Fall 2014

Studies of Systems with Nonholonomic Constraints: the Segway and the Chaplygin Sleigh

Joseph Troy Lee Tuttle
Purdue University

Recommended Citation
https://docs.lib.purdue.edu/open_access_theses/386

Follow this and additional works at: https://docs.lib.purdue.edu/open_access_theses
Part of the Aerospace Engineering Commons, and the Mechanical Engineering Commons

This document has been made available through Purdue e-Pubs, a service of the Purdue University Libraries. Please contact epubs@purdue.edu for additional information.
This is to certify that the thesis/dissertation prepared

By  Joseph Troy Lee Tuttle

Entitled
Studies of Systems with Nonholonomic Constraints: the Segway and the Chaplygin Sleigh

For the degree of  Master of Science in Aeronautics and Astronautics

Is approved by the final examining committee:

Arthur E. Frazho

Inseok Hwang

Martin J. Corless

To the best of my knowledge and as understood by the student in the Thesis/Dissertation Agreement, Publication Delay, and Certification/Disclaimer (Graduate School Form 32), this thesis/dissertation adheres to the provisions of Purdue University’s “Policy on Integrity in Research” and the use of copyrighted material.

Arthur E. Frazho

Approved by Major Professor(s):  

Approved by: Weinong Chen  10/20/2014

Head of the Department Graduate Program  Date
STUDIES OF SYSTEMS WITH NONHOLONOMIC CONSTRAINTS: THE
SEGWAY AND THE CHAPLYGIN SLEIGH

A Thesis
Submitted to the Faculty
of
Purdue University
by
Joseph T. Tuttle

In Partial Fulfillment of the
Requirements for the Degree
of
Master of Science in Aeronautics and Astronautics

December 2014
Purdue University
West Lafayette, Indiana
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>LIST OF FIGURES</th>
<th>iv</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYMBOLS</td>
<td>viii</td>
</tr>
<tr>
<td>ABBREVIATIONS</td>
<td>xii</td>
</tr>
<tr>
<td>ABSTRACT</td>
<td>xiii</td>
</tr>
<tr>
<td>1 Introduction</td>
<td>1</td>
</tr>
<tr>
<td>2 Review of Nonholonomic Lagrangian Mechanics</td>
<td>3</td>
</tr>
<tr>
<td>3 Segway</td>
<td>6</td>
</tr>
<tr>
<td>3.1 Introduction to the Segway</td>
<td>6</td>
</tr>
<tr>
<td>3.2 Setup to Equations of Motion</td>
<td>6</td>
</tr>
<tr>
<td>3.3 Nonholonomic Constraints</td>
<td>15</td>
</tr>
<tr>
<td>3.4 Deriving the Equations of Motion</td>
<td>19</td>
</tr>
<tr>
<td>3.5 Voltage with Back EMF</td>
<td>25</td>
</tr>
<tr>
<td>3.6 Transformation of Voltage Inputs</td>
<td>26</td>
</tr>
<tr>
<td>3.7 Reference Commands Applied to Velocity and Heading Angle Rate</td>
<td>30</td>
</tr>
<tr>
<td>3.8 Inclusion of Measurement and Noise</td>
<td>34</td>
</tr>
<tr>
<td>3.9 Parameters of the Built Segway</td>
<td>38</td>
</tr>
<tr>
<td>3.10 Stabilization of Segway (Zero Reference Commands)</td>
<td>45</td>
</tr>
<tr>
<td>3.11 Test of Constant Velocity and Heading Angle Rate</td>
<td>54</td>
</tr>
<tr>
<td>3.12 Simulations of Following a Given Path</td>
<td>57</td>
</tr>
<tr>
<td>3.13 Reference Commands Applied to Arc Length and Heading Angle</td>
<td>62</td>
</tr>
<tr>
<td>3.14 Conclusions</td>
<td>63</td>
</tr>
<tr>
<td>4 Chaplygin Sleigh with a Spinning Disc</td>
<td>66</td>
</tr>
<tr>
<td>4.1 Introduction to the Chaplygin Sleigh</td>
<td>66</td>
</tr>
<tr>
<td>4.2 Derivation of Equations of Motion Without Friction</td>
<td>67</td>
</tr>
</tbody>
</table>
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1 Segway in an upright position.</td>
<td>7</td>
</tr>
<tr>
<td>3.2 Angular positions of each wheel.</td>
<td>10</td>
</tr>
<tr>
<td>3.3 Latest version of the built Segway.</td>
<td>39</td>
</tr>
<tr>
<td>3.4 Real-time data of $l$ versus time plot for stabilizing (zero reference command).</td>
<td>46</td>
</tr>
<tr>
<td>3.5 Real-time data of $v$ versus time for stabilizing (zero reference command).</td>
<td>46</td>
</tr>
<tr>
<td>3.6 Real-time data of $\psi$ versus time for stabilizing (zero reference command).</td>
<td>47</td>
</tr>
<tr>
<td>3.7 Real-time data of $\dot{\psi}$ versus time for stabilizing (zero reference command).</td>
<td>47</td>
</tr>
<tr>
<td>3.8 Real-time data of $\theta$ versus time for stabilizing (zero reference command).</td>
<td>47</td>
</tr>
<tr>
<td>3.9 Real-time data of $\dot{\theta}$ versus time for stabilizing (zero reference command).</td>
<td>47</td>
</tr>
<tr>
<td>3.10 Simulation of $l$ versus time plot for stabilizing (zero reference command).</td>
<td>48</td>
</tr>
<tr>
<td>3.11 Simulation of $v$ versus time for stabilizing (zero reference command).</td>
<td>48</td>
</tr>
<tr>
<td>3.12 Simulation of $\psi$ versus time for stabilizing (zero reference command).</td>
<td>49</td>
</tr>
<tr>
<td>3.13 Simulation of $\dot{\psi}$ versus time for stabilizing (zero reference command).</td>
<td>49</td>
</tr>
<tr>
<td>3.14 Simulation of $\theta$ versus time for stabilizing (zero reference command).</td>
<td>49</td>
</tr>
<tr>
<td>3.15 Simulation of $\dot{\theta}$ versus time for stabilizing (zero reference command).</td>
<td>49</td>
</tr>
<tr>
<td>3.16 Simulation of motor input voltage versus time for stabilizing (zero reference command).</td>
<td>50</td>
</tr>
<tr>
<td>3.17 Comparison of real-time data and simulation of $l$ versus time plot for stabilizing (zero reference command).</td>
<td>51</td>
</tr>
<tr>
<td>3.18 Comparison of real-time data and simulation of $v$ versus time for stabilizing (zero reference command).</td>
<td>51</td>
</tr>
<tr>
<td>3.19 Comparison of real-time data and simulation of $\theta$ versus time for stabilizing (zero reference command).</td>
<td>52</td>
</tr>
<tr>
<td>3.20 Comparison of real-time data and simulation of $\dot{\theta}$ versus time for stabilizing (zero reference command).</td>
<td>52</td>
</tr>
<tr>
<td>Figure</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>------</td>
</tr>
<tr>
<td>3.21</td>
<td>53</td>
</tr>
<tr>
<td>3.22</td>
<td>53</td>
</tr>
<tr>
<td>3.23</td>
<td>53</td>
</tr>
<tr>
<td>3.24</td>
<td>53</td>
</tr>
<tr>
<td>3.25</td>
<td>55</td>
</tr>
<tr>
<td>3.26</td>
<td>55</td>
</tr>
<tr>
<td>3.27</td>
<td>55</td>
</tr>
<tr>
<td>3.28</td>
<td>55</td>
</tr>
<tr>
<td>3.29</td>
<td>56</td>
</tr>
<tr>
<td>3.30</td>
<td>56</td>
</tr>
<tr>
<td>3.31</td>
<td>60</td>
</tr>
<tr>
<td>3.32</td>
<td>60</td>
</tr>
<tr>
<td>3.33</td>
<td>60</td>
</tr>
<tr>
<td>3.34</td>
<td>60</td>
</tr>
<tr>
<td>3.35</td>
<td>61</td>
</tr>
<tr>
<td>3.36</td>
<td>61</td>
</tr>
<tr>
<td>4.1</td>
<td>66</td>
</tr>
<tr>
<td>4.2</td>
<td>75</td>
</tr>
<tr>
<td>4.3</td>
<td>80</td>
</tr>
<tr>
<td>4.4</td>
<td>80</td>
</tr>
<tr>
<td>4.5</td>
<td>80</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>4.6</td>
<td>Simulation of $\dot{x}$ versus time of constant input without friction.</td>
</tr>
<tr>
<td>4.7</td>
<td>Simulation of $\dot{y}$ versus time of constant input without friction.</td>
</tr>
<tr>
<td>4.8</td>
<td>Simulation of $x$ versus time of constant input without friction.</td>
</tr>
<tr>
<td>4.9</td>
<td>Simulation of $y$ versus time of constant input without friction.</td>
</tr>
<tr>
<td>4.10</td>
<td>Simulation of distance versus time of constant input without friction.</td>
</tr>
<tr>
<td>4.11</td>
<td>Simulation of $xy$ position of constant input without friction.</td>
</tr>
<tr>
<td>4.12</td>
<td>Simulation of $v$ versus time for heading angle controller.</td>
</tr>
<tr>
<td>4.13</td>
<td>Simulation of $\dot{\theta}$ versus time for heading angle controller.</td>
</tr>
<tr>
<td>4.14</td>
<td>Simulation of $\theta$ versus time for heading angle controller.</td>
</tr>
<tr>
<td>4.15</td>
<td>Simulation of input versus time for heading angle controller.</td>
</tr>
<tr>
<td>4.16</td>
<td>Simulation of $xy$ position for heading angle controller.</td>
</tr>
<tr>
<td>4.17</td>
<td>Simulation of $v$ versus time with friction.</td>
</tr>
<tr>
<td>4.18</td>
<td>Simulation of $\dot{\theta}$ versus time with friction.</td>
</tr>
<tr>
<td>4.19</td>
<td>Simulation of $xy$ position with friction.</td>
</tr>
<tr>
<td>4.20</td>
<td>Simulation of $v$ versus time for constant heading with friction.</td>
</tr>
<tr>
<td>4.21</td>
<td>Simulation of $\dot{\theta}$ versus time for constant heading with friction.</td>
</tr>
<tr>
<td>4.22</td>
<td>Simulation of $\theta$ versus time for constant heading with friction.</td>
</tr>
<tr>
<td>4.23</td>
<td>Simulation of input versus time for constant heading with friction.</td>
</tr>
<tr>
<td>4.24</td>
<td>Simulation of $xy$ position for constant heading with friction.</td>
</tr>
<tr>
<td>4.25</td>
<td>Simulation of $v$ versus time for sine and heading angle controller.</td>
</tr>
<tr>
<td>4.26</td>
<td>Simulation of $\dot{\theta}$ versus time for sine and heading angle controller.</td>
</tr>
<tr>
<td>4.27</td>
<td>Simulation of $\theta$ versus time for sine and heading angle controller.</td>
</tr>
<tr>
<td>4.28</td>
<td>Simulation of input versus time for sine and heading angle controller.</td>
</tr>
<tr>
<td>4.29</td>
<td>Simulation of $xy$ position for sine and heading angle controller. vspace5.21 mm.</td>
</tr>
<tr>
<td>4.30</td>
<td>Simulation of $v$ versus time for sinusoidal input.</td>
</tr>
<tr>
<td>4.31</td>
<td>Simulation of $\dot{\theta}$ versus time for sinusoidal input.</td>
</tr>
<tr>
<td>4.32</td>
<td>Simulation of $\theta$ versus time for sinusoidal input.</td>
</tr>
<tr>
<td>Figure</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>------</td>
</tr>
<tr>
<td>4.33 Simulation of $\omega$ versus $v$ phase plot for sinusoidal input.</td>
<td>105</td>
</tr>
<tr>
<td>4.34 Simulation of $\omega$ versus $v$ phase plot for sinusoidal input (close up).</td>
<td>105</td>
</tr>
<tr>
<td>4.35 Simulation of $\dot{x}$ versus time for sinusoidal input.</td>
<td>105</td>
</tr>
<tr>
<td>4.36 Simulation of $\dot{y}$ versus time for sinusoidal input.</td>
<td>105</td>
</tr>
<tr>
<td>4.37 Simulation of $x$ versus time for sinusoidal input.</td>
<td>106</td>
</tr>
<tr>
<td>4.38 Simulation of $y$ versus time for sinusoidal input.</td>
<td>106</td>
</tr>
<tr>
<td>4.39 Simulation of $xy$ position for sinusoidal input.</td>
<td>106</td>
</tr>
<tr>
<td>4.40 Simulation of fast Fourier transform of the velocities.</td>
<td>107</td>
</tr>
<tr>
<td>4.41 Simulation of fast Fourier transform of the positions.</td>
<td>107</td>
</tr>
<tr>
<td>4.42 Simulation of $v$ versus time for sinusoidal input (small mean input).</td>
<td>108</td>
</tr>
<tr>
<td>4.43 Simulation of $y$ versus $x$ for sinusoidal input for 120 seconds.</td>
<td>109</td>
</tr>
<tr>
<td>4.44 Simulation of $y$ versus $x$ for sinusoidal input with $\omega_u = 1$.</td>
<td>109</td>
</tr>
<tr>
<td>4.45 Simulation of $y$ versus $x$ for sinusoidal input with $\omega_u = 0.3$.</td>
<td>109</td>
</tr>
<tr>
<td>4.46 Simulation of $y$ versus $x$ for sinusoidal input with $\omega_u = \frac{5}{4}\dot{\theta}_s$.</td>
<td>110</td>
</tr>
<tr>
<td>4.47 Simulation of $y$ versus $x$ for sinusoidal input with $\omega_u = \frac{1}{3}\dot{\theta}_s$.</td>
<td>111</td>
</tr>
<tr>
<td>A.1 Simulink model without noise (Segway).</td>
<td>127</td>
</tr>
<tr>
<td>A.2 Simulink model with noise (Segway).</td>
<td>128</td>
</tr>
<tr>
<td>B.1 Simulink model for constant input (Chaplygin Sleigh).</td>
<td>136</td>
</tr>
</tbody>
</table>
SYMBOLS

The following are general symbols that are used throughout the paper. Some symbols have multiple listed meanings that depend on the context.

Latin symbols:

- \( a \) half axle length
- \( a \) rose length parameter
- \( A \) state matrix of state space representation
- \( B \) input matrix of state space representation
- \( C \) output matrix of state space representation
- \( D \) feedforward matrix of state space representation
- \( \vec{e} \) error vector
- \( F(\cdot) \) generalized forces coefficient matrix of \( (\cdot) \)
- \( g \) Earth’s gravitational constant
- \( h \) height above center of mass
- \( H \) Pfaffian non-holonomic constraint matrix
- \( J \) moment of inertia (matrix and scalar)
- \( k \) voltage constant
- \( K \) feedback matrix
- \( l \) arc length
- \( L \) Lagrangian
- \( m \) total mass
- \( m(\cdot) \) mass component in reference to \( (\cdot) \)
- \( M(\cdot) \) mass matrix in reference to \( (\cdot) \)
- \( n \) general size of matrix or vector
rose pedal number parameter

\( p \) unconstrained general coordinates

\( P \) steady state solution to Riccati equation

\( q \) constrained general coordinates

\( Q \) state weight matrix
   solution to Riccati equation

\( r \) radial length

\( \vec{r} \) radial vector
   reference command

\( R \) controllability matrix
   input weight matrix

\( R_{(1)}^{(2)} \) frame rotation from the (1) frame to the (2) frame

\( S \) skew-symmetric matrix

\( t \) time

\( T \) total kinetic energy

\( T_{(\cdot)} \) torque input in reference to (\( \cdot \))

\( \bar{u} \) input matrix

\( v \) velocity in scalar form

\( \vec{v} \) velocity in vector form

\( V \) potential energy

\( V_{(\cdot)} \) voltage input in reference to (\( \cdot \))

\( \bar{w} \) random vector

\( \vec{x} \) position vector
   state space vector

\( \hat{x} \) \( x \) unit vector
   estimated state vector

\( \hat{y} \) \( y \) unit vector

\( \hat{z} \) \( z \) unit vector
Greek symbols:
\(\alpha\) angular position of Segway’s left wheel
\(\beta\) angular position of Segway’s right wheel
\(\beta_i\) leftover terms in reference to (\cdot)
\(\zeta\) shorthand expression for simplification purpose
\(\eta\) shorthand expression for simplification purpose
\(\theta\) the second Euler angle that is often known as the pitch angle
\(\lambda\) eigenvalue
\(\bar{\lambda}\) Lagrangian multiplier vector
\(\sigma\) standard deviation
\(\phi\) the third Euler angle that is often known as the roll angle
\(\Phi\) basis for nullspace of Pfaffian nonholonomic constraint matrix
\(\varphi\) angular velocity in body coordinates
\(\psi\) the first Euler angle that is often known as the yaw angle
\(\Psi\) polar angle
\(\omega\) angular velocity in inertial coordinates

general angular velocity

Superscripts and subscripts:
(\cdot)^b, (\cdot)_b main body frame
(\cdot)_c constant term
circle
(\cdot)_d difference
(\cdot)_e voltage constant
(\cdot)_f free speed
(\cdot)^i, (\cdot)_i inertial frame
(\cdot)^l, (\cdot)_l left wheel body frame
(\cdot)_L linear
(\cdot)_p unconstrained generalized coordinates
\(\cdot_q\) constrained generalized coordinates
\(\cdot^r, \cdot_r\) right wheel body frame
\(\cdot_r\) reference command
\(\cdot_{rot}\) rotational
\(\cdot_s\) state command
\(\text{sum}\) sampling rate
\(\cdot_{ss}\) steady state
\(\cdot_{st}\) stall
\(\cdot_t\) linear secular term
\(\cdot_{tr}\) translational
\(\cdot_v\) back electromagnetic force constant
\(\cdot_w\) wheel, or axial of wheel
\(\cdot_{wt}\) transverse of wheel
\(\cdot_x\) \(x\) axis
\(\cdot_y\) \(y\) axis
\(\cdot_z\) \(z\) axis
\(\cdot_0\) initial
\(\cdot^\ast\) transpose
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DOF</td>
<td>Degrees of Freedom</td>
</tr>
<tr>
<td>EMF</td>
<td>Electromagnetic Force</td>
</tr>
<tr>
<td>LQR</td>
<td>Linear Quadratic Regulator</td>
</tr>
<tr>
<td>PD</td>
<td>Proportional and Derivative</td>
</tr>
<tr>
<td>PSD</td>
<td>Power Spectrum Density</td>
</tr>
</tbody>
</table>

In this thesis, two systems with nonholonomic systems are investigated: the Segway and the Chaplygin sleigh. Using Lagrangian mechanics, the constrained nonlinear equations of motion for both systems are derived. By use of the nullspace of the constraint matrices, the unconstrained equations of motion can be obtained. For the Segway, these equations are linearized about a zero equilibrium state, placed into state space form and decoupled. A feedback controller is designed about the velocity and heading angle rate reference commands. To compare to the real data from the built Segway, measurement noise was also included in the model. Experimental data is taken for the case of both zero and constant reference commands. The data is then compared to the simulated results. The model is shown to be satisfactory, but better parameter measurements of the Segway is needed for a more conclusive comparison. The unconstrained equations of motion for the Chaplygin sleigh can not be linearized. Thus Lyapunov stability theory was used for analysis. The Chaplygin sleigh with constant input was shown to spiral outward and settle into a circle. If a PD feedback controller was designed about the heading angle, then the Chaplygin sleigh would be driven to the angle, but would eventually coast to a stop. From simulations, the addition of a sinusoidal component appears to move in the desired direction without slowing down. A sinusoidal component was also added to a constant input to result in roulette like paths in the simulation. Future investigation would require a more definite analysis of the sinusoidal term in the input.
1. INTRODUCTION

For each type of physical system, its behavior can be described by a set of equations. Known as the equations of motion, these equations are derived from the most fundamentals laws of physics applied to the system. However, most systems have constraints that adds complexity in deriving these equations.

By definition, constraints are considered any restrictions that would limit or influence the motion in anyway. For example, if two parts of a system are linked together by a pin, then there is a constraint acting on the system. This constraint can be expressed as a mathematical relationship in which the position of the pin on both parts are equivalent. This is an example of a holonomic constraint, in which the constraint is described as a function of the generalized coordinates of the system. The term generalized coordinates refers to the list of parameters that define the configuration of the system at each instant of time, and are either spacial or angular positions. Although there is no unique set of generalized coordinates, one set can be more convenient to describe the system than another set.

Although a system can be described by many possible choices of generalized coordinates and constraint equations, the degrees of freedom is fixed for that particular system. The degrees of freedom, or DOF, is the minimum number of parameters that can be used to describe the system. The degrees of freedom is equal to difference between the size of any set of generalized coordinates and the number of constraints between those generalized coordinates. If the holonomic constraint can be rearranged such that one generalized coordinate can be expressed as a function of the other generalized coordinates, then that generalized coordinate can be substituted out, and the number of constraint equations is decreased by one. This can be repeated to further simplify the derivation of the equations of motion. After the equations of motion are
derived for the other generalized coordinates, then that generalized coordinate can be related to the others through the constraint.

However, there are some constraints that are instead nonholonomic. The nonholonomic constraint is a constraint that is described as a function between the time derivatives of the generalized coordinates, and cannot be integrated into a holonomic constraint. The nonholonomic constraint cannot be eliminated before the derivation of the equations of motion, and thus must be considered during the derivation. This method will be described later in the section. The two most common nonholonomic constraints are the skate constraint that applies to a skate blade and the rolling with slipping constraint that applies to both rolling wheels and balls. The skate constraint constricts the motion of the blade such that it cannot move perpendicular to the direction of its orientation. The rolling with slipping constraint requires that the instantaneous velocity of the ground contact with respect to the inertial frame to be zero.

There are many systems that have these constraints applied. In particular, we will investigate two different systems that have at least one of these constraints. First, we will investigate the equations of motion of the Segway and design a feedback controller to use in a built form of the Segway. Next, we will derive the equations of motion for the Chaplygin sleigh with a spinning disc attached. Using Lyapunov stability theory, several types of controllers under different scenarios will be analyzed. The investigation of both these systems represents a wide variation of the behavior a nonholonomic constraint can produce.
2. REVIEW OF NONHOLONOMIC LAGRANGIAN MECHANICS

To derive the equations of motion for systems with nonholonomic constraints, we will use Lagrangian mechanics. Recall that the Lagrangian $L$ is defined as the difference between the kinetic energy $T$ and the potential energy $V$, that is

$$ L = T - V. \quad (2.1) $$

The Lagrangian $L = L(q, \dot{q})$ is a function of the generalized coordinates, $q$, and its time derivative.

If there is no forces acting on the system, including constraint forces, then the equation of motion are given by

$$ \frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = 0. \quad (2.2) $$

For our purposes, the left-hand side of Equation (2.2) can be organized into the form of

$$ \frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = M_q \ddot{q} - \beta_q \quad (2.3) $$

where $M$ is a positive matrix, and $\beta_q = \beta(q, \dot{q})$. The subscript $q$ of $M_q$ and $\beta_q$ emphasizes that they are of the system with the $q$ generalized coordinates.

If there are forces acting on the system, then they are added to the right-hand side of Equation (2.2). For example, if there are inputs acting on the system, then Equation (2.2) becomes

$$ M_q \ddot{q} - \beta_q = F_q \vec{u}. $$

The term $F_q \vec{u}$ is the generalized input forces, such that $F_q$ is the input coefficient matrix, and $\vec{u}$ is the vector of inputs applied to the system.
In addition to the input forces, there are constraint forces acting on the system. As stated, the constraints acting on the system are nonholonomic. They can be expressed in Pfaffian form such that

\[ H \dot{q} = 0. \] (2.4)

The matrix \( H = H(q) \) is referred to as the constraint matrix. Following Flannery [1], the constraint forces are given by \( H^* \bar{\lambda} \), where \( \bar{\lambda} \) is the vector of Lagrange multipliers. Adding the constraint forces, the equations of motion become

\[ M_q \ddot{q} = \beta_q + H^* \bar{\lambda} + F_q \bar{u}. \] (2.5)

Equation (2.5) can be referred to as the constrained equations of motion.

The final step involves eliminating the Lagrange multipliers from Equation (2.5). First, find a set of basis \( \{ \varphi_1, \varphi_2, ..., \varphi_m \} \) for the nullspace of \( H \), where \( m \) is the nullity of \( H \). Then the matrix \( \Phi = [\varphi_1, \varphi_2, ..., \varphi_m] \) is a one to one matrix from \( \mathbb{R}^m \) onto the nullspace of \( H \) satisfying

\[ H \Phi = 0. \]

By taking the transpose, we obtain \( \Phi^* H^* = 0 \). Therefore, the Lagrange multipliers can be eliminated by multiplying Equation (2.5) by \( \Phi^* \) to result in

\[ \Phi^* M_q \ddot{q} = \Phi^* \beta_q + \Phi^* H^* \bar{\lambda} + \Phi^* F_q \bar{u} \]

\[ = \Phi^* \beta_q + \Phi^* F_q \bar{u}. \] (2.6)

Because the solution must satisfy \( H \dot{q} = 0 \), we can look for solutions of the form

\[ \dot{q} = \Phi \dot{p}. \] (2.7)

By taking the derivative of Equation (2.7) we obtain

\[ \ddot{q} = \Phi \ddot{p} + \Phi \dot{p}. \] (2.8)

Substituting Equation (2.8) into Equation (2.6), the following can be obtained

\[ \Phi^* M_q \Phi \ddot{p} = \Phi^* \beta_q - \Phi^* M_q \Phi \dot{p} + \Phi^* F_q \bar{u}. \] (2.9)
Equation (2.9) can be placed into the form of

\[ M_p \ddot{p} = \beta_p + F_p \ddot{u}. \]  

(2.10)

This leads to a new mass matrix \( M_p \), \( \beta_p \), and force matrix \( F_p \) given by

\[
M_p = \Phi^* M_q \Phi \\
\beta_p = \Phi^* \beta_q - \Phi^* M_q \dot{\Phi} \dot{p} \\
F_p = \Phi^* F_q.
\]

Since \( M_q \) is positive, then \( M_p \) must also be a positive matrix. Because Equation (2.10) does not have constraints, they will referred to as the unconstrained equations of motion. The generalized coordinates \( p \) will also be referred to the unconstrained generalized coordinates.

Because there are an infinite number of possible matrices \( \Phi \), there results in an infinite number of possible \( p \) as well. Therefore, there is an art to choosing an ideal \( p \) such that the generalized coordinates have a physical meaning.
3. SEGWAY

3.1 Introduction to the Segway

For our purposes, the Segway is defined as an inverted pendulum that balances on two wheels connected by an axle. Each wheel is independently controlled and powered by an electric motor. A general drawing of a Segway is given in Figure 3.1. When aligned along the inertial frame, the coordinate system is given such that the origin is set at the middle of the Segway’s base, the $x$ direction is towards the front of the Segway, the $y$ direction is through the left wheel, and the $z$ direction is up along the vertical.

As part of the investigation of the Segway, an actual Segway was built based on the description above. A state space model is first found from the equation of motions. A feedback controller is then designed with reference commands applied to the velocity and heading angle rate. To resemble the actual Segway, measurement noise is taken into account through a linear steady state Kalman Filter. The results of the actual Segway are then compared to the simulations as a final step in the investigation.

3.2 Setup to Equations of Motion

To derive the equations of motion, the Segway is divided into three main segments, the body and the two wheels.
Fig. 3.1. Segway in an upright position.

In the case of the Segway, the wheels are assumed to not lift off the ground. As a result, the wheel axis is restricted to remain in the inertial $xy$ frame, shown in Figure 3.1. If the main body is first rotated about the $z$ axis by $\psi$, then about the $y$ axis by $\theta$, then all possible orientations will follow this constraint. The orientation of the main body with respect to the inertial frame can be described by the Euler sequence

$$R_{ib} = R_z(\psi) R_y(\theta).$$

where $R_{ib}$ is the rotation matrix from the body to the inertial frame. The constraint imposed by the wheel reduces the degrees of freedom of the rotation by one, thus resulting in only two Euler angles. The rotation matrix in Equation (3.1) also transforms vectors in the main body coordinates into inertial coordinates. If aviation terminology is used, $\theta$ and $\psi$ are the pitch and yaw angles respectively, with the roll
angle forced to zero. For frame rotations from the body to the inertial frame, the individual rotation matrices about the $y$ and $z$ axis are given as

$$R_y(\theta) = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix}$$

$$R_z(\psi) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (3.2)$$

The rotation matrix from the inertial to the body frame is denoted as $R_{i}^{b}$ and is equal to the transpose of $R_{i}^{b}$, that is $R_{i}^{b} = R_{i}^{b *}$. The angular velocity of the main body can be found through the relation

$$\dot{R}_{b}^{i} = S_{\omega}^{b}R_{b}^{i} = R_{b}^{i *}S_{\omega b}$$

where $\omega^{b}$ is the angular velocity in inertial coordinates, and $\omega b$ is the angular velocity in the body coordinates. The matrix $S_{\omega}^{b}$ is the skew-symmetric matrix with respect to $\omega^{b}$. In general, if

$$\omega = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}, \quad \text{then} \quad S_{\omega} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix}. \quad (3.4)$$

By inverting the rotation matrices in Equation (3.3), the two skew-symmetric matrices $S_{\omega}^{b}$ and $S_{\omega b}$ are found by

$$S_{\omega}^{b} = \dot{R}_{b}^{i}R_{b}^{i *}$$

$$S_{\omega b} = R_{b}^{i *}\dot{R}_{b}^{i}. \quad (3.5)$$

By consulting Equation (3.4), Equation (3.5) leads to the angular velocities of the Segway’s main body

$$\omega^{b} = \begin{bmatrix} -\dot{\theta} \sin(\psi) \\ \dot{\theta} \cos \psi \\ \dot{\psi} \end{bmatrix}. \quad (3.6)$$
and

\[
\omega^b = \begin{bmatrix}
-\psi \sin(\theta) \\
\dot{\theta} \\
\psi \cos(\theta)
\end{bmatrix}.
\] (3.7)

The two angular velocities are also related by

\[
\omega^b = R^b_i \omega^b
\]

as a rotation matrix acts as a transformation matrix from one set of coordinates to another.

In addition to the main body frame, there are separate frames attached to the centers of the left and right wheels. In particular, the wheels are attached to the motor shaft, while the motor housing is attached to the Segway. Therefore, there are angular positions that can be defined as the relative angle between the motor shaft and housing. These angular positions are equivalent to the relative angle of the wheel with respect to the Segway. They are denoted as \(\alpha\) and \(\beta\) for the left and right wheel respectively, as shown in Figure 3.2. That is, there is a cusp on the shaft, and the angle from the cusp to the housing is \(\alpha\) and \(\beta\). As a result, the rotation matrices from the left and right wheel frames to the inertial frame are

\[
R^i_l = R^i_b R^b(y(\alpha)) \quad \text{and} \quad R^i_r = R^i_b R^b(y(\beta)).
\] (3.8)
Fig. 3.2. Angular positions of each wheel.

Because of the order of the rotations in Equation (3.8), the final two rotations are about the $y$ axis, such that the corresponding angles can be added as

$$R_l = R_z(\psi)R_y(\theta + \alpha) \quad \text{and} \quad R_r = R_z(\psi)R_y(\theta + \beta). \quad (3.9)$$

In other words, because the angle from the housing to the vertical is $\theta$, the total angle from the cusp to the vertical is $(\theta + \alpha)$ and $(\theta + \beta)$.

The rotation matrices in Equation (3.9) are then applied to Equation (3.5), such that

$$S_{\omega_l} = \dot{R}_l R^*_l \quad \text{and} \quad S_{\omega_r} = \dot{R}_r R^*_r \quad (3.10)$$

The resulting angular velocities of the left and right frame in inertial coordinates are

$$\omega_l = \begin{bmatrix} -\dot{\psi} + \dot{\psi}_l \sin(\psi) \\ (\dot{\psi} + \dot{\psi}_l) \cos(\psi) \\ \dot{\psi} \end{bmatrix} \quad \text{and} \quad \omega_r = \begin{bmatrix} -(\dot{\psi} + \dot{\psi}_r) \sin(\psi) \\ (\dot{\psi} + \dot{\psi}_r) \cos(\psi) \\ \dot{\psi} \end{bmatrix}. \quad (3.11)$$
The same result is obtained if $\theta$ is replaced with $(\theta + \alpha)$ for the left wheel and $(\theta + \beta)$ for the right wheel in Equation (3.6). Similar to Equation (3.7), the angular velocities for the left and right wheel in their frame’s coordinates are

\[
\varpi^l = \begin{bmatrix}
-\dot{\psi} \sin(\theta + \alpha) \\
\dot{\theta} + \dot{\alpha} \\
\dot{\psi} \cos(\theta + \alpha)
\end{bmatrix} \quad \text{and} \quad \varpi^r = \begin{bmatrix}
-\dot{\psi} \sin(\theta + \beta) \\
\dot{\theta} + \dot{\beta} \\
\dot{\psi} \cos(\theta + \beta)
\end{bmatrix}.
\]  

(3.12)

With the angular velocities of both the body and the two wheels known, the velocity of the center of mass for each segment is determined next. To do so, their positions must be related to the origin of the inertial frame. Suppose the midpoint of the axle is given as

\[
\vec{x} = \begin{bmatrix} x \\ y \\ 0 \end{bmatrix}
\]

such that the main body frame’s origin lies at this point. If the Segway is considered symmetric about the $xz$ and $yz$ planes, then the center of mass’s position for the main body is at a height $h$ along the main body’s $z$ axis, as shown in Figure 3.1. The center of mass of the main body is expressed as $\vec{x}_b = \vec{x} + h\hat{z}^b$ where $\hat{z}^b$ is the $z$ unit vector of the main body frame. The position of the center of mass can be expressed in inertial coordinates as

\[
\vec{x}_b = \begin{bmatrix} x \\ y \\ 0 \end{bmatrix} + R^b_r \begin{bmatrix} 0 \\ 0 \\ h \end{bmatrix} = \begin{bmatrix} x + h \cos(\psi) \sin(\theta) \\ y + h \sin(\psi) \sin(\theta) \\ h \cos(\theta) \end{bmatrix}. 
\]  

(3.13)

Differentiating the center of mass’s position yields the velocity

\[
\dot{\vec{x}}_b = \begin{bmatrix} \dot{x} \\ \dot{y} \\ -h \sin(\theta) \dot{\theta} \end{bmatrix} + \begin{bmatrix} -\sin(\psi) \\ \cos(\psi) \\ 0 \end{bmatrix} h \dot{\psi} \sin(\theta) + \begin{bmatrix} \cos(\psi) \\ \sin(\psi) \\ 0 \end{bmatrix} h \cos(\theta) \dot{\theta}.
\]  

(3.14)
The norm of the velocity squared is given by

\[ \| \dot{\vec{x}}_b \|^2 = \dot{x}^2 + \dot{y}^2 + h^2 \sin(\theta)^2 \dot{\theta}^2 + h^2 \sin(\theta)^2 \dot{\psi}^2 + h^2 \cos(\theta)^2 \dot{\theta}^2 \]

\[ + 2 (-\dot{x} \sin(\psi) + \dot{y} \cos(\psi)) h \sin(\theta) \dot{\psi} \]

\[ + 2 (\dot{x} \cos(\psi) + \dot{y} \sin(\psi)) h \cos(\theta) \dot{\theta}. \]

This simplifies to

\[ \| \dot{\vec{x}}_b \|^2 = \dot{x}^2 + \dot{y}^2 + h^2 \dot{\theta}^2 + h^2 \sin(\theta)^2 \dot{\psi}^2 \]

\[ + 2 (-\dot{x} \sin(\psi) + \dot{y} \cos(\psi)) h \sin(\theta) \dot{\psi} \]

\[ + 2 (\dot{x} \cos(\psi) + \dot{y} \sin(\psi)) h \cos(\theta) \dot{\theta}. \] (3.15)

Let \( \vec{x}_r \) be the position for the center of mass of the right wheel. As shown in Figure (3.1), its position of the right wheel is related to the Segway’s origin by \( \vec{x}_r = \vec{x} - a \hat{y}_b \) where \( a \) is the half the length of the axle connecting the wheels and \( \hat{y}_b \) is the unit vector of the body frame in the \( y \) direction. In inertial coordinates \( \vec{x}_r \) becomes

\[ \vec{x}_r = \begin{bmatrix} x + a \sin(\psi) \\ y - a \cos(\psi) \\ 0 \end{bmatrix}. \] (3.16)

The velocity is given by

\[ \dot{\vec{x}}_r = \begin{bmatrix} \dot{x} + a \cos(\psi) \dot{\psi} \\ \dot{y} + a \sin(\psi) \dot{\psi} \\ 0 \end{bmatrix}. \] (3.17)

The norm squared, \( \| \dot{\vec{x}}_r \|^2 \), is determined as

\[ \| \dot{\vec{x}}_r \|^2 = \dot{x}^2 + \dot{y}^2 + a^2 \dot{\psi}^2 + 2a (\cos(\psi) \dot{x} + \sin(\psi) \dot{y}) \dot{\psi}. \] (3.18)

The left wheel is treated in the same fashion, except it is on the opposite side of the body. Therefore, the center of mass, \( \vec{x}_l \), is given by \( \vec{x}_l = \vec{x} + a \hat{y}_b \), resulting in

\[ \vec{x}_l = \begin{bmatrix} x - a \sin(\psi) \\ y + a \cos(\psi) \\ 0 \end{bmatrix}. \] (3.19)
when expressed in inertial coordinates. The velocity of the left wheel is

\[
\dot{\mathbf{x}}_l = \begin{bmatrix}
\dot{x} - a \cos(\psi) \dot{\psi} \\
\dot{y} - a \sin(\psi) \dot{\psi} \\
0
\end{bmatrix}.
\] (3.20)

Moreover, the norm squared, \( \|\dot{\mathbf{v}}_l\|^2 \), is given by

\[
\|\dot{\mathbf{x}}_l\|^2 = \dot{x}^2 + \dot{y}^2 + a^2 \dot{\psi}^2 - 2a (\cos(\psi) \dot{x} + \sin(\psi) \dot{y}) \dot{\psi}.
\] (3.21)

With the velocities and angular velocities of each component known, the Lagrangian can be calculated. The Lagrangian is the kinetic energy \( T \) minus the potential energy \( V \), that is

\[
L = T - V.
\] (3.22)

The kinetic energy is the sum of the translational energy \( T_{tr} \) and the rotational kinetic energy about the center of mass \( T_{rot} \). The translational energy for the system is given by

\[
T_{tr} = \frac{1}{2} \left( m_b \|\dot{\mathbf{v}}_b\|^2 + m_w \|\dot{\mathbf{v}}_l\|^2 + m_w \|\dot{\mathbf{v}}_r\|^2 \right)
\] (3.23)

where the mass of the main body and each wheel is denoted as \( m_b \) and \( m_w \) respectively. The total mass is also given as

\[
m = m_b + 2m_w.
\] (3.24)

The rotational energy is determined by

\[
T_{rot} = \frac{1}{2} \left( \omega^b^* J_b \omega^b + \omega^l^* J_{wh} \omega^l + \omega^r^* J_{wh} \omega^r \right)
\] (3.25)

If the main body is assumed to be symmetric about the \( xy \), \( xz \), and \( yz \) plane, then its moment of inertia \( J_b \) is expressed as

\[
J_b = \begin{bmatrix}
J_{b,x} & 0 & 0 \\
0 & J_{b,y} & 0 \\
0 & 0 & J_{b,z}
\end{bmatrix}
\] (3.26)
If the wheel is assumed to be symmetric about the \( y \) axis, then its moment of inertia, \( J_{wh} \) is given by

\[
J_{wh} = \begin{bmatrix}
J_{wt} & 0 & 0 \\
0 & J_w & 0 \\
0 & 0 & J_{wt}
\end{bmatrix}
\]

where \( J_w \) is the axial moment of inertia, and \( J_{wt} \) is the transverse moment of inertia.

By applying the angular velocities from Equations (3.7) and (3.12), Equation (3.25) becomes

\[
T_{rot} = \frac{1}{2} \left( J_{b,x} \sin^2(\theta) \dot{\psi}^2 + J_{b,y} \dot{\theta}^2 + J_{b,z} \cos^2(\theta) \dot{\psi}^2 \right) \\
+ \frac{1}{2} \left( 2J_{wt} \dot{\psi}^2 + J_w \left( \dot{\theta} + \dot{\alpha} \right)^2 + J_w \left( \dot{\theta} + \dot{\beta} \right)^2 \right). \tag{3.27}
\]

In order to simplify the expression, set

\[
J_x = J_{b,x} + 2J_{wt} \\
J_y = J_{b,y} \\
J_z = J_{b,z} + 2J_{wt}.
\]

Then Equation (3.27) becomes

\[
T_{rot} = \frac{1}{2} \left( J_x \sin^2(\theta) \dot{\psi}^2 + J_y \dot{\theta}^2 + J_z \cos^2(\theta) \dot{\psi}^2 \right) \\
+ \frac{1}{2} \left( J_w \left( \dot{\theta} + \dot{\alpha} \right)^2 + J_w \left( \dot{\theta} + \dot{\beta} \right)^2 \right). \tag{3.28}
\]

Because the height of the wheel’s center of mass remains constant, they do not contribute to the potential energy. Therefore, the potential energy for the Segway is given by the product of the main body’s mass, the gravitational constant, and the height of the main body’s center of mass. Consulting Equation (3.13), the total potential energy is given by

\[
V = m_\theta g h \cos(\theta). \tag{3.29}
\]
Therefore, the Lagrangian is found as

\[
L = \frac{1}{2} m_b \left( \dot{x}^2 + \dot{y}^2 + h^2 \dot{\theta}^2 + h^2 \dot{\psi}^2 \sin(\theta)^2 \right) \\
+ m_b (\cos(\psi) \dot{x} + \sin(\psi) \dot{y}) h \sin(\theta) \dot{\psi} \\
+ m_b (\cos(\psi) \dot{x} + \sin(\psi) \dot{y}) h \cos(\theta) \dot{\theta} \\
+ m_w \left( \dot{x}^2 + \dot{y}^2 + a^2 \dot{\psi}^2 \right) \\
+ \frac{1}{2} \left( J_x \sin^2(\theta) \dot{\psi}^2 + J_y \dot{\theta}^2 + J_z \cos^2(\theta) \dot{\psi}^2 \right) \\
+ \frac{1}{2} \left( J_w \left( \dot{\theta} + \dot{\alpha} \right)^2 + J_{\omega} \left( \dot{\theta} + \dot{\beta} \right)^2 \right) \\
- m_b g h \cos(\theta). \tag{3.30}
\]

### 3.3 Nonholonomic Constraints

The first nonholonomic constraints are obtained from the knowledge that the wheels must move perpendicular to the axle. First, let us consider the right wheel. The velocity, \( \dot{\mathbf{r}} \), from Equation (3.17) is orthogonal to the axle direction

\[
\begin{bmatrix}
- \sin(\psi) \\
\cos(\psi) \\
0
\end{bmatrix}^*.
\]

Hence, the scalar product of the two vectors must be zero, that is

\[
0 = \left( \dot{x} + a \cos(\psi) \dot{\psi} \right) \left( - \sin(\psi) \right) + \left( \dot{y} + a \sin(\psi) \dot{\psi} \right) \cos(\psi)
= -\dot{x} \sin(\psi) + \dot{y} \cos(\psi).
\]

This leads to the Pfaffian constraint

\[
\begin{bmatrix}
- \sin(\psi) \\
\cos(\psi)
\end{bmatrix}^* \begin{bmatrix}
\dot{x} \\
\dot{y}
\end{bmatrix} = 0. \tag{3.31}
\]

Equation (3.31) shows that \( \begin{bmatrix} \dot{x} & \dot{y} \end{bmatrix}^* \) is in the nullspace of \( \begin{bmatrix} - \sin(\psi) & \cos(\psi) \end{bmatrix}^* \). As a result, Equation (3.31) is the nonholonomic constraint corresponding to the condition
that the right wheel cannot move sideways. The solution to Equation (3.31) can be expressed as

\[
\begin{bmatrix}
\dot{x} \\
\dot{y}
\end{bmatrix} = \begin{bmatrix}
\cos(\psi) \\
\sin(\psi)
\end{bmatrix} v
\]

(3.32)

where \(v\) is some function. Because \(\dot{x}^2 + \dot{y}^2 = v^2\), the function \(v\) can be interpreted as the velocity of the center of the Segway’s base, with \(\psi\) as the heading angle.

Now apply the same nonholonomic constraint for the left wheel. The velocity \(\dot{x}_l\) is also orthogonal the the axle. Hence,

\[
0 = \left(\dot{x} - a \cos(\psi) \dot{\psi}\right)(-\sin(\psi)) + \left(\dot{y} - a \sin(\psi) \dot{\psi}\right) \cos(\psi)
= -\dot{x} \sin(\psi) + \dot{y} \cos(\psi).
\]

(3.33)

This is precisely the same nonholonomic Pfaffian constraint as Equation (3.31). The repeating of this constraint equation offers no new information. In addition, the constraint equation also does not depend on \(a\), which suggests that the motion would be constrained in the same manner for any length of the axle.

The other constraint equations result from the non-slipping conditions between each wheel and the ground. This constraint requires the instantaneous bottom of the wheel to have a velocity of zero. The constraint results in the following equations

\[
\omega_l \times \vec{r} + \dot{x}_l = 0 \\
\omega_r \times \vec{r} + \dot{x}_r = 0
\]

(3.34)

where \(\vec{r}\) is the radial vector pointing down, given by

\[
\vec{r} = \begin{bmatrix} 0 & 0 & -r \end{bmatrix}
\]
while $r$ is the radius of the wheel. The angular and translational velocities about the center of mass in the inertial frame are given by

$$
\omega^l = \begin{bmatrix}
-\sin(\psi)(\dot{\theta} + \dot{\alpha}) & \cos(\psi)(\dot{\theta} + \dot{\alpha}) & \dot{\psi} \\
0 & 0 & 0
\end{bmatrix}^T
$$

$$
\omega^r = \begin{bmatrix}
-\sin(\psi)(\dot{\theta} + \dot{\beta}) & \cos(\psi)(\dot{\theta} + \dot{\beta}) & \dot{\psi} \\
0 & 0 & 0
\end{bmatrix}^T
$$

$$
\hat{x}_l = \begin{bmatrix}
\dot{x} - a \cos(\psi)\dot{\psi} & \dot{y} - a \sin(\psi)\dot{\psi} & 0
\end{bmatrix}^T
$$

$$
\hat{x}_r = \begin{bmatrix}
\dot{x} + a \cos(\psi)\dot{\psi} & \dot{y} + a \sin(\psi)\dot{\psi} & 0
\end{bmatrix}^T
$$

(3.35)

The angular and translational velocities were previously derived in Equations (3.11), (3.17), and (3.20).

The total velocity is the sum of the translational velocity of the wheel’s center plus the rotational velocity about the center. The center of the wheel is also the center of mass for both wheels. Notice that both the angular velocities are in the inertial frame. This is required as the no slipping condition is taken with respect to the inertial frame. Taking the cross product in Equation (3.34) results in six equations, two of which result in zero under all conditions. The four remaining constraint equations are

$$
\dot{x} - a \cos(\psi)\dot{\psi} - r \cos(\psi)\dot{\theta} - r \cos(\psi)\dot{\alpha} = 0
$$

$$
\dot{y} - a \sin(\psi)\dot{\psi} - r \sin(\psi)\dot{\theta} - r \sin(\psi)\dot{\alpha} = 0
$$

$$
\dot{x} + a \cos(\psi)\dot{\psi} - r \cos(\psi)\dot{\theta} - r \cos(\psi)\dot{\beta} = 0
$$

$$
\dot{y} + a \sin(\psi)\dot{\psi} - r \sin(\psi)\dot{\theta} - r \sin(\psi)\dot{\beta} = 0.
$$

(3.36)

Let $q$ be the constrained generalized coordinates, defined by

$$
q = \begin{bmatrix}
x & y & \psi & \theta & \alpha & \beta
\end{bmatrix}^T
$$

(3.37)
Then the nonholonomic constraints in Equations (3.31) and (3.36) can be collectively be represented in the Pfaffian nonholonomic form as \( H\dot{q} = 0 \). The constraint matrix \( H \) is given by

\[
H = 
\begin{bmatrix}
-\sin(\psi) & \cos(\psi) & 0 & 0 & 0 & 0 \\
1 & 0 & -a \cos(\psi) & -r \cos(\psi) & -r \cos(\psi) & 0 \\
0 & 1 & -a \sin(\psi) & -r \sin(\psi) & -r \sin(\psi) & 0 \\
1 & 0 & a \cos(\psi) & -r \cos(\psi) & 0 & -r \cos(\psi) \\
0 & 1 & a \sin(\psi) & -r \sin(\psi) & 0 & -r \sin(\psi)
\end{bmatrix}
\tag{3.38}
\]

where \( H \) is a matrix from \( \mathbb{R}^6 \) into \( \mathbb{R}^5 \).

Note that the first row is linearly dependent of the second row and third row. This is shown by multiplying the second row by \(-\sin(\psi)\) and the third row by \(\cos(\psi)\) and adding them together. To be precise,

\[
(-\sin(\psi)) \begin{bmatrix} 1 & 0 & -a \cos(\psi) & -r \cos(\psi) & -r \cos(\psi) & 0 \end{bmatrix} + \\
(\cos(\psi)) \begin{bmatrix} 0 & 1 & -a \sin(\psi) & -r \sin(\psi) & -r \sin(\psi) & 0 \end{bmatrix}
= \begin{bmatrix} -\sin(\psi) & \cos(\psi) & 0 & 0 & 0 & 0 \end{bmatrix}.
\]

which yields the first row. Because \( \cos(\psi) \) and \( \sin(\psi) \) can never be simultaneously zero, the first row is a linear combination of the second and third row. Hence, the rank of \( H \) reduces from a maximum of five to four. The first row is also a linear combination of the fourth and fifth row, shown by

\[
(-\sin(\psi)) \begin{bmatrix} 1 & 0 & a \cos(\psi) & -r \cos(\psi) & 0 & -r \cos(\psi) \end{bmatrix} + \\
(\cos(\psi)) \begin{bmatrix} 0 & 1 & a \sin(\psi) & -r \sin(\psi) & 0 & -r \sin(\psi) \end{bmatrix}
= \begin{bmatrix} -\sin(\psi) & \cos(\psi) & 0 & 0 & 0 & 0 \end{bmatrix}.
\]

Once again, the rank decreases from a maximum of four to three. Due to the placement of the 1s and 0s of the first two columns, there is no other linear dependence relationships between the other rows. Therefore, the rank of \( H \) is shown to be three.
3.4 Deriving the Equations of Motion

With the constraints accounted for, the equations of motion are then derived. Applying Lagrangian’s equations, the resulting operation for each generalized coordinate is

\[
\frac{d}{dt} \frac{\partial L}{\partial \dot{x}} - \frac{\partial L}{\partial x} = (m_b + 2m_w)\ddot{x} - m_b h \sin(\psi) \sin(\theta) \ddot{\psi} + m_b h \cos(\theta) \cos(\psi) \ddot{\theta}
- m_b h \cos(\psi) \sin(\theta) \dot{\psi}^2 - 2m_b h \sin(\psi) \cos(\theta) \dot{\psi} \dot{\theta}
- m_b h \sin(\theta) \cos(\psi) \dot{\theta}^2
\]

\[
\frac{d}{dt} \frac{\partial L}{\partial \dot{y}} - \frac{\partial L}{\partial y} = (m_b + 2m_w)\ddot{y} + m_b h \cos(\psi) \sin(\theta) \ddot{\psi} + m_b h \sin(\psi) \sin(\theta) \ddot{\theta}
- m_b h \sin(\theta) \sin(\psi) \dot{\theta}^2 + 2m_b h \cos(\theta) \dot{\psi} \ddot{\theta}
- m_b h \sin(\psi) \sin(\theta) \dot{\psi}^2
\]

\[
\frac{d}{dt} \frac{\partial L}{\partial \dot{\psi}} - \frac{\partial L}{\partial \psi} = -m_b h \sin(\psi) \sin(\theta) \ddot{x} + m_b h \cos(\psi) \sin(\theta) \ddot{y}
+ \left( (J_x + m_b h^2) \sin^2(\theta) + J_z \cos^2(\theta) + 2m_w a^2 \right) \ddot{\psi}
+ 2(J_x - J_z + m_b h^2) \sin(\theta) \cos(\theta) \dot{\theta} \dot{\psi}
\]

\[
\frac{d}{dt} \frac{\partial L}{\partial \dot{\theta}} - \frac{\partial L}{\partial \theta} = m_b h \cos(\theta) \cos(\psi) \ddot{x} + m_b h \cos(\theta) \sin(\psi) \ddot{y}
+ \left( J_y + m_b h^2 + 2J_w \right) \ddot{\theta} + J_w \ddot{\alpha} + J_w \ddot{\beta}
- (J_x - J_z + m_b h^2) \sin(\theta) \cos(\theta) \dot{\psi}^2 - m_b g h \sin(\theta)
\]

\[
\frac{d}{dt} \frac{\partial L}{\partial \dot{\alpha}} - \frac{\partial L}{\partial \alpha} = J_w (\ddot{\theta} + \dot{\alpha})
\]

\[
\frac{d}{dt} \frac{\partial L}{\partial \dot{\beta}} - \frac{\partial L}{\partial \beta} = J_w (\ddot{\theta} + \dot{\beta}).
\]

(3.39)

The expressions can be organized into the following form

\[
\frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = M_q \ddot{q} - \beta_q
\]

(3.40)
where $M_b$ is the mass matrix and $\beta_q$ are the leftover term. The mass matrix is found as

$$M_q = \begin{bmatrix} M_{q1} & M_{q2} \end{bmatrix}$$

$$M_{q1} = \begin{bmatrix} m & 0 \\ 0 & m \\ -m_b h \sin(\psi) \sin(\theta) & m_b h \cos(\psi) \sin(\theta) \\ m_b h \cos(\psi) \cos(\theta) & m_b h \sin(\psi) \cos(\theta) \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$M_{q2} = \begin{bmatrix} -m_b h \sin(\psi) \sin(\theta) & m_b h \cos(\psi) \cos(\theta) & 0 & 0 \\ m_b h \cos(\psi) \sin(\theta) & m_b h \sin(\psi) \cos(\theta) & 0 & 0 \\ (J_x + m h^2) \sin(\theta)^2 + J_z \cos^2(\theta) + 2 m_w a^2 & 0 & 0 & 0 \\ 0 & J_y + m_b h^2 + 2 J_w & J_w & J_w \end{bmatrix}$$

and the leftover terms are

$$\beta_q = \begin{bmatrix} m_b h \cos(\psi) \sin(\theta) \dot{\psi} - 2 m_b h \sin(\psi) \cos(\theta) \dot{\psi} \dot{\theta} + m_b h \sin(\theta) \cos(\psi) \dot{\theta}^2 \\ m_b h \sin(\theta) \sin(\psi) \dot{\theta}^2 - 2 m_b h \cos(\theta) \cos(\psi) \dot{\psi} \dot{\theta} + m_b h \sin(\psi) \sin(\theta) \dot{\psi}^2 \\ -2(J_x - J_z + m_b h^2) \cos(\theta) \sin(\theta) \dot{\theta} \\ (J_x - J_z + m_b h^2) \dot{\psi}^2 \cos(\theta) \sin(\theta) + m_b g h \sin(\theta) \\ 0 \\ 0 \end{bmatrix}.$$
It is noted that $\beta_q$ can also be formatted as

$$
\beta_q = \begin{bmatrix}
\cos(\psi) \\
\sin(\psi) \\
0 \\
0 \\
0 \\
0 \\
0 \\
m_b h (\dot{\psi}^2 + \dot{\theta}^2) \sin(\theta) + 2 m_b h \dot{\psi} \dot{\theta} \cos(\theta)
\end{bmatrix}
$$

\begin{align}
\begin{bmatrix}
0 \\
0 \\
-2(J_x - J_z + m_b h^2) \cos(\theta) \sin(\theta) \dot{\psi} \dot{\theta} \\
(J_x - J_z + m_b h^2) \dot{\psi}^2 \cos(\theta) \sin(\theta) + m_b g h \sin(\theta)
\end{bmatrix}.
\end{align} \tag{3.43}

The inputs acting on the system are the torque on the left wheel and right wheel. They are denoted as $T_l$ and $T_r$ respectively. These torques are generated by motors attached to the Segway and are applied to the motor shaft. Therefore, the left and right motors apply a generalized force to $\alpha$ and $\beta$ respectively. The generalized forces are expressed in the form of $F_q \vec{u}$ where $\vec{u}$ is the inputs of the system, and $F_q$ is the input coefficient matrix. They are given as

$$
F_q = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^* 
$$

$$
\vec{u} = \begin{bmatrix} T_l \\ T_r \end{bmatrix}
$$

The equations of motion are given by

$$
M_q \ddot{q} = \beta_q + H^* \lambda + F_q \vec{u} \tag{3.44}
$$

where $\lambda$ is the vector of the Lagrange multipliers corresponding to the forces $H^* \lambda$ arising from the Pfaffian constraint $H \dot{q} = 0$. The Lagrange multipliers are eliminated
by finding a matrix $\Phi$ whose range equals the nullspace of $H$. Let Equation (3.44) be known as the constrained equations of motion.

Consider the one to one matrix $\Phi$ given by

$$
\Phi = \begin{bmatrix}
\cos(\psi) & 0 & 0 \\
\sin(\psi) & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1 \\
\frac{1}{r} & -\frac{a}{r} & -1 \\
\frac{1}{r} & \frac{a}{r} & -1
\end{bmatrix}.
$$

(3.45)

It can be verified that $H\Phi = 0$. Because the rank of $H$ equals three, it follows that the range of $\Phi$ equals the nullspace of $H$. Since the solution $q$ satisfies $H\dot{q} = 0$, the solution can be expressed as a linear combination of the column vectors of $\Phi$. That is, there is some set of generalized coordinates $p$ such that

$$
\dot{q} = \Phi \dot{p}.
$$

(3.46)

Note that the first two rows of $\Phi$ in Equation (3.45) is similar to Equation (3.32), suggesting that the first variable in $\dot{p}$ is the velocity $v$. The third and fourth rows of $\Phi$ returns the second and third elements of $\dot{p}$. They are also set equal to the third and fourth element of $\dot{q}$, which are $\dot{\psi}$ and $\dot{\theta}$ respectively. Therefore

$$
\begin{bmatrix}
\dot{x} \\
\dot{y} \\
\dot{\psi} \\
\dot{\theta} \\
\dot{\alpha} \\
\dot{\beta}
\end{bmatrix} = \begin{bmatrix}
\cos(\psi) & 0 & 0 \\
\sin(\psi) & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1 \\
\frac{1}{r} & -\frac{a}{r} & -1 \\
\frac{1}{r} & \frac{a}{r} & -1
\end{bmatrix} \begin{bmatrix}
v \\
\dot{\psi} \\
\dot{\theta}
\end{bmatrix}.
$$

(3.47)

As a result, the derivative of the unconstrained generalized coordinates are

$$
\dot{p} = \begin{bmatrix}
v \\
\dot{\psi} \\
\dot{\theta}
\end{bmatrix}.
$$

(3.48)
Without loss of generality, let

\[
p = \begin{bmatrix} l \\ \psi \\ \theta \end{bmatrix}
\]

where

\[ l = \int_0^t v \, d\tau \quad \text{and} \quad \psi = \int_0^t \dot{\psi} \, d\tau. \]

The state \( l \) can be thought of as the arc length traveled by the Segway and \( \psi \) is the heading angle. Both \( l \) and \( \psi \) are defined to be zero at a time of zero. In other words, \( l_0 = 0 \) and \( \psi_0 = 0 \).

Because \( H \Phi = 0 \), taking the transpose results in \( \Phi^* H^* = 0 \) as well. Multiplying the constrained equations of motion in Equation (3.44) by \( \Phi^* \) yields

\[
\Phi^* M_q \ddot{q} = \Phi^* \beta_q + \Phi^* F_q \ddot{u}.
\]

(3.49)

Taking the derivative of Equation (3.46) through the product rule results in

\[
\dot{q} = \dot{\Phi} \dot{p} + \Phi \dddot{p}.
\]

When substituted in, Equation (3.49) becomes

\[
\Phi^* M_q \left( \dot{\Phi} \dot{p} + \Phi \dddot{p} \right) = \Phi^* \beta_q + \Phi^* F_q \ddot{u}.
\]

(3.50)

Equation (3.50) can be represented in the form

\[
M_p \ddot{p} = \beta_p + F_p \ddot{u}.
\]

(3.51)

The matrices \( M_p, \beta_p, \) and \( F_p \) corresponding to the generalized coordinates \( p \) are given by

\[
M_p = \Phi^* M_q \Phi \\
\beta_p = \Phi^* \left( \beta_q - M_q \dot{\Phi} \dot{p} \right) \\
F_p = \Phi^* F_q.
\]

(3.52)
Carrying out the expressions in Equation (3.52), the unconstrained equations of motion in Equation (3.51) become

\[
\begin{bmatrix}
  m + \frac{2J_w}{r^2} & 0 & m_b h \cos(\theta) \\
  0 & (J_x + m_b h^2) \sin^2(\theta) + J_z \cos^2(\theta) + \frac{2J_a a^2}{r^2} + 2m_w a^2 & 0 \\
  m_b h \cos(\theta) & 0 & J_y + m_b h^2
\end{bmatrix}
\begin{bmatrix}
  \dot{v} \\
  \dot{\psi} \\
  \dot{\theta}
\end{bmatrix}
= \begin{bmatrix}
  m_b h \left(\dot{\psi}^2 + \dot{\theta}^2\right) \sin(\theta) \\
  -2(J_x - J_z + m h^2) \cos(\theta) \sin(\theta)\dot{\psi}\dot{\theta} - m_b h \sin(\theta)v\dot{\psi} \\
  (J_x - J_z + m_b h^2) \cos(\theta) \sin(\theta)\dot{\psi}^2 + m_b g h \sin(\theta)
\end{bmatrix}
\begin{bmatrix}
  \frac{1}{r} & \frac{1}{r} \\
  -\frac{a}{r} & \frac{a}{r} \\
  -1 & -1
\end{bmatrix}
\begin{bmatrix}
  T_l \\
  T_r
\end{bmatrix}.
\] (3.53)

The matrices in Equation (3.53) are linearized about the zero equilibrium point. The linearized equations of motion are

\[
\begin{bmatrix}
  m + \frac{2J_w}{r^2} & 0 & m_b h \\
  0 & J_z + \frac{2J_a a^2}{r^2} + 2m_w a^2 & 0 \\
  m_b h & 0 & J_y + m_b h^2
\end{bmatrix}
\begin{bmatrix}
  \dot{v} \\
  \dot{\psi} \\
  \dot{\theta}
\end{bmatrix}
= \begin{bmatrix}
  \frac{1}{r} & \frac{1}{r} \\
  -\frac{a}{r} & \frac{a}{r} \\
  -1 & -1
\end{bmatrix}
\begin{bmatrix}
  T_l \\
  T_r
\end{bmatrix}.
\] (3.54)

As seen, Equation (3.54) can be decoupled into two systems. The first system is about \( v \) and \( \dot{\theta} \) where

\[
\begin{bmatrix}
  m + \frac{2J_w}{r^2} & m_b h \\
  m_b h & J_y + m_b h^2
\end{bmatrix}
\begin{bmatrix}
  \dot{v} \\
  \dot{\theta}
\end{bmatrix}
= \begin{bmatrix}
  0 \\
  m_b g h \theta
\end{bmatrix}
+ \begin{bmatrix}
  \frac{1}{r} & \frac{1}{r} \\
  -\frac{a}{r} & \frac{a}{r} \\
  -1 & -1
\end{bmatrix}
\begin{bmatrix}
  T_l \\
  T_r
\end{bmatrix}.
\] (3.55)

Note the linearized system in Equation (3.55) is about the sum of the torques. The remaining system is about \( \dot{\psi} \) and is given by

\[
(J_z + \frac{2J_a a^2}{r^2} + 2m_w a^2) \ddot{\psi} = \begin{bmatrix}
-\frac{a}{r} & \frac{a}{r}
\end{bmatrix}
\begin{bmatrix}
  T_l \\
  T_r
\end{bmatrix}.
\] (3.56)
The linearized system in Equation (3.56) instead is about the difference of the torques. The systems in Equations (3.55) and (3.56) can be placed into the form of

\[
\begin{bmatrix}
\dot{v} \\
\ddot{\theta}
\end{bmatrix} = \beta_s + \mathbf{F}_s \ddot{u}
\]

\[
M_d \ddot{\psi} = \beta_d + \mathbf{F}_d \ddot{u}.
\]

The decoupling of the systems into two separate systems simplifies the system as a whole. As a result, the two sets of states can be treated independent of one another.

Note that if the Segway was constrained such that \( \psi = 0 \) then the Segway reduces to an inverted pendulum in one direction. Equation (3.53) would then reduce to

\[
\begin{bmatrix}
\frac{2 J_w}{r^2} & m_b h \cos(\theta) \\
mb h \cos(\theta) & J_y + m_b h^2
\end{bmatrix}
\begin{bmatrix}
\dot{v} \\
\dot{\theta}
\end{bmatrix} = \begin{bmatrix}
m_b h \dot{\theta} \sin(\theta) \\
mb g h \sin(\theta)
\end{bmatrix} + \begin{bmatrix}
\frac{1}{r} \\
-1
\end{bmatrix} T
\]

where \( T_l = T_r = \frac{1}{2} T \). Note that the second row of Equation (3.53) forces the left and right torques to be equal. Equation (3.57) is equivalent to the equations of motion that would be derived directly for an inverted pendulum on a cart moving in one direction (assuming the cart itself is massless, but not the wheels themselves).

### 3.5 Voltage with Back EMF

The torques on the left and right wheel only arise when a voltage is sent to the motors. In a more realistic sense, the motors generate a torque that is counteracted by a back electromagnetic force. The back electromagnetic force is proportional to \( \dot{\alpha} \) and \( \dot{\beta} \) for the left and right motors respectively. The expressions for the left and right torques are thus given as

\[
T_l = k_v V_l - k_v \dot{\alpha}
\]
\[
T_r = k_v V_r - k_v \dot{\beta}
\]

(3.58)
where $k_e$ and $k_v$ are the voltage and back EMF coefficients respectively and $V_l$ and $V_r$ is the voltage supplied to the left and right motors respectively. Applying the relationship from Equation (3.47), the expressions for $\dot{\alpha}$ and $\dot{\beta}$ are

\[
\dot{\alpha} = \frac{1}{r} v - \frac{a}{r} \dot{\psi} - \dot{\theta}
\]
\[
\dot{\beta} = \frac{1}{r} v + \frac{a}{r} \dot{\psi} - \dot{\theta}.
\]

Equation (3.58) become

\[
T_l = k_e V_l - k_v \left( \frac{1}{r} v - \frac{a}{r} \dot{\psi} - \dot{\theta} \right)
\]
\[
T_r = k_e V_r - k_v \left( \frac{1}{r} v + \frac{a}{r} \dot{\psi} - \dot{\theta} \right).
\]

Equation (3.59) can be substituted into Equations (3.55) and (3.56) to yield

\[
\begin{bmatrix}
  m + \frac{2 J_w}{r^2} & m b h & m b h \\
  m b h & J_y + m b h^2
\end{bmatrix}
\begin{bmatrix}
  \ddot{v} \\
  \ddot{\theta}
\end{bmatrix}
= \begin{bmatrix}
  -2 k_v \left( \frac{v}{r^2} - \frac{\dot{\theta}}{r} \right) \\
  m_v g h \theta + 2 k_v \left( \frac{v}{r} - \dot{\theta} \right)
\end{bmatrix}
+ k_e \begin{bmatrix}
  \frac{1}{r} & \frac{1}{r} \\
  -1 & -1
\end{bmatrix}
\begin{bmatrix}
  V_l \\
  V_l
\end{bmatrix}
\]

(3.60)

\[
(J_z + \frac{2 J_w a^2}{r^2} + 2 m_w a^2) \ddot{\psi} = -2 k_v \frac{a^2}{r^2} \dot{\psi} + k_e \begin{bmatrix}
  -\frac{a}{r} \\
  \frac{a}{r}
\end{bmatrix}
\begin{bmatrix}
  V_l \\
  V_l
\end{bmatrix}
\]

where the back EMF terms are moved to the $\beta_p$ vector for both systems. Due to the input coefficient matrix, the $\dot{\psi}$ term does not appear in the first system. Likewise, $v$ and $\dot{\theta}$ do not appear in the second. As a result, the systems still remain decoupled.

### 3.6 Transformation of Voltage Inputs

As stated before, the time derivatives of the $v$, $\theta$, and $\dot{\theta}$ are linearly dependent only on those three states and the sum of the voltages, $V_l$ and $V_r$. Similarly, the time derivative of the $\dot{\psi}$ is linearly dependent only on that state and the difference of $V_l$ and $V_r$. If the sum and difference of the voltages are denoted as $V_s$ and $V_d$, then

\[
V_s = V_l + V_r
\]
\[
V_d = V_l - V_r
\]
or
\[
\begin{bmatrix}
V_s \\
V_d
\end{bmatrix} =
\begin{bmatrix}
1 & 1 \\
1 & -1
\end{bmatrix}
\begin{bmatrix}
V_l \\
V_r
\end{bmatrix}.
\] (3.61)

By taking the inverse of the square matrix in Equation (3.61), we obtain
\[
\begin{bmatrix}
V_l \\
V_r
\end{bmatrix} =
\begin{bmatrix}
\frac{1}{2} & \frac{1}{2} \\
\frac{1}{2} & -\frac{1}{2}
\end{bmatrix}
\begin{bmatrix}
V_s \\
V_d
\end{bmatrix}.
\] (3.62)

If the transformation of Equation (3.62) is applied such that the inputs \( \vec{u} \) is now
\[
\vec{u} =
\begin{bmatrix}
V_s \\
V_d
\end{bmatrix}
\]
then the new input coefficient matrix for both systems becomes
\[
F_s =
\begin{bmatrix}
\frac{1}{r} & \frac{1}{r} \\
-1 & -1
\end{bmatrix}
\begin{bmatrix}
\frac{1}{2} & \frac{1}{2} \\
\frac{1}{2} & -\frac{1}{2}
\end{bmatrix}
\begin{bmatrix}
\frac{1}{r} & 0 \\
-1 & 0
\end{bmatrix}
\]
\[
F_d = k_e \begin{bmatrix}
-\frac{a}{r} \\
\frac{a}{r}
\end{bmatrix}
\begin{bmatrix}
\frac{1}{2} & \frac{1}{2} \\
\frac{1}{2} & -\frac{1}{2}
\end{bmatrix}
\begin{bmatrix}
0 & -\frac{a}{r}
\end{bmatrix}.
\] (3.63)

As expected, the first system is dependent only on \( V_s \) and the second system is dependent only on \( V_d \). Therefore, the linearized equations of motion simplify to
\[
\begin{bmatrix}
m + \frac{2J_w}{r^2} & m_b h \\
m_b h & J_y + m_b h^2
\end{bmatrix}
\begin{bmatrix}
\dot{v} \\
\dot{\theta}
\end{bmatrix}
= \begin{bmatrix}
-2k_v \left( \frac{\dot{v}}{r^2} - \frac{\dot{\theta}}{r} \right) \\
m_b h \dot{\theta} + 2k_v \left( \frac{\dot{v}}{r} - \dot{\theta} \right)
\end{bmatrix} + k_e \begin{bmatrix}
\frac{1}{r} \\
-1
\end{bmatrix}
V_s
\]
\[
\begin{bmatrix}
J_z + \frac{2J_w a^2}{r^2} + 2m_a a^2
\end{bmatrix}
\ddot{\psi} = -2k_v \frac{a^2}{r^2} \dot{\psi} - k_e \frac{a}{r} V_d.
\] (3.64)

If the linearized equations of motion are multiplied by the inverse of the mass matrix, then Equation (3.64) becomes
\[
\begin{bmatrix}
\dot{v} \\
\dot{\theta}
\end{bmatrix}
= \frac{2k_v}{r} \begin{bmatrix}
- (J_y + m_b h (h + r)) r \\
((m_b h + m r) r + 2J_w)
\end{bmatrix}
\begin{bmatrix}
\dot{v} - r \dot{\theta} \\
\dot{\theta}
\end{bmatrix}
+ \frac{k_e}{\zeta}
\begin{bmatrix}
r(J_y + m_b h (h + r)) \\
-(m_b h + m r) r + 2J_w
\end{bmatrix}
V_s
\]
\[
\dot{\psi} = -\frac{2k_v a^2}{\eta} \dot{\psi} - \frac{k_e a r}{\eta} V_d
\] (3.65)
where
\[ \zeta = 2J_w(J_y + m_b h^2) + (2m_b m_w h^2 + m J_y) r^2 \]
\[ \eta = J_x r^2 + 2a^2 (J_w + m_w r^2). \]

We then place the linearized systems in Equation (3.65) into state space representations. Let the first set of states include the velocity, pitch angle, and the pitch angle rate. These states are depicted as
\[
\vec{x}_s = \begin{bmatrix} v \\ \theta \\ \dot{\theta} \end{bmatrix}
\] (3.66)
and are related to the sum of the torques. They will be referred to as the sum states. The second set of states include only the heading angle, or
\[
\vec{x}_d = \dot{\psi}.
\] (3.67)

The second set relies only on the difference of torques. It will be referred to as the difference state. As a result, the two separate state spaces are represented as
\[
\begin{align*}
\dot{\vec{x}}_s &= A_s \vec{x}_s + B_s V_s \\
\dot{\vec{x}}_d &= A_d \vec{x}_d + B_d V_d.
\end{align*}
\] (3.68)
The state space coefficient matrices are

\[
A_s = \begin{bmatrix}
-\frac{2k_v(J_y + m_b h (h + r))}{\zeta} & -\frac{m_b^2 g h^2 r^2}{\zeta} & \frac{2k_v r (J_y + m_b h (h + r))}{\zeta} \\
0 & 0 & \frac{2k_v ((m_b h + m r) r + 2J_w)}{\zeta} \\
\frac{2k_v ((m_b h + m r) r + 2J_w)}{\zeta} & \frac{m_b g h (2J_w + m r)}{\zeta} & 1
\end{bmatrix}
\]

\[
B_s = \frac{k_e}{\zeta} \begin{bmatrix}
  r (J_y + m_b h (h + r)) \\
  0 \\
  -\left( (m_b h + m r) r + 2J_w \right)
\end{bmatrix}
\]

\[
A_d = -\frac{2k_v a^2}{\eta}
\]

\[
B_d = -\frac{k_e a r}{\eta}
\]

\[
\zeta = 2J_w (J_y + m_b h^2) + (2m_b m_w h^2 + m J_y) r^2
\]

\[
\eta = J_z r^2 + 2a^2 (J_w + m_w r^2).
\]

(3.69)

It is noted that the first column of \(A_s\) is in the range of \(B_s\). The same is automatically true for \(A_d\) and \(B_d\), as they are scalar. This observation will be important in the next section.

Because the states are decoupled, the controllability of the states can be found more easily. Because the difference state space is scalar, the system is automatically controllable, so long as \(B_d\) is nonzero. To determine the controllability of the sum states system, the rank of the controllability matrix, \(R\), is determined. By definition, the controllability matrix for the pair \(A_s\) and \(B_s\) is given by

\[
R_s = \begin{bmatrix}
  B_s & A_s B_s & A^2_s B_s
\end{bmatrix}
\]

as the size of the the system is three. Because of the single input for each set, the controllability matrix has full rank if and only if the determinant of the matrix is not zero. The determinant of \(R_s\) is determined to be

\[
\det(R_s) = \frac{k_e^3 m_b g h r ((m_b h + m r) r + 2J_w)^2}{(2J_w (J_y + m_b h^2) + (2m_b m_w h^2 + m J_y) r^2)^3}.
\]

Because all of the parameters are strictly positive, the determinant of \(R_s\) is nonzero. Therefore, both systems are controllable.
3.7 Reference Commands Applied to Velocity and Heading Angle Rate

Suppose that there exist a single input controllable system in the form of

\[ \dot{x} = Ax + Bu \]  \hspace{1cm} (3.70)

with the property in which the first column of \( A \) is in the range of \( B \). It is desired for the system to be driven to some reference command about the first element. If the reference command is denoted as \( r_1 \), then the reference vector for the set of states is given as

\[ r = \Pi r_1 \]

where

\[ \Pi = \begin{bmatrix} 1 & 0 & \ldots & 0 \end{bmatrix}^* \]

such that \( \Pi \) has the length equal to the number of states.

Let us apply an integrated error feedback controller for the system. In general, the error is defined as

\[ \vec{e} = r - x \]

such that state space representation in Equation (3.70) is equivalent to

\[ \dot{\vec{e}} = A(\vec{e} - r) - Bu. \hspace{1cm} (3.71) \]

If a feedback controller is designed only about the error, such that \( u = K\vec{e} \), then there will be steady state error if \( A\vec{r} \neq 0 \). Therefore, a new variable \( e_0 \) is defined such that its derivative is equal to the first entry of the error vector. This is represented as

\[ \dot{e}_0 = e_1. \hspace{1cm} (3.72) \]

Doing so expands Equation (3.71) into

\[
\begin{bmatrix}
\dot{e}_0 \\
\dot{e}
\end{bmatrix} =
\begin{bmatrix}
0 & \Pi^* \\
0 & A
\end{bmatrix}
\begin{bmatrix}
e_0 \\
e
\end{bmatrix} -
\begin{bmatrix}
0 \\
A\Pi r_1
\end{bmatrix} -
\begin{bmatrix}
0 \\
B
\end{bmatrix} u.
\hspace{1cm} (3.73)
\]
To prove that the system in Equation (3.73) is controllable, let us apply the PBH test. The test states that a pair \((A, B)\) is controllable if and only if \([\lambda I - A \ B]\) has full rank for all \(\lambda\). If the test is applied to the system in Equation (3.73), then it must be shown that the matrix

\[
M_{PBH} = \begin{bmatrix}
\lambda & -\Pi^* & 0 \\
0 & \lambda I - A & B
\end{bmatrix}
\]

has full rank. It is given that the pair \((A, B)\) is controllable. If \(\lambda \neq 0\), then it is possible to multiply the first column by \(-1/\lambda\) and add it to the second column to result in the new matrix

\[
N_{PBH} = \begin{bmatrix}
\lambda & 0 & 0 \\
0 & \lambda I - A & B
\end{bmatrix}.
\]

The matrix \(N_{PBH}\) therefore has full rank as the first column is linearly independent of the remaining rows.

For the case of when \(\lambda = 0\), then Equation (3.74) becomes

\[
M_{PBH} = \begin{bmatrix}
0 & -\Pi^* & 0 \\
0 & -A & B
\end{bmatrix}.
\]

Because the first column of \(A\) is in the range of \(B\), then no other columns of \(A\) can be in the range of \(B\). As such, all columns past the second columns are linearly independent. In addition, the placeholder of the single 1 in \(\Pi^*\) prevents any column operations of \(M_{PBH}\) of Equation (3.75) to result in the zero column. In accordance, \(M_{PBH}\) has full rank. By the PBH test, the system in (3.73) is controllable.

Therefore, a feedback controller is designed to include the new variable, such that

\[
u = K_0 e_0 + K \bar{e}.
\]

By applying the feedback controller, Equation (3.73) becomes

\[
\begin{bmatrix}
\dot{e}_0 \\
\dot{\bar{e}}
\end{bmatrix} = \begin{bmatrix}
0 & \Pi^* \\
-BK_0 & A - BK
\end{bmatrix} \begin{bmatrix}
e_0 \\
\bar{e}
\end{bmatrix} = \begin{bmatrix}
0 \\
A \Pi r_1
\end{bmatrix}.
\]
Let $K$ and $K_0$ be chosen such that the above system is stable. Then $\dot{e}_0$ and $\dot{\vec{e}}$ converges to zero and Equation (3.77) converges to a steady state. As a result

$$0 = \begin{bmatrix} \begin{bmatrix} 0 & \Pi^* \\ -BK_0 & A - BK \end{bmatrix} & \begin{bmatrix} e_0(\infty) \\ \vec{e}(\infty) \end{bmatrix} \end{bmatrix} - \begin{bmatrix} 0 \\ A\Pi r_1 \end{bmatrix}. \quad (3.78)$$

Because the system is controllable, there exist a unique solution to Equation (3.78). To find the solution, let us first consider the first row in Equation (3.78), which is equivalent to

$$\Pi^*\vec{e}(\infty) = e_1 = 0.$$ 

As desired, the first state converges to the reference command.

Now consider the second row of equations in Equation (3.78)

$$-BK_0e_0(\infty) + (A - BK)\vec{e}(\infty) = A\Pi r_1. \quad (3.79)$$

Recall the first column of $A$ is in the range of $B$. In other words, there exist a constant $\gamma$ such that

$$A\Pi = B\gamma. \quad (3.80)$$

Then Equation (3.79) can be rewritten as

$$(A - BK)\vec{e}(\infty) = B(r_1\gamma + K_0e_0(\infty)). \quad (3.81)$$

From Equation (3.81), it can be seen that

$$e_0(\infty) = -\frac{\gamma}{K_0}r_1$$

$$\vec{e} = 0$$

is a possible solution. Because a unique solution must exist, then Equation (3.82) is the unique solution to Equation (3.78) if $K_0 \neq 0$. As seen, $e_0(\infty) = 0$ only if $r = 0$ or $\gamma = 0$. The latter can only be true if the the first column of $A$ is zero.

Now let us refer back to the Segway. From Equation (3.69) it is seen that the first column of $A_s$ is in the range of $B_s$. The same is automatically true for $A_d$ and $B_d$ as
they are both scalar. Therefore, the results from the general case can be applied to the Segway.

In accordance, the feedback controller from Equation (3.76) is expressed as

\[ V_s = K_{s,0}e_{s,0} + K_se_s \]
\[ V_d = K_{d,0}e_{d,0} + K_de_d. \]

Allow the feedback gain matrices to be denoted as

\[ K_{s,0} = K_l \]
\[ K_s = \begin{bmatrix} K_v & K_\theta & K_\theta \end{bmatrix} \]
\[ K_{d,0} = K_\psi \]
\[ K_d = K_{\dot{\psi}}. \] (3.83)

The \( \gamma \) from Equation (3.80) is determined for both sets of states as

\[ \gamma_s = -\frac{2k_v}{K_ik_er} \]
\[ \gamma_d = \frac{2ak_v}{K_\psi k_er}. \]

As a result, Equation (3.82) yields the steady state errors \( e_0(\infty) \) as

\[ e_{s,0}(\infty) = \frac{2k_v}{K_ik_er}v_r \]
\[ e_{d,0}(\infty) = -\frac{2ak_v}{K_\psi k_er}\dot{\psi}_r. \] (3.84)

Suppose, we are interested in the behavior of \( l \) and \( \psi \). If the reference commands are constant, then

\[ e_{s,0} = \int_0^t (v_r - v) \, d\tau = v_rt - l \]
\[ e_{d,0} = \int_0^t (\dot{\psi}_r - \dot{\psi}) \, d\tau = \dot{\psi}_rt - \psi. \] (3.85)

Notice that the steady state errors in Equation (3.84) are also zero if \( k_v = 0 \). In other words, there would be no steady state error if there is no back EMF. However
this is unrealistic, and therefore there will be steady state error for nonzero reference commands. From Equation (3.85) it is seen that for constant reference command that

\[ l \approx v_r \left( t - \frac{2k_v}{K_i k_e r} \right) \]
\[ \psi \approx \dot{\psi}_r \left( t + \frac{2ak_v}{K_k k_e r} \right). \] (3.86)

The symbol \( \approx \) denotes that the relationship is asymptotic. As time increases, then the steady state errors become less significant.

To determine the path of the Segway given constant reference commands, recall from the constraint equations that

\[ \dot{x}(t) = v(t) \cos(\psi(t)) \]
\[ \dot{y}(t) = v(t) \sin(\psi(t)). \]

If \( v \) and \( \dot{\psi} \) converges to \( v_r \) and \( \dot{\psi}_r \), then the above equations become

\[ \dot{x}(t) = v_r \cos(\dot{\psi}_r t + \psi_c) \]
\[ \dot{y}(t) = v_r \sin(\dot{\psi}_r t + \psi_c) \] (3.87)

for some phase shift \( \psi_c \). As seen from Equation (3.87), the Segway is expected to follow a circular path such that the radius \( r_c \) and angular velocity \( \omega_c \) is

\[ r_c = \frac{v_r}{\dot{\psi}_r} \]
\[ \omega_c = \dot{\psi}_r. \] (3.88)

### 3.8 Inclusion of Measurement and Noise

Until now, the Segway was stabilized and controlled with full knowledge of the states and with no noise, as is the ideal case. In a more realistic sense, sensors would be required to perform measurements. These measurements are then fed into a Kalman filter to estimate the states. The actual Segway itself uses an extended Kalman filter to estimate the states for any general nonlinear system. For the simulations however,
a steady state linear Kalman filter is used instead. The linear Kalman filter is more limited, but serves the purpose of estimating the states in the simulations.

Although there are many sensors that can be used on the Segway, let us focus on two most vital sensors that contribute most to the measurements. They are the three axis gyroscopes and the position encoders on the motors. The gyroscopes are able to measure the angular rates in terms of the body frame, or \( \dot{\theta} \) and \( \dot{\psi} \). The roll rate, \( \dot{\phi} \) can also be measured, but is not necessary as the Segway is not allowed to move about the main body \( x \) axis. The position encoders on each motor measure the angles \( \alpha \) and \( \beta \) for the left and right wheel, allowing their rates to be estimated. To relate the velocity to the angular rates, first take the bottom three rows of the matrix in Equation (3.47), such that

\[
\begin{bmatrix}
\dot{\theta} \\
\dot{\alpha} \\
\dot{\beta}
\end{bmatrix} =
\begin{bmatrix}
0 & 0 & 1 \\
\frac{1}{r} & -\frac{a}{r} & -1 \\
\frac{1}{r} & \frac{a}{r} & -1
\end{bmatrix}
\begin{bmatrix}
v \\
\dot{\psi} \\
\dot{\theta}
\end{bmatrix}
\]

which can be inverted to result in

\[
\begin{bmatrix}
v \\
\dot{\psi} \\
\dot{\theta}
\end{bmatrix} =
\begin{bmatrix}
r & \frac{r}{2} & \frac{r}{2} \\
0 & -\frac{r}{2a} & \frac{r}{2a} \\
1 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
\dot{\theta} \\
\dot{\alpha} \\
\dot{\beta}
\end{bmatrix}.
\]

Therefore, the velocity is given as

\[
v = r \left( \dot{\theta} + \frac{\dot{\alpha}}{2} + \frac{\dot{\beta}}{2} \right)
\]

Because \( \dot{\theta} \) is expected to average around zero as well as generate a significant amount of measurement noise, \( \dot{\theta} \) is neglected, such that the velocity is estimated as

\[
v \approx \frac{r}{2} \left( \dot{\alpha} + \dot{\beta} \right).
\]

As a result, Equation (3.89) is used by the Segway to estimate the velocity in real time.

There are other sensors that can be used in conjunction with the gyroscopes and position encoders to improve the estimate, but do not contribute as much.
For the simulation, measurement noise will be added to $v$, $\dot{\theta}$, and $\dot{\psi}$ to imitate measurements taken by the Segway. The measurement noise is white Gaussian noise with zero mean and the given variances, $\sigma_v^2$ and $\sigma_\omega^2$ for the position encoder and gyroscope respectively. The angular rate $\omega$ applies to both $\dot{\theta}$ and $\dot{\psi}$. A band-limited white noise is applied with sampling time rate $t_s$, such that the Power Spectrum Density (PSD) for the white noise is

$$PSD = \sigma^2 t_s.$$ (3.90)

For general linear systems, the output is given in the form of

$$\vec{y} = C\vec{x} + D\vec{w}$$ (3.91)

where $\vec{w}$ is a Gaussian white noise vector such that the mean is zero and the variance is the identity. For the simulations, the three outputs are given as

$$y_1 = v + \sigma_v^2 w_1$$
$$y_2 = \dot{\theta} + \sigma_\omega^2 w_2$$
$$y_3 = \dot{\psi} + \sigma_\omega^2 w_3.$$ 

Therefore the corresponding coefficient matrices for the output vector in Equation (3.91) are given as

$$C_s = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$D_s = \begin{bmatrix} \sigma_v^2 \\ 0 \\ 0 \end{bmatrix}$$

$$C_d = \begin{bmatrix} 1 \end{bmatrix}$$

$$D_d = \begin{bmatrix} \sigma_\omega^2 \end{bmatrix}$$ (3.92)

If the state noise is ignored, then a steady state Kalman filter with a feedback controller is applied such that the state estimator, $\hat{x}$, is determined by

$$\dot{\hat{x}} = A\hat{x} + B\vec{u} + L(\vec{y} - C\hat{x})$$ (3.93)
where

\[ L = PC^*(DD^*)^{-1}. \tag{3.94} \]

The matrix \( P \) is found by solving for \( Q \) in the following steady state continuous algebraic Riccati equation

\[ \dot{Q} = AQ + QA^* - QC^*(DD^*)^{-1}CQ \tag{3.95} \]

and taking the limit of \( Q \) as time goes to infinity, such that

\[ P = \lim_{t \to \infty} Q(t). \]

For the Segway, the \( A \) matrices in Equation (3.95) are taken from the state space representations in Equation (3.69). The estimated states \( \hat{l} \) and \( \hat{\psi} \) are found by integrating \( \hat{v} \) and \( \hat{\dot{\psi}} \) with respect to time. The omission of the state noise leads to the \( P \) matrix corresponding to \( \dot{\psi} \) to result in zero. This is due to the damping back EMF term in \( A_d \) in Equation (3.69), which causes \( A_d \) to be negative. Because \( A_d \) is scalar, Equation (3.95) becomes

\[ \dot{Q}_d = 2A_dQ_d - \frac{C_d^2}{D_d^2}Q_d^2. \tag{3.96} \]

If \( \dot{Q}_d \) is set to zero to find the steady state values of \( Q_d \), then the two possible solutions for \( P_d \) are

\[ P_d = 0 \quad \text{and} \quad P_d = \frac{2A_dD_d^2}{C_d^2}. \tag{3.97} \]

However, because \( Q_d \) cannot be negative and \( A_d \) is known to be negative, \( \dot{Q}_d \) cannot be positive as a result of Equation (3.96). Therefore, the solution is driven to the first solution in Equation (3.97) where \( P_d = 0 \) instead of the second solution (which is negative). If \( A_d \) was positive, then \( P_d \) would be driven to the second solution (which would be positive). The consequences of \( P_d \) being zero results in Equation (3.93) for the heading angle rate reducing to

\[ \ddot{\psi} = A_d\dot{\psi} + B_dV_d \tag{3.98} \]
neglecting any measurements of $\dot{\psi}$ taken by the gyroscope. This is equivalent to the state space representation in Equation (3.68) and results in

$$\hat{\dot{\psi}} = \dot{\psi}.$$ 

This is not the case for the other set of states, $\vec{x}_s$. Because $\vec{x}_s$ is more significant to the stability of the Segway than $\dot{\psi}$, the noise in these states are of greater concern. However, this is an example of the limits of a linear steady state Kalman filter.

The feedback controller is now designed about the estimated states, such that

$$V_s = K_l e_{s,0} + K_s \vec{e}_s$$
$$V_d = K_s e_{d,0} + K_d \vec{e}_d$$

where $e_0$ and $\vec{e}$ are now defined as

$$e_{s,0} = \int_0^t (v_r - \dot{v}) \, d\tau$$
$$\vec{e}_s = \begin{bmatrix} v_r - \dot{v} \\ -\dot{\theta} \\ -\ddot{\theta} \end{bmatrix}$$
$$e_{d,0} = \int_0^t (\dot{\psi}_r - \dot{\psi}) \, d\tau$$
$$\vec{e}_d = \psi_r - \dot{\psi}$$

and the same feedback matrix is applied from Equation (3.83).

With the inclusion of the linear steady state Kalman filter, the simulations can provide a more realistic, but limited prediction of the Segway.

### 3.9 Parameters of the Built Segway

With the theory of the Segway established, the Segway can be built and compared to simulations. The Segway itself was built from components that can be bought online for a reasonable price. Several versions were required until a satisfactory design was made. The motor components included the Pololu 12V brushed DC motors with a
18.75:1 gearbox ratio connected to BaneBot wheels. The Orion Robotics RoboClaw 2 x 15A Motor Controller was used to control both motors. The entire Segway is powered by a single Turnigy 2.2A 3 cell 11.1V Lipo battery.

The main controller is the PX4 Pixhawk designed by the Computer Vision and Geometry Lab, Autonomous Systems Lab, and Automatic Control Laboratory of ETH Zurich (Swiss Federal Institute of Technology), along with many individual contributors. James Goppert of Purdue University is one of the individuals who assisted in the coding of the 3DR Pixhawk during development, and was responsible for modifying the code to be compatible with the Segway.

![Fig. 3.3. Latest version of the built Segway.](image)

Pictured above is the latest version of the built Segway used. Several designs was used, including the use of other motors. However, it was found that the previous
motors were too slow to respond. As suggested by James Goppert, one solution was to increase the $J_y$ moment of inertia, such that the Segway falls at a slower rate and gives the motors more time to respond. The moment of inertia can be increased by designing the Segway to be much taller. As a drawback, more torque is required for stability and thus more powerful motors were required.

From Figure 3.3, the motors are attached to each wheel at the Segway’s base. Not pictured is the battery, which is attached to the base through velcro. The motor controller is on the next level up, and is connected to both the motors and the the PX4 controller, which rest on the next level up. The PX4 controller is located at the center of mass to allow the sensors to be more effective at measuring. The next level up is a platform that makes the Segway more rigid, but also can carry a payload if necessary. On the top level are 4 door springs, to help protect the Segway if it falls over.

In order for the Segway to be simulated, the parameters must be estimated. The masses $m_b$ and $m_w$ could be measured directly with a scale. The lengths $a$ and $r$ were also measured directly from a ruler. The height of the center of mass, $h$, required finding the pivot point where along the $z$ axis where the Segway remain balanced. This was accomplished by placing the Segway on a table and continuously bringing the Segway further from the table until it would barely start tipping.

The moments of inertia are more challenging to measure, and instead is estimated by assuming the Segway is a rectangular body of uniform mass. The lengths along the $x$, $y$, and $z$ axis, denoted as $l_b$, $w_b$, and $h_b$ are measured and recorded as

\[
l_b = 0.0508 \text{ m} \\
w_b = 0.1524 \text{ m} \\
h_b = 0.9462 \text{ m}.
\]
As a result, the moment of inertia is estimated as

\[ J_x = \frac{m}{12} (h_b^2 + w_b^2) \]
\[ J_y = \frac{m}{12} (h_b^2 + l_b^2) \]
\[ J_z = \frac{m}{12} (l_b^2 + w_b^2) \]

resulting in the values given in Equation (3.100). Note that \( h_b \) is significantly greater than \( l_b \) and \( w_b \) and thus the moments of inertia about the \( x \) and \( y \) axis are similar to each other, while the moments of inertia about the \( z \) axis is much smaller. Although the error in these estimates are expected to be significant, they are a much better estimate than if the system was assumed to be a point mass body.

As a result the parameters of the physical body are estimated as

\[ m_b = 2.313 \text{ kg} \]
\[ m_w = 0.141 \text{ kg} \]
\[ h = 0.254 \text{ m} \]
\[ a = 0.165 \text{ m} \]
\[ r = 0.0615 \text{ m} \]
\[ J_x = 0.1986 \text{ mboxkg} \ast \text{ m}^2 \]
\[ J_y = 0.1942 \text{ kg} \ast \text{ m}^2 \]
\[ J_z = 0.0056 \text{ kg} \ast \text{ m}^2 \]
\[ J_w = 0.00025 \text{ kg} \ast \text{ m}^2. \] (3.100)

The voltage coefficients, \( k_e \) and \( k_f \), can be determined from the stall torque, \( T_{st} \), and the free run speed, \( \omega_f \). Recall that the general expression for the torque in terms of the voltage is

\[ T = k_e V - k_f \omega \] (3.101)
where $\omega$ is the angular velocity of the motor. The stall torque is the maximum torque that can be generated by the motor and occurs when the angular velocity is at zero such that Equation (3.101) becomes

$$T_{st} = k_e V.$$  (3.102)

The free run speed is the angular velocity of the motor if there was no load, such that Equation (3.101) becomes

$$0 = k_e V - k_v \omega_f.$$  (3.103)

If Equation (3.102) is substituted in Equation (3.103), then

$$T_{st} = k_v \omega_f.$$  (3.104)

As mentioned previously, the motors used are the Pololu 12V brushed DC motors with a 18.75:1 gearbox ratio. They are documented to have a free run speed, $\omega_f$, of 500 rpm and a stall torque, $T_{st}$, of 84 oz-in. From these specifications, the voltage coefficients are found through Equations (3.102) and (3.104) as

$$k_e = \frac{T_{st}}{V} = \frac{84 \text{ oz-in}}{12 \text{ V}} = 0.04943 \frac{\text{N*m}}{\text{V}}$$

$$k_v = \frac{T_{st}}{\omega_f} = \frac{84 \text{ oz-in}}{500 \text{ rpm}} = 0.01133 \frac{\text{kg*m}^2}{\text{s}}.$$  (3.105)

The velocity measurement is primarily based on the position encoders of the motors. The velocity is also measured by the accelerometers, but the encoders give a much more accurate estimate. The encoders are documented to measure 64 clicks per revolution of the motor shaft. If given the radius of the wheel, the maximum arc length from either click registering is

$$s = \frac{2\pi}{64}$$

and can be thought of the maximum error allowed. Because the angular rate is estimated by taking the difference between two angular positions and dividing by the difference in time, the error is doubled to $2s$. To estimate the maximum velocity error,
the error is multiplied by the maximum angular frequency. The maximum angular frequency is related to the free run speed by

\[ f_m = \frac{w_f}{2\pi}. \]

Therefore, the maximum velocity error is estimated as

\[ v_e = 2s \ast f_m \approx 0.100. \]

The purpose of the previous calculations is to find a reasonable order of magnitude for the velocity error. The error is then squared to given an estimate of the velocity measurement variance. The measurement variance for the gyroscopes is already given in the pre-existing code for the extended Kalman Filter written for the PX4 autopilot controller. Overall, the measurement variances are estimated as

\[ \sigma_v^2 = 0.010 \text{ (m/s)}^2 \]
\[ \sigma_\omega^2 = 0.008 \text{ (rad/s)}^2. \]  

(3.106)

The sampling time of the PX4 is estimated to be at least

\[ t_s = 0.001 \text{ s.} \]  

(3.107)

For the built Segway, the controller is also designed such that the states are decoupled. However, the controller is setup as stages where a closed loop is first formed about a smaller subset of the states. This subset is typically chosen to be the more vital states to the stability of the states. Specifically, the closed loop that determines \( V_s \) is first built about \( \theta \) and \( \dot{\theta} \) with zero reference commands on both and gains \( K^s_\theta \) and \( K^s_{\dot{\theta}} \). It is then given that

\[ V_s = 12 \left( -K^s_\theta \theta - K^s_{\dot{\theta}} \dot{\theta} \right). \]

The preference of the coefficient 12 is related to the battery’s voltage of 12V. Once these gains are tuned such that the Segway is able to remain about a zero pitch angle,
a reference command is fed into $\theta$, which is a function of the velocity error and its integral. This can be expressed as

$$V_s = 12 \left( K^s_\theta (\theta_r - \theta) - K^s_\dot{\theta} \right)$$

$$\theta_r = K^\theta_v (v_r - v) + K^\theta_l \int_0^t (v_r - v) \, d\tau.$$  

The gains $K^\theta_v$ and $K^\theta_l$ are then tuned for a zero velocity reference command. For $\vec{x}_d$, the process is similar but simpler is there are less states. The controller for the input $V_d$ is designed as

$$V_d = K^d_\psi (\dot{\psi}_r - \dot{\psi}) + K^d_\dot{\psi} \int_0^t (\dot{\psi}_r - \dot{\psi}) \, d\tau.$$  

As a result, the following relationship results between the Segway gains and the feedback gain matrix

$$K_l = -12 K^s_\theta K^\theta_l$$
$$K_v = -12 K^s_\theta K^\theta_v$$
$$K_\theta = -12 K^s_\theta$$
$$K_{\dot{\theta}} = -12 K^s_{\dot{\theta}}$$
$$K_\psi = -12 K^d_\psi$$
$$K_{\dot{\psi}} = -12 K^d_{\dot{\psi}}.$$  

After some fine tuning, the final set of gains are

$$K^s_\theta = 10$$
$$K^s_{\dot{\theta}} = 1$$
$$K^\theta_l = 0.2$$
$$K^\theta_v = 0.2$$
$$K^d_\psi = 0.2$$
$$K^d_{\dot{\psi}} = 0.2.$$  

(3.108)
From Equation (3.108), the Segway gains in Equation (3.109) become

\[ K_l = -24 \]
\[ K_s = \begin{bmatrix} -24 & -120 & -12 \end{bmatrix} \]
\[ K_\psi = -2.4 \]
\[ K_d = -2.4. \]

If the tuned feedback gain matrix in Equation (3.110) is fed back into the predicted model, then the resulting eigenvalues of the matrix \( A - BK \) are determined as

\[ \lambda_s = \{-1.33 \pm 8.69i, -0.79 \pm 1.34i\} \]
\[ \lambda_d = \{-0.68, -27.8\}. \]

These eigenvalues are stable, but not considered optimal. However, remember that the actual Segway is not as ideal as the simulations and therefore is has a smaller range of possible gains.

### 3.10 Stabilization of Segway (Zero Reference Commands)

The first scenario to be tested is the case when the reference commands for both the velocity and the heading angle rate is zero. In other words, the Segway is only stabilizing itself with a velocity, pitch, pitch rate, and heading angle rate going to zero. Referring to Equation (3.86), the arc length and heading angle are also driven to the initial values under zero reference commands.

With some initial disturbance, the Segway is powered on a carpeted surface and left to ran for about 60 seconds. The data of the states is logged from QGroundControl, which is the ground control station program used to track the Segway. Note that the angles and their rates are sampled at a frequency of approximately 5 Hz, and the
velocity and arc length is sampled at approximately 1 Hz. The initial conditions are recorded as

\[
\vec{x}_{s,0} = \begin{bmatrix} 0 \text{ m} \\ -0.07 \text{ m/s} \\ -9.13 \text{ deg} \\ -15.17 \text{ deg/s} \end{bmatrix}, \\
\vec{x}_{d,0} = \begin{bmatrix} -7.94 \text{ deg} \\ -1.29 \text{ deg/s} \end{bmatrix}.
\]

The plots of the real-time data is plotted below in Figures 3.4 - 3.9.

Fig. 3.4. Real-time data of \( l \) versus time plot for stabilizing (zero reference command).

Fig. 3.5. Real-time data of \( v \) versus time for stabilizing (zero reference command).
As shown in Figures 3.8 and 3.9, the pitch $\theta$ and its rate $\dot{\theta}$ are shown to settle within 1.0° and 12 deg/s about zero respectively after the initial disturbance. The velocity $v$ and $\dot{\psi}$ from Figures 3.5 and 3.7 are also shown to go to 0.1 m/s and 12 deg/s respectively about zero. However, the arc length $s$ and heading angle $\psi$ from
Figures 3.4 and 3.6 have not settled as much as the other state, but appears to be converging to within 0.05 m about 0.05 m and 1.0° about −2.5° respectively. It is important to note that when the Segway is powered on, the initial arc length and heading angle is reported at zero. The data is recorded shortly after, but not after the Segway is slightly off from this position. Therefore, the Segway is always driven to a heading angle and arc length of zero when there is no reference commands. As shown, these states were not driven to zero as desired. Although small, there clearly is a difference. This offset is thought to be due to the rolling friction between the wheel and the carpet and Coulomb friction in general. In total, the Segway does a satisfactory job of remaining stable, although not about the origin.

Now allow the same initial conditions to be carried out for the simulations for 60 seconds. The simulations are ran on MATLAB and Simulink using the nonlinear model from Equation (3.53). Plotted are the states itself, the estimated states, and the states if there was no noise in the system.

Fig. 3.10. Simulation of $l$ versus time plot for stabilizing (zero reference command).

Fig. 3.11. Simulation of $v$ versus time for stabilizing (zero reference command).
Fig. 3.12. Simulation of $\psi$ versus time for stabilizing (zero reference command).

Fig. 3.13. Simulation of $\dot{\psi}$ versus time for stabilizing (zero reference command).

Fig. 3.14. Simulation of $\theta$ versus time for stabilizing (zero reference command).

Fig. 3.15. Simulation of $\dot{\theta}$ versus time for stabilizing (zero reference command).
As shown, the estimator follows the state very well, with exception to \( \psi \) and \( \dot{\psi} \). As mentioned, the omission of state noise results in the steady state Kalman Filter to ignore the measurements of \( \dot{\psi} \). Shown in Figures 3.12 and 3.13, there is no noise whatsoever in these states. Notice the simulated estimator returns \( \psi \) to its initial value, as if there was no noise, but the simulated state itself has a slight steady state error.

Remember that the decoupling is only possible under the linear assumption. However, there is still coupling when the nonlinear model is considered. Although it is at its smallest when the system is near equilibrium, the coupling still exist. As a result, the measurement noise present in the other states have an impact in \( \psi \) and \( \dot{\psi} \).

As shown, the error accumulates to about a 0.1° difference. Although this error is small, it will differ much more for a realistic nonlinear model of the Segway. This therefore demonstrates the limitations of the linear steady state Kalman Filter, which relies only on the linear model of the Segway. As mentioned, The PX4 Pixhawk was designed with a general extended Kalman filter, such the estimated states of \( \dot{\psi} \) and \( \dot{\psi} \) are now dependent on the measurements of the gyroscope.
In addition, the voltage is plotted from the simulation in Figure 3.16. As shown, the voltage for the left and right wheel are nearly identical as the motion in the simulation is mainly in one direction. The voltage is shown to reach a maximum of about 10 volts when stabilizing in the beginning and settles to a voltage of less than 1 volt. The voltage of the Segway was not recorded from QGroundControl.

Now the simulated and actual measured states of $l$, $v$, $\theta$ and $\dot{\theta}$ are plotted together in Figures 3.17 - 3.20.

![Fig. 3.17. Comparison of real-time data and simulation of $l$ versus time plot for stabilizing (zero reference command).](image)

![Fig. 3.18. Comparison of real-time data and simulation of $v$ versus time for stabilizing (zero reference command).](image)
Fig. 3.19. Comparison of real-time data and simulation of $\theta$ versus time for stabilizing (zero reference command).

Fig. 3.20. Comparison of real-time data and simulation of $\dot{\theta}$ versus time for stabilizing (zero reference command).

For the states $v$, $\theta$, and $\dot{\theta}$, the simulations appear provide a very good estimate of the amount noise present in the system, as seen in Figures (3.18), (3.19) and (3.20). For the arc length, the error of the simulation seems to be about half the magnitude of the real data, as seen in Figure (3.17).

A significant difference between the simulated and real results is their response within the first few seconds. There is shown to be a greater amount of overshoot in the simulated model. This is better demonstrated by zooming in the first 10 seconds for $l$, $v$ $\theta$ and $\dot{\theta}$ in Figures 3.21 - 3.24.
Fig. 3.21. Comparison of real-time data and simulation of $l$ versus time plot for stabilizing (zero reference command).

Fig. 3.22. Comparison of real-time data and simulation of $v$ versus time for stabilizing (zero reference command).

Fig. 3.23. Comparison of real-time data and simulation of $\theta$ versus time for stabilizing (zero reference command).

Fig. 3.24. Comparison of real-time data and simulation of $\dot{\theta}$ versus time for stabilizing (zero reference command).

As shown in each of the figures, the settling time appears to be roughly the same. The overshoot appears to stand out the most in $v$ and $\dot{\theta}$ in Figures 3.22 and 3.24. For $l$ and $\theta$, the overshoot is roughly twice as much in the simulations in comparison the
the real results. However, a greater sampling frequency would be beneficial to better determine the amount of overshoot for the built Segway, especially, for $l$ and $v$.

It is believed that if the parameters were better estimated, particularly in $J_y$, then the difference in the simulations results would be considerable less. However, the simulations still reflect the behavior of the built Segway fairly well. Suppose that the built Segway is now given nonzero reference commands.

### 3.11 Test of Constant Velocity and Heading Angle Rate

The Segway is also capable of handling velocity and heading rate reference commands through a RC transmitter. Using the right control stick of the Spektrum DX5e transmitter, the velocity and heading angle rate reference commands can be sent to the Segway. The $x$ direction of the control stick determines $\dot{\psi}_r$ and the $y$ direction of the control stick determines $v_r$. The magnitudes of $v_r$ and $\dot{\psi}_r$ are limited at 0.2 m/s and 25 deg/s respectively to reduce the risk of the Segway becoming unstable.

Suppose that the reference commands are set at the maximum possible values of $v_r = 2$ m/s and $\dot{\psi}_r = 25$ deg/s, and ran for about 50 seconds. The initial conditions are recorded as

\[
\vec{x}_{s,0} = \begin{bmatrix}
0.37 \text{ m} \\
0 \text{ m/s} \\
5.84 \text{ deg} \\
-1.95 \text{ deg/s}
\end{bmatrix}
\]

\[
\vec{x}_{d,0} = \begin{bmatrix}
0 \text{ deg} \\
-1.29 \text{ deg/s}
\end{bmatrix}
\]

The plots of the real-time data and the simulated data are plotted below in Figures 3.25 - 3.30.
Fig. 3.25. Comparison of real-time data and simulation of \( I \) versus time plot for constant reference commands.

Fig. 3.26. Comparison of real-time data and simulation of \( v \) versus time for constant reference commands.

Fig. 3.27. Comparison of real-time data and simulation of \( \psi \) versus time for constant reference commands.

Fig. 3.28. Comparison of real-time data and simulation of \( \dot{\psi} \) versus time for constant reference commands.
As seen in Figure 3.25 and 3.26, the actual Segway was first stabilizing itself before following the reference commands for the first 17 seconds. The simulation expected this stabilization to happen much earlier. However, the Segway was then able to maintain an average velocity of about 0.2 m/s very well. This is better shown in Figure 3.25, where the simulation and experimental plots are nearly parallel, such that the slope of both lines are approximately equal to each other. The actual velocity is still varying quite greatly, but averages to about 0.2 m/s.

The Segway also did an excellent job in maintaining a constant heading angle rate, as shown in Figures 3.27 and 3.28. Like before, the Segway required some time to stabilize itself before focusing on the reference command. As seen in Figure 3.28, this time is also about 17 seconds compared to the simulation’s 5 seconds. Before then, $\dot{\psi}$ reaches to both 75 deg/s and $-70$ deg/s. The heading angle rate still fluctuates greatly at about 20 deg/s such that the plot in Figure 3.27 is not as linear as it was in Figure 3.25. However, the Segway is still able to follow an average heading angle rate of 25 deg/s very well.
The biggest surprise is the results of the pitch and pitch rate in Figures 3.29 and 3.30. From the given model, the Segway is expected to go to a zero pitch angle, even with a constant reference command. This is shown in the simulation plot of 3.29. However, the Segway is shown to steady itself in the first 17 seconds before settling within 1° about 1.5°. The simulation also appears to have the same noise, but settles within 1° about 0° instead. The pitch rate in Figure 3.30 appears to overlap very well, where they both settle to 0 deg/s. The actual Segway appears to have greater spikes reaching up to 15 deg/s at certain times.

Overall, the Segway exceed expectations in following the constant reference commands with a velocity of 0.2 m/s and a heading angle rate of 25 deg/s. The simulations are able to model the actual Segway in many regards except the pitch angle. Although 1.5° is very small and not noticeable, this steady state error was not expected in the predicted model. Like before, this is thought to be due to unaccounted sources of error in the model, such as rolling friction.

3.12 Simulations of Following a Given Path

Up to now only constant reference commands were given to Segway, as allowed by theory. However, the built Segway can be given reference commands that are not constant through use of the RC transmitter. This allows the Segway to travel along a path as the commands are adjusted in real time. What about a predetermined path that could be programmed ahead of time into the Segway?

If the reference commands are a function of time, then it indeterminable of how the Segway will perform. The slower the reference commands change with time, the more likely the Segway is able to follow the reference commands. Let us investigate the possibility of giving the Segway velocity and heading angle rate reference commands such that it attempts to follow a given path.
Suppose that the Segway is desired to follow some predetermined path in which the 
\( x \) and \( y \) coordinates are functions of time. From the constraint equation in Equation (3.32), the reference command of velocity can be related to the \( x \) and \( y \) velocities as
\[
v_r = \sqrt{\dot{x}_p^2 + \dot{y}_p^2} \tag{3.111}
\]
where \( \dot{x}_p \) and \( \dot{x}_p \) is the \( x \) and \( y \) velocities of the desired path. The command of the heading angle is also related to the \( x \) and \( y \) velocities by the expression
\[
\tan(\psi_r) = \frac{\dot{y}_p}{\dot{x}_p}. \tag{3.112}
\]
Taking the implicit derivative of Equation (3.112) results in
\[
\frac{\dot{\psi}_r}{\cos(\psi_r)^2} = \frac{\dot{x}_p \ddot{y}_p - \dot{y}_p \ddot{x}_p}{\dot{x}_p^2} \tag{3.113}
\]
Because \( \dot{x}_p = v_s \cos(\psi) \), Equation (3.113) becomes
\[
\dot{\psi}_r = \frac{\ddot{x}_p \dot{y}_s - \ddot{y}_p \dot{x}_s}{\dot{x}_p^2 + \dot{y}_p^2} = \frac{\ddot{x}_p \dot{y}_p - \ddot{y}_s \dot{x}_p}{v_r^2}. \tag{3.114}
\]
Suppose the path is in polar coordinates instead of Cartesian, such that
\[
x_p = r_p \cos(\Psi_p) \nonumber
\]
\[
y_p = r_p \sin(\Psi_p) \tag{3.115}
\]
where \( r_p \) and \( \Psi_p \) is the radius and polar angle of the desired path. Then the velocities and accelerations in the \( x \) and \( y \) directions are
\[
\dot{x}_p = \dot{r}_p \cos(\Psi_p) - r_s \dot{\Psi}_p \sin(\Psi_p) \nonumber
\]
\[
\dot{y}_p = \dot{r}_p \sin(\Psi_p) + r_s \dot{\Psi}_p \cos(\Psi_p) \nonumber
\]
\[
\ddot{x}_p = \ddot{r}_p \cos(\Psi_p) - 2 \dot{r}_p \dot{\Psi}_p \sin(\Psi_p) - r_p \ddot{\Psi}_p \sin(\Psi_p) - r_p \dot{\Psi}_p^2 \cos(\Psi_p) \nonumber
\]
\[
\ddot{y}_p = \ddot{r}_p \sin(\Psi_p) + 2 \dot{r}_p \dot{\Psi}_p \cos(\Psi_p) + r_p \ddot{\Psi}_p \cos(\Psi_p) - r_p \dot{\Psi}_p^2 \sin(\Psi_p). \tag{3.116}
\]
Applying Equations (3.116) to Equations (3.111) and (3.114) results in the reference commands as a function of \( r_s \) and \( \Psi_s \)
\[
v_r = \sqrt{\dot{r}_p^2 + \dot{r}_p \dot{\Psi}_p^2 + \dot{r}_p^2 \dot{\Psi}_p^2} \nonumber
\]
\[
\dot{\psi}_r = \frac{r_p \left( \dot{r}_p \dot{\Psi}_p - \dot{r}_p \dot{\Psi}_p \right) + \dot{\Psi}_p \left( r_p \dot{r}_p^2 + 2 r_p \dot{r}_p \dot{\Psi}_p + r_p \dot{\Psi}_p^2 \right)}{\dot{r}_p^2 + \left( r_p \dot{\Psi}_p \right)^2}. \tag{3.117}
\]
As an example, consider a path in which both the velocity and heading rate commands are a function of time, such as a rose. By the mathematical definition, a rose is characterized by the polar equation

\[
\begin{align*}
    r_p(t) &= a \cos(n \omega t) \\
    \Psi_p(t) &= \omega t.
\end{align*}
\]  

(3.118)

By applying Equation (3.117), the reference commands for the velocity and the heading rate are

\[
\begin{align*}
    v_r(t) &= a \omega \sqrt{1 + (n^2 - 1) \sin^2(n \omega t)} \\
    \dot{\psi}_r(t) &= \frac{\omega \left( (n^2 + 1) + (n^2 - 1) \sin^2(n \omega t) \right)}{1 + (n^2 - 1) \sin^2(n \omega t)}.
\end{align*}
\]  

(3.119)

Note that the commands are functions of time and thus the Segway can roughly follow the path if the commands change slowly over time. Using the same parameters as before, allow the conditions for the rose be

\[
\begin{align*}
    n &= 2 \\
    a &= 1 \\
    \omega &= 4 \text{ deg/s}
\end{align*}
\]  

(3.120)

such that a full completion takes 90 seconds and the rose has four pedals. The initial conditions are chosen as

\[
\begin{align*}
    \vec{x}_{s,0} &= \begin{bmatrix} 0 \text{ m} \\ 0.05 \text{ m/s} \\ -6 \text{ deg} \\ 3 \text{ deg/s} \end{bmatrix} \\
    \vec{x}_{d,0} &= \begin{bmatrix} 0 \text{ deg} \\ 3 \text{ deg/s} \end{bmatrix}.
\end{align*}
\]

From the simulations, the following results are obtained and presented below in Figures 3.31 - 3.36.
Fig. 3.31. Simulation of $v$ versus time for a four petaled rose.

Fig. 3.32. Simulation of $\dot{\psi}$ versus time for a four petaled rose.

Fig. 3.33. Simulation of $\theta$ versus time for a four petaled rose.

Fig. 3.34. Simulation of $\dot{\theta}$ versus time for a four petaled rose.
Fig. 3.35. Simulation of $xy$ position for a four petaled rose.

Fig. 3.36. Simulation of motor input voltage versus time for a four petaled rose.

If there is no noise, the velocity changes from about 0.07 m/s to 0.14 m/s repeatedly in Figure 3.31. Similarly, the heading rate changes from 8 deg/s to 20 deg/s in Figure 3.32. However, the Segway is required to obtain stability first before following the given path. Once the Segway settled to a small $\theta$, then it was able to follow the remainder of the desired path fairly well. As a result, the rose was both off center and tilted due to this delay. Overall, there was some success to tracing out a rose path, but caution must be taken to how fast the reference commands change with time.

Therefore it seems possible that the Segway can follow the path of a rose with these given parameters. As of now, the Segway responds to slowly to changes in the velocity and heading angle commands. However, this is assumed to be greatly due to the limitation of performance in the current motors and other components. If improved, then I believe that the investigation in this section could be applied to the built Segway. The coding of the Segway would also have to modified to better accommodate the input of a predetermined path. This implementation is very feasible if more time was permitted.
3.13 Reference Commands Applied to Arc Length and Heading Angle

Another possible future investigation is to drive the Segway to a given arc length and heading angle. To do so, consider the approach taken with the velocity and heading angle rate commands. However, the variable is defined about the error of \( l \) and \( \psi \) instead, such that

\[
e_{s,0} = l_r - l
\]
\[
e_{d,0} = \psi_r - \psi.
\]

Then the resulting system for both states becomes

\[
\begin{bmatrix}
\dot{e}_0 \\
\dot{\vec{e}}
\end{bmatrix}
= \begin{bmatrix}
0 & \Pi^* \\
0 & A
\end{bmatrix}
\begin{bmatrix}
e_0 \\
\vec{e}
\end{bmatrix}
- \begin{bmatrix}
0 \\
B
\end{bmatrix}
\begin{bmatrix}
u
\end{bmatrix}.
\]

(3.121)

As shown previously, the system in Equation (3.121) is a controllable system. By applying the same feedback controller of \( u = K_0 e_0 + K \vec{e} \), Equation (3.121) becomes

\[
\begin{bmatrix}
\dot{e}_0 \\
\dot{\vec{e}}
\end{bmatrix}
= \begin{bmatrix}
0 & \Pi^* \\
-BK_0 & A - BK
\end{bmatrix}
\begin{bmatrix}
e_0 \\
\vec{e}
\end{bmatrix}.
\]

(3.122)

In steady state, Equation (3.122) results in

\[
\begin{bmatrix}
0 \\
0
\end{bmatrix}
= \begin{bmatrix}
0 & \Pi^* \\
-BK_0 & A - BK
\end{bmatrix}
\begin{bmatrix}
e_0(\infty) \\
\vec{e}(\infty)
\end{bmatrix}.
\]

(3.123)

Because the square matrix in Equation (3.123) is invertible, both \( e_0(\infty) = 0 \) and \( \vec{e}(\infty) = 0 \). Therefore, there is no steady state error in the states if the model is assumed to be linear.

As desired, the Segway is driven to some given arc length and heading angle. It is important to note that the arc length is not equivalent to the net distance. If a waypoint \((x_r, y_r)\) was given in inertial coordinates such that \( l_r = \sqrt{x_r^2 + y_r^2} \), then the Segway would fall short of the waypoint. In addition, \( \psi_r \) is the final heading angle of the Segway and not the angle between the waypoint and the \( x \) axis. However, if the Segway does not deviate too much from a straight line during stabilization, then the
final position and waypoint is expected to be considerably close to one another. This is not guaranteed to be true in all cases.

However, what is the biggest concern is that when the reference $l_r$ is given, the error $l_r - l$ would roughly be at its largest at the initial point, assuming $l(0) = 0$. The greater the error, the more the controller will focus on driving the Segway to the arc length $l_r$ than keeping $\theta$ near zero. This is problematic when considering that the stability of $\theta$ is vital for the linearized approximations to be valid. Therefore, it is best to keep $l_r$ relatively small. It is possible for a string of waypoints to be set such that the waypoints are close together such that these conflicts are minimized. Doing so requires that the reference commands to repeatedly change once the Segway draws close to each waypoint. This is a strong possibility that can be implemented into the built Segway in the future.

In addition, it is possible to mix and match the feedback controllers from both this and the last section. For example, it is possible to set the reference commands about the velocity and heading angle.

### 3.14 Conclusions

With the use of Lagrangian mechanics, the nonlinear model of the Segway is obtained. The equations of motion were then linearize about a zero pitch angle equilibrium, and decoupled to simplify the model. A feedback controller was then designed for the Segway to follow a given constant velocity and heading angle rate. To imitate more realistic conditions, measurement noise was included in the model of the Segway. A steady state Kalman Filter was designed for the simulations, but was shown to be greatly limited to estimating the heading angle and its rate.

With the feedback controller designed, a Segway was built from relatively expensive parts and controlled through a RC transmitter. The Segway was first tested with an initial disturbance under zero reference commands, otherwise known as the stabilization scenario. The results were recorded for the states and compared to the
simulation results under the same conditions. Overall, the simulations were able to give a good representation of the actual results, but differed in some respects. This included the steady state error in the heading angle and arc length. The simulated results also overshot more, suggesting that the measured parameters used in the simulation could be more accurate. However, the noise in the simulation was very close to the error in the actual results.

The Segway was also given constant velocity and heading angle rate reference commands after an initial disturbance. The results were again compared to the simulated results. The overshoot in the simulated results were also greater than the actual results. The actual results also revealed that the Segway would have a small steady error in the pitch angle, which was not predicted in the simulated results. However, the Segway did an excellent job of following the reference commands once it stabilized from the initial disturbance. The simulated results were once again effective in reflecting the amount of state noise in the actual results. The possibility of following velocity and heading angle commands that changed with time was investigated in the simulations with a path of a rose. The Segway appears to be able to follow the rose very well, so long as the reference commands change slowly with time. However, time was limited such before a predetermined path was programmed into the Segway. This option is possible for some time in the future.

However, it is important that the Segway was compared to ideal conditions. The Segway was limited by the rigidness of the structure, in which the simulations were based on a rigid body. The simulation also did not include delays such as those the wheels overcome with a change in direction. For the price and availability of the Segway’s components, the limitations were completely reasonable. However, the motors could be greatly improved to provide a faster response. More importantly, the simulation’s parameters were based on rough measurements of the built Segway and could be improved to be much more accurate, especially the moments of inertia.

Future work would involve the investigation of these limitations and their inclusion in the simulation models through further system identification. Although they
may not be eliminated, accounting for these limitations can make a dramatic difference in the performance of the Segway. The system identification would also yield better estimates for the parameters, which were based on rough measurements and documented information. Doing so would allow the simulations to have a better representation of the actual results. In addition, the use of the arc length and heading angle reference commands could prove very useful in setting waypoints, and is worthwhile to further investigate in the future. Overall, the Segway performed well under all given scenarios and exceeded all expectations.
4. CHAPLYGIN SLEIGH WITH A SPINNING DISC

4.1 Introduction to the Chaplygin Sleigh

The next system to be investigated is the Chaplygin sleigh, which also has a nonholonomic constraint. The Chaplygin sleigh (or Chaplygin beanie) is defined as a body platform on top of a fixed wheel, placed towards the front of the body. The other points of contact between the body and the floor are typically ball bearings, such that the motion is not constrained by these contacts. However, the front wheel is modeled as a nonholonomic constraint. To drive the Chaplygin sleigh, a horizontal spinning disc is placed at the center of mass of the main body. The main body also includes the wheel and bearings. Our work on the Chaplygin sleigh with the rotating disc was inspired by M.J. Fairchild [2].

A basic drawing of the Chaplygin sleigh is given in Figure 4.1. Here, \( x \) and \( y \) are the inertial coordinate of the sleigh’s center of mass. The disc has a moment of inertia of \( J_d \) about the \( z \) axis and an angular position of \( \psi \) with respect to the body. The front wheel is placed a distance \( l \) from the center of mass. The main body is at an angle of \( \theta \) with respect to the inertial frame. The angle \( \theta \) can be thought of as
the heading angle. The main body has a moment of inertia of $J$ about the $z$ axis. The total mass of both the main body and disc is denoted as $m$. For the case of the Chaplygin sleigh, the moment of inertia of the front wheel and the bearings will be approximated as zero and ignored.

4.2 Derivation of Equations of Motion Without Friction

To derive the equations of motion, Lagrangian mechanics will be applied. The first step in doing so is to find the kinetic and potential energy. The kinetic energy is the sum of the translational and rotational motion of the center of mass with respect to the inertial frame. Because the disc is both symmetric and spins about the center of mass of the main body, the spinning of the disc does not effect the center of mass of the entire system. Therefore, the total translation energy of the system is given as

$$T_{tr} = \frac{1}{2} m \left( \dot{x}^2 + \dot{y}^2 \right).$$

In addition, the angular velocity of both the main body and disc is always along the $z$ axis. Therefore, the rotational energy is given as

$$T_{rot} = \frac{1}{2} J \dot{\theta}^2 + \frac{1}{2} J_d \left( \dot{\psi} + \dot{\theta} \right)^2.$$

Taking the sum of the translational and rotation energy results in the total kinetic energy, or

$$T = \frac{1}{2} m \left( \dot{x}^2 + \dot{y}^2 \right) + \frac{1}{2} J \dot{\theta}^2 + \frac{1}{2} J_d \left( \dot{\psi} + \dot{\theta} \right)^2. \quad (4.1)$$

Because the body doesn’t move vertically, there is no change in potential energy. As a result, the potential energy is zero, or

$$V = 0.$$

The Lagrangian is then found as

$$L = T - V$$

$$= \frac{1}{2} m \left( \dot{x}^2 + \dot{y}^2 \right) + \frac{1}{2} J \dot{\theta}^2 + \frac{1}{2} J_d \left( \dot{\psi} + \dot{\theta} \right)^2. \quad (4.2)$$
The nonholonomic constraints are obtained by mathematically expressing the possible motion of the front wheel as a skate. From Figure 4.1, the inertial coordinates of the wheel with respect to the center of mass are

\[ x_w = x - l \cos(\theta) \]
\[ y_w = y - l \sin(\theta) \]  

(4.3)

where \( x_w \) and \( y_w \) are the \( x \) and \( y \) positions of the front wheel respectively. Taking the derivative of Equation (4.3) results in the velocity of the front wheel

\[ \dot{x}_w = \dot{x} + l \dot{\theta} \sin(\theta) \]
\[ \dot{y}_w = \dot{y} - l \dot{\theta} \cos(\theta). \]  

(4.4)

Because the wheel cannot move sideways with respect to the wheel, the velocity in the perpendicular direction is zero. In other words, \( [\dot{x}_w \dot{y}_w]^* \) is orthogonal to \( [\sin(\theta) \ - \cos(\theta)]^* \). As a result, the dot product between the two vectors is zero, such that

\[ 0 = \begin{bmatrix} x_w & y_w \end{bmatrix} \begin{bmatrix} \sin(\theta) \\ - \cos(\theta) \end{bmatrix} \\
= (\dot{x} + l \dot{\theta} \sin(\theta)) \sin(\theta) + (\dot{y} - l \dot{\theta} \cos(\theta)) (- \cos(\theta)) \]

which simplifies to

\[ \dot{x} \sin(\theta) - \dot{y} \cos(\theta) + l \dot{\theta} = 0. \]  

(4.5)

Now consider the generalized coordinates of the Chaplygin sleigh given in the order of

\[ q = \begin{bmatrix} x \\ y \\ \theta \\ \psi \end{bmatrix}. \]  

(4.6)
Then the constraint equation in Equation (4.5) can be expressed in the form of

\[ H\ddot{q} = 0 \quad \text{where} \quad H = \begin{bmatrix} \sin(\theta) & -\cos(\theta) & l & 0 \end{bmatrix}. \quad (4.7) \]

As mentioned previously, constraints of the form \( H(q)\dot{q} = 0 \) are, by definition, Pfaffian nonholonomic constraint.

The equations of motion of the system are then derived using Lagrangian dynamics. Lagrange’s equations of motion for each generalized coordinate becomes

\[
\frac{d}{dt} \frac{\partial L}{\partial \dot{x}} - \frac{\partial L}{\partial x} = m\ddot{x} \\
\frac{d}{dt} \frac{\partial L}{\partial \dot{y}} - \frac{\partial L}{\partial y} = m\ddot{y} \\
\frac{d}{dt} \frac{\partial L}{\partial \dot{\theta}} - \frac{\partial L}{\partial \theta} = (J + J_d)\ddot{\theta} + J_d\ddot{\psi} \\
\frac{d}{dt} \frac{\partial L}{\partial \dot{\psi}} - \frac{\partial L}{\partial \psi} = J_d\ddot{\theta} + J_d\ddot{\psi}. \quad (4.8)
\]

Lagrange’s equations can be written in the form of

\[
\frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = M_q\ddot{q} - \beta_q \quad (4.9)
\]

where \( M \) is the mass matrix and \( \beta \) are the leftover terms. From Equation (4.8), the mass matrix is

\[
M_q = \begin{bmatrix}
m & 0 & 0 & 0 \\
0 & m & 0 & 0 \\
0 & 0 & J + J_d & J_d \\
0 & 0 & J_d & J_d
\end{bmatrix}. \quad (4.10)
\]

In this case, there are no leftover terms, such that \( \beta_q = 0 \).

Equation (4.9) is set equal to the forces that act on the system

\[
M_q\ddot{q} - \beta_q = H^*\lambda + F_qu. \quad (4.11)
\]

Here \( H^*\lambda \) are the forces due to the nonholonomic constraint, \( H\dot{q} = 0 \), with \( \lambda \) corresponding to the Lagrange multiplier. The generalized forces due to the inputs is given
in the form of $F_q u$, where $u$ corresponds to the input and $F_q$ is the input coefficient matrix.

Because the only input on the system is the torque applied to the disc $T_d$ the input is expressed as

$$u = T_d$$

and the input coefficient matrix is

$$F_d = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}.$$

Equation (4.11) is therefore simplified to

$$M_q \ddot{q} = H^* \lambda + F_q T_d. \quad (4.12)$$

The Lagrange multiplier, $\lambda$ can be eliminated by finding a basis for the null space of $H$. To be specific, first notice the rank of $H$ equals one. Hence the dimensions of the nullspace equals three. Therefore, there exist a matrix $\Phi = [\varphi_1, \varphi_2, \varphi_3]$ whose range equals the nullspace of $H$. In particular

$$H \Phi = 0$$

or equivalently

$$\Phi^* H^* = 0.$$

The matrix $\Phi$ allows us to define a new set of generalized coordinates, $p$, through the transformation

$$\dot{q} = \Phi \dot{p}. \quad (4.13)$$
The new coordinates $p$ can be viewed as the unconstrained generalized coordinates. Although there are an infinite number of possible choices for $\Phi$, the matrix we chose is given as

$$
\Phi = \begin{bmatrix}
\cos(\theta) & -l \sin(\theta) & 0 \\
\sin(\theta) & l \cos(\theta) & 0 \\
0 & 1 & 0 \\
0 & 0 & 1
\end{bmatrix}.
$$

(4.14)

The expression $\dot{q} = \Phi \dot{p}$ becomes

$$
\begin{bmatrix}
\dot{x} \\
\dot{y} \\
\dot{\theta} \\
\dot{\psi}
\end{bmatrix} =
\begin{bmatrix}
\cos(\theta) & -l \sin(\theta) & 0 \\
\sin(\theta) & l \cos(\theta) & 0 \\
0 & 1 & 0 \\
0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
\dot{p}_1 \\
\dot{p}_2 \\
\dot{p}_3
\end{bmatrix}.
$$

(4.15)

From the bottom two rows of $\Phi$ in Equation (4.15), the second and third unconstrained coordinates, $p_2$ and $p_3$ can be chosen as $\theta$ and $\psi$ respectively. The first two equations of Equation (4.15) are carried out as

$$
\dot{x} = \ddot{x}_w - l \ddot{\theta} \sin(\theta)
$$

$$
\dot{y} = \ddot{y}_w + l \ddot{\theta} \cos(\theta).
$$

(4.16)

By rewriting Equation (4.4) to

$$
\dot{x} = \dot{x}_w - l \dot{\theta} \sin(\theta)
$$

$$
\dot{y} = \dot{y}_w + l \dot{\theta} \cos(\theta).
$$

(4.17)

Comparing Equation (4.16) to (4.4), we see that

$$
\ddot{x}_w = p_1 \cos(\theta)
$$

$$
\ddot{y}_w = p_2 \sin(\theta).
$$

(4.18)
As shown in Equation (4.18), \( \dot{p}_1 \) is shown to be the velocity of the front wheel, and is denoted as \( v \) such that

\[
\begin{align*}
\dot{x}_w &= v \cos(\theta) \\
\dot{y}_w &= v \sin(\theta).
\end{align*}
\] (4.19)

From Equation (4.17) and (4.19), the velocity \( v \) can be related to both the \( x \) and \( y \) velocities of the front wheel and center of mass by

\[
v = \dot{x} \cos \theta + \dot{y} \sin \theta
= \dot{x}_w \cos \theta + \dot{y}_w \sin \theta.
\]

The final expression for \( \dot{p} \) is given as

\[
\dot{p} = \begin{bmatrix} v \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}.
\] (4.20)

To transform the equations of motion from the \( q \) set of generalized coordinates into the \( p \) set, recall that \( \Phi^* H^* = 0 \). Multiplying the left hand side of \( M_q \ddot{q} = H^* \lambda + F_q T_d \) by \( \Phi^* \) yields \( \Phi^* M_q \ddot{q} = \Phi^* F_q T_d \). Additionally, taking the derivative of \( \dot{q} = \Phi \dot{p} \) results in

\[
\ddot{q} = \Phi \ddot{p} + \dot{\Phi} \dot{p}.
\]

Substituting the above into \( \Phi^* M_q \ddot{q} = \Phi^* F_q T_d \) yields \( \Phi^* M_q \left( \Phi \ddot{p} + \dot{\Phi} \dot{p} \right) = \Phi^* F_q T_d \). This can be simplified into the form of

\[
M_p \ddot{p} = \beta_p + F_p T_d
\] (4.21)

where

\[
M_p = \Phi^* M_q \Phi \\
\beta_p = -\Phi^* M_q \dot{\Phi} \dot{p} \\
F_p = \Phi^* F_q.
\] (4.22)
Carrying out each of the expressions in Equation (4.22) results in

\[ M_p = \begin{bmatrix} m & 0 & 0 \\ 0 & J + J_d + ml^2 & J_d \\ 0 & J_d & J_d \end{bmatrix} \]

\[ \beta_p = -\Phi^* M_q \dot{\phi} = \begin{bmatrix} ml\dot{\theta}^2 \\ -mlv\dot{\theta} \\ 0 \end{bmatrix} \]

\[ F_p = \Phi^* F_q = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \] (4.23)

Substituting the coefficient matrices in Equation (4.23) into Equation (4.21) becomes

\[ \begin{bmatrix} m & 0 & 0 \\ 0 & J + J_d + ml^2 & J_d \\ 0 & J_d & J_d \end{bmatrix} \begin{bmatrix} \dot{v} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} ml\dot{\theta}^2 \\ -mlv\dot{\theta} \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ T_d \end{bmatrix}. \] (4.24)

Multiplying Equation (4.24) by the inverse of the mass matrix results in the unconstrained equations of motion

\[ \dot{v} = l\dot{\theta}^2 \] (4.25)

\[ \ddot{\theta} = -\frac{mlv\dot{\theta} + T_d}{J + ml^2} \] (4.26)

\[ \ddot{\psi} = \frac{J_d mlv\dot{\theta} + (J + J_d + ml^2)T_d}{(J + ml^2)J_d}. \] (4.27)

### 4.3 Analysis of Equations of Motion

The equations of motion in Equations (4.25) - (4.26) reduce to the classic equation of motions for the Chaplygin sleigh when there is no input, such that \( T_d = 0 \). The classic equation of motions are given in Bloch [3] in which the angular momentum and projected linear momentum of the wheel, denoted as \( \alpha_1 \) and \( \alpha_2 \) are used instead.
of the wheel velocity and heading angle. The relationship between the two sets of states are given as

\[ \alpha_1 = (J + ml^2)\dot{\theta} \]
\[ \alpha_2 = mv. \]  

(4.28)

It is noted that \( J + ml^2 \) is the moment of inertia of the Chaplygin sleigh about the wheel. By substituting in Equations (4.28) into Equations (4.25) and (4.26) and letting \( T_d = 0 \), we obtain the following equations of motion given in Bloch [3], that is

\[ \dot{\alpha}_1 = -\frac{l\alpha_1\alpha_2}{J + ml^2} \]
\[ \dot{\alpha}_2 = \frac{ml\alpha_2^3}{(J + ml^2)^2}. \]  

(4.29)

Suppose Equations (4.25) and (4.26) are combined into a single equation. This is accomplished by multiplying Equations (4.25) by \( mv \) and (4.26) by \(- (J + ml^2)\dot{\theta} \). As a result, the following relationship is obtained

\[ mv\ddot{v} = -(J + ml^2)\dot{\theta}\ddot{\theta} - T_d\dot{\theta} = mlv\dot{\theta}^2. \]

It then follows that

\[ mv\ddot{v} + (J + ml^2)\dot{\theta}\ddot{\theta} + T_d\dot{\theta} = 0. \]  

(4.30)

Integrating Equation (4.30) results in

\[ \frac{1}{2}mv^2 + \frac{1}{2}(J + ml^2)\dot{\theta}^2 + \int T_d\dot{\theta} \, dt = C \]  

(4.31)

where \( C \) is an integration constant. Equation (4.31) is very similar to the law of energy conservation, but is not in the true sense. Because \( v \) is the velocity of the wheel and not the center of mass, \( \frac{1}{2}mv^2 \) is not the translational energy. In addition \( J + ml^2 \) is the moment of inertia about the wheel by the parallel axis theorem. Then \( \frac{1}{2}(J + ml^2)\dot{\theta}^2 \) could be thought of as the rotational energy if the body was rotating about the wheel. Only if \( l = 0 \) would Equation (4.31) be valid as the conservation of
energy. Doing so would result in both Equations (4.25) and (4.26) to be equal to zero. The remaining term, $\int T_d \dot{\theta}$, represents the work of the input applied to the system.

Once again, consider the classical case of the Chaplygin sleigh, allow $T_d = 0$. Equation (4.31) then becomes an equation of an ellipse, such that

$$\frac{1}{2} m v^2 + \frac{1}{2} (J + ml^2) \dot{\theta}^2 = C.$$ 

If depicted on a phase portrait, then the trajectory follows an arc of an ellipse. For more information on the trajectory, refer back to Equations (4.25) and (4.26).

Equation (4.25) states that $\dot{v}$ must always be nonnegative, and is only zero when $\dot{\theta}$ is also zero. Therefore, if $v$ is initially positive, then it will remain positive. If $v$ is positive, then it is shown in Equation (4.26) that $\ddot{\theta}$ will have the opposite sign of $\dot{\theta}$ and is only zero when $\dot{\theta}$ is also zero. Therefore $\dot{\theta}$ converges to zero. If $\dot{\theta} = 0$, then Equation (4.25) states that $\dot{v} = 0$ and thus $v$ is a constant. As a result, the Chaplygin sleigh is driven to steady state.

Thus, the trajectory on the phase portrait begins at the initial point and follows an elliptical path to the positive $v$ axis. For example, consider the case when $m = J + ml^2$ such that the ellipse becomes a circle. Let the initial conditions be $\dot{\theta}_0 = 0$ and $v_0 = 1$. Then it is estimated that the final value will be $\dot{\theta}_f = 1$ and $v_f = 0$. After simulating the system for a considerable time, the phase portrait is given below.

![Phase portrait of Chaplygin sleigh without input.](image)

Fig. 4.2. Phase portrait of Chaplygin sleigh without input.
As shown, the phase portrait follows what the path described earlier. The phase portrait also agrees with the phase portrait given by Bloch [3].

The Chaplygin sleigh is also an example of a system that provides no useful information if linearized about the equilibrium point of zero. To do so would result in equations that are equal to zero. Although some analysis has been carried out for the case with no input, it is somewhat limited. The equations of motion with input are analyzed in more detail with the use of Lyapunov stability theory in the next section.

4.4 Constant Input Without Friction

Suppose that a constant input is applied to the Chaplygin sleigh in the case where there is no friction. An important result is that \( \dot{\theta} \) goes to zero. For notation purposes, let \( \dot{\theta} \) be denoted as \( \omega \) and the constant \( T_d \) be denoted as \( u \). Equations (4.25) and (4.26) are rewritten in terms of \( \omega \)

\[
\dot{v} = l\omega^2 \tag{4.32}
\]

\[
(J + m l^2)\dot{\omega} = -ml\omega v - u. \tag{4.33}
\]

By differentiating Equation (4.33), the resulting equation is

\[
(J + m l^2)\ddot{\omega} = -ml(\dot{v}\dot{\omega} + \omega\ddot{v}). \tag{4.34}
\]

Equation (4.32) can be substituted into the second right hand term of Equation (4.33) to result in

\[
(J + m l^2)\ddot{\omega} = -ml(v\dot{\omega} + l\omega^3). \tag{4.35}
\]

Equation (4.35) can be rearranged and multiplied by \( \dot{\omega} \) to produce

\[
(J + m l^2)\dot{\omega}\ddot{\omega} + m l^2\dot{\omega}\omega^3 = -mlv\dot{\omega}. \tag{4.36}
\]

Consider the Lyapunov function\(^1\) given by

\[
V(\omega, \dot{\omega}) = \frac{1}{2}(J + m l^2)\dot{\omega}^2 + \frac{1}{4}m l^2\dot{\omega}\omega^4. \tag{4.37}
\]

\(^1\)found due to Professor Martin Corless of Purdue University
The function is positive definite and therefore is a valid Lyapunov function. The derivative of the Lyapunov function is

\[
\frac{dV}{dt} = (J + ml^2)\dot{\omega}\ddot{\omega} + ml^2\dot{\omega}\dot{\omega}^3
\]

and matches the lefthand side of Equation (4.35). Therefore,

\[
\frac{dV}{dt} = -mlv\dot{\omega}^2
\]  \hspace{1cm} (4.38)

is also true. If the initial condition \( v(0) = 0 \) is imposed, then Equation (4.32) states that \( \dot{v} \geq 0 \) and hence \( v \) is never negative. If the input \( u \) is not zero, then Equation (4.33) implies that \( \omega \) cannot be zero for all time. As a result, \( \dot{v} \) cannot be zero for all time and \( v \) is greater than zero for any nonzero input. The right hand side of Equation (4.38) is less than zero, with exception to the equilibrium. In other words,

\[
\dot{V}(\omega, \dot{\omega}) < 0 \quad \forall (\omega, \dot{\omega}) \neq 0
\]

\[
\dot{V}(\omega, \dot{\omega}) = 0 \quad \text{iff} \quad (\omega, \dot{\omega}) = 0.
\]

Because \( v \) is always positive, \( v(t) = 0 \) for \( t > 0 \) is no longer possible, \( v = 0 \) no longer belongs to the set of solutions given by LaSalle’s Theorem. However, \( \omega \) and \( \dot{\omega} \) does converge to zero as \( t \) tends to infinity by the theory of Lyapunov stability. As a consequence of Equation (4.32), \( \dot{v} \) will also tend to zero as time goes to infinity.

Recall that \( \dot{\omega} \) and \( \omega \) is shown to converge to zero for a constant input. Solving for \( \omega \) in Equation (4.32) and substituting into Equation (4.33) results in

\[
(J + ml^2)\dot{\omega} = -ml\sqrt{\frac{\dot{v}}{l}v} - u.
\]  \hspace{1cm} (4.39)

Because \( \dot{\omega} \) converges to zero for large \( t \). The lefthand side of Equation (4.39) asymptotically becomes zero such that

\[
(J + ml^2)\dot{\omega} \approx 0.
\]

Then the righthand side of Equation (4.39) asymptotically becomes zero such that

\[
-ml\sqrt{\frac{\dot{v}}{l}v} - u \approx 0
\]
which becomes
\[ \dot{v} v^2 \approx \frac{u^2}{m^2 l}. \]  
(4.40)

Through implicit integration, the following approximation for \( v(t) \) is obtained
\[ v(t) \approx \left( \frac{3u^2}{m^2 l} (t - t_s) \right)^{\frac{1}{3}} \]  
(4.41)

where \( t_s \) is an integration constant.

By substituting Equation (4.41) into Equation (4.32) and solving for \( \omega \), it can be shown for large \( t \) that
\[ \dot{\theta}(t) \approx - \left( \frac{u}{3ml^2 (t - t_s)} \right)^{\frac{1}{3}} \]  
(4.42)
\[ \theta(t) \approx - \left( \frac{9u}{8ml^2 (t - t_s)^2} \right)^{\frac{1}{3}} + \theta_s \]  
(4.43)

where \( \theta_s \) is another integration constant. Because \( (t - t_s)^{-\frac{1}{3}} \to 0 \), \( \dot{\theta} \) converges to zero as \( t \) tends to infinity.

From Equations (4.14), the velocity of the center of mass in the \( x \) and \( y \) directions can be written in the form of
\[ \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} v \\ \dot{\theta} \end{bmatrix}. \]  
(4.44)

By substituting in Equations (4.41) and (4.42), Equation (4.44) results in
\[ \begin{aligned} \dot{x}(t) &\approx \left( \frac{1}{m^2 l (t - t_s)} \right)^{\frac{1}{3}} \left[ (3u^2 (t - t_s)^2)^{\frac{1}{3}} \cos(\theta(t)) + \frac{1}{3} (9uml^2)^{\frac{1}{3}} \sin(\theta(t)) \right] \\ \dot{y}(t) &\approx \left( \frac{1}{m^2 l (t - t_s)} \right)^{\frac{1}{3}} \left[ (3u^2 (t - t_s)^2)^{\frac{1}{3}} \sin(\theta(t)) - \frac{1}{3} (9uml^2)^{\frac{1}{3}} \cos(\theta(t)) \right]. \end{aligned} \]  
(4.45)

Equations (4.45) can be integrated to become
\[ \begin{aligned} x(t) &\approx \left( \frac{1}{m} \right)^{\frac{1}{3}} \left[ - (9u (t - t_s)^2)^{\frac{1}{3}} \sin(\theta(t)) + 3 (ml^2)^{\frac{1}{3}} \cos(\theta(t)) \right] + x_s \\ y(t) &\approx \left( \frac{1}{m} \right)^{\frac{1}{3}} \left[ (9u (t - t_s)^2)^{\frac{1}{3}} \cos(\theta(t)) + 3 (ml^2)^{\frac{1}{3}} \sin(\theta(t)) \right] + y_s. \end{aligned} \]  
(4.46)
If \( x_s = 0 \) and \( y_s = 0 \), then the distance \( d \) from the origin, can be calculated independent of \( \theta \). In this case, the orthogonal nature of the terms of Equations (4.46) yield

\[
d(t) = \left( \frac{l}{m} \right)^{\frac{1}{3}} \left[ 3 \left( 3u^2 (t - t_s)^4 \right)^{\frac{1}{3}} + 9 \left( m^2 l^4 \right)^{\frac{1}{3}} \right]^{\frac{1}{3}}. \tag{4.47}
\]

Note the only parameters in the equations of motion (4.41) to (4.47) are \( m, l \), and \( u \). Refer back to conservation equation in Equation (4.31) for the case of a constant input. If all the states are initially zero, then the constant of integration \( C \) will also be zero. Therefore, Equation (4.31) becomes

\[
\frac{1}{2}mv^2 + \frac{1}{2} (J + ml^2) \dot{\theta}^2 + u\theta = 0. \tag{4.48}
\]

Suppose the asymptotic approximations of Equations (4.41) - (4.43) are applied. It is known that \( \dot{\theta} \) converges to zero, while \( v \) and \( \theta \) does not. Therefore, Equation (4.48) reduces to

\[
\frac{1}{2}mv^2 + u\theta \approx 0 \tag{4.49}
\]

for large \( t \). If Equations (4.41) and (4.43) are substituted in, Equation (4.49) becomes

\[
\frac{1}{2}m \left( \frac{3u^2}{m^2 l} (t - t_s) \right)^{\frac{2}{3}} + u \left( - \left( \frac{9u}{8ml^2} (t - t_s)^2 \right)^{\frac{1}{3}} + \theta_s \right) \approx 0. \tag{4.50}
\]

If Equation (4.49) is true, then the lefthand side of Equation (4.50) is simplifies to zero if \( \theta_s = 0 \). As a result, there is one less unknown in the asymptotic equations under zero initial conditions.

To simulate the equations of motion without friction, let the parameters be

\[
m = 2 \text{ kg} \]
\[
l = 0.2 \text{ m} \]
\[
J = 0.025 \text{ kg}^*\text{m}^2 \]
\[
J_d = 0.008 \text{ kg}^*\text{m}^2 \tag{4.51}
\]

and the input set constant at

\[
u = 4 \text{ Nm}. \tag{4.52}
\]
With no prior information, let $t_s = 0$. Figures 4.3 - 4.11 are the plots of the simulated solution versus the asymptotic solution derived from Equations (4.41) to (4.47) over 20 seconds.

Fig. 4.3. Simulation of $v$ versus time of constant input without friction.

Fig. 4.4. Simulation of $\dot{\theta}$ versus time of constant input without friction.

Fig. 4.5. Simulation of $\theta$ versus time of constant input without friction.
Fig. 4.6. Simulation of $\dot{x}$ versus time of constant input without friction.

Fig. 4.7. Simulation of $\dot{y}$ versus time of constant input without friction.

Fig. 4.8. Simulation of $x$ versus time of constant input without friction.

Fig. 4.9. Simulation of $y$ versus time of constant input without friction.
As seen, the asymptotic solutions are very close to the simulated results. The greatest differences between the simulated and asymptotic results occur during the transient phase, but converges quickly in this example. For example, it is shown in Figure 4.4 that the simulation of $\omega = \dot{\theta}$ reaches a minimum of about -7.4 rad/s at about 0.26 seconds before increasing, while the asymptotic solution begins from negative infinity. However, the two plots converge relatively quickly at around 4 seconds. The others figures demonstrates this idea of convergence after the some transient behavior.

Due to the sinusoidal nature of the $x$ and $y$ positions and velocities, the corresponding shifts can be estimated. The shifts account for the periodicity are seen in the simulated distance function of Figure 4.10, but the result still follows the shape of the asymptotic solution nicely. What is of most interest is the $xy$ position of the Chaplygin sleigh, as it undergoes an outward spiral motion from the origin, as shown in Figure 4.11.
4.5 Constant Heading Without Friction

Suppose a feedback controller is applied to the Chaplygin sleigh such that it is driven in the direction of a fixed angle $\theta_r$ from the $x$ axis. If a proportional and derivative (PD) feedback controller is applied to Equation (4.26) such that

$$T_d = k_p(\theta - \theta_r) + k_d\dot{\theta}$$

where $k_p$ and $k_d$ are greater than zero, then the resulting two differential equations are

$$\dot{\theta} = l\dot{\omega}^2$$

$$(J + ml^2)\ddot{\theta} = -(mlv + k_d)\dot{\theta} + k_p(\theta - \theta_r)$$

To prove stability, the Lyapunov function is applied again. Similar to the constant input case, the derivatives of Equation (4.54) are taken

$$\dot{v} = 2l\omega \dot{\omega}$$

$$(J + ml^2)\ddot{\omega} = -(mlv + k_d)\dot{\omega} - ml\dot{v}\omega - k_p\omega$$

where $\dot{\theta}$ is denoted by $\omega$ again. As before, multiply the first equation in Equation (4.55) by $\dot{v}$ and the second by $\dot{\omega}$. Doing so allows the two equations to be combined and reorganized into

$$(J + ml^2)\ddot{\omega}\dot{\omega} + \frac{1}{2}mv\ddot{\omega} + k_p\omega\dot{\omega} = -(mlv + k_d)\dot{\omega}^2.$$  

Motivated by (4.57), the Lyapunov function is found to be

$$V(\dot{\omega}, \omega, \dot{v}) = \frac{1}{2}(J + ml^2)\dot{\omega}^2 + \frac{1}{4}mv^2 + \frac{1}{2}k_p\omega^2.$$  

The derivative of the Lyapunov function is equal to right hand side of Equation (4.57), that is

$$\frac{dV}{dt} = -(mlv + k_d)\dot{\omega}^2.$$  


As previously shown, $v$ is always greater than zero after the initial state for nonzero input. The gains $k_p$ and $k_d$ must also be greater than zero for the Lyapunov function to be well defined. Note that this Lyapunov functions shows that $\dot{v}$ converges to zero (which was not true for a constant input). The states $\ddot{\theta}$ and $\dot{\theta}$ also converges to zero. Because these states all converge to zero, the input converges to zero as well, as required by the equations of motion. From Equation (4.53), it is then shown that the heading angle goes to the desired angle, or

$$\theta \to \theta_r.$$ 

Using the same parameters from the previous simulation, as stated in Equation (4.51), simulate the application of the PD controller with the gains set at

$$k_p = 0.2 \text{ N*m/rad}$$

$$k_d = 0.1 \text{ N*m*s/rad}$$

and a desired heading angle of

$$\theta_r = -45^\circ.$$ 

If the simulation is allowed to run for 15 seconds, then the following plots are obtained.

![Fig. 4.12. Simulation of $v$ versus time for heading angle controller.](image-url)
As seen in Figures 4.13 and 4.14, \( \theta \) went to a constant of about negative 45 degrees as desired, and the sleigh goes in a straight line, which is shown in Figure 4.16. However, the input also tends to zero such that the sleigh coasts at a constant \( v \) of about 0.1 m/s, as read from Figure 4.12. If friction were to be added, it is expected
that \( v \) would no longer be constant and be driven to zero as energy is dissipated from the system. Therefore, the addition of friction will be investigated next.

### 4.6 Equations of Motion With Friction

In this section friction is added to the Chaplygin sleigh. It is assumed that there are two types of friction that exist between the sleigh and the floor, translation and rotational. Assume the dissipation energy lost to each type of friction is expressed as

\[
D_{tr} = \frac{1}{2} \zeta_d (\dot{x}^2 + \dot{y}^2) \quad \text{and} \quad D_{rot} = \frac{1}{2} \zeta_r \dot{\theta}^2.
\]

The Rayleigh dissipation function, \( D \), is the sum of translation and rotational dissipation energy

\[
D = \frac{1}{2} \zeta_d (\dot{x}^2 + \dot{y}^2) + \frac{1}{2} \zeta_r \dot{\theta}^2
\]  

(4.60)

In this case, Lagrange’s equations become

\[
\frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} + \frac{\partial D}{\partial \dot{q}} = M_q \ddot{q} - \beta_q.
\]  

(4.61)

The mass matrix remains the same as in Equation (4.10), that is

\[
M_q = \begin{bmatrix}
m & 0 & 0 & 0 \\
0 & m & 0 & 0 \\
0 & 0 & J + J_d & J_d \\
0 & 0 & J_d & J_d
\end{bmatrix}
\]

However, there are leftover terms, where

\[
\beta_q = \begin{bmatrix}
-\zeta_d & 0 & 0 & 0 \\
0 & -\zeta_d & 0 & 0 \\
0 & 0 & -\zeta_r & 0 \\
0 & 0 & 0 & 0
\end{bmatrix} \dot{q}.
\]

The constrained equations of motion are given once again by

\[
M_q \ddot{q} = \beta_q + H^* \lambda + F_q T_d
\]
where
\[
H = \begin{bmatrix}
\sin(\theta) & -\cos(\theta) & l & 0
\end{bmatrix}
\]
\[
F_q = \begin{bmatrix}
0 & 0 & 0 & 1
\end{bmatrix}^*.
\]

Using \( \Phi^*H^* = 0 \), we obtain \( \Phi^*M_q\ddot{q} = \Phi^*\beta_q + \Phi^*F_qT_d \). Note that the constraint equations are the same as before, such that \( \Phi \) remains the same as the case with no friction. Recall that \( \dot{q} = \Phi\dot{p} \) and \( \ddot{q} = \Phi\ddot{p} + \Phi\dot{p} \). Hence,
\[
(\Phi^*M_q\Phi)\ddot{p} = \Phi^*\beta_q - \Phi^*M_q\Phi\dot{p} + \Phi^*F_qT_d.
\]
or
\[
\begin{bmatrix}
m & 0 & 0 \\
0 & J + J_d + ml^2 & J_d \\
0 & J_d & J_d
\end{bmatrix} \ddot{p} = \begin{bmatrix}
-\zeta_d & ml\dot{\theta} & 0 \\
-ml\dot{\theta} & -(\zeta_r + \zeta_d l^2) & 0 \\
0 & 0 & 0
\end{bmatrix} \ddot{p} + \begin{bmatrix}
0 \\
0 \\
T_d
\end{bmatrix}.
\]

Taking the inverse of the mass matrix results in the equations of motion
\[
\dot{v} = -\frac{\zeta_d v}{m} + l\dot{\theta}^2 
\tag{4.62}
\]
\[
\ddot{\theta} = -\frac{(mlv + \zeta_r + \zeta_d l^2)\dot{\theta} + T_d}{J + ml^2} 
\tag{4.63}
\]
\[
\ddot{\psi} = \frac{(Jdmlv + \zeta_r + \zeta_d l^2)\dot{\theta} + (J + J_d + ml^2)T_d}{(J + ml^2)J_d}. 
\tag{4.64}
\]

Taking \( \zeta_d = 0 \) and \( \zeta_r = 0 \) reduces the equations of motion in Equations (4.62) - (4.64) to the frictionless case in Equations (4.25) - (4.27).

### 4.7 Constant Input With Friction

As with the case with no friction, suppose that a constant input is applied to the system. As before, let \( \omega = \dot{\theta} \) and \( u = T_d \). Rearranging Equations (4.62) and (4.63), the equations of motion with respect to \( v \) and \( \omega \) can be written as
\[
m\dot{v} + \zeta_d v = ml\omega^2 
\tag{4.65}
\]
\[
(J + ml^2)\dot{\omega} + (\zeta_r + \zeta_d l^2)\omega + ml\omega v = -u. 
\tag{4.66}
\]
By taking the time derivative of both Equations (4.65) and (4.66), the following are obtained

\[ m\ddot{v} + \zeta_d \dot{v} = 2ml\omega \dot{\omega} \tag{4.67} \]

\[ (J + ml^2)\ddot{\omega} + (\zeta_r + \zeta_d^2)\dot{\omega} + ml\dot{\omega}v + ml\omega \dot{v} = 0. \tag{4.68} \]

By multiplying Equation (4.68) by \( \dot{\omega} \), Equation (4.67) can be substituted in as the last parenthetical term on the left hand side.

\[ (J + ml^2)\ddot{\omega}\dot{\omega} + (\zeta_r + \zeta_d^2 + mlv)\dot{\omega}^2 + \frac{1}{2}(m\ddot{v} + \zeta_d \dot{v})\dot{v} = 0 \tag{4.69} \]

Rearranging the terms results in

\[ (J + ml^2)\ddot{\omega}\dot{\omega} + \frac{1}{2}m\ddot{v}\dot{v} = -(\zeta_r + \zeta_d^2 + mlv)\dot{\omega}^2 - \frac{1}{2}\zeta_d \dot{v}^2. \tag{4.70} \]

Integrating the left hand side of Equation (4.70) with respect to time leads to the desired Lyapunov function

\[ V(\dot{\omega}, \dot{v}) = \frac{1}{2}(J + ml^2)\dot{\omega}^2 + \frac{1}{4}m\dot{v}^2. \tag{4.71} \]

Notice that the right hand side of Equation (4.70) is equal to the derivative of the Lyapunov function \( V \) such that

\[ \frac{dV}{dt} = -(\zeta_r + \zeta_d^2 + mlv)\dot{\omega}^2 - \frac{1}{2}\zeta_d \dot{v}^2. \tag{4.72} \]

To apply the Lyapunov stability theorem, \( v \) must shown to be positive. Equation (4.65) may be solved as a first order linear equation with the following solution

\[ v(t) = e^{-\frac{\zeta_d}{m}t}v(0) + e^{-\frac{\zeta_d}{m}t}\int_0^t e^{\frac{\zeta_d}{m}\tau}l\omega(\tau)^2 d\tau. \tag{4.73} \]

Assuming \( v(0) \) equals zero, \( v \) will always be greater than zero after the initial state due to the \( \omega^2 \) as stated before. By applying the Lyapunov stability theory, the functions \( \dot{v} \) and \( \dot{\omega} \) are both shown to converge to zero. Referring back to Equation (4.65) and substituting \( \dot{v} = 0 \) shows that for large \( t \)

\[ v \approx \frac{ml}{\zeta_d} \omega^2. \tag{4.74} \]
With knowledge of $\dot{\omega}$ and $\dot{v}$ being driven to zero, Equation (4.74) is substituted into Equation (4.66) to show that for large $t$

$$\left(\frac{m^2l^2}{\zeta_d}\right)\omega^3 + (\zeta_r + \zeta_d l^2)\omega + u \approx 0.$$  \hspace{1cm} (4.75)

Equation (4.75) is in the form of a depressed cubic and is known to have one real root. From Cardano’s method, the real root for a general depressed cubic is given by

$$x^3 + ax + b = 0$$

$$x = \left(-\frac{b}{2} + \sqrt{\frac{b^2}{4} + \frac{a^3}{27}}\right)^{\frac{1}{3}} + \left(-\frac{b}{2} - \sqrt{\frac{b^2}{4} + \frac{a^3}{27}}\right)^{\frac{1}{3}}.$$ \hspace{1cm} (4.76)

Applying the above solution to Equation (4.75) yields the limit

$$\omega_\infty = \lim_{t\to\infty} z(t)$$

$$= \left(-\frac{\zeta_d u}{2m^2l^2} + \sqrt{\frac{\zeta_d^2 u^2}{4m^2l^4} + \frac{(\zeta_r \zeta_d + \zeta_d^2 l^2)^3}{27m^6l^6}}\right)^{\frac{1}{3}} + \left(-\frac{\zeta_d u}{2m^2l^2} - \sqrt{\frac{\zeta_d^2 u^2}{4m^2l^4} + \frac{(\zeta_r \zeta_d + \zeta_d^2 l^2)^3}{27m^6l^6}}\right)^{\frac{1}{3}}.$$ \hspace{1cm} (4.77)

When enough time has passed to assume that $\omega$ becomes constant, $\theta$ can be approximated as

$$\theta(t) \approx \int \omega_\infty dt = \omega_\infty t + \theta_s$$ \hspace{1cm} (4.78)

where $\theta_s$ is an integration constant. Recall that from Equation (4.44) that the $x$ and $y$ velocities of the center of mass and are given again as

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} \omega \\ \dot{l}\theta \end{bmatrix}$$ \hspace{1cm} (4.79)

with the magnitude of the total velocity found as

$$\sqrt{\dot{x}^2 + \dot{y}^2} = \sqrt{v^2 + l^2\dot{\theta}^2}.$$ 

Because $v$ and $\dot{\theta}$ both converge constant, the magnitude also converges to a constant. In addition the velocities in the $x$ and $y$ directions are orthogonal to each other by
definition. When Equation (4.78) is applied, Equation (4.79) can be expressed as the equations of a circle with a velocity of $v_c$ and some phase shift $\phi_s$.

\[
\dot{x} \approx -v_c \sin(\omega_\infty t + \phi_s) \\
\dot{y} \approx v_c \cos(\omega_\infty t + \phi_s) \\
v_c = \sqrt{v_\infty^2 + l^2 \omega_\infty^2} = \frac{\omega_\infty l}{\zeta_d} \sqrt{m^2 \omega_\infty^2 + \zeta_d^2}.
\]

(4.80)

The phase shift $\phi_s$ is due to both the combination of the two trigonometric terms into a single trigonometric term in Equation (4.79) and the integration constant $\theta_s$.

Integrating Equations (4.80) results in the position equations

\[
x = \frac{v_c}{\omega_\infty} \cos(\omega_\infty t + \phi_s) + x_c = r_c \cos(\omega_\infty t + \phi_s) + x_c \\
y = \frac{v_c}{\omega_\infty} \sin(\omega_\infty t + \phi_s) + y_c = r_c \sin(\omega_\infty t + \phi_s) + y_c \\
r_c = \frac{v_c}{\omega_\infty} = \frac{l}{\zeta_d} \sqrt{m^2 \omega_\infty^2 + \zeta_d^2}.
\]

(4.81)

Equations (4.81) form a circle of radius $r_c$ about some center $(x_c, y_c)$. Although the radius can be calculated from the given parameters, the center can only be known if the history of the velocities during the transient phase are known.

Consider the same parameters as the last example from Equation (4.51). In addition, set the friction coefficients as

\[
\zeta_d = 0.1 \text{ kg/s} \\
\zeta_r = 0.1 \text{ kg/s} * \text{ m}^2
\]

The resulting estimated values of $\omega_\infty$ and $v_\infty$ are calculated as

\[
v_\infty = 7.1958 \quad \text{and} \quad \omega_\infty = -1.3412.
\]

Figures 4.17 - 4.19 show the results of the following case compared against the asymptotic solution over 30 seconds.
As seen in Figures 4.17 and 4.18, $v$ and $\omega$ reach a steady state value after a relatively short amount of time, and the asymptotic estimate is very accurate. An important observation from Figure 4.19 is that the sleigh spirals outward until settling into a circle. If there was no friction, the sleigh would spiral out indefinitely, as well as approaching an infinite velocity, as shown previously. Although the center is slightly off from the origin, the radius appears to be very accurate.
4.8 Heading Controller with Friction

Suppose the feedback controller that drives the Chaplygin sleigh in the direction of a fixed constant angle $\theta_r$ is applied to the friction case. From Equation (4.63), if the same PD feedback controller is applied such that

$$T_d = k_p(\theta - \theta_r) + k_d\dot{\theta}$$

(4.82)

where $k_p$ and $k_d$ are greater than zero. The resulting two differential equations are

$$m\ddot{v} = -\zeta_d v + ml\dot{\theta}^2$$

$$\left(J + ml^2\right)\ddot{\omega} = -(mlv + l^2\zeta_d + \zeta_r + k_d)\dot{\theta} + k_p(\theta - \theta_r)$$

(4.83)

The Lyapunov function is applied in a similar fashion as before. The derivatives of Equation (4.83) are taken, such that

$$m\ddot{v} = -\zeta_d \dot{v} + 2ml\dot{\omega}\omega$$

$$\left(J + ml^2\right)\ddot{\omega} = -(\zeta_d l^2 + \zeta_r + mlv + k_d)\dot{\omega} - ml\dot{v}\omega - k_p\omega$$

(4.84)

(4.85)

where $\dot{\theta}$ is replaced by $\omega$ again for notation purposes. As before, multiply the first equation in Equation (4.84) by $\dot{v}$ and the second by $\dot{\omega}$. The two equations can be combined and reorganized into

$$\left(J + ml^2\right)\ddot{\omega}\dot{\omega} + \frac{1}{2}m\dddot{v}\dot{v} + k_p\omega\dot{\omega} = -(\zeta_d l^2 + \zeta_r + mlv + k_d)\dot{\omega}^2 - \frac{1}{2}\zeta_d \dot{v}^2.$$  

(4.86)

The corresponding Lyapunov function is

$$V(\dot{\omega}, \omega, \dot{v}) = \frac{1}{2}(J + ml^2)\dot{\omega}^2 + \frac{1}{4}m\dddot{v}^2 + \frac{1}{2}k_p\omega^2.$$  

(4.87)

The derivative of the Lyapunov function is equal to right hand side of Equation (4.86), that is

$$\frac{dV}{dt} = -(\zeta_d l^2 + \zeta_r + mlv + k_d)\dot{\omega}^2 - \frac{1}{2}\zeta_d \dot{v}^2.$$  

(4.88)

From previous arguments, if $v(0) = 0$, $v$ is always greater than zero after the initial state for all nonzero input. The gains $k_p$ and $k_d$ must also be greater than zero for the Lyapunov function to be well defined.
As desired, the states $\dot{v}, \ddot{\theta},$ and $\dot{\theta}$ converge to zero. As before, the input converges to zero as well, as required by the equations of motion. From Equation (4.82), it is then shown that the heading angle goes to the desired angle, or

$$\theta \to \theta_r.$$ 

As a result of the translational damping coefficient, $v$ also converges to zero. This is because the input is driven to zero as the sleigh aligns itself in the right direction. From that point, the sleigh coasts until it comes to a stop due to friction.

Suppose, under the same system parameters as before in Equation (4.51), the following command heading angle

$$\theta_r = -45^\circ$$

is desired, with the gains set to

$$k_p = 0.2 \text{ Nm/rad}$$

$$k_d = 0.1 \text{ Nms/rad.}$$

The simulation is ran for 15 seconds and the results of both with and without friction are plotted in Figures 4.20 - 4.24.

![Fig. 4.20. Simulation of $v$ versus time for constant heading with friction.](image-url)
As seen in Figure 4.22, the desired heading angle is achieved. Thus the input is driven to zero, which is demonstrated in Figure 4.23. Figure 4.20 shows that $\dot{v}$ also decays to zero as expected. As a result, the Chaplygin sleigh is driven to a significantly smaller distance than the case without friction, as shown in Figure 4.24.
This behavior was not an issue in the previous no friction case, but now is if the sleigh is desired to reach some waypoint some considerable distance away.

Suppose that a sinusoidal input is also added to the PD feedback controller such that the input does not decay to zero. The input is given as

\[ T_d = A_u \sin(\omega_u t) + k_p(\theta - \theta_r) + k_d \dot{\theta}. \]  

(4.90)

The addition of the sinusoidal term complicates the behavior of the system, and no Lyapunov function has yet been shown to be effective. However, suppose the controller is applied with the sine wave parameters

\[ A_u = 0.5 \text{ N} \]
\[ \omega_u = 1.5 \text{ rad/s} \]  

(4.91)

and simulated for 15 seconds. The controller is then compared to the previous PD controller in Figures 4.25 - 4.29 with the same gains.

![Graph](image)

**Fig. 4.25.** Simulation of \( v \) versus time for sine and heading angle controller.
As expected, the sine controller results in a sinusoidal motion for each of the states, while the heading angle controller has the states tend to the desired value. However, what is the most important difference between the two controllers is that $v$ does not tend to zero for the sine controller, and instead averages out to about 1.4 m/s. This
is critical as the $y$ versus $x$ plot shows that the heading controller does not advance the sleigh very far, while the sine controller is able to move in the desired direction over a much larger direction. However, the average of $v$ was not directly controlled and changes under different parameters. Although the sine controller shows promise, it is not yet viable as a predictable controller until further investigated and analyzed.

4.9 Sinusoidal Plus Constant Input With Friction

As shown, the additional of a sinusoidal component to the input results in very interesting behavior of the Chaplygin sleigh. Let us simulate other possible scenarios with a sinusoidal component in the input. Consider the case where a sinusoidal and constant input is applied such that

$$T_d = A_u \sin(\omega_u t) + u_s$$  \hspace{1cm} (4.92)

where $A_u$ is the amplitude, $\omega_u$ is the angular velocity, and $u_s$ is the average input. By running a few simulations it was found that the solutions appeared to settle into a closed trajectory. The trajectory would be periodic, where $v(t) = v(t + T)$ and $\omega(t) = \omega(t + T)$ for some minimum period $T$, and thus would be a limit cycle. If true, then $\dot{v}(t) = \dot{v}(t + T)$ and $\dot{\omega}(t) = \dot{\omega}(t + T)$ must also be true for the limit cycle to exist. If the equations of motion are notated such that

$$\dot{v} = f(v, \omega)$$  \hspace{1cm} (4.93)
$$\dot{\omega} = g(v, \omega, u)$$  \hspace{1cm} (4.94)

then the conditions for the limit cycle are only met if $u(t) = u(t + T)$. This is true if $T = 2\pi/\omega_u$ for the given input in Equation (4.92). Suppose the limit cycle can be expressed as a Fourier series of the form

$$v = \sum_{n=1}^{\infty} A_{vn} \sin(n\omega_u t + \phi_{vn}) + v_s$$
$$\theta = \sum_{n=1}^{\infty} A_{\theta n} \sin(n\omega_u t + \phi_{\theta n}) + \theta_s + \dot{\theta}_s t$$  \hspace{1cm} (4.95)
where $A_n$ is the amplitude and $\phi_n$ is the phase angle. Note that $v(t) = v(t + T)$ and $\omega(t) = \omega(t + T)$ is true for Equation (4.95). From the fast Fourier transforms of the states, it was found that $\omega_u$ was the most significant of the angular velocities. Based on this observation, propose an estimated solution in which the Fourier series is truncated to

$$v = A_v \sin(\omega_u t + \phi_v) + v_s$$

$$\theta = A_\theta \sin(\omega_u t + \phi_\theta) + \dot{\theta}_s t. \quad (4.96)$$

If $A_u = 0$, then the case is reduced to constant input with friction, and thus $A_v = 0$ and $A_\theta = 0$, and the mean values have been previously solved as

$$\dot{\theta}_s = \left( -\frac{\zeta_d u_s}{2m^2 l^2} + \sqrt{\frac{\zeta_d^2 u_s^2}{4m^4 l^4} + \left(\frac{\zeta_r \zeta_d + \zeta_d^2 l^2}{27 m^6 l^6}\right)} \right) + \left( -\frac{\zeta_d u_s}{2m^2 l^2} - \sqrt{\frac{\zeta_d^2 u_s^2}{4m^4 l^4} + \left(\frac{\zeta_r \zeta_d + \zeta_d^2 l^2}{27 m^6 l^6}\right)} \right) \frac{1}{3}$$

$$v_s = \frac{ml}{\zeta_d} \dot{\theta}_s^2. \quad (4.97)$$

Equations (4.96) are applied to the equations of motion in Equations (4.62) and (4.63) to result in

$$A_v \omega_u \cos(\omega_u t + \phi_v) = -\frac{\zeta_d}{m} (A_v \sin(\omega_u t + \phi_v) + v_s) + l (A_\theta \omega_u \cos(\omega_u t + \phi_\theta) + \dot{\theta}_s) \frac{2}{J + ml^2}$$

$$-A_\theta \omega_u^2 \sin(\omega_u t + \phi_\theta) = -\frac{ml}{J + ml^2} (A_v \sin(\omega_u t + \phi_v) + v_s) (A_\theta \omega_u \cos(\omega_u t + \phi_\theta) + \dot{\theta}_s) + \frac{ml}{J + ml^2} + \frac{(l^2 \zeta_d + \zeta_r) (A_\theta \omega_u \cos(\omega_u t + \phi_\theta) + \dot{\theta}_s) + (A_u \sin(\omega_u t) + u_s)}{J + ml^2}. \quad (4.98)$$

Suppose the results from the constant input case were applied such that the following was true

$$0 = -\frac{\zeta_d}{m} v_s + l \dot{\theta}_s^2$$

$$0 = -\frac{ml v_s \dot{\theta}_s + (l^2 \zeta_d + \zeta_r) \dot{\theta}_s + u_s}{J + ml^2}. \quad (4.99)$$

Then Equation (4.99) can be applied to simplify Equations (4.98). In order to solve for the amplitudes and phase shifts, the squared and product terms involving trigono-
metric terms, \( \frac{1}{2}l(A_\theta \omega_u \cos(\omega_u t + \phi_\theta))^2 \) and \( ml(A_v \sin(\omega_u t + \phi_v))(A_\theta \omega_u \cos(\omega_u t + \phi_\theta)) \), are ignored for now. The two arising equations reduce to

\[
0 = A_v \omega_u \cos(\omega_u t + \phi_v) - \frac{\zeta_d}{m} A_v \sin(\omega_u t + \phi_v) + 2A_\theta \omega_u \dot{\theta}_s \cos(\omega_u t + \phi_\theta) \\
0 = (J + ml^2)A_\theta \omega_u^2 \sin(\omega_u t + \phi_\theta) - (mlv_s + l^2 \zeta_d + \zeta_r)A_\theta \omega_u \cos(\omega_u t + \phi_\theta) - ml\dot{\theta}_s A_v \sin(\omega_u t).
\]  

(4.100)

With four variables \((A_v, A_\theta, \phi_v, \phi_\theta)\), there must be four independent equations. The first two equations are found by setting \(t = 0\) for Equations (4.100). This results in

\[
0 = A_v \omega_u \cos(\phi_v) - \frac{\zeta_d}{m} A_v \sin(\phi_v) + 2A_\theta \omega_u \dot{\theta}_s \cos(\phi_\theta) \\
0 = (J + ml^2)A_\theta \omega_u^2 \sin(\phi_\theta) - (mlv_s + l^2 \zeta_d + \zeta_r)A_\theta \omega_u \cos(\phi_\theta) - ml\dot{\theta}_s A_v \sin(\phi_v).
\]  

(4.101)

The next set are found by setting \(t = \frac{\pi}{2\omega_u}\) such that Equation (4.100) then becomes

\[
0 = -A_v \omega_u \sin(\phi_v) - \frac{\zeta_d}{m} A_v \cos(\phi_v) - 2A_\theta \omega_u \dot{\theta}_s \sin(\phi_\theta) \\
0 = (J + ml^2)A_\theta \omega_u^2 \cos(\phi_\theta) + (mlv_s + l^2 \zeta_d + \zeta_r)A_\theta \omega_u \sin(\phi_\theta) - ml\dot{\theta}_s A_v \cos(\phi_v) - A_u.
\]  

(4.102)

To transform Equations (4.101) and (4.102) into a different form, multiply Equations (4.101) by \(\cos(\phi_v)\) and Equations (4.102) by \(-\sin(\phi_v)\). The next set of equations are found by multiplying Equation (4.101) by \(\sin(\phi_v)\) and (4.102) by \(\cos(\phi_v)\). Doing so acts like a rotation by \(\phi_v\) and simplifies the expressions such that the first and second equations in Equations (4.101) and (4.102) become

\[
0 = -\omega_u \left(A_v - 2A_\theta l \dot{\theta}_s \cos(\phi_v - \phi_\theta)\right) \\
0 = -\frac{\zeta_d}{m} A_v + 2A_\theta l \dot{\theta}_s \sin(\phi_v - \phi_\theta)
\]  

(4.103)

and

\[
0 = -A_\theta \omega_u^2 (J + ml^2) \sin(\phi_v - \phi_\theta) - A_\theta \omega_u (mlv_s + l^2 \zeta_d + \zeta_r) \cos(\phi_v - \phi_\theta) + A_u \sin(\phi_v) \\
0 = A_\theta \omega_u^2 (J + ml^2) \cos(\phi_v - \phi_\theta) - A_\theta \omega_u (mlv_s + l^2 \zeta_d + \zeta_r) \sin(\phi_v - \phi_\theta) - A_v ml \dot{\theta}_s - A_u \cos(\phi_v).
\]  

(4.104)
From Equation (4.103), the sine and cosine of the phase shift differences are solved as

$$\sin(\phi_v - \phi_\theta) = \frac{A_v \zeta_d}{2 A_\theta m \omega_u \dot{\theta}_s}$$

$$\cos(\phi_v - \phi_\theta) = \frac{A_v}{2 A_\theta l \dot{\theta}_s}$$  \hspace{1cm} (4.105)

and the difference is found through the tangent

$$\tan(\phi_v - \phi_\theta) = \frac{\zeta_d}{m \omega_u}$$  \hspace{1cm} (4.106)

which is expressed in known parameters. The quadrant of the difference is known through the signs of the sine and cosine. Additionally, the Pythagorean theorem can be applied to Equation (4.105) to solve for $A_\theta$ as a function of $A_v$ by

$$A_\theta = \frac{\sqrt{m^2 \omega^2_u + \zeta^2_d}}{|2 m l \omega_u \dot{\theta}_s|} A_v$$  \hspace{1cm} (4.107)

where the absolute value is applied such that the amplitude is positive.

Applying the results in Equation (4.105), the same procedure can be repeated for Equation (4.104). The sine and cosine of $\phi_v$ are

$$\sin(\phi_v) = \frac{A_v \omega_u ((J + ml^2) \zeta_d + m (ml v_s + l^2 \zeta_d + \zeta_r))}{2 A_v m l \dot{\theta}_s}$$

$$\cos(\phi_v) = \frac{A_v \left(m \omega^2_u (J + ml^2) - (ml v_s + l^2 \zeta_d + \zeta_r) \zeta_d - 2 m^2 l^2 \dot{\theta}_s^2\right)}{2 A_v m l \dot{\theta}_s}$$  \hspace{1cm} (4.108)

Dividing the two functions results in the tangent function, expressed as

$$\tan(\phi_v) = \frac{\omega_u ((J + ml^2) \zeta_d + m (ml v_s + l^2 \zeta_d + \zeta_r))}{m \omega^2_u (J + ml^2) - (ml v_s + l^2 \zeta_d + \zeta_r) \zeta_d - 2 m^2 l^2 \dot{\theta}_s^2}.$$  \hspace{1cm} (4.109)

The amplitude of $v$ can be solved in terms of the amplitude of the input function by applying the Pythagorean theorem to Equation (4.108), such that

$$A_v = \frac{|2 m l \dot{\theta}_s| A_u}{\sqrt{\eta}}$$

$$\eta = \left(\omega_u ((J + ml^2) \zeta_d + m (ml v_s + l^2 \zeta_d + \zeta_r))\right)^2 + \left(m \omega^2_u (J + ml^2) - (ml v_s + l^2 \zeta_d + \zeta_r) \zeta_d - 2 m^2 l^2 \dot{\theta}_s^2\right)^2$$  \hspace{1cm} (4.110)
With \( A_v \) calculated, the previous equations are used to solve for \( A_\theta, \phi_v, \) and \( \phi_\theta \). Note that the above expression contains both \( \dot{\theta}_s \) and \( v_s \), which are functions of \( u_s \).

Recall that the squared and product terms were ignored to solve for the amplitudes and phase shifts. Suppose that the average of two terms are taken over a full period, such that

\[
\frac{\omega_u}{2\pi} \int_0^{2\pi} l(A_\theta \omega_u \cos(\omega_u t + \phi_\theta))^2 = \frac{1}{2} l A_v^2 \omega_u^2
\]

\[
\frac{\omega_u}{2\pi} \int_0^{2\pi} ml(A_v \sin(\omega_u t + \phi_v))(A_\theta \omega_u \cos(\omega_u t + \phi_\theta)) = \frac{1}{2} A_v A_\theta ml \sin(\phi_v - \phi_\theta). \tag{4.111}
\]

These ignored terms contribute to some shifting of the mean values \( v_s \) and \( \dot{\theta}_s \), such that Equation (4.99) becomes

\[
0 = -\frac{\zeta_d}{m} v_s + l \dot{\theta}_s^2 + \frac{1}{2} l A_\theta^2 \omega_u^2
\]

\[
0 = -\frac{mlv_s \dot{\theta}_s + (l^2 \zeta_d + \zeta_f) \dot{\theta}_s + u_s + \frac{1}{2} A_v A_\theta ml \sin(\phi_v - \phi_\theta)}{J + ml^2}. \tag{4.112}
\]

Using the previously obtained expressions for the amplitude and phase shifts, the new mean values can be solved for through the depressed cubic once again. The new mean values can then be applied again to update the amplitude and phase shifts. Therefore, a recursive process is created in hopes that the values converge to some value under the assumption that there is convergence. In other words, the values are "tuned" until there is little change in the values.

The \( x \) and \( y \) states can be estimated as well. Recall that from the constraint equations, the \( \dot{x} \) and \( \dot{y} \) expressions are

\[
\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} v \\ l \dot{\theta} \end{bmatrix}. \tag{4.113}
\]

Because \( \theta \) is either increasing or decreasing at a constant rate with some sinusoidal motion, suppose that it is assumed to be linear such that

\[
\theta(t) \approx \dot{\theta}_s t \tag{4.114}
\]
and the velocities are expressed as
\[
\dot{x}(t) = (A_v \sin(\omega_u t + \phi_v) + v_s) \cos(\dot{\theta}_s t) - l(A_\theta \omega_u \cos(\omega_u t + \phi_\theta) + \dot{\theta}_s) \sin(\dot{\theta}_s t)
\]
\[
\dot{y}(t) = (A_v \sin(\omega_u t + \phi_v) + v_s) \sin(\dot{\theta}_s t) + l(A_\theta \omega_u \cos(\omega_u t + \phi_\theta) + \dot{\theta}_s) \cos(\dot{\theta}_s t).
\] (4.115)

Integrating Equation (4.115) leads to
\[
x = \frac{v_s}{\dot{\theta}_s} \sin(\dot{\theta}_s t) + l \cos(\dot{\theta}_s t) - \frac{A_v \cos((\omega_u - \dot{\theta}_s)t + \phi_v)}{2(\omega_u - \dot{\theta}_s)} - \frac{A_v \cos((\omega_u + \dot{\theta}_s)t + \phi_v)}{2(\omega_u + \dot{\theta}_s)}
\]
\[
+ \frac{A_\theta \omega_u l \cos((\omega_u - \dot{\theta}_s)t + \phi_\theta)}{2(\omega_u - \dot{\theta}_s)} - \frac{A_\theta \omega_u l \cos((\omega_u + \dot{\theta}_s)t + \phi_\theta)}{2(\omega_u + \dot{\theta}_s)}
\]
\[
y = \frac{-v_s}{\dot{\theta}_s} \cos(\dot{\theta}_s t) + l \sin(\dot{\theta}_s t) + \frac{A_v \sin((\omega_u - \dot{\theta}_s)t + \phi_v)}{2(\omega_u - \dot{\theta}_s)} - \frac{A_v \sin((\omega_u + \dot{\theta}_s)t + \phi_v)}{2(\omega_u + \dot{\theta}_s)}
\]
\[
+ \frac{A_\theta \omega_u l \sin((\omega_u - \dot{\theta}_s)t + \phi_\theta)}{2(\omega_u - \dot{\theta}_s)} + \frac{A_\theta \omega_u l \sin((\omega_u + \dot{\theta}_s)t + \phi_\theta)}{2(\omega_u + \dot{\theta}_s)}.
\] (4.116)

Therefore, the path of the Chaplygin sleigh can be estimated from the \(x\) and \(y\) positions in Equation (4.116).

With the estimated behavior of each of the states, suppose the the Chaplygin sleigh is now simulated with the controller in Equation (4.92). As an example, apply the same parameters in the friction case, where
\[
J_d = 0.008 \text{ kg} \cdot \text{m}^2
\]
\[
J = 0.025 \text{ kg} \cdot \text{m}^2
\]
\[
m = 2 \text{ kg}
\]
\[
l = 0.2 \text{ m}
\]
\[
u = 4 \text{ Nm}
\]
\[
\zeta_d = 0.1 \text{ kg/s}
\]
\[
\zeta_r = 0.1 \text{ kg/s} \cdot \text{m}^2
\]

and suppose that the amplitude and angular velocity of the controller to be
\[
A_u = 2 \text{ Nm}
\]
\[
\omega_u = 3 \text{ rad/s}.
\]
For this case, the tuned constant shifts are calculated as

\[ v_s = 7.4897 \]
\[ \dot{\theta}_s = -1.2906 \]

which differs somewhat from the case with no sinusoidal input (constant input) of

\[ v_s = 7.1958 \]
\[ \dot{\theta}_s = -1.3412 \]

that were calculated earlier and used as the initial shifts for the recursive process.
The tuned amplitudes and the phase shifts are also calculated as

\[ A_v = 0.1107 \]
\[ A_\theta = 0.2144 \]
\[ \phi_v = -1.6269 \]
\[ \phi_\theta = 1.4980. \]

Notice that the magnitude of the phase shifts are fairly close to \( \pi/2 = 1.5708. \)

The following plots are given for a simulation of 30 seconds. The simulated results are plotted against two types of estimated approximations. The first is simply called "Single" in the plot legend and refers to the approximation with the bare minimum of one step of recurrence. The second is called "Tuned" in the plot legend as ten steps of recurrence are applied.
As shown, the amplitude for \( \dot{\theta} \) didn’t require too much tuning, but the amplitude for \( v \) did. The phase shifts are very accurate, as the crests are aligned nicely with the simulated results. For \( \theta \), the shift \( \theta_s \) is required, but cannot be determined analytically. Note that the tuning was essential to get a more precise \( \dot{\theta}_s \) such that the simulated and tuned \( \theta \) are parallel to another, and do not intersect, as shown in Figure 4.32.
The phase plot shown in Figures 4.33 and 4.34 indicates that the sleigh settles into a limit cycle as predicted. The shape in this case tends to be elliptical, in which the amplitudes, $A_v$ and $\omega_u A_{\theta}$, are the semi-minor and semi-major axis, and the mean values, $v_s$ and $\dot{\theta}_s$ are the coordinates of the center.
The analytical approximate solutions for $\dot{x}$ and $\dot{y}$ also appear to be very close to the simulated results. However, the analytical and simulated results begin to differ for $x$ and $y$. Even if the shifts in the $x$ and $y$ directions are corrected for, the amplitudes appear to be changing over time. As mentioned, the expected solution of $v$ and $\theta$ would have sinusoidal terms with integer multiples of $\omega_u$, such as $2\omega_u$, $3\omega_u$, and so on. It is assumed that the amplitudes for the larger angular velocities are less
significant. However, \( \dot{x} \) and \( \dot{y} \) become much more complicated with the product of sinusoidal components in Equation (4.115). As a result, the possible angular velocities includes the integer multiples shifted by \( \dot{\theta}_s \) both in the positive and negative direction. In addition, there is also the angular velocity of \( \dot{\theta}_s \) that arises. Remember, this is based on the addition of the assumption that \( \theta \) can be treated like a linear function. However, the \( x \) and \( y \) position analytical approximate solutions become even more complicated. The integration divides each trigonometric term by these shifted angular velocities, as shown in Equation (4.116). Although the amplitudes is assumed to go to zero for higher \( n \), the corresponding term can be significant if the shifted angular velocity is close to zero.

This is shown in the fast Fourier transform plots below. The plots of the positions are similar to their respective velocities for most of the shifted angular velocities. However, the amplitudes increase greatly in the position functions compared to the velocity functions near zero frequencies.

As shown the approximate solutions look very promising. However, it is very important to note that the solutions were based on several assumptions and may not be accurate for all cases. Although the results were favorable for the given conditions,
the estimated solution can vary significantly under different conditions. From further investigation, this was found to be the case when the mean input, $u_s$, is significantly smaller than the amplitude of the input, $A_u$. In other words

$$\frac{u_s}{A_u} \approx 0$$

For example, consider the $v$ plot for when

$$u_s = 0.05$$
$$A_u = 2.$$  

![Simulation of $v$ versus time for sinusoidal input (small mean input).](image)

Fig. 4.42. Simulation of $v$ versus time for sinusoidal input (small mean input).

The analytical results do not approach the simulated results, even with tuning. Therefore, caution must be taken for smaller mean inputs in comparison to the amplitude.

Now consider Figure 4.39 again and notice that although the analytical solution is circular, the simulation results in something resembling curves can be created from a spirograph. These curves are generally referred to as roulettes. Suppose that the curves themselves are referred to as This is better shown by extending the simulation to 120 seconds.
The roulettes created by a spirograph requires trigonometric terms of different angular velocities. As described earlier, if \( n\omega_u \pm \dot{\theta}_s \) is close to zero, the trigonometric term will stand out. The other main term corresponds to the first two terms in Equations (4.116). Different roulettes arise by changing the angular velocity \( \omega_u \), especially to some value less than the magnitude of \( \dot{\theta}_s \). In the example, \( \dot{\theta}_s \approx -1.3 \text{ rad/sec} \), although it varies slightly through the recurrence process. Suppose that \( \omega_u = 1 \text{ rad/sec} \) and the simulation is ran for 360 seconds.

---

**Fig. 4.43.** Simulation of \( y \) versus \( x \) for sinusoidal input for 120 seconds.

**Fig. 4.44.** Simulation of \( y \) versus \( x \) for sinusoidal input with \( \omega_u = 1 \).

**Fig. 4.45.** Simulation of \( y \) versus \( x \) for sinusoidal input with \( \omega_u = 0.3 \).
The most spectacular results are when the angular velocity is set to some improper fraction of the $\dot{\theta}_s$. For example, let $\omega_u = \frac{5}{4}\dot{\theta}_s$.

![Simulation of $y$ versus $x$ for sinusoidal input with $\omega_u = \frac{5}{4}\dot{\theta}_s$.](image)

As seen, the roulette has five pedals. The same is true for an improper fraction of the form

$$\omega_u = \frac{m}{m-1}\dot{\theta}_s.$$  \hspace{1cm} (4.117)

This is because $\omega_u - \dot{\theta}_s = \frac{\dot{\theta}}{m-1}$ and becomes the angular velocity that stands out. This results in a plot similar to a rose with $m$ pedals. However, if $\dot{\theta}_s$ is divided by an integer, then the center moves as the sleigh continues to move in a spiral. One example is if $\omega_u = \frac{1}{3}\dot{\theta}_s$.
This is considered as a degenerate case where the motion is no longer periodic and is due to $n\omega_u - \dot{\theta}_s \approx 0$ when $n = m$. Because the limit of $\frac{\sin(\alpha)}{\alpha}$ goes to one as $\alpha$ goes to zero, the centers of $x$ and $y$ moves in some direction related to the amplitudes and phase shifts of the $m$th term of the Fourier series.

4.10 Conclusions and Future Work

As shown, the Chaplygin sleigh has a very interesting behavior when a constant input is applied. If there is no friction, then the states asymptotically follows a solution that involves the time raised to a fractional power. Additionally, the Chaplygin sleigh undergoes an outward spiral. If friction is included, then the states settle to a constant value, and the spiral settles into a circle.

A proportional and derivative feedback controller can also be applied to the Chaplygin sleigh to follow a desired heading angle. Although the desired heading angle was achieved, the sleigh would coast at a constant wheel velocity in the no friction case. If friction was added, then the Chaplygin sleigh would slow down until a stop. This is an issue if it was desired for the Chaplygin sleigh to reach some given distance.
It was proposed that adding a sinusoidal component to the feedback controller would eliminate the velocity from going to zero. The simulations show great promise to this approach, as the mean velocity is no longer zero, and the mean heading angle is the desired heading angle. However, little is analytical known about the resulting behavior. This includes the resulting amplitudes of the sinusoidal behavior of the states, as well as the mean velocity.

Because adding a sinusoidal component resulted in very interesting behavior, the same idea was applied to a constant input. From several assumptions, some approximate solutions were found. Although the solutions are fairly accurate for some simulations, there would be cases where the solutions are inaccurate. One such example is when the constant input is small compared to the amplitude of the sinusoidal component. However, the Chaplygin sleigh was shown to yield very interested results in the simulations, especially the positions plots. They would resemble the curves created by a spirograph. The type of curves could be roughly determined from the approximate solutions.

The Chaplygin sleigh proved to be a very unique system. Although simple in design, the nonholonomic constraint resulted in equations of motion that yielded very interesting behavior under several types of behavior. However, there is still more to be determined about applying sinusoidal input that can be very promising. Although it is very challenging, I am highly motivated to continue pursuing a deeper analysis to the behavior of the Chaplygin sleigh.
LIST OF REFERENCES
LIST OF REFERENCES


APPENDICES
A. SEGWAY’S MATLAB CODE

A.1 Parameters and Matrices

The following code gives values to the Segway’s parameters and calculates the state space coefficient matrices.

```
% Joseph Tuttle
% Param_and_Matrices_7.m
%
% Defines parameters and matrices used in the simulations

m = 2.595;
mw = 0.141;
h = 0.254;
r = 0.0615;
a = 0.165;
Jw = 0.00025;
g = 9.81;
Jx = 0.1986;
Jy = 0.1942;
Jz = 0.0056;

mp = m - 2*mw;
%
% Motor Parameters
```
kemf = 0.04928;
kv = 0.01133;

% Measurement Noise
tc = 0.001;

velerr = 0.001;
gyroerr = 0.008;

v1 = velerr*tc;
v2 = gyroerr*tc;
v3 = gyroerr*tc;

% State Noise

w1 = 0;
w2 = 0;
w3 = 0;

%% State Space Coefficient Matrices

delta = 2*Jw*(Jy+mp*h^2)+(2*mp*mw*h^2+m*Jy)*r^2;
mesa = Jz*r^2+2*a^2*(mw*r^2+Jw) ;

% Reduced States (4 states)

% sum states
As4 = [-2*kv*(Jy+mp*h*(h+r))/delta,-mp^2*g*h^2*r^2/delta,...
2*kv*r*(Jy+mp*h*(h+r))/delta;...
... 0,0,1;...
...  
\[ 2 \cdot kv \cdot (mp \cdot h + m \cdot r) \cdot r + 2 \cdot Jw) / \text{delta}, \ldots \]
\[-2 \cdot kv \cdot (mp \cdot h + m \cdot r) \cdot r + 2 \cdot Jw) / \text{delta} \];

Bs4 = \[ r \cdot kemf \cdot \left( Jy + mp \cdot h \cdot (h + r) \right) / \text{delta}; \ldots \]
\[ 0; \ldots \]
\[-kemf \cdot (mp \cdot h + m \cdot r) \cdot r + 2 \cdot Jw) / \text{delta} \];

% difference states
Ad4 = \[-2 \cdot kv \cdot a^2 / \text{mesa};\]
Bd4 = \[-a \cdot r \cdot kemf / \text{mesa};\]

% Expanded States (6 states)

% sum states
As6 = \[0, 1, 0, 0; \ldots \]
\[0, 2 \cdot kv \cdot (Jy + mp \cdot h \cdot (h + r)) / \text{delta}, -mp^2 \cdot g \cdot h^2 \cdot r^2 / \text{delta}, \ldots \]
\[2 \cdot kv \cdot r \cdot (Jy + mp \cdot h \cdot (h + r)) / \text{delta}; \ldots \]
\[0, 0, 0, 1; \ldots \]
\[0, 2 \cdot kv \cdot (mp \cdot h + m \cdot r) \cdot r + 2 \cdot Jw) / \text{delta}, \ldots \]
\[mp \cdot g \cdot h \cdot (2 \cdot Jw + m \cdot r^2) / \text{delta}, -2 \cdot kv \cdot (mp \cdot h + m \cdot r) \cdot r + 2 \cdot Jw) / \text{delta} \];

Bs6 = \[0; \ldots \]
\[r \cdot kemf \cdot (Jy + mp \cdot h \cdot (h + r)) / \text{delta}; \ldots \]
\[0; \ldots \]
\[-kemf \cdot (mp \cdot h + m \cdot r) \cdot r + 2 \cdot Jw) / \text{delta} \];

% difference states
Ad6 = \[0, 1; \ldots \]
\[0, -2 \cdot kv \cdot a^2 / \text{mesa};\]
Bd6 = \[0; \ldots \]
\[-a \cdot r \cdot kemf / \text{mesa};\]
Kalman Filter Coefficient Matrices

Cs = [1 0 0; 0 0 1];
[nsl,ns2] = size(Cs);
Ds = [velerr 0; 0 gyroerr];
BBs = zeros(3,3);

Ps = care(As4',Cs'*Ds*(-1)',BBs);
KsJ = [-24 -24 -120 -12];
KdJ = [-2.4 -2.4];

Ls = Ps*Cs'*Ds*(-1);

% A,B,C,D matrices that go into Simulink state space block
Aes = As4-Ls*Cs;
Bes = [Ls, Bs4];
Ces = eye(3);
Des = zeros(3,nsl+1);

Cd = [1];
[nd1,nd2] = size(Cd);
Dd = [gyroerr];
BBd = zeros(1,1);
Pd = care(Ad4',Cd'*Dd'^(-1)',BBd);
Ld = Pd*Cd'*Dd'^(-1);

% A,B,C,D matrices that go into Simulink state space block
Aed = Ad4-Ld*Cd;
Bed = [Ld, Bd4];
Ced = eye(1);
Ded = zeros(1,nd1+1);

% Gains
KsJ = [-24 -24 -120 -12];
KdJ = [-2.4 -2.4];
\[ K_s = K_s J; \]
\[ K_d = K_d J; \]

\%
\[ \text{eig}(A s_6 - B s_6 \times K_s J) \]
\%
\[ \text{eig}(A d_6 - B d_6 \times K_d J) \]
A.2 Setup to Simulations

The following code defines the initial conditions and the reference commands that are to be used for the simulation. The code also initializes the simulation and plots the results.

```matlab
% Joseph Tuttle
% % setup SEGway_nonlinear_7_2.m
clc
close all
clear all

% x_s = [s, v, theta, theta dot]
% x_d = [psi, psi dot]

v_ref = 0;
psidot_ref = 0;

% Realtime
% v_ref = 0.2;
% psidot_ref = 25*pi/180;

xr_s = [v_ref, 0, 0];
xr_d = [psidot_ref];

xr = [v_ref, psidot_ref, 0, 0];

ICs = [0, 0.05, -6*pi/180, 3*pi/180]';
```
ICd = [0, 3*pi/180]';

% From Test Data (stability)
ICs = [0, -0.0680678, -0.159346, -0.264732]';
ICd = [-0.1384837, -0.0224624]';

% % From Test Data (circle)

% ICs = [0.368614, 0, 0.101909, -0.034043]';
% ICd = [0, -0.00460424]';

% ICd = [5*pi/180, 3*pi/180]';
x_s0 = [ICs(2), ICs(3), ICs(4)]';
x_d0 = [ICd(2)]';
x_s0 = [ICs(2), ICd(2), ICs(3), ICs(4)];
x_s0 = rand(4,1);
x_e_s0 = x_s0;
x_e_d0 = x_d0;

xp_0 = 0;
yp_0 = 0;

tfinal = 50;  % Total Time

Param_and_Matrices_7; % obtains parameters and matrices

%% Simulates both the cases with noise and without noise

sim('segway_nonlinear_nonoise_7_2');
t2 = t;
x2 = x;
sim('segway_nonlinear_noise_7_2');
%% Plots all the simulated plots

figure
plot(t,x(:,1))
hold on
plot(t,xe(:,1),'r')
plot(t2,x2(:,1),'m')
legend('State', 'Estimator','No Noise')
xlabel('Time (s)')
ylabel('Arc Length (m)')
% title('Distance vs Time')
grid on;

figure
plot(t,x(:,2))
hold on
plot(t,xe(:,2),'r')
plot(t2,x2(:,2),'m')
legend('State', 'Estimator','No Noise')
xlabel('Time (s)')
ylabel('Velocity (m/s)')
% title('Velocity vs Time')
grid on;

figure
plot(t,x(:,3)*180/pi)
hold on
plot(t,xe(:,3)*180/pi,'r')
plot(t2,x2(:,3)*180/pi,'m')
legend('State', 'Estimator','No Noise')
xlabel('Time (s)')
ylabel('Psi (deg)')
% title('Psi vs Time')
grid on;

figure
plot(t,x(:,4)*180/pi)
hold on
plot(t,xe(:,4)*180/pi,'r')
plot(t2,x2(:,4)*180/pi,'m')
legend('State', 'Estimator','No Noise')
xlabel('Time (s)')
ylabel('Psi Dot (deg/s)')
% title('Psi Dot vs Time')
grid on;

figure
plot(t,x(:,5)*180/pi)
hold on
plot(t,xe(:,5)*180/pi,'r')
plot(t2,x2(:,5)*180/pi,'m')
legend('State', 'Estimator','No Noise')
xlabel('Time (s)')
ylabel('Theta (deg)')
% title('Theta vs Time')
grid on;

figure
plot(t,x(:,6)*180/pi)
hold on
plot(t,xe(:,6)*180/pi,'r')
plot(t2,x2(:,6)*180/pi,'m')
legend('State', 'Estimator','No Noise')
xlabel('Time (s)')
ylabel('Theta Dot (deg/s)')
% title('Theta Dot vs Time')
grid on;

figure
% axis square
plot(xp,yp)
hold on;
plot(xpa,ypa,'r')
xlabel('X Position')
ylabel('Y Position')
legend('Simulated', 'Desired')
% title('Y Position vs X Position')
axis equal
grid on;

figure
plot(t,input)
xlabel('Time (s)')
ylabel('Voltage (V)')
legend('Left Wheel','Right Wheel')
% title('Voltage vs Time')
grid on;

%%

Stab_2_Data

% Circle_1_Data

figure
plot(t2,x1)
hold on
plot(t,xe(:,1),'r')
legend('Experimental', 'Simulated')
xlabel('Time (s)')
ylabel('Arc Length (m)')
% title('Distance vs Time')
grid on;

figure
plot(t3,vel1)
hold on
plot(t,xe(:,2),'r')
legend('Experimental', 'Simulated')
xlabel('Time (s)')
ylabel('Velocity (m/s)')
% title('Velocity vs Time')
grid on;

figure
plot(t1,yaw2)
hold on
plot(t,xe(:,3)*180/pi,'r')
legend('Experimental', 'Simulated')
xlabel('Time (s)')
ylabel('Psi (deg)')
% title('Psi vs Time')
grid on;

figure
plot(t1,yawdot1)
hold on
plot(t,xe(:,4)*180/pi,'r')
legend('Experimental', 'Simulated')
xlabel('Time (s)')
ylabel('Psi Dot (deg/s)')
% title('Psi Dot vs Time')
grid on;

figure
plot(t1,thetal)
hold on
plot(t,xe(:,5)*180/pi,'r')
legend('Experimental', 'Simulated')
xlabel('Time (s)')
ylabel('Theta (deg)')
% title('Theta vs Time')
grid on;

figure
plot(t1,thetadot1)
hold on
plot(t,xe(:,6)*180/pi,'r')
legend('Experimental', 'Simulated')
xlabel('Time (s)')
ylabel('Theta Dot (deg/s)')
% title('Theta Dot vs Time')
grid on;
A.3 Simulink Model Without Noise

The Simulink model includes the nonlinear plant dynamics and the feedback controller about the velocity and heading angle rate commands. No measurement noise is included in the system. It also simulates the path of the Segway and the desired path. All the states and other variables are sent to the workspace.

Fig. A.1. Simulink model without noise (Segway).
A.4 Simulink Model With Noise

The model is similar to the previous model, but with measurement noise included. Steady state linear Kalman filters are applied to both sets of states and the estimated error is fed back into the controller. All the states and other variables are sent to the workspace.

Fig. A.2. Simulink model with noise (Segway).
B. CHAPLYGIN SLEIGH’S MATLAB CODE

B.1 Setup to Simulation with Constant Input

The setup simulates the case of applying a constant input to the Chaplygin Sleigh. The code includes all the parameters and initial conditions to be applied to the simulation. The code also initializes the simulation and plots the results. The friction parameters can be set to zero for the case with no friction.

```matlab
% Chaplygin_Friction
% 08/31/2012

clc;
close all;
clear all;

% Body Parameters
Jd = 0.008;
J = 0.025;
m = 2;
l = 0.2;

% Constant Input
u = 4;

tfinal = 45;
```
% Friction Parameters

w = 0.00;
% zetat = 0.1;
zetat = 0;
% zetar = 0.1;
zetar = 0;
zeta4 = 0.0;
gamma = 0.0;

zeta = zetat;
zeta3 = zetar;

% Initial conditions

xdot0 = 0;
ydot0 = 0;
psidot0 = 0;

x0 = 0;
y0 = 0;
theta0 = 0;
psio = 0;

% Analytical Solution Initial Conditions

ts = 0;
thetas = 0;
xs = 0;
ys = 0;

% Initial Condition Relations
\[ v_0 = x_{\dot{0}} \cos(\theta_0) + y_{\dot{0}} \sin(\theta_0); \]

\[ z_0 = (-x_{\dot{0}} \sin(\theta_0) + y_{\dot{0}} \cos(\theta_0))/l; \quad \text{% Due to constraint} \]

\[ \text{xs0} = [v_0, z_0, \psi_{\dot{0}}]; \]

%% Initialize Simulation through Simulink

\text{sim('Chaplygin Friction sub');}

%% Analytical Results with no friction

\text{if zetat ~= 0 & zetar ~= 0}

\[ zp1 = -zetat * u / (2 * m^2 * l^2); \]

\[ zp2 = (zetat^2 * u^2 / (4 * m^4 * l^4)) + ((zetar * zetat + zetat^2 * l^2)^3) / (27 * m^6 * l^6)); \]

\[ zp3 = (zp1 - (zp2)^{(1/2)}); \]

\[ za = (zp1 + (zp2)^{(1/2)})^{(1/3)} + \text{sign}(zp3) * \text{abs}(zp3^{(1/3)}) \]

\[ va = m * l / zetat * za^2 \]

\[ amp = za * l / zetat * (m^2 * za^2 + zetat^2)^(1/2); \]

\[ xda = - amp * \sin(za * t); \]

\[ yda = amp * \cos(za * t); \]

\[ rad = amp / za; \]

\[ xa = rad * \cos(za * t + \theta); \]

\[ ya = rad * \sin(za * t + \theta); \]

%% Plot Figures (No Friction}

\text{figure}

\text{hold on}

\text{plot(t,x(:,1))}

\text{plot(t,va,'r')}

\text{xlabel('Time (s)')}

\text{legend('Simulation','Analytical')}

\text{ylabel('v (m/s)')}

\text{figure}
hold on
plot(t,x(:,2))
plot(t,za,'r')
xlabel('Time (s)')
legend('Simulation','Analytical')
ylabel('z (rad/s)')

figure
hold on
plot(t,yp_dot)
plot(t,yda,'r')
xlabel('Time (s)')
legend('Simulation','Analytical')
ylabel('y derivative (m/s)')

figure
hold on
plot(t,xp)
plot(t,xa,'r')
xlabel('Time (s)')
legend('Simulation','Analytical')
ylabel('x (m)')

figure
hold on
plot(xp,yp)
plot(xa,ya,'r')
xlabel('x (m)')
legend('Simulation','Analytical')
ylabel('y (m)')

else

%%% Analytical Results with Friction
va = (3*\(u^2/(m^2*l)\)*(t-ts))^\((1/3)\);
za = -(\(u/(3*m*l^2*(t-ts))\))^\((1/3)\);
thetaa = -(9*\(u/(8*m*l^2)*(t-ts)^2\))^\((1/3)\)+thetas;
xda = (1./(m^2*l*(t-ts)))^\((1/3)\).*((3*u^2*(t-ts)^2)^\((1/3)\).*\(\cos(\thetaa)\)...
   +1/3*(9*u*m*l^2)^\((1/3)\).*\(\sin(\thetaa)\);
yda = (1./(m^2*l*(t-ts)))^\((1/3)\).*((3*u^2*(t-ts)^2)^\((1/3)\).*\(\sin(\thetaa)\)...
   -1/3*(9*u*m*l^2)^\((1/3)\).*\(\cos(\thetaa)\);
xa = (1/m)^\((1/3)\)*(-9*u*(t-ts)^2)^\((1/3)\).*\(\sin(\thetaa)\)...
   +3*(m*l^2)^\((1/3)\).*\(\cos(\thetaa)\)-1))+xs;
ya = (1/m)^\((1/3)\)*((9*u*(t-ts)^2)^\((1/3)\).*\(\cos(\thetaa)\)...
   +3*(m*l^2)^\((1/3)\).*\(\sin(\thetaa)\))+ys;
da = (1/m)^\((1/3)\)*((3*(3*u^2*(t-ts)^4)^\((1/3)\)+9*(m^2*l^4)^\((1/3)\))^\((1/2)\);

d = sqrt(xp.^2+yp.^2);

%% Plots (Friction)
figure
hold on
plot(t,x(:,1))
plot(t,va,'r')
xlabel('Time (s)')
legend('Simulation','Analytical')
ylabel('v (m/s)')
figure
hold on
plot(t,x(:,2))
plot(t,za,'r')
xlabel('Time (s)')
legend('Simulation','Analytical')
ylabel('z (rad/s)')
figure
hold on
plot(t,theta)
plot(t,thetaa,'r')
xlabel('Time (s)')
legend('Simulation','Analytical')
ylabel('Theta (rad)')

figure
hold on
plot(t,xp_dot)
plot(t,xda,'r')
xlabel('Time (s)')
legend('Simulation','Analytical')
ylabel('x derivative (m/s)')

figure
hold on
plot(t,yp_dot)
plot(t,yda,'r')
xlabel('Time (s)')
legend('Simulation','Analytical')
ylabel('y derivative (m/s)')

figure
hold on
plot(t,xp)
plot(t,xa,'r')
xlabel('Time (s)')
legend('Simulation','Analytical')
ylabel('x (m)')

figure
hold on
plot(t,yp)
plot(t,ya,'r')
xlabel('Time (s)')
legend('Simulation','Analytical')
ylabel('y (m)')

figure
hold on
plot(t,d)
plot(t,da,'r')
xlabel('Time (s)')
legend('Simulation','Analytical')
ylabel('Distance (m)')

figure
hold on
plot(xp,yp)
plot(xa,ya,'r')
xlabel('x (m)')
legend('Simulation','Analytical')
ylabel('y (m)')

end
B.2 Simulink Model

The simulink model for the Chaplygin sleigh includes the constant input fed into the plant dynamics. It also collects the states and outputs it to the workspace.

Fig. B.1. Simulink model for constant input (Chaplygin Sleigh).
C. EXAMPLE OF BUILT SEGWAY CONTROLLER CODE

Given below is the main code of the built Segway controller written by James Goppert. As described, the controller is broken into a series of inner loops.

```cpp
#include "BlockSegwayController.hpp"

// px4
#include <geo/geo.h>
#include <drivers/drv_hrt.h>

BlockSegwayController::BlockSegwayController() :
    SuperBlock(NULL, "SEG"),

    // subscriptions
    _att(&getSubscriptions(), ORB_ID(vehicle_attitude), 3),
    _pos(&getSubscriptions(), ORB_ID(vehicle_global_position), 3),
    _posCmd(&getSubscriptions(), ORB_ID(position_setpoint_triplet), 3),
    _localPos(&getSubscriptions(), ORB_ID(vehicle_local_position), 3),
    _localPosCmd(&getSubscriptions(), ...
    ORB_ID(vehicle_local_position_setpoint), 3),
    _manual(&getSubscriptions(), ORB_ID(manual_control_setpoint), 3),
    _status(&getSubscriptions(), ORB_ID(vehicle_status), 3),
    _param_update(&getSubscriptions(), ORB_ID(parameter_update), 1000), ...
    // limit to 1 Hz
    _encoders(&getSubscriptions(), ORB_ID(encoders), 10),...
    // limit to 100 Hz

    // publications
    _attCmd(&getPublications(), ORB_ID(vehicle_attitude_setpoint)),
```
ratesCmd(&getPublications(), ORB_ID(vehicle_rates_setpoint)),
_globalVelCmd(&getPublications(), ... ORB_ID(vehicle_global_velocity_setpoint)),
_actuators(&getPublications(), ORB_ID(actuator_controls_1)),

_yaw2r(this, "YAW2R"),
_r2v(this, "R2V"),
_th2v(this, "TH2V"),
_q2v(this, "Q2V"),
_x2vel(this, "X2VEL"),
_vel2th(this, "VEL2TH"),
_thLimit(this, "TH_LIM"),
_velLimit(this, "VEL_LIM"),
_thStop(this, "TH_STOP"),
_sysIdAmp(this, "SYSID_AMP"),
_sysIdFreq(this, "SYSID_FREQ"),
_attPoll(),
_timeStamp(0)
{
    orb_set_interval(_att.getHandle(), 10);...
    // set attitude update rate to 100 Hz (period 10 ms)
    _attPoll.fd = _att.getHandle();
    _attPoll.events = POLLIN;
}

void BlockSegwayController::update() {
    // wait for a sensor update, check for exit condition every 100 ms
    if (poll(&_attPoll, 1, 100) < 0) return; // poll error

    uint64_t newTimeStamp = hrt_absolute_time();
    float dt = (newTimeStamp - _timeStamp) / 1.0e6f;
    _timeStamp = newTimeStamp;

    // check for sane values of dt
// to prevent large control responses
if (dt > 1.0f || dt < 0) return;

// set dt for all child blocks
setDt(dt);

// check for new updates
if (paramUpdate.updated()) updateParams();

// get new information from subscriptions
updateSubscriptions();

// default all output to zero unless handled by mode
for (unsigned i = 2; i < NUM_ACTUATOR_CONTROLS; i++)
    _actuators.control[i] = 0.0f;

// only update guidance in auto mode
if (_status.main_state == MAIN_STATE_AUTO) {
    // update guidance
}

// commands for inner stabilization loop
float thCmd = 0; // pitch command
float rCmd = 0; // yaw rate command
float yawCmd = 0; ...
// always point north for now, can use localPosCmd.yaw later
float velCmd = 0; // velocity command

// syste id
float t = _timeStamp/1.0e6;

// modes that track position
if (_status.main_state == MAIN_STATE_AUTO ||
    _status.main_state == MAIN_STATE_EASY ||
    _status.main_state == MAIN_STATE_SEATBELT) {
// the position to track
float localPosCmdX = 0;

// auto mode follows waypoints
if (_status.main_state == MAIN_STATE_AUTO) {
    localPosCmdX = _localPosCmd.x;
    // system id with square wave
} else if (_status.main_state == MAIN_STATE_EASY) {
    float sineWave = sinf(2*M_PI_F*_sysIdFreq.get()*t);
    float squareWave = 0;
    if (sineWave > 0) {
        squareWave = _sysIdAmp.get();
    } else {
        squareWave = -_sysIdAmp.get();
    }
    localPosCmdX = squareWave;
    // system id with sine wave
} else if (_status.main_state == MAIN_STATE_SEATBELT) {
    float sineWave = sinf(2*M_PI_F*_sysIdFreq.get()*t);
    localPosCmdX = sineWave;
}

// track the position command
velCmd = ...
    _velLimit.update(_x2vel1.update(localPosCmdX - _localPos.x));
// negative sign since need to lean in negative pitch to move forward
thCmd = _thLimit.update(_vel2th.update(velCmd - _localPos.vx));
float yawError = yawCmd - _att.yaw;
// wrap yaw error to between -180 and 180
if (yawError > M_PI_F/2) yawError = yawError - 2*M_PI_F;
if (yawError < -M_PI_F/2) yawError = yawError + 2*M_PI_F;
rCmd = _yaw2r.update(yawError);

// manual mode
} else if (status.main_state == MAIN_STATE_MANUAL) {
    rCmd = _manual.roll;
    velCmd = _manual.pitch * velLimit.getMax();
}

// negative sign since need to lean in negative pitch to move forward
thCmd = -thLimit.update(vel2th.update(velCmd - localPos.vx));

// compute control for pitch
float controlPitch = _th2v.update(thCmd - _att.pitch) - _q2v.update(_att.pitchspeed);

// compute control for yaw
float controlYaw = _r2v.update(rCmd - _att.yawspeed);

// output scaling by manual throttle
controlPitch *= _manual.throttle;
controlYaw *= _manual.throttle;

// attitude set point
_attCmd.timestamp = _timeStamp;
_attCmd.pitch_body = thCmd;
_attCmd.roll_body = 0;
_attCmd.yaw_body = yawCmd;
_attCmd.R_valid = false;
_attCmd.q_d_valid = false;
_attCmd.q_e_valid = false;
_attCmd.thrust = 0;
_attCmd.roll_reset_integral = false;
_attCmd.update();

// rates set point
_ratesCmd.timestamp = _timeStamp;
_ratesCmd.roll = 0;
_ratesCmd.pitch = 0;
\_ratesCmd.yaw = rCmd;
\_ratesCmd.thrust = 0;
\_ratesCmd.update();

// global velocity set point
\_globalVelCmd.vx = velCmd;
\_globalVelCmd.vy = 0;
\_globalVelCmd.vz = 0;
\_globalVelCmd.update();

// send outputs if armed and pitch less than shut off pitch
// than shut off pitch
if (\_status.arming.state == ARMING\_STATE\_ARmed &&
    fabsf(\_att.pitch) < \_thStop.get() ) {
    // controls
    \_actuators.timestamp = \_timeStamp;
    \_actuators.control[0] = 0; // roll
    \_actuators.control[1] = controlPitch; // pitch
    \_actuators.control[2] = controlYaw; // yaw
    \_actuators.control[3] = 0; // thrust
    \_actuators.update();
} else {
    // controls
    \_actuators.timestamp = \_timeStamp;
    \_actuators.control[0] = 0; // roll
    \_actuators.control[1] = 0; // pitch
    \_actuators.control[2] = 0; // yaw
    \_actuators.control[3] = 0; // thrust
    \_actuators.update();
}
VITA
Joseph Tuttle was born and raised in Ramsey, Minnesota. After graduation from high school in 2007, he enrolled in the undergraduate program at the University of Minnesota in Minneapolis, Minnesota. He graduated with a Bachelor of Science degree in Aerospace Engineering and Mechanics with Summa Cum Laude honors in 2011. In addition, he obtained a minor in Mathematics and Astrophysics. During that time, he was a teaching assistant in the University of Minnesota Talented Youth Mathematics Program (UMTYMP). He was then a Guidance, Navigation, and Control Engineer Intern at Goodrich/United Technologies Corporation in Orlando, Florida during the summer of 2012. Quickly after the internship, he attended the master’s program at Purdue University in West Lafayette, Indiana. His field was Dynamics and Controls with a minor in Astrodynamics and Space Applications. He was funded as a Research Assistant under Professor Arthur Frazho, focusing on systems with non-holonomic constraints, and building a Segway for use in class lab demonstrations. He is projected to defend his master’s thesis in November and graduate with a Master’s degree in December.