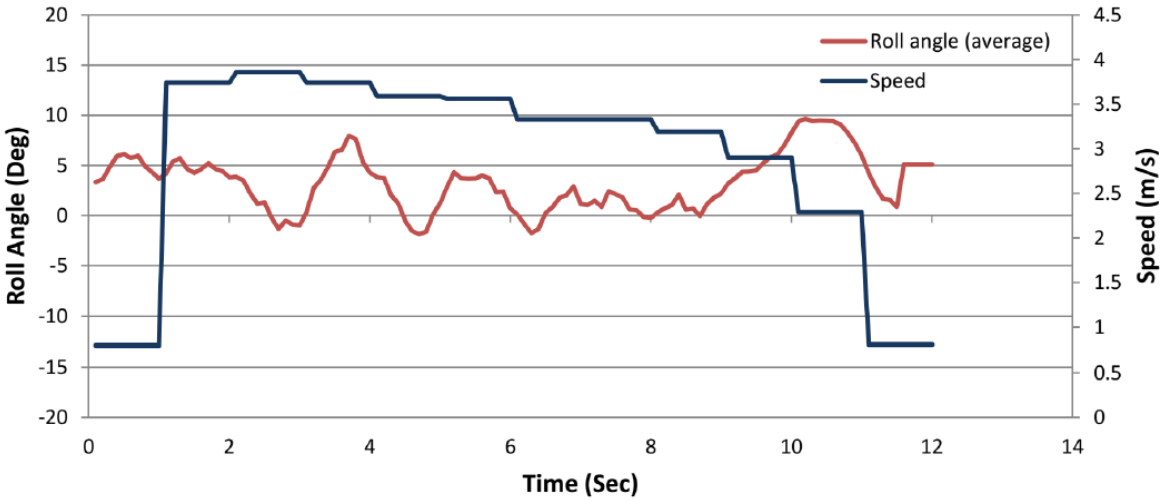

Figure 12. Lean angle and speed time profile for the *3 m/s* experiment

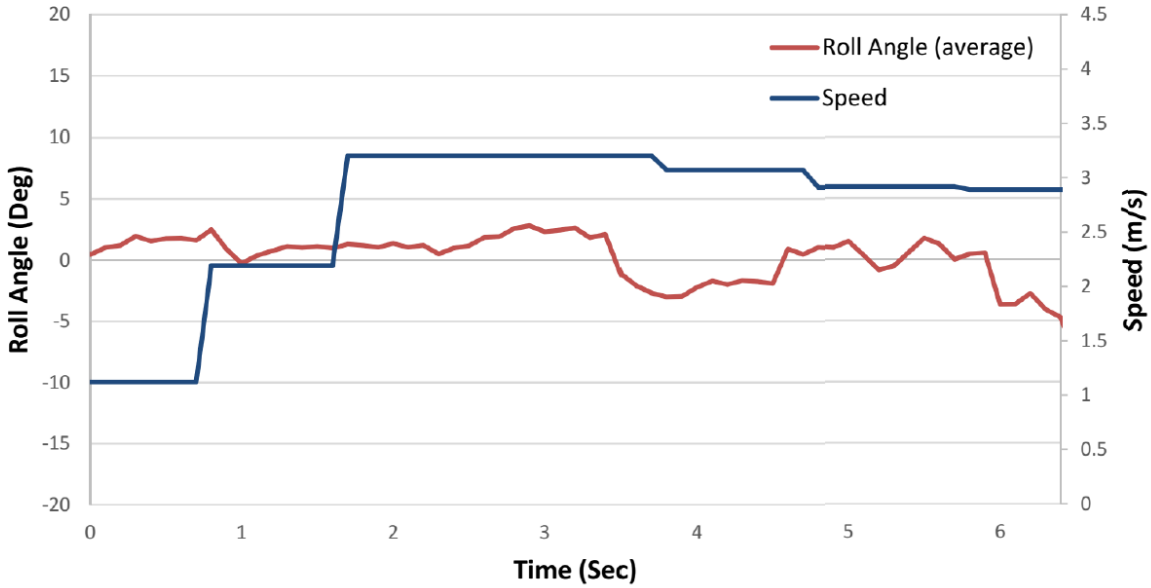

Figure 13. Lean angle and speed time profile for the *4 m/s* experiment

Chapter 5: Self-balancing System Stability

In Karnopp's Vehicle Dynamics, Stability, and Control, he models bicycle dynamics using a 2nd-order system. The transfer function of this model,

$$\frac{\theta}{\delta} = \frac{-K(\tau_2 s + 1)}{{\tau_1}^2 s^2 - 1}$$

where

$$\tau_1 = \frac{I_1 + mh^2}{mgh},$$

$$\tau_2 = \frac{b}{U},$$

$$K = \frac{U^2}{g(a+b)},$$

relates a moving bicycle's steering angle to lean angle. The lean angle, θ, is the output and δ is the steering angle or input to the second-order model [3]. The constants *a* and *b* are the lengths from the front and back wheel, respectively, to the center of gravity, and *U* represents the

minimum speed at which the bike can stay balanced. These are variables of the bicycle in terms of gravity, center of mass locations, and moving speed. For stability, our controller was designed to read the output of the bike model, the lean angle, as input to the control block minus the desired lean angle with a reference of zero degree lean, at which the bicycle is unstable at speeds under 6 m/s. Karnopp integrates the simplest possible control to the model, defining the steering angle as a constant *G* multiplied by the lean angle error, or the difference between the actual lean angle and our desired reference [8]:

$$\delta = G(\theta - \theta_d).$$

The new closed-loop transfer function directly relates the bicycle's lean angle to the desired reference lean angle [8]:

$$\frac{\theta}{\theta_d} = \frac{GK(\tau_2 s + 1)}{{\tau_1}^2 s^2 + GK\tau_2 s + GK - 1}.$$

To examine the system's stability, the eigenvalues of the system are found:

$$\lambda_{1,2} = \frac{\pm\sqrt{G^2K^2\tau_2^2 - 4GK\tau_1^2 + 4\tau_1^2} - GK\tau_2}{2\tau_1^2}.$$

For the system to be stable, the real components of $\lambda_{1,2}$ must be on the left-hand side of the complex plane:

$$\lambda_{1,2} < 0$$

$$\frac{GK\tau_2}{2\tau_1^2} > \frac{\sqrt{G^2K^2\tau_2^2 - 4GK\tau_1^2 + 4\tau_1^2}}{2\tau_1^2}$$

$$GK > 1$$

The above condition only states that if the system is stable, the product of the proportional gain *G* and *K* should be greater than 1. Pole-zero and root locus plots are displayed for the preliminary experiments conducted on the system, where a standard PID controller with a single set of gains succeeds in balancing the bicycle system. Those gains, experimentally tuned, were:

$$k_p = 10, k_i = 0.01, k_d = 13.3.$$

Using a PID controller, rather than a single gain controller to stabilize the system, the new system model should include the PID block, depicted in Figure 13. Figure 15 shows the system's pole-zero plot with the controller gains $[k_p, k_i, k_d] = [10, 0.01, 13.3]$ and for a speed of *1 m/s*, with the poles on the left-hand side of the complex plane, confirming a stable controlled system. Root locus plots reconfirm the minimum gain required for stability at different speeds.

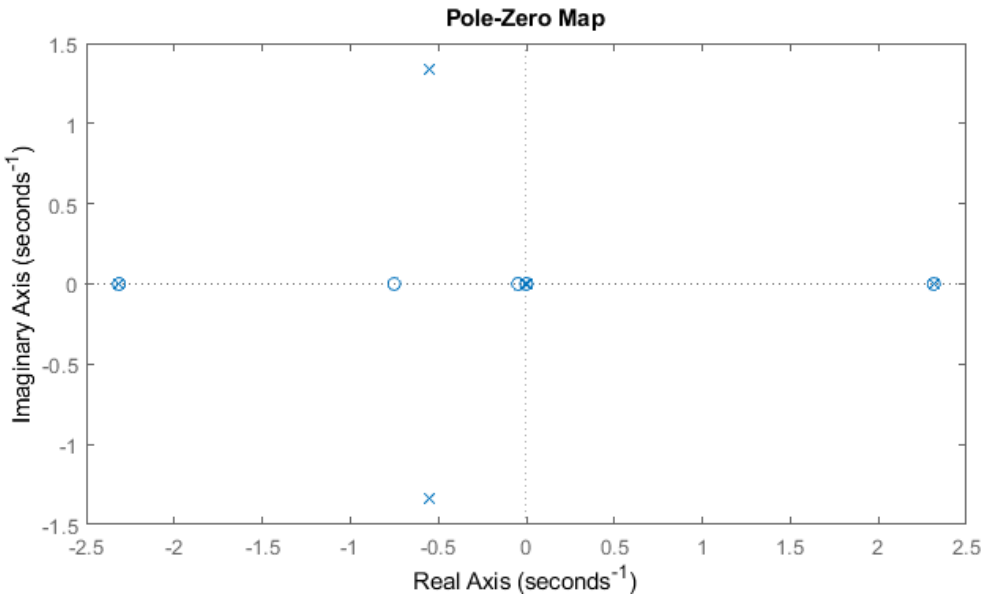

Figure 14. Controlled system pole-zero plot (1 m/s)

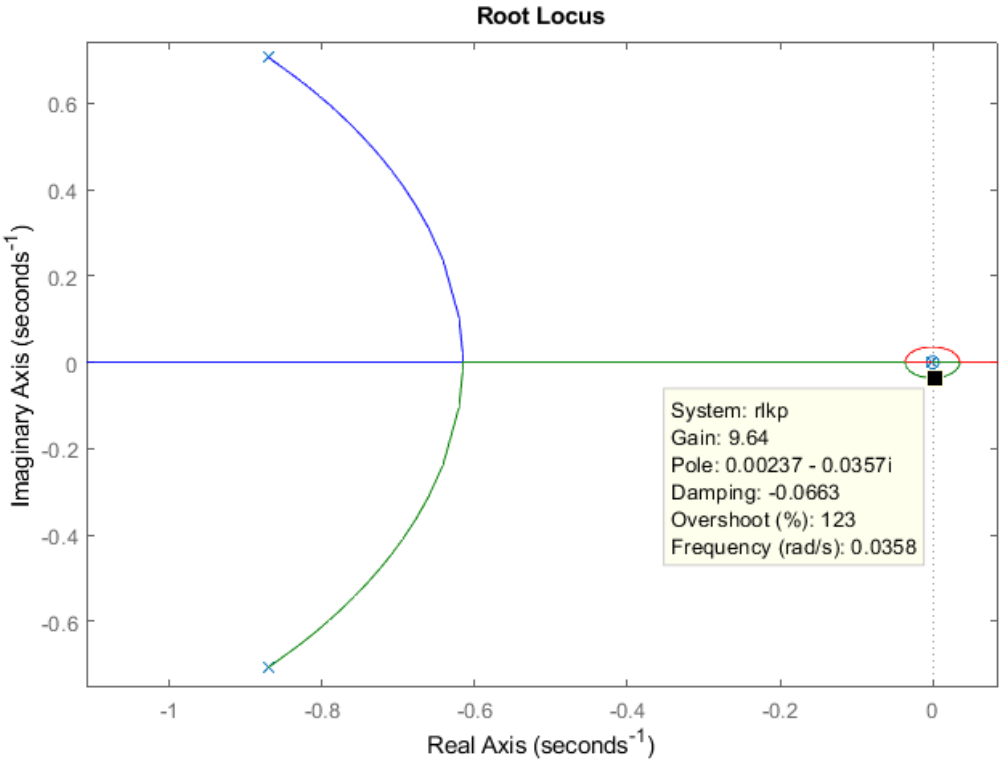

Figure 15. Root locus plot for k_p with k_d = 13.3 and k_i = 0.01at a speed equal to 1 m/s

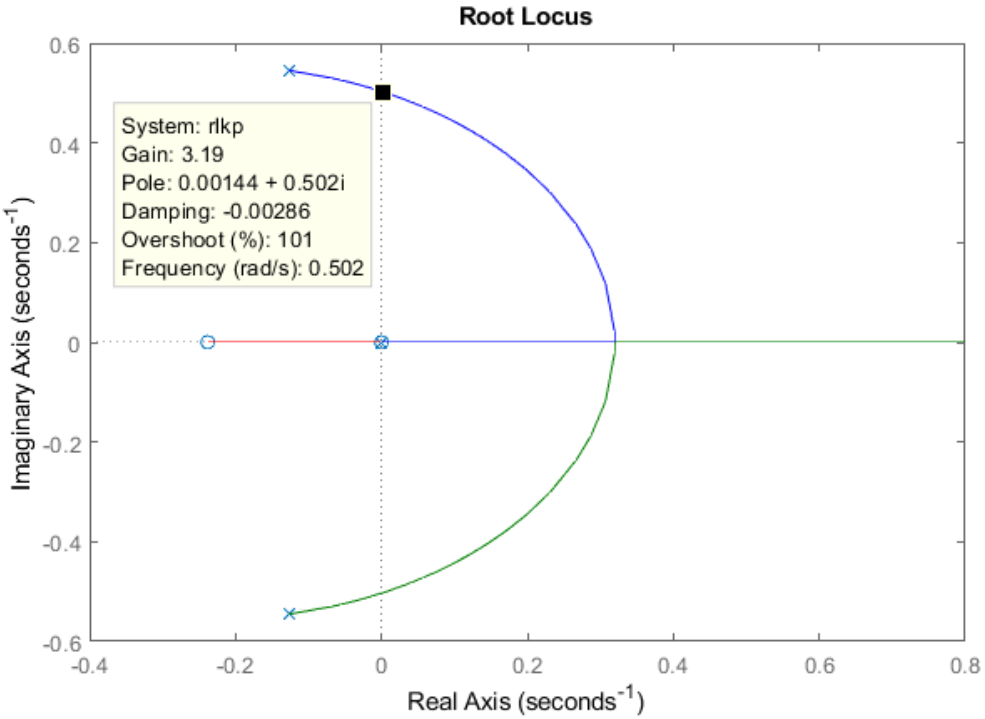

Figure 16. Root locus plot for k_p with k_d = 13.3 and k_i = 0.01at a speed equal to 5 m/s

The first root locus plot, displayed in Figure 16, shows the trajectory for poles of the system as k_p increases. A k_p value higher than 9.64 would make the system less stable, since the pole labeled in green would be on the right-hand side of the complex plane. Similarly, the root locus shown in Figure 17 indicates that a k_p value higher than 3.19 would cause instability, dragging both poles to the right hand of the complex plane. This only reflects that at higher speeds, the system does not require a very high gain.

Chapter 6: Conclusion & Future Works

This thesis covered the design and control of a semi-autonomous, self-balancing bicycle using an electric linear actuator as the primary balancer of the system. Open-loop tests are conducted on the system and its components to identify system parameters and relate them accordingly. Using the law of cosines, a model for steering dynamics can be applied in future development of the system. PID control was the basis of stabilizing the system. At first, a single PID controller with select gains succeeds in stabilizing the system at low and medium speeds. A gain scheduling control system where PID gains are scheduled depending on the bicycle lean then succeeds in balancing the system at medium range speeds. Finally, stability plots confirm the system's self-balancing ability at different speeds.

Bibliography

[1] Schwabb, A. L. and Meijaard, J., "A review on bicycle dynamics and rider control," *Vehicle System Dynamics* 51(7), 1059-1090 (2013).

[2] Yetkin, H., Kalouche, S., Vernier, M., Colvin, G., Redmill, K., and Ozguner, U., "Gyroscopic stabilization of an Unmanned Bicycle," *Stanford University*.

[3] Ruijs, P. and Pacejka, H., "Recent research in lateral dynamics of motorcycles," *Vehicle System Dynamics* **15**(sup1), 467-480 (1986).

[4] Saguchi, T., Yoshida, K., and Takahashi, M., "Stable running control of autonomous bicycle robot," *Nihon Kikai Gakkai Ronbunshu, C Hen/Trans. Japan Soc. Mech. Eng., Part C* **73**(7), 2036-2041 (2007).

[5] Andreo, D., Cerone, V., Dzung, D., and Regruto, D., "Experimental results on LPV stabilization of a riderless bicycle," *Proc. 2009 American Contr. Conf.*, 3124-3129 (2009).

[6] Michini, B. and Torrez, S., "Autonomous stability control of a moving bicycle," *Proc. Amer. Inst. Aeronaut. Astronaut.*, 1-10 (2006).

[7] Sup., F., Bohara, A., and Goldfarb, M., "Design and control of a powered transfemoral prosthesis," *Int. J. Robot. Res.* **27**(2), 263-273 (2008).

[8] Karnopp, D., [*Vehicle dynamics, stability, and control*], CRC Press, 2nd ed. (2013).