1-1-2006

Automatic steering control system model and simulation to analyze performance of off-road vehicles subject to sloped terrain and vehicle parameter changes

B Scott Evans
Iowa State University

Automatic steering control system model and simulation to analyze performance of off-road vehicles subject to sloped terrain and vehicle parameter changes

by

B. Scott Evans

A thesis submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Major: Mechanical Engineering

Program of Study Committee:
James E. Bernard, Major Professor
Greg R. Luecke
Brian L. Steward

Iowa State University

Ames, Iowa

2006

Graduate College
Iowa State University

This is to certify that the master's thesis of

B. Scott Evans

has met the thesis requirements of Iowa State University

Signatures have been redacted for privacy

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1. INTRODUCTION

This thesis provides the technical development and analysis of a vehicle dynamics and automatic steering control model for use in a computer simulation. The objective of this study is to create a simulation to allow fast and robust controller design and to quickly evaluate the controller and vehicle performance when subjected to a sloped terrain. While the basic setup of the model enables it to be adaptive to any type of on or off-road vehicle with front, rear, or 4-wheel steer configurations, the detailed derivation looks at the response and characteristics of a rear steer combine harvester.

In addition to analyzing the response on a sloped terrain, the simulation also considers changes in certain vehicle parameters. Vehicle response to a range of longitudinal speed, center of gravity position, and vehicle weight demonstrates the flexibility of the simulation to analyze several system components.

Chapter 2 discusses the motivation and background for this study and gives an overview of the dynamics and controller models. Chapter 3 provides the details and derivations of these models and looks at the characteristics of the system. Chapter 4 examines some results of the simulation and discusses the system response for a range of parameters. Chapter 5 presents conclusions and a brief discussion of future work involving the development and validation of the model. The Appendix provides details of the model derivations and simulation code.

CHAPTER 2. BACKGROUND

2.1 Motivation

Precision navigation guidance has been a longstanding topic of great interest to agricultural researchers [8]. Precisely planted crop and accurately driven harvesting equipment leads to an increase in crop yield by reducing operator error and fatigue [6, 8, 13]. A successful guidance system yields desirable response on different terrains and for a variety of vehicle parameters, such as weight, speed, and center of gravity (CG) position [7].

Many guidance methods have been explored, but with advances in GPS technology, one common and effective type of guidance system uses GPS tracked vehicle position to maintain a desired path, commonly a straight line, as determined by the operator [5, 8, 9, 19]. The perpendicular distance of the vehicle from the path defines the lateral or off-track error, while the angle of the vehicle centerline with respect to the path is called the heading error. A steering controller calculates the steer angle to minimize the lateral and heading errors.

The ability of the guidance system to follow the path depends on the controlled vehicle system. Many researchers have used a simplified kinematic model [8, 9, 19] to define the behavior of the vehicle. Although these models have been adequate for many situations, they do not allow modeling of a sloped terrain [6]. This study presents a modeling approach that uses the lateral forces created by the tire slip and the external force associated with the terrain slope to develop the equations of motion and controller design. This allows theoretical

assessment of the effects of changes in speed, mass, CG position, and tires, all beyond the capability of a model based on kinematic assumptions.

The model can also be used as the underlying dynamics and control for an operator-in-the-loop virtual vehicle simulator application, such as those mentioned by Baack [2] and Norris et al [17]. These types of simulations eliminate the necessity and associated cost of a full-scale prototype [17]. Simulations such as these could allow the engineer to visualize the operation and behavior and make real time changes. It also facilitates operator training and familiarity with the guidance and steering system.

2.2 Overview of Model

This section describes the controller design, the basic vehicle dynamics model, and the associated tire model. Table 2.1 provides nomenclature which follows standard SAE convention that will be used throughout the study. Chapter 3 presents the derivation of the models discussed in the following sections.

Table 2.1 Nomenclature of vehicle and simulation parameters

Parameter	Description
a	Distance from front axle to CG
b	Distance from rear axle to CG
c_c	Cornering coefficient
$C_{\alpha f}$	Cornering stiffness of front tire
$C_{\alpha R}$	Cornering stiffness of rear tire
$F_{y,ext}$	Lateral force due to side slope
F_{yf}	Lateral force on front tires
F_{yR}	Lateral force on rear tires
I	Yaw moment of intertia *(about the z-axis)*
L	Wheel base
M	Vehicle mass
r	Vehicle yaw rate
u	Longitudinal velocity
v	Lateral velocity
V_v	Velocity vector of vehicle
W	Weight of vehicle
W_f	Weight on front wheels
W_R	Weight on rear wheels
x	Vehicle longitudinal axis *(x-direction)*
X	Vehicle position in world coordinate system *(longitudinal trajectory)*
y	Vehicle lateral axis *(y-direction)*
Y	Vehicle position in world coordinate system *(lateral trajectory)*
α_f	Front wheel slip angle
α_R	Rear wheel slip angle
β	Sideslip angle
δ_f	Front steer angle
δ_R	Rear steer angle
θ	Angle of terrain slope
ψ	Vehicle heading

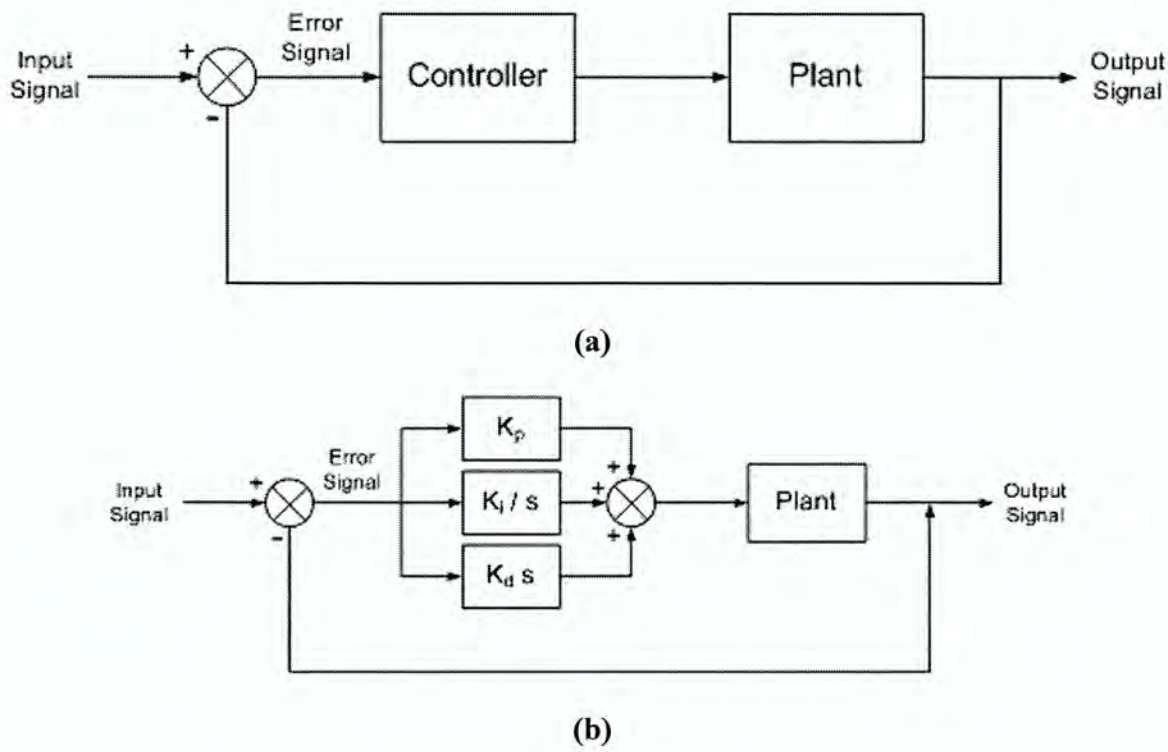


Figure 2.1 (a) Basic feedback controller block diagram (b) PID controller

2.2.1 Controller Model and Implementation

Control systems provide "an output or response for a given input or stimulus [4, p2]." The input, usually a desired response, is compared with the output, or actual response, creating an error signal that passes through the controller and system model (or plant) to generate a new output (see Figure 2.1a). This feedback controller aims at driving the error signal to the zero. Many methods of control have been studied and applied [8]. Some examples include: a hybrid controller incorporating non-linear "bang-bang" control when the error was large and Linear Quadratic Regulator control when the error became small [9], an active steering controller using fuzzy logic [16], and a controller based on optimal control theory [6]. This automatic steering control system uses a traditional approach, the proportional-plus-integral-

plus-derivative (PID) controller (see Figure 2.1b). This particular control method feeds a proportion of the error, the derivative of the error, and the integral of the error to the plant. The derivative term improves transient response and the integral term improves steady state error [4], both essential in the precision control of an agricultural vehicle. The proportional term contributes to both to some extent. The variables K_p, K_i, and K_d represent the controller gains that are manipulated to obtain the desired transient response and steady state characteristics. This study uses $K_{p\psi}$, $K_{i\psi}$, $K_{d\psi}$, and K_{pY}, K_{iY}, K_{dY} to represent the gains for the respective heading and lateral errors.

For the automatic steering control model in this study, two PID controllers command the rear steer angle needed to maintain the vehicle path based on the lateral and heading error. The steer angle is arbitrarily chosen to be in units of degrees; units of radians would simply result in a different set of controller gain values. The commanded steer angle is assumed to be a step input, resulting in large initial state variable derivative terms. Appendix A discusses a comparison to a steering lag model, which provides a more realistic steering response. The desired path is the X-axis of the world coordinate system. This simplifies calculations and allows the lateral error to equal the vehicle world position, Y, and the heading error to equal the heading angle, ψ, as defined by the world coordinate system shown in Figure 2.2. Eqn. 2.1 shows the controller command as the linear combination of Y and ψ. The commanded steer angle for a positive lateral error is positive for a rear wheel steer vehicle and negative for a front wheel steer configuration. In other words, the commanded steer angle is towards the desired path for a front steer vehicle and away from the desired path for a rear steer vehicle. There is no saturation for the steer angle in order to maintain linearity.

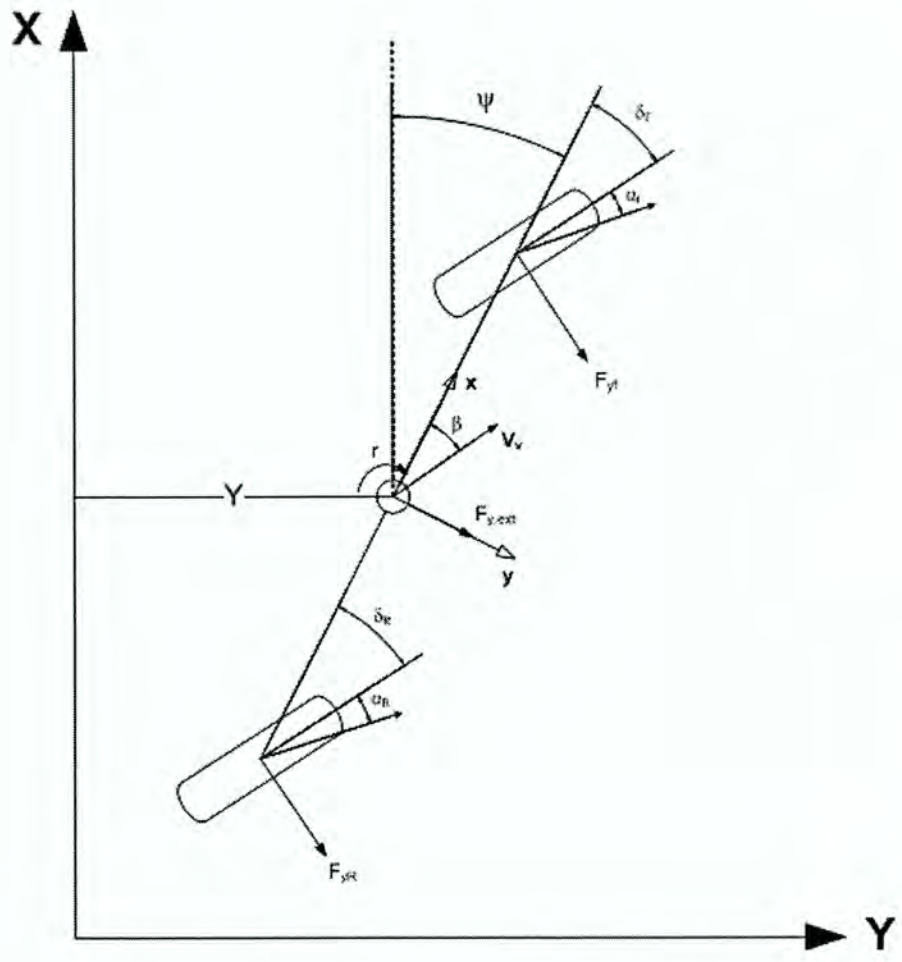


Figure 2.2 Earth fixed coordinate system showing vehicle orientation

$$\delta_R = K_{p\psi}\psi + \frac{K_{i\psi}}{s}\psi + K_{d\psi}s\psi + K_{pY}Y + \frac{K_{iY}}{s}Y + K_{dY}sY$$

$$\delta_R = \left(\frac{K_{d\psi}s^2 + K_{p\psi}s + K_{i\psi}}{s}\right)\psi + \left(\frac{K_{dY}s^2 + K_{pY}s + K_{iY}}{s}\right)Y \qquad (2.1)$$

2.2.2 Tire Model

Previous studies [8, 9, 19] have used a kinematic model, which assumes the velocity of each tire is in the direction it faces. In practice, however, there is a slip angle, α (see Figure 2.2),

defined as the angle between the direction of tire travel and the direction of tire heading [1]. In other words, the tire velocity is not in the direction the tire faces. Introducing the slip angle increases the complexity of the model and requires additional vehicle parameters. A basic vector relationship results in the expressions for the slip angles,

$$\alpha_f = \frac{v + ar}{u} - \delta_f \tag{2.2a}$$

$$\alpha_R = \frac{v - br}{u} - \delta_R \tag{2.2b}$$

The difference in heading and travel direction causes the region of the tire in contact with the ground, so-called the contact patch, to deflect and generate a force along the y-axis, termed the lateral force, F_y. The lateral force builds as the tire rolls, thus considerable sideslip lag could exist for low speeds [20]. To maintain simplicity, this model assumes no lag in lateral force buildup. The relationship between the slip angle and lateral force is determined experimentally and is shown in a carpet plot, such as the example in Figure 2.3.

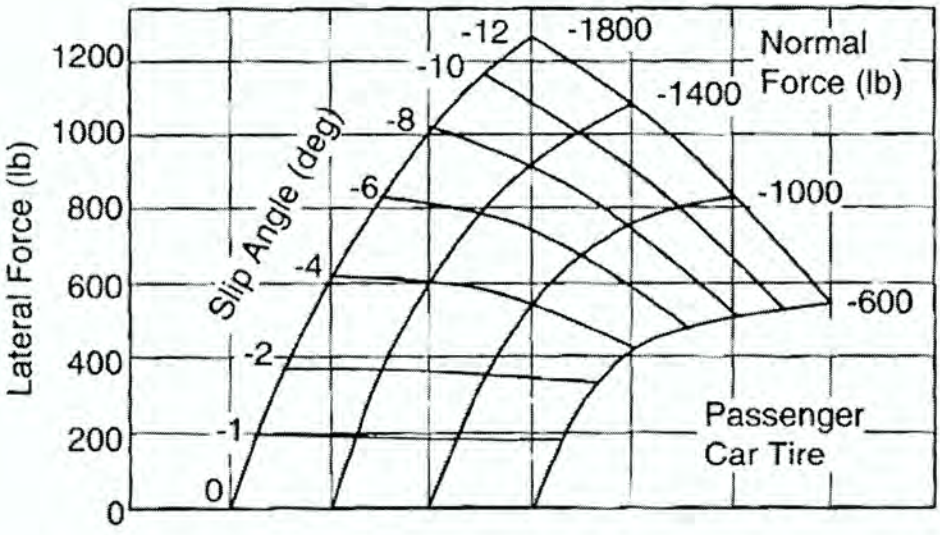


Figure 2.3 Example of carpet plot showing lateral force due to slip angle [1 p351]

The slope of the curve at zero slip angle is known as the cornering stiffness, C_α. The cornering stiffness is an essential parameter in regards to the behavior of the vehicle. Eqns. 2.3a and 2.3b show the relationship between the cornering stiffness and the lateral forces. A negative sign is added to maintain SAE convention [1], which calls for a negative slip angle to result in a positive lateral force.

$$F_{yf} = -C_{\alpha f}\alpha_f \tag{2.3a}$$

$$F_{yR} = -C_{\alpha R}\alpha_R \tag{2.3b}$$

Cornering stiffness data for on-road tires is well documented; however, lateral forces and slip for off-road vehicles are greatly affected by the nature of the soil [10], making it difficult to obtain cornering data for agricultural tires. In fact, tire companies do not collect and maintain cornering data for off-road tires, which has led to much research regarding methods to determine the lateral forces and related cornering stiffness values. Hun and Kim [16] looked at estimating the tire forces by relating the lateral forces to vehicle roll. Feng et al., [13, 14] used a series of field tests to obtain the cornering stiffness of a certain tractor tire combination. Metz, [10], presented data for the cornering coefficient of off-road tires for various ground conditions and summarized the range to be 0.03 to 0.09 deg^{-1}. Eqn. 2.4 [10] defines the cornering coefficient, c_c, as the ratio of cornering stiffness per normal load, or W_f and W_R respectively. The variable z denotes front or rear position. This relationship is used to find the cornering stiffness values $C_{\alpha f}$ and $C_{\alpha R}$.

$$c_{cz} = \frac{C_{\alpha z}}{W_z} \tag{2.4}$$

Chapter 4 examines the effect on system response of tire parameters exhibiting cornering coefficients in the range presented by Metz.

2.2.3 Vehicle Dynamics

The vehicle dynamics in this study derives from a linear yaw-plane bicycle model (see Figure 2.4). The model assumes no motion and acceleration along the z-axis, the axis normal to the ground, and eliminates roll and pitch. The model also does not consider steering lag or tire slip lag. It lumps all wheels into one for each axle; thus, the front and rear cornering stiffness is the sum of the stiffness values for each tire on the axle.

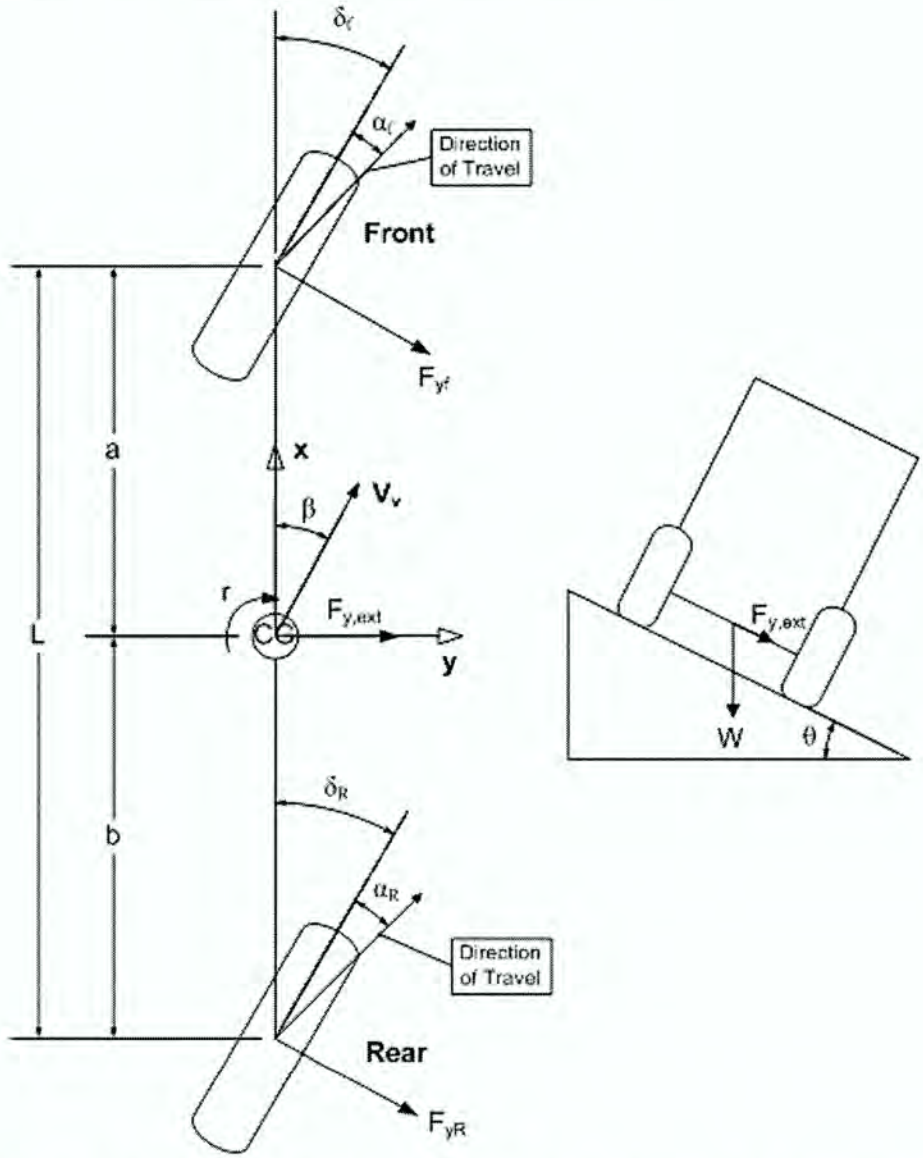


Figure 2.4 Free body diagram of bicycle model and side slope force

The longitudinal velocity, u, is the along the vehicle x-axis and the lateral velocity, v, is along the vehicle y-axis. In order to maintain linearity, this model assumes a constant forward velocity.

As indicated by Figure 2.4, the sideslip angle, β, is the angle between the vehicle x-axis and the direction of travel of the mass center [1]. Using small angle assumptions to preserve linearity results in the expression

$$\beta = \tan^{-1}\left(\frac{v}{u}\right) = \frac{v}{u} \tag{2.5}$$

One of the main interests in this study is to consider the effects of a sloped terrain on the automatic guidance control system. The right hand diagram in Figure 2.4 shows how the model incorporates the slope into the governing equations. Downhill slope is to the right of the vehicle, or along the positive y-axis. The component of the vehicle weight in the y-direction acts as an external force at the CG.

$$F_{y,ext} = W\sin(\theta) \tag{2.6}$$

A sum of the moments about the CG results in the first equation of motion shown below. The second equation of motion is the result of a sum of the forces in the y-direction. Chapter 3 defines the constants $a_{1\text{-}4}$ and $b_{1\text{-}4}$.

$$\dot{v} + a_1 v + a_2 r = a_3\delta_f + a_4\delta_R + \frac{W}{M}\sin(\theta) \tag{2.7}$$

$$\dot{r} + b_1 v + b_2 r = b_3\delta_f + b_4\delta_R \tag{2.8}$$

The vehicle states of interest for the automatic steering control are Y and ψ. They are functions of the other state variables, r and v. Relating the yaw angle and vehicle velocities results in the derivative of Y

$$\dot{Y} = u\sin(\psi) + v\cos(\psi) \tag{2.9a}$$

which for small angles is reduced to

$$\dot{Y} = u\psi + v \tag{2.9b}$$

The yaw angle is given by

$$\dot{\psi} = r \tag{2.10}$$

2.2.4 Overall system model

This study analyzes automatic steering control system in three ways, each of which confirms the calculations of the other models. A MATLAB® written time domain state representation uses a higher order Runge-Kutta method to numerically integrate the state variables. The controller commands the steer angle as described section 2.2.1 by a separate controller function called in the numerical integration code. The MATLAB® version of the model allows initial conditions of both Y and ψ and allows non-linearities of state equations and controller steer angle saturation to be easily introduced and examined. Manipulating Eqns. 2.7 through 2.10 gives the resulting system, or plant, in matrix form.

$$\begin{bmatrix} \dot{r} \\ \dot{v} \\ \dot{\psi} \\ \dot{Y} \end{bmatrix} = \begin{bmatrix} -b_2 & -b_1 & 0 & 0 \\ -a_2 & -a_1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & u & 0 \end{bmatrix} \begin{bmatrix} r \\ v \\ \psi \\ Y \end{bmatrix} + \begin{bmatrix} b_3 & b_4 & 0 \\ a_3 & a_4 & W/M \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_f \\ \delta_R \\ \theta \end{bmatrix} \tag{2.11}$$

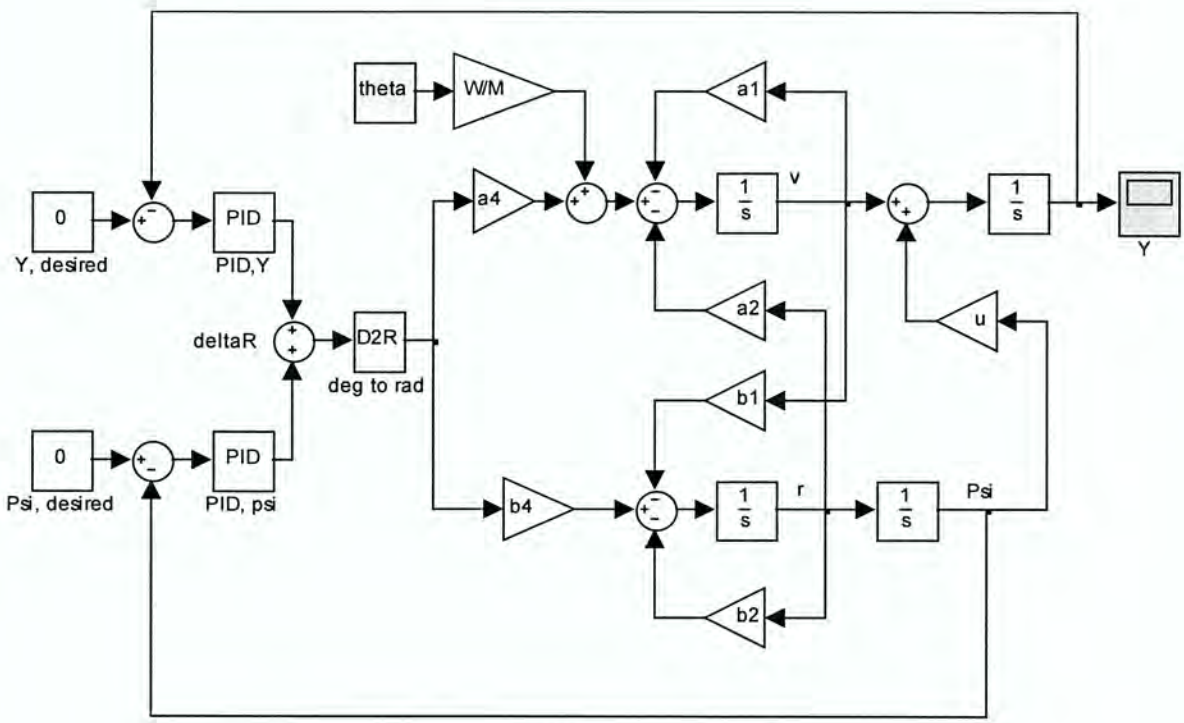


Figure 2.5 Simulink® block diagram of plant and controller before simplification

A Simulink® block diagram representing the system in the Laplace domain is setup using the same state equations (see Fig. 2.5). A block diagram representation allows the input signal to be easily followed through the system to the output. The third representation uses block diagram algebra to reduce the system to a single transfer function. The closed-loop system roots are found using the denominator of the transfer function as are the generalized root locus for each controller gain. Generalized root locus analysis is a useful tool because it enables graphical inspection of the controller response characteristics for different gain values. The next chapter contains the details of the entire system model, including steady state characteristics, transfer function derivation, and expressions for the constants used in this chapter.

CHAPTER 3. MODEL DEVELOPMENT

3.1 Derivation of vehicle dynamics model

The following subsections contain brief derivations of the final equations. The step-by-step derivations can be found in Appendix B. The free body diagram of the vehicle presented in Chapter 2 is shown again here as Figure 3.1.

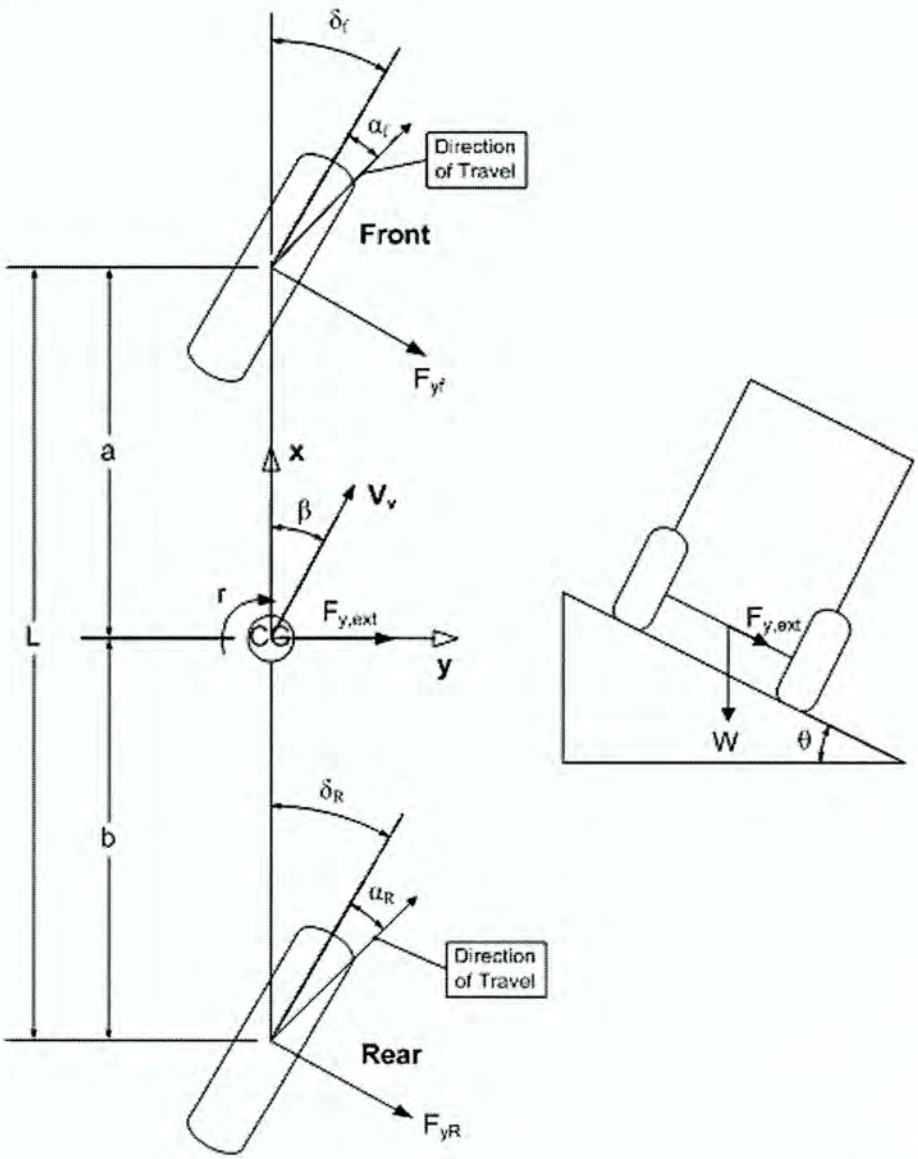


Figure 3.1 Free body diagram of bicycle model and side slope force

To maintain the linearity of the model, small steer approximations are used for determining the contribution of lateral forces. Summing the forces in the *y*-direction gives the following relationship.

$$F_{yf} + F_{yR} + F_{y,ext} = Ma_y \tag{3.1}$$

The a_y term includes the derivative of the lateral velocity (v), one of the state variables. This is derived by looking at the velocity vector of the entire vehicle, V_v [11].

$$\vec{V}_v = u\vec{i} + v\vec{j} \tag{3.2}$$

Differentiate $\vec{V}_v$ to get the acceleration vector $\vec{A}$, which consists of a_x and a_y.

$$\vec{A} = \frac{du}{dt}\vec{i} + \frac{dv}{dt}\vec{j} + u\frac{d\vec{i}}{dt} + v\frac{d\vec{j}}{dt} \tag{3.3}$$

Additionally,

$$\frac{d\vec{i}}{dt} = r\vec{k} \times \vec{i} = r\vec{j}$$
$$\frac{d\vec{j}}{dt} = r\vec{k} \times \vec{j} = -r\vec{i}$$

therefore, the following equations give us the formula for a_y, using dot notation.

$$\vec{A} = (\dot{u} - vr)\vec{i} + (\dot{v} + ur)\vec{j} \tag{3.4}$$

$$a_y = \dot{v} + ur \tag{3.5}$$

Substituting the expressions for F_{yf}, F_{yR}, and $F_{y,ext}$ into Eqn. 3.1 and rearranging gives,

$$-C_{\alpha f}\alpha_f - C_{\alpha R}\alpha_R + W\sin(\theta) = M\dot{v} + Mur \tag{3.6}$$

Substituting for α_f, and α_R and collecting like terms results in Eqn. 3.7.

$$M\dot{v} + \left(\frac{C_{\alpha f}}{u} + \frac{C_{\alpha R}}{u}\right)v + \left(Mu + \frac{C_{\alpha f}a}{u} - \frac{C_{\alpha R}b}{u}\right)r = C_{\alpha f}\delta_f + C_{\alpha R}\delta_R + W\theta \tag{3.7}$$

Further algebraic manipulation results in Eqn. 3.8, providing the first state equation as presented earlier in Chapter 2,

$$\dot{v} + a_1 v + a_2 r = a_3 \delta_f + a_4 \delta_R + \frac{W}{M}\theta \tag{3.8}$$

where,

$$\begin{aligned} a_1 &= \frac{C_{\alpha f} + C_{\alpha R}}{Mu} \\ a_2 &= \frac{Mu^2 + C_{\alpha f} a - C_{\alpha R} b}{Mu} \\ a_3 &= \frac{C_{\alpha f}}{M} \\ a_4 &= \frac{C_{\alpha R}}{M} \end{aligned} \tag{3.8a}$$

The second state equation is derived by initially summing the moments about the CG, resulting in

$$F_{yf} a - F_{yR} b = I\dot{r} \tag{3.9}$$

The yaw moment of inertia, I, is approximated using the widely accepted formula [11, 12], where M is the mass of the vehicle and L is the wheel base.

$$I = \frac{ML^2}{4} \tag{3.10}$$

Substituting for values of lateral forces, the second governing equation of motion becomes

$$I\dot{r} + \left(\frac{C_{\alpha f} a^2 + C_{\alpha R} b^2}{u} \right) r + \left(\frac{C_{\alpha f} a - C_{\alpha R} b}{u} \right) v = C_{\alpha f} a \delta_f - C_{\alpha R} b \delta_R \tag{3.11}$$

Or, rearranging,

$$\dot{r} + b_1 v + b_2 r = b_3 \delta_f + b_4 \delta_R \tag{3.12}$$

where,

$$\begin{aligned} b_1 &= \frac{C_{\alpha f} a - C_{\alpha R} b}{Iu} \\ b_2 &= \frac{C_{\alpha f} a^2 + C_{\alpha R} b^2}{Iu} \\ b_3 &= \frac{C_{\alpha f} a}{I} \\ b_4 &= \frac{-C_{\alpha R} b}{I} \end{aligned} \tag{3.12a}$$

The remaining two state equations and overall system matrix representation are repeated here for continuity.

$$\dot{Y} = u\psi + v \tag{3.13}$$

$$\dot{\psi} = r \tag{3.14}$$

$$\begin{bmatrix} \dot{r} \\ \dot{v} \\ \dot{\psi} \\ \dot{Y} \end{bmatrix} = \begin{bmatrix} -b_2 & -b_1 & 0 & 0 \\ -a_2 & -a_1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & u & 0 \end{bmatrix} \begin{bmatrix} r \\ v \\ \psi \\ Y \end{bmatrix} + \begin{bmatrix} b_3 & b_4 & 0 \\ a_3 & a_4 & W/M \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_f \\ \delta_R \\ \theta \end{bmatrix} \tag{3.15}$$

3.2 Roots and Stability of System

Stability is a critical component of vehicle design and operation. This section discusses the stability of the vehicle alone. The stability of the controlled vehicle system is discussed in later sections and chapters.

To solve for the roots of the system, the right hand side of Eqns. 3.8 and 3.12 are set to zero and the equations are put in the Laplace domain yielding,

$$(s+a_1)V + a_2 R = 0 \tag{3.16}$$

$$(s+b_2)R + b_1 V = 0 \tag{3.17}$$

Solving Eqn. 3.17 for R, substituting for R in Eqn. 3.16 gives,

$$(s+a_1)V - a_2\left(\frac{b_1}{s+b_2}\right)V = 0 \tag{3.18}$$

Now, dividing both sides by V, multiplying by $(s + b_2)$ and collecting terms of s results in the characteristic equation

$$s^2 + (a_1 + b_2)s + (a_1 b_2 - a_2 b_1) = 0 \tag{3.19}$$

Using the quadratic formula, the roots of the characteristic equation become

$$s = \frac{-(a_1 + b_2) \pm \sqrt{(a_1 + b_2)^2 - 4(a_1 b_2 - a_2 b_1)}}{2} \tag{3.20}$$

System stability exists only for negative real roots, either two real values or complex conjugate pairs. The real values for complex roots are always negative since

$$a_1 + b_2 = \left(\frac{C_{\alpha f} + C_{\alpha R}}{Mu} + \frac{C_{\alpha f} a^2 + C_{\alpha R} b^2}{Iu}\right) > 0 \tag{3.21}$$

For two real roots, instability occurs when

$$(a_1 + b_2) < \sqrt{(a_1 + b_2)^2 - 4(a_1 b_2 - a_2 b_1)} \tag{3.22}$$

After some substitution for constants and algebraic manipulation, Eqn. 3.22 becomes

$$1 + \frac{Ku^2}{Lg} < 0 \tag{3.23a}$$

$$\frac{Ku^2}{Lg} < -1 \tag{3.23b}$$

where K is the understeer gradient defined as

$$K = \frac{W_f}{C_{\alpha f}} - \frac{W_R}{C_{\alpha R}} \tag{3.24}$$

Eqn. 3.23b occurs when $K < 0$ since all other variables are positive. In that case, the system is unstable when

$$K < \frac{-Lg}{u^2} \tag{3.25}$$

This happens when u is above the so-called critical speed

$$u_{crit} = \sqrt{\frac{Lg}{|K|}} \tag{3.26}$$

In summary, the vehicle is stable for a positive understeer gradient; however, for $K < 0$, so-called oversteer, the vehicle is unstable when $u > u_{crit}$. For on-road vehicles, K is always positive and there is no critical speed; however, for off-road vehicles, such as agricultural equipment, a typical K and related critical speed are difficult to pinpoint because of the difference in tires, weight, and CG positions. Chapter 4 shows simulation for a wide range of K values to demonstrate the robust capabilities of the control system.

3.3 Steady state solutions

The solutions to the equations of motion at steady state are interesting to examine because they offer a quick insight to the final vehicle orientation and steer angle position by calculating the steady state yaw rate (r_{ss}), sideslip (β_{ss}), and steer angle, (δ) to follow a given path. At steady state, the derivative terms of Eqns. 3.8 and 3.12 go to zero and the system can be defined in matrix form by Eqn. 3.28.

$$\begin{bmatrix} a_1 & a_2 \\ b_1 & b_2 \end{bmatrix} \begin{bmatrix} v_{ss} \\ r_{ss} \end{bmatrix} = \begin{bmatrix} a_3\delta_f + a_4\delta_R + \dfrac{W}{M}\theta \\ b_3\delta_f + b_4\delta_R \end{bmatrix} \tag{3.28}$$

Using Cramer's rule to solve for r_{ss} and substituting values in for the constants gives

$$r_{ss} = \frac{u/L\left[\left(\delta_f - \delta_R\right) + K\theta\right]}{1 + \dfrac{Ku^2}{Lg}} \tag{3.29}$$

The resulting steady state value agrees with the literature [1] when the side slope, θ, is zero. For a curved path,

$$\frac{r}{u} = \frac{1}{R} \tag{3.30}$$

where R is the radius of the steady state turn. Substituting this relationship into the steady state yaw rate (Eqn. 3.29) and rearranging gives the steady state steer equation

$$\delta_f - \delta_R = \frac{L}{R} + K\left(a_y - \theta\right) \tag{3.31}$$

where a_y is the lateral acceleration in g's defined by

$$a_y = \frac{r^2 R}{g} = \frac{u^2}{Rg} \tag{3.32}$$

Note that tracking a straight line requires R to equal infinity and a_y to equal zero. This yields

$$(\delta_f - \delta_R) = -K\theta \tag{3.33}$$

This is a very important result, as it shows that kinematic analysis, which requires $(\delta_f - \delta_R) = 0$ to track a straight line, is not sufficient for a sloped terrain. In the case of a rear steer vehicle such as the combine, kinematic analysis concludes the steady state steer angle is zero when tracking a straight line on a hill, whereas Eqn. 3.33 clearly shows that with the inclusion of the side slope forces,

$$\delta_R = K\theta \tag{3.34}$$

Another key steady state characteristic to examine is the steady state sideslip angle, which shows the final vehicle orientation. From the definition of sideslip, if $\beta_{ss} \neq 0$, then the vehicle heading is not pointed along the desired navigation path. The desired path in this study is the X-axis; therefore, in this case

$$\beta_{ss} = -\psi_{ss} \tag{3.35}$$

Steady state vehicle orientation is important in agricultural applications. Figure 3.2 shows two combine orientations: a combine tracking a straight line with no sideslip and a combine tracking a line with steady state sideslip. The so-called header track, T_h, determines the width of the harvesting path. In the presence of sideslip, the effective header track decreases while the overall path width increases. Since the header is no longer properly aligned with the crop rows, it may potentially damage the crop or be unable to harvest correctly. This will occur when the sideslip is large and the outside edges of the header or the vehicle tires run over the crop.

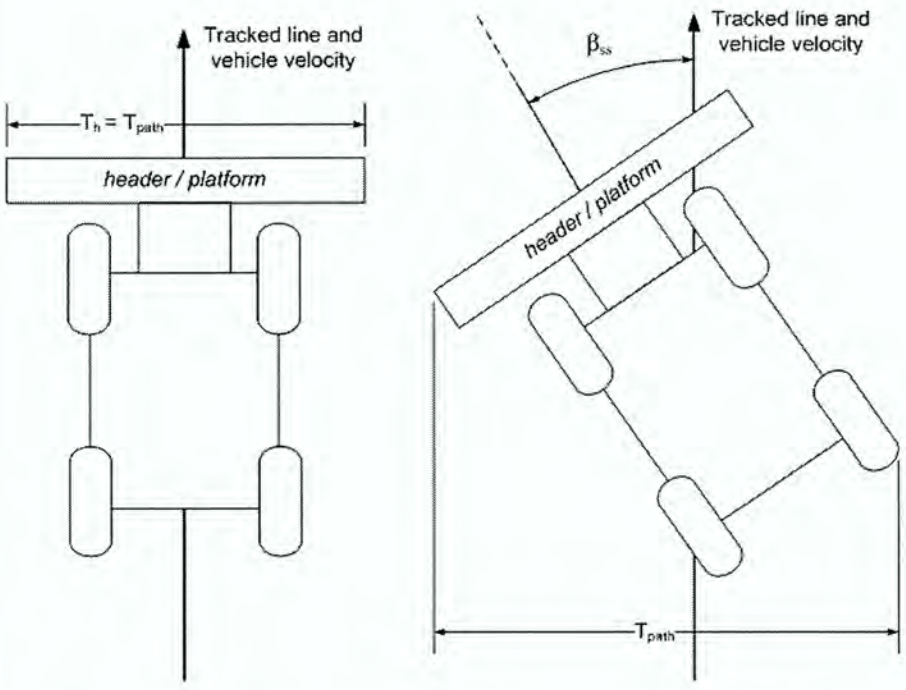


Figure 3.2 Combine orientation tracking line with and without steady state sideslip

The sideslip is found by determining steady state lateral velocity and then applying Eqn. 2.5. To find the steady state when tracking a straight line, begin by setting r_{ss} to zero in Eqn. 3.28. Solving for steady state lateral sideslip in this manner results in

$$\beta_{ss} = \frac{a_3}{a_1}\delta_f + \frac{a_4}{a_1}\delta_R + \frac{W/M}{a_1}\theta$$

$$\beta_{ss} = \frac{C_{\alpha f}}{C_{\alpha f} + C_{\alpha R}}\delta_f + \frac{C_{\alpha R}}{C_{\alpha f} + C_{\alpha R}}\delta_R + \frac{W}{C_{\alpha f} + C_{\alpha R}}\theta \tag{3.36}$$

and

$$\beta_{ss} = \frac{b_3}{b_1}\delta_f + \frac{b_4}{b_1}\delta_R$$

$$\beta_{ss} = \frac{C_{\alpha f} a}{C_{\alpha f} a - C_{\alpha R} b}\delta_f - \frac{C_{\alpha R} b}{C_{\alpha f} a - C_{\alpha R} b}\delta_R \tag{3.37}$$

Note that Eqn. 3.37 is not an appropriate expression when $K = 0$ because the denominator of both the δ_f and δ_R terms goes to zero.

To demonstrate the value of this steady state equation, consider the specific case of the rear steer combine tracking a straight line on a side slope. Setting δ_f to zero and substituting the resulting expression for δ_R from Eqn. 3.34 gives

$$\beta_{ss} = \left(\frac{C_{\alpha R} K + W}{C_{\alpha f} + C_{\alpha R}} \right) \theta \tag{3.38}$$

With a little more algebraic manipulation, this becomes

$$\beta_{ss} = \frac{W_f}{C_{\alpha f}} \theta \tag{3.39}$$

or, using the definition of the cornering coefficient,

$$\beta_{ss} = \frac{1}{c_{cf}} \theta \tag{3.40}$$

Note that units for all variables must be in terms of radians. The variable c_{cf} represents the cornering coefficient as the sum of both left and right front tires. The importance of this steady state sideslip and side slope relationship is that given a sloped terrain, the vehicle will not face straight down the desired path. Some sideslip must exist in the presence of the slope and the amount present is solely determined by the tire parameters and side slope angle. A typical expected range for β_{ss} would be between 0 and 4 degrees for slopes up to 10% and c_{cf} values in the Metz range. These results show the steady state solutions for steer angle and sideslip are independent of the controller; thus, controllers could be swapped to obtain different transient responses, but the steady state values will remain the same.

3.4 System Transfer Functions

3.4.1 Derivation using block diagrams

An advantage of using block diagrams is that they can be easily manipulated to create a simpler system representation. This leads to the formation of a transfer function, which is a useful tool to look at system characteristics and response to an input. Many steps are needed in order to arrive at the final transfer function and this section focuses on the major development. A step by step reduction process is shown in Appendix B. Figure 3.3 redisplays the general block diagram representation of the entire system shown in Figure 2.5. Note that only the rear steer angle feeds into the vehicle dynamics for the specific case of the combine.

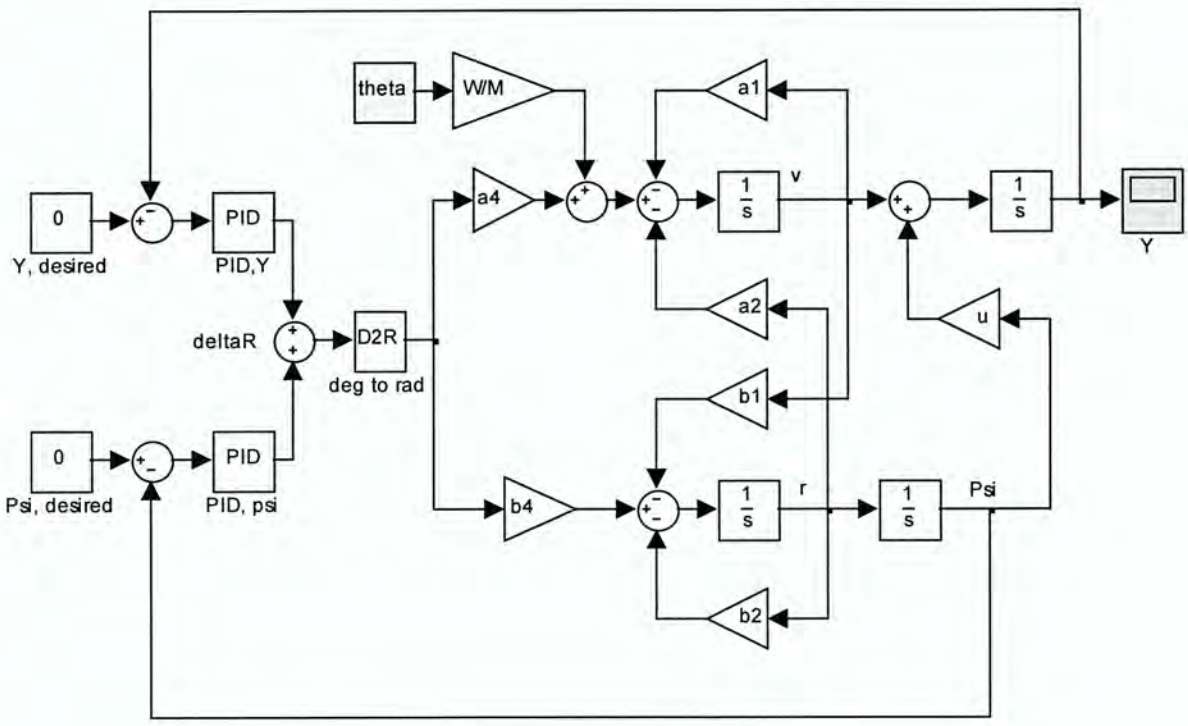


Figure 3.3 Simulink® block diagram representation before simplification

Further system reduction requires all state variables to be related to the input. Eqns. 3.41 and 3.42 show the relationship between r and v obtained by redefining the state equations (Eqns. 3.8 and 3.12) in the Laplace domain and solving for R and V.

$$V = -\frac{a_2}{(s+a_1)}R + \frac{a_3}{(s+a_1)}\delta_f + \frac{a_4}{(s+a_1)}\delta_R + \frac{W/M}{(s+a_1)}\theta \tag{3.41}$$

$$R = -\frac{b_1}{(s+b_2)}V + \frac{b_3}{(s+b_2)}\delta_f + \frac{b_4}{(s+b_2)}\delta_R \tag{3.42}$$

Substituting the expression for V into R and redrawing the diagram to reflect this change results in the new system configuration shown in Figure 3.4.

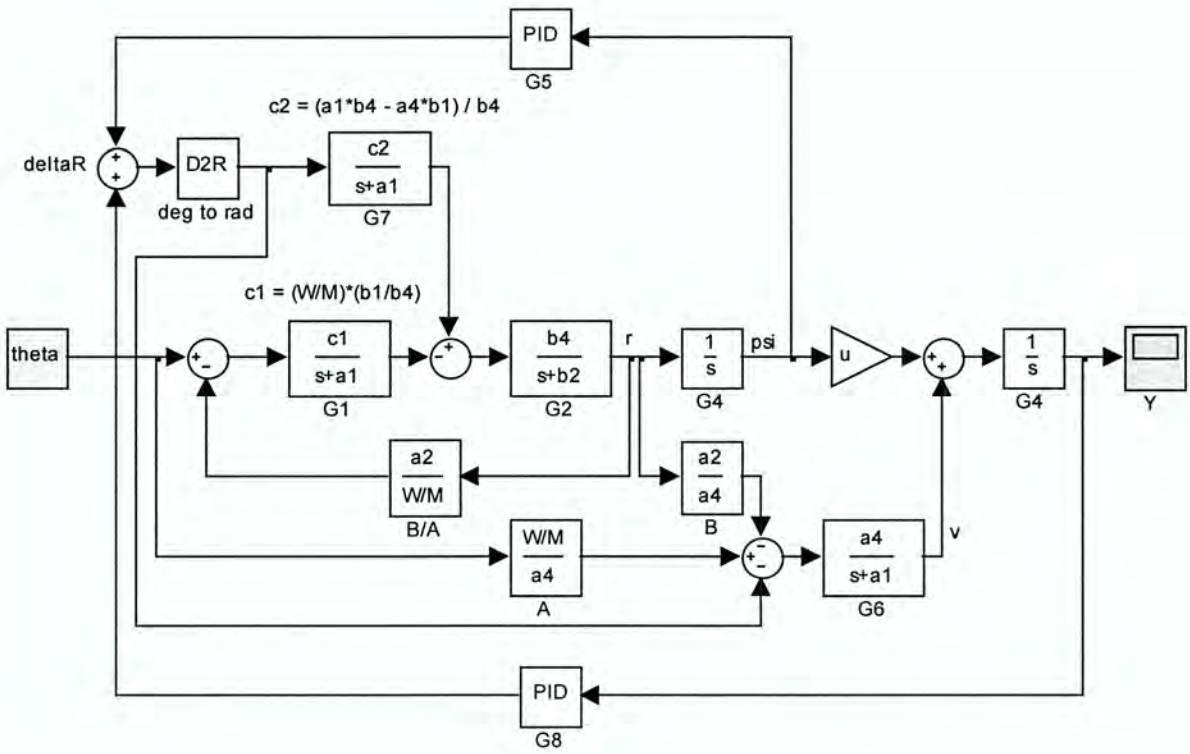


Figure 3.4 Overall vehicle steering control system block diagram

The rear steer summing junction can be eliminated by rearranging output signal to feed through to other summing junctions in the diagram. Likewise, the summing junction to the left of block G6 can be eliminated. Using the labels underneath the blocks (G1 through G8, A, B, B/A) and the constants c1 and c2 shown in Figure 3.4, the diagram is reconfigured in pure symbolic form with r and v incorporated into the overall system as functions of the input Y and the output θ (see Figure 3.5). Note the D2R (degrees to radians) conversion block has been lumped into blocks G5 and G8.

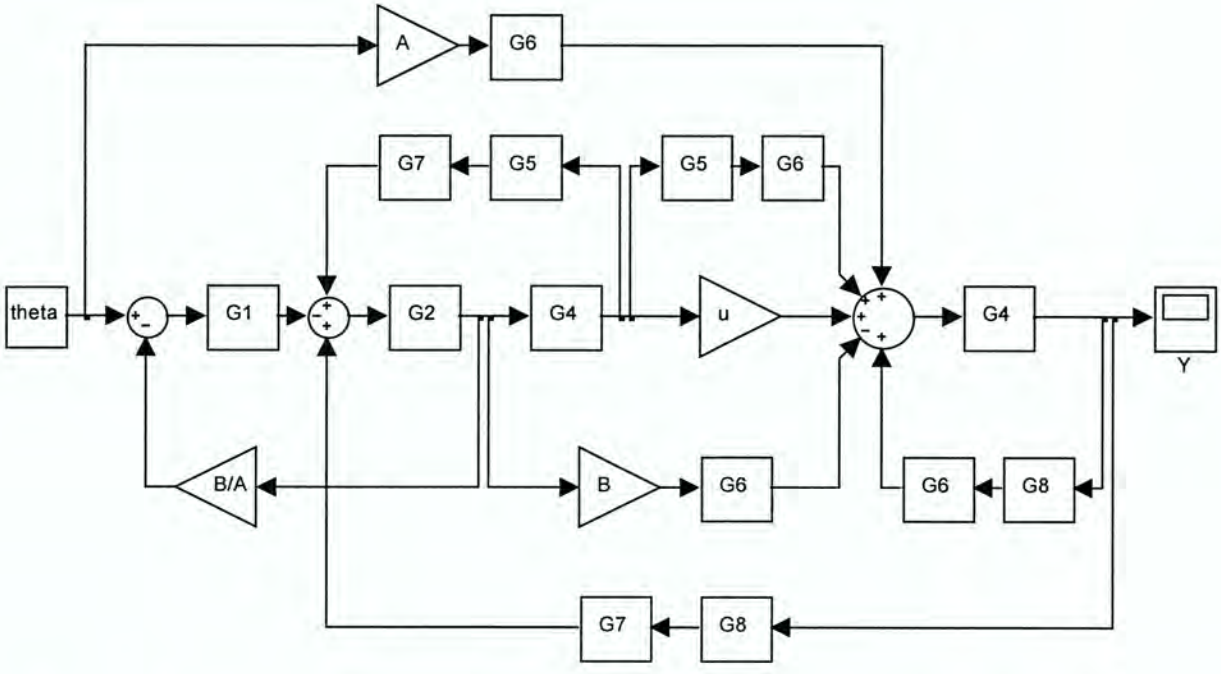


Figure 3.5 System redrawn with state variables as functions of the output and input

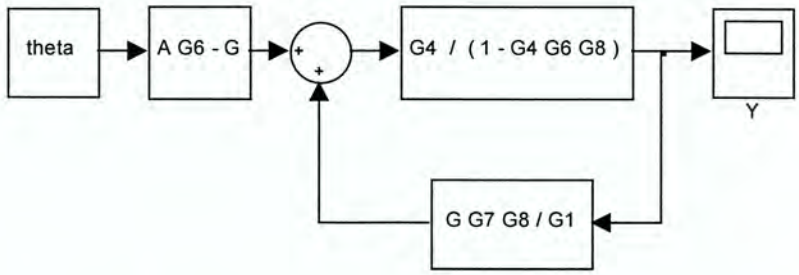


Figure 3.6 Feedback loop shown symbolically for Y/θ

A series of block diagram algebra operations, which are shown in their entirety in the Appendix, leads to the simple feedback loop shown in Figure 3.6. Reducing the feedback loop to a single transfer function gives the final input/output relationship shown in two forms in Eqn. 3.43 and 3.45.

$$\frac{Y}{\theta} = \frac{AG_1G_4G_6 - GG_1G_4}{G_1 - G_1G_4G_6G_8 - GG_4G_7G_8} \tag{3.43}$$

where

$$G = \frac{G_1G_2G_4u - G_1G_2G_4G_5G_6 - BG_1G_2G_6}{1 - G_1G_2\dfrac{B}{A} - G_2G_4G_5G_7} \tag{3.44}$$

Many symbolic variables are used to maintain the readability of the equations. The complexity of combining all variables is left to the computer simulation. The next chapter shows the numeric representation of the final system with substituted values for the vehicle and controller parameters.

Replacing $G_{1\text{-}8}$ with their respective transfer functions and simplifying the numerator and denominator results in

$$\frac{Y}{\theta} = \frac{g\left(s^4 + NN_3 s^3 + NN_2 s^2 + NN_1 s + NN_0\right)}{s^6 + DD_6 s^5 + DD_5 s^4 + DD_4 s^3 + DD_3 s^2 + DD_2 s + DD_1} \tag{3.45}$$

where,

$$\begin{aligned} NN_3 &= K_{d\psi} m_1 + z_1 \\ NN_2 &= K_{p\psi} m_1 + K_{d\psi} z_2 + z_3 \\ NN_1 &= K_{p\psi} z_2 + K_{i\psi} m_1 + z_4 \\ NN_0 &= K_{i\psi} z_2 \end{aligned} \tag{3.45a}$$

$$\begin{aligned} DD_6 &= K_{d\psi} m_1 + K_{d\gamma} m_2 + m_3 \\ DD_5 &= K_{d\psi} m_4 + K_{p\psi} m_1 + K_{p\gamma} m_2 + K_{d\gamma} m_5 + m_6 \\ DD_4 &= K_{p\psi} m_4 + K_{i\psi} m_1 + K_{d\psi} m_7 + K_{i\gamma} m_2 + K_{d\gamma} m_8 + K_{p\gamma} m_5 + m_9 \\ DD_3 &= K_{i\psi} m_4 + K_{p\psi} m_7 + K_{d\gamma} m_{10} + K_{p\gamma} m_8 + K_{i\gamma} m_5 \\ DD_2 &= K_{i\psi} m_7 + K_{p\gamma} m_{10} + K_{i\gamma} m_8 \\ DD_1 &= K_{i\gamma} m_{10} \end{aligned} \tag{3.45b}$$

and the constants $m_{1\text{-}10}$ and $z_{1\text{-}4}$ are constants containing $a_{1\text{-}4}$ and $b_{1\text{-}4}$ defined in Section 3.1 by Eqn. 3.8a and 3.12a. The variable x represents the degrees to radians conversion for δ_R.

$$\begin{aligned} z_1 &= a_1 + b_2 \\ z_2 &= -x a_1 b_4 \\ z_3 &= a_1 b_2 - b_1 u \\ z_4 &= -b_1 u a_1 \end{aligned} \tag{3.45c}$$

$$
\begin{aligned}
m_1 &= -xb_4 \\
m_2 &= -xa_4 \\
m_3 &= 2a_1 + b_2 \\
m_4 &= xa_4b_1 - 2xa_1b_4 \\
m_5 &= -xa_1a_4 - xa_4b_2 - xb_4u + xa_2b_4 \\
m_6 &= 2a_1b_2 - a_2b_1 + a_1^2 \\
m_7 &= xa_1a_4b_1 - xa_1^2b_4 \\
m_8 &= -xa_1a_4b_2 + xa_4b_1u - 2xa_1b_4u + xa_1a_2b_4 \\
m_9 &= a_1^2b_2 - a_1a_2b_1 \\
m_{10} &= xa_1a_4b_1u - xa_1^2b_4u
\end{aligned}
\tag{3.45d}
$$

This final transfer function gives the lateral error of the steering control system when applied to a step input of slope assuming no initial conditions of Y or ψ. Overall system response requires the combination of an impulse response to initial condition and a step response to the sloped terrain. For this study, only the response to an initial lateral error, Y_0, was derived. Adding the initial lateral error is a simple modification of the final transfer function. The input now becomes Y_0 and θ goes to zero. The initial condition is added to the state equation for $\dot{Y}$ (Eqn. 3.13) as shown by taking the Laplace transform of Eqn. 3.13 and modifying the system block diagram in Figure 3.3 to represent this change. Eqn. 3.46 and Figure 3.7 demonstrate these changes.

$$
Y = \frac{1}{s}\left(u\psi + v + Y_0\right) \tag{3.46}
$$

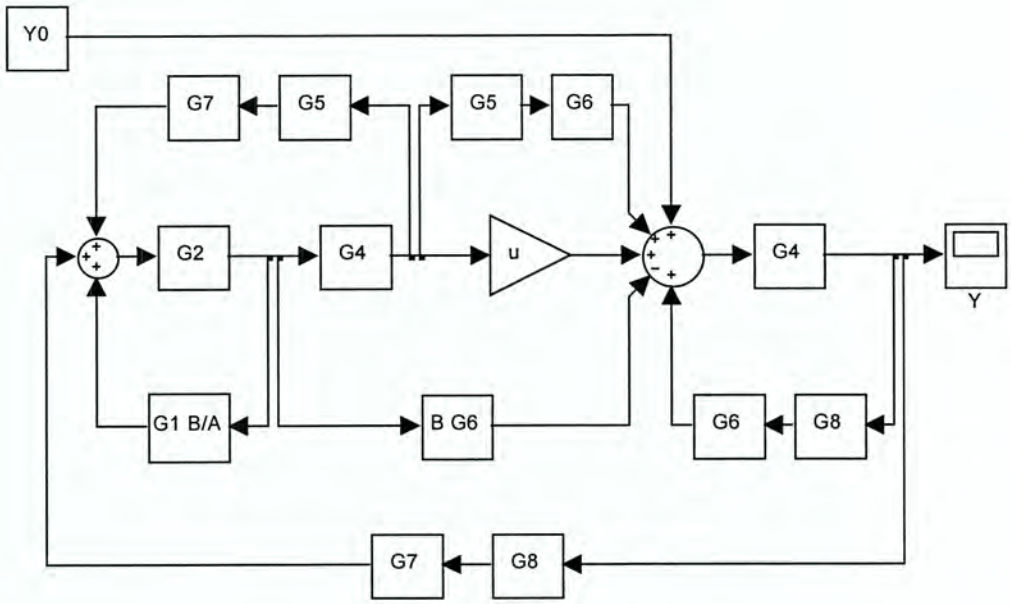


Figure 3.7 System block diagram for initial lateral error

This system representation reduces to a very similar final feedback loop as shown in Figure 3.6, with an input of Y_0 rather than $\theta(AG_6 - G)$. The ultimate result is also very similar; only the numerator of the final transfer function differs, as shown below.

$$\frac{Y}{Y_0} = \frac{s^5 + N_4 s^4 + N_3 s^3 + N_2 s^2 + N_1 s + N_0}{s^6 + DD_6 s^5 + DD_5 s^4 + DD_4 s^3 + DD_3 s^2 + DD_2 s + DD_1} \tag{3.47}$$

where

$$\begin{aligned} N_4 &= K_{d\psi} m_1 + m_3 \\ N_3 &= K_{p\psi} m_1 + K_{d\psi} m_4 + m_6 \\ N_2 &= K_{p\psi} m_4 + K_{i\psi} m_1 + K_{d\psi} m_7 + m_9 \\ N_1 &= K_{p\psi} m_7 + K_{i\psi} m_4 \\ N_0 &= K_{i\psi} m_7 \end{aligned} \tag{3.47a}$$

3.4.2 Generalized root locus of controller gains

The concept of generalized root locus isolates one particular system parameter and examines the closed-loop effect on the system as the parameter changes [4]. The objective is to factor out the desired parameter and rearrange the denominator to resemble an open-loop transfer function form. With the open-loop form, the root locus of the desired parameter can be drawn and the system response to a change in this parameter can be observed. The following equations present the generalized open-loop transfer functions (TK_{zz}) for each controller gain, where the subscript *zz* represents the type of gain and the associated error. The root locus for each gain is drawn from these equations. Note that since the denominator of Eqn. 3.47 is the same as Eqn. 3.45, the generalized root locus for the controller gains will be the same for both input/output relationships; i.e., the same gains will result in the same overall transient response.

$$TK_{p\psi} = \frac{K_{p\psi}\left(m_1 s^4 + m_4 s^3 + m_7 s^2\right)}{s^6 + DD_6 s^5 + \left(DD_5 - K_{p\psi} m_1\right)s^4 + \left(DD_4 - K_{p\psi} m_4\right)s^3 + \left(DD_3 - K_{p\psi} m_7\right)s^2 + DD_2 s + DD_1} \tag{3.48}$$

$$TK_{i\psi} = \frac{K_{i\psi}\left(m_1 s^3 + m_4 s^2 + m_7 s\right)}{s^6 + DD_6 s^5 + DD_5 s^4 + \left(DD_4 - K_{i\psi} m_1\right)s^3 + \left(DD_3 - K_{i\psi} m_4\right)s^2 + \left(DD_2 - K_{i\psi} m_7\right)s + DD_1} \tag{3.49}$$

$$TK_{d\psi} = \frac{K_{d\psi}\left(m_1 s^5 + m_4 s^4 + m_7 s^3\right)}{s^6 + \left(DD_6 - K_{d\psi} m_1\right)s^5 + \left(DD_5 - K_{d\psi} m_4\right)s^4 + \left(DD_4 - K_{d\psi} m_7\right)s^3 + DD_3 s^2 + DD_2 s + DD_1} \tag{3.50}$$

$$TK_{pY} = \frac{K_{pY}\left(m_2 s^4 + m_5 s^3 + m_8 s^2 + m_{10} s\right)}{s^6 + DD_6 s^5 + \left(DD_5 - K_{pY} m_2\right)s^4 + \left(DD_4 - K_{pY} m_5\right)s^3 + \left(DD_5 - K_{pY} m_8\right)s^2 + \left(DD_2 - K_{pY} m_{10}\right)s + DD_1}$$

(3.51)

$$TK_{iY} = \frac{K_{iY}\left(m_2 s^3 + m_5 s^2 + m_8 s + m_{10}\right)}{s^6 + DD_6 s^5 + DD_5 s^4 + \left(DD_4 - K_{iY} m_2\right)s^3 + \left(DD_3 - K_{iY} m_5\right)s^2 + \left(DD_2 - K_{iY} m_8\right)s}$$

(3.52)

$$TK_{dY} = \frac{K_{dY}\left(m_2 s^5 + m_5 s^4 + m_8 s^3 + m_{10} s^2\right)}{s^6 + \left(DD_6 - K_{dY} m_2\right)s^5 + \left(DD_5 - K_{dY} m_5\right)s^4 + \left(DD_4 - K_{dY} m_8\right)s^3 + \left(DD_3 - K_{dY} m_{10}\right)s^2 + DD_2 s + DD_1}$$

(3.53)

CHAPTER 4. SIMULATION RESULTS AND DISCUSSION

4.1 Introduction

Each of the three of system representations described in Section 2.2.4 could interchangeably be used to simulate the system and achieve comparable results. This chapter focuses on the use of the transfer function and MATLAB® numerical integration models to analyze the system. Appendix A provides a brief discussion regarding comparisons of results with the Simulink® model. The chapter is broken into several sections and subsections describing the use of the models to design the controller, analyze the response to typical values of combine vehicle parameters, and compare the response between uphill and downhill initial conditions.

The simulations show the response of a combine moving at a constant forward velocity of 10 mph subjected to a five degree slope. The desired path is a straight line track directly across the slope. The combine has an initial positive lateral error of 10 feet downhill from the track and no initial heading error. Table 4.1 defines the values for the base vehicle parameters used to design the controller and compare simulations. The forward speed represents a typical harvest speed. Vehicle dimensions and weight are average values for the John Deere Single-Tine Separation (STS) combines. The cornering stiffness values of the tires are approximate and are based on conditions explained in Section 2.2.2. John Deere steering controller specifications state the lateral error must be within 40% of total vehicle header track width for the system to engage. A 10 foot initial lateral error reflects this specification. Unless otherwise stated, all simulations use the baseline parameters given in Table 4.1.

Table 4.1 Vehicle and simulation baseline parameters

Parameter	Value and units
a	2.3 ft
b	9.2 ft
c_c	0.06 for front tires, 0.06 for rear tires (deg^{-1})
$C_{\alpha f}$	1632 lbs/deg
$C_{\alpha R}$	408 lbs/deg
I	34,911 slugs-ft^2
L	11.5 ft
M	1055.9 slugs
u	10 mph
W	34,000 lbs including header, empty grain tank
W_f	27,200 lbs
W_R	6,800 lbs
Y_0	10 ft
θ	5 degrees, 8.7% grade

4.2 Controller design

This study uses the transfer function and the associated generalized root loci for the controller gains to design the system controller. The root locus is a plot of the collection of possible roots for the system shown on the complex plane. These plots show how the stability and response of the system changes as the value of the gain changes. The controller is designed by selecting gain values determined from the root locus to yield desirable system behavior. For a navigation guidance system, little to no overshoot and fast settling time are transient response characteristics of importance. Settling time is the time required for the transient oscillations to reach and stay within 2% of the steady state value [4]. The MATLAB Control Systems Toolbox includes an interactive root locus plotter that provides information about damping, percent overshoot, and gain values for each root location. This

tool helps determine gains that yield stable and desirable system behavior. The design process requires multiple runs, as parameter modifications change the individual generalized root locus plots.

The final system transfer functions from Eqns. 3.45 and 3.47 show that in order to have zero steady state lateral error, the integral heading gain must be set to zero. At steady state, the s terms go to zero and the input output relationships become

$$\frac{Y}{\theta} = \frac{K_{i\psi} z_2}{K_{iY} m_{10}}$$

and

$$\frac{Y}{Y_0} = \frac{K_{i\psi} m_7}{K_{iY} m_{10}}$$

If $K_{i\psi}$ is not equal to zero, $Y = P\theta$, where P is some non-zero constant. This would seem to represent a tradeoff between obtaining zero lateral error and zero steady state error; however, the steady state solutions derived in Chapter 2 show that steady state sideslip, thus steady state heading does not depend on the controller (see Eqns. 3.35 and 3.40).

In summary, no controller will compensate for the steady state heading error on sloped terrain for a rear steer vehicle, as the error is related only to the slope angle, the tire cornering stiffness, and load on the tire. Zero steady state heading error would require a four wheel steer vehicle. The ability to control all wheels would result in lateral forces that would create moments about the CG to straighten the vehicle orientation and is a topic for another study. The simulations in this study focus on obtaining zero steady state off-track error and then

observing the steady state heading error. Results show that steady state heading error is minimal for typical vehicle parameters and slope conditions.

The initial simulation sets $K_{i\psi}$ to zero and all controller gain values to one. These values are arbitrarily selected to serve as a baseline for design. Examination of the root loci shows that the system is unstable for this combination. The interactive root locus plots help find gains that lead to stable behavior and desirable transient response. Each individual generalized root locus shows the response to changes in only the selected gain. All other controller and vehicle parameters are held constant and are equal to the value set at the beginning of the simulation. For example, in the initial simulation, the generalized root locus plot for K_{pY} shows the system response as K_{pY} ranges from zero to infinity while $K_{i\psi}$ equals zero, all other gains equal one, and vehicle parameters remain constant. Multiple runs and root loci observations lead to a set of optimized gain values.

Table 4.2 shows a set of optimized controller gain values obtained from the root locus analysis. Figures 4.1 and 4.2 show the final generalized root locus for the controller gain and Figure 4.5 shows the final overall vehicle response to the parameters in Table 4.1.

Table 4.2 Optimized controller gain values

Heading Gains		***Lateral Gains***	
$K_{p\psi} =$	74.00	$K_{pY} =$	3.70
$K_{i\psi} =$	0.00	$K_{iY} =$	0.05
$K_{d\psi} =$	0.00	$K_{dY} =$	1.30

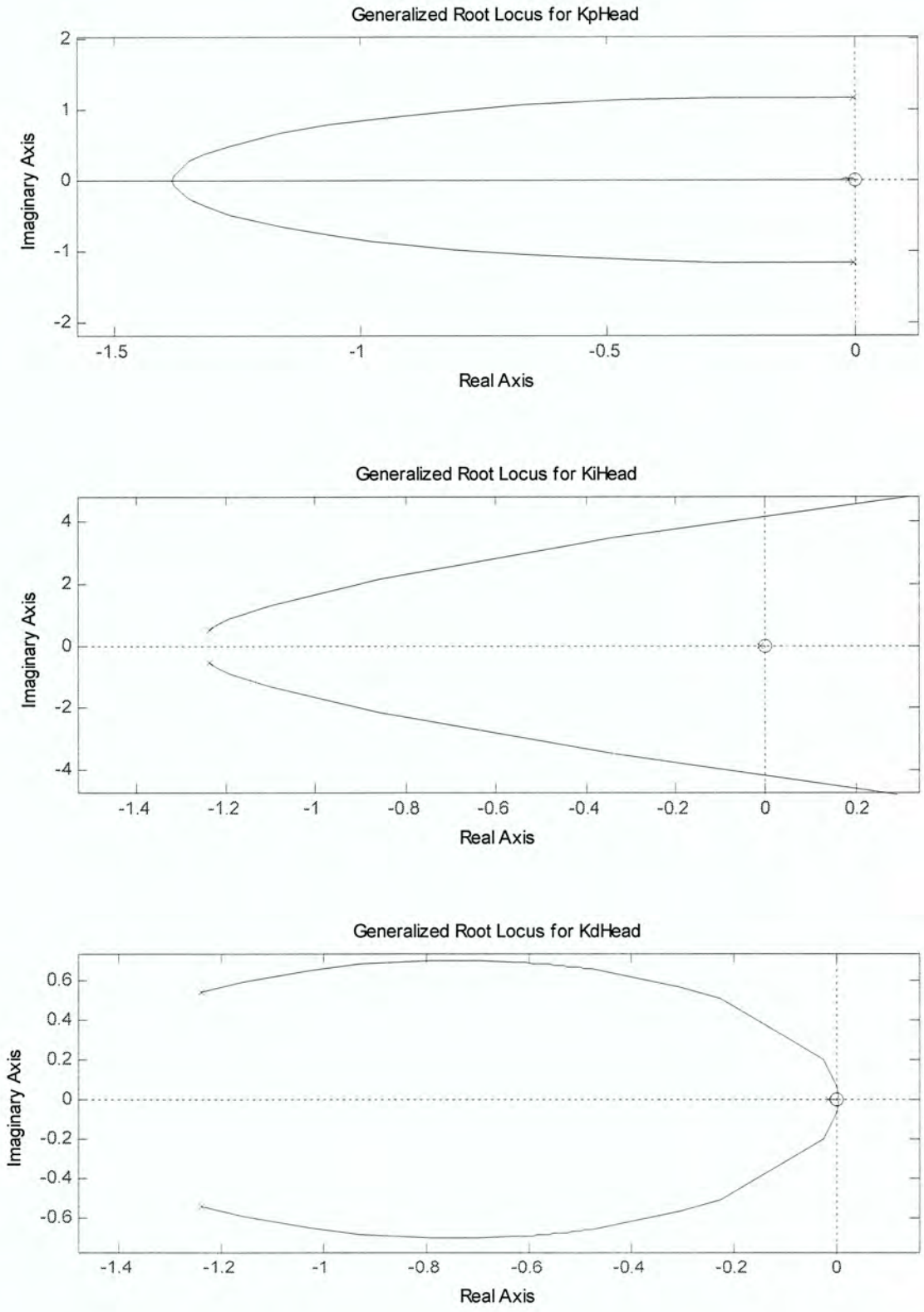


Figure 4.1 Final generalized root locus for heading error controller gains

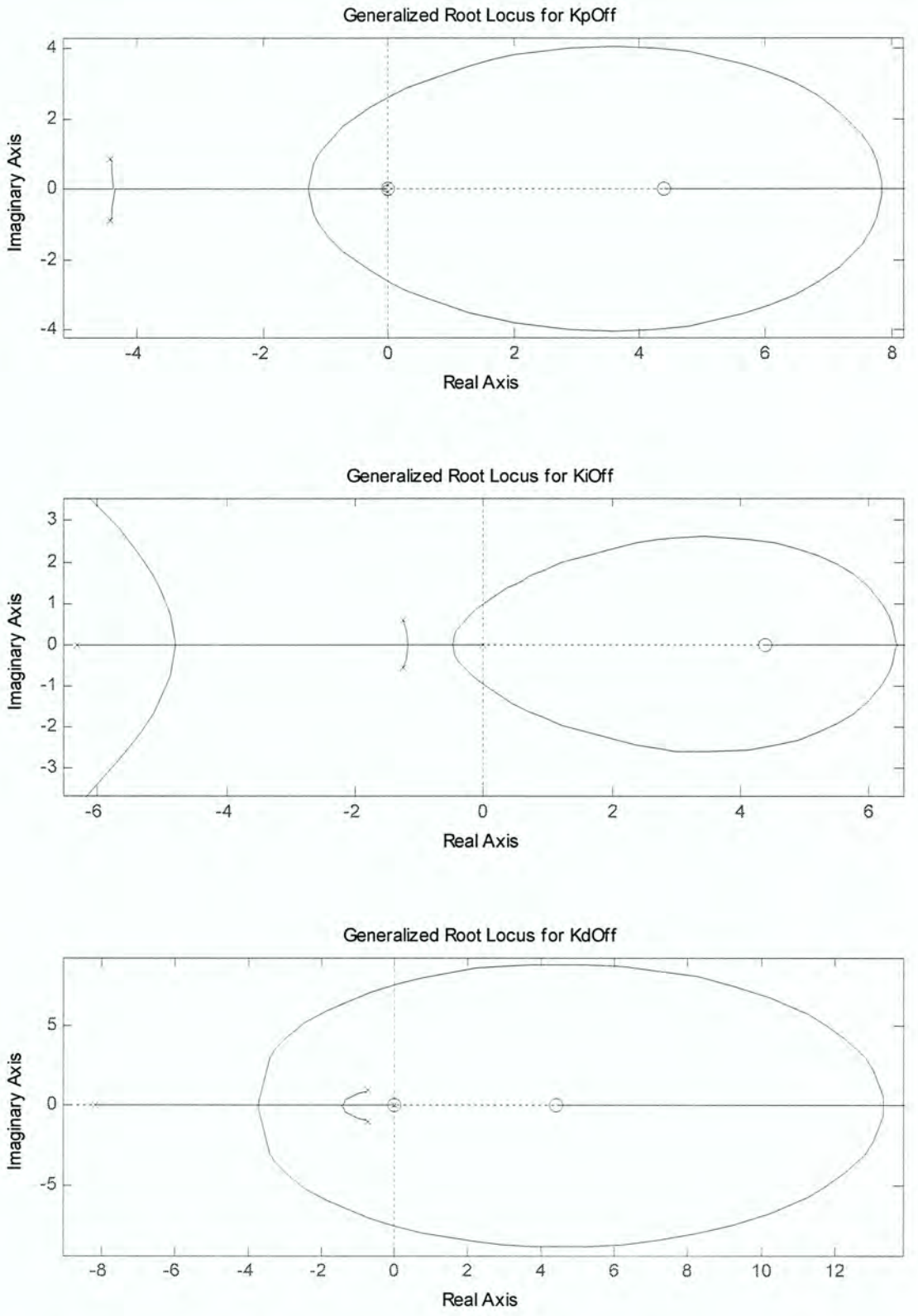


Figure 4.2 Final generalized root locus for lateral error controller gains

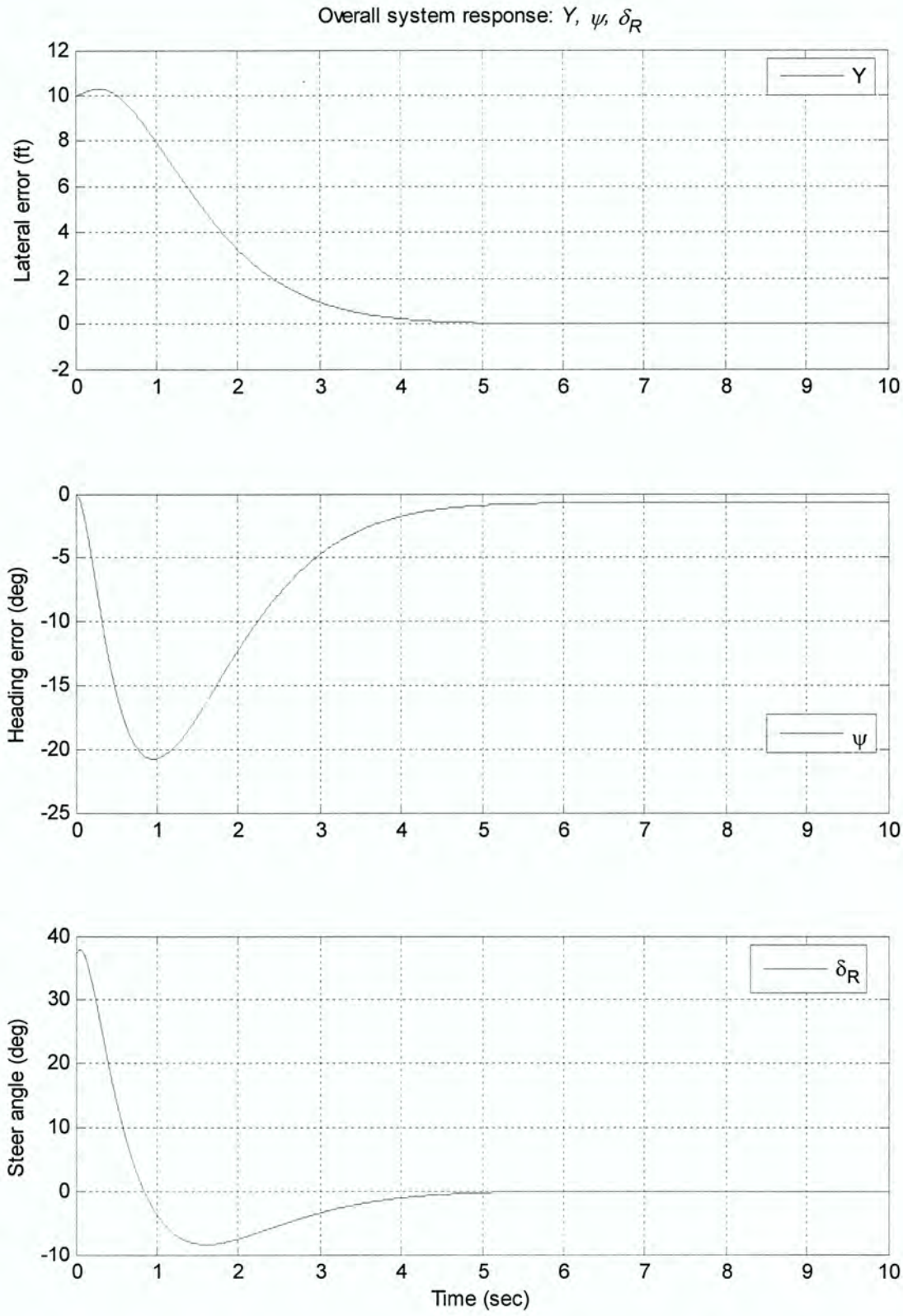


Figure 4.3 Final system response with initial downhill *Y* of 10 ft. on 5 deg slope

The final root locus plots show that while holding all other gains constant at the optimized values shown in Table 4.2, any gain value of $K_{p\psi}$ results in a stable system. The same is true for $K_{d\psi}$ as well. The plot for $K_{i\psi}$ shows that if an integral gain for heading is used, values greater than 644 would cause the system to be unstable. Likewise, system instability occurs for values of K_{pY} greater than 18.7, K_{iY} greater than 4.8, and K_{dY} greater than 13.2. Keep in mind that this is true only if all other gains remain constant at the optimized values in Table 4.2. Although the system is stable for the above mentioned range of each individual gain, transient response will vary significantly.

The final controller design yields desirable navigation controller response. The settling time is just under 5 seconds with no overshoot and no steady state off-track error. The steady state heading error is -0.73 degrees. The sharp initial slope results from the assumed instantaneous steer angle as mentioned in Section 2.2.1 and shown in the bottom of Figure 4.3. The CG position rise above the initial condition is due to the fact that the vehicle is rear steer.

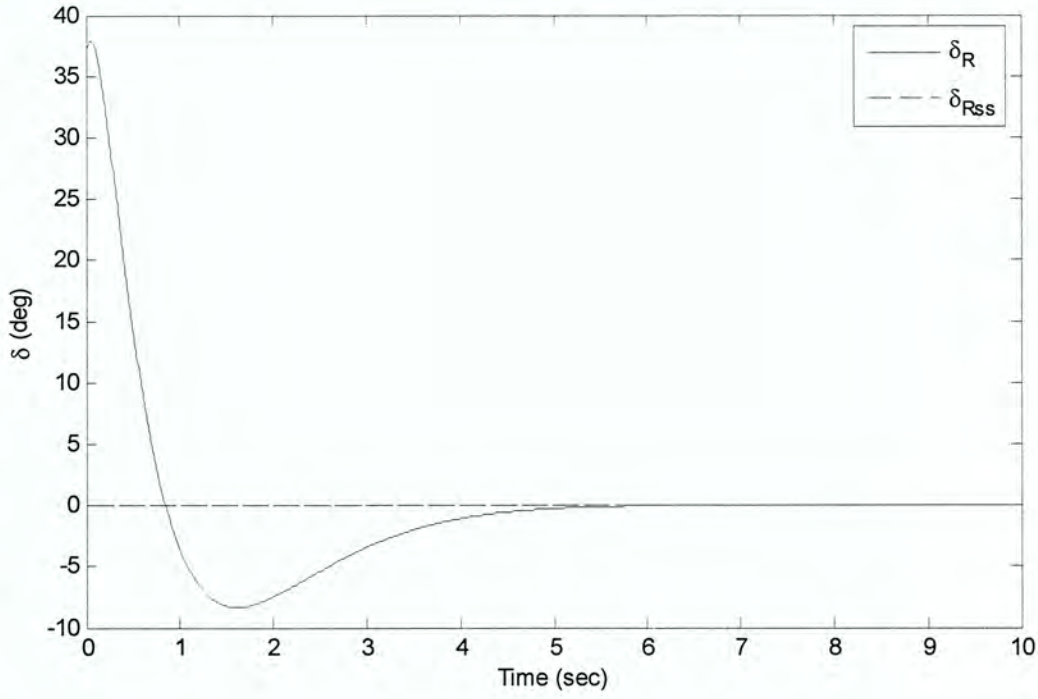
When the result of the transfer function model is compared to the result from the numerical integration, there is a slight difference which is discussed in Appendix A, Section A.1. Moving forward, it is sufficient to say that the difference is related to software algorithms and not mathematical dissimilarities. Both models may be and are used in this study with confidence.

4.3 Steady state equations comparison

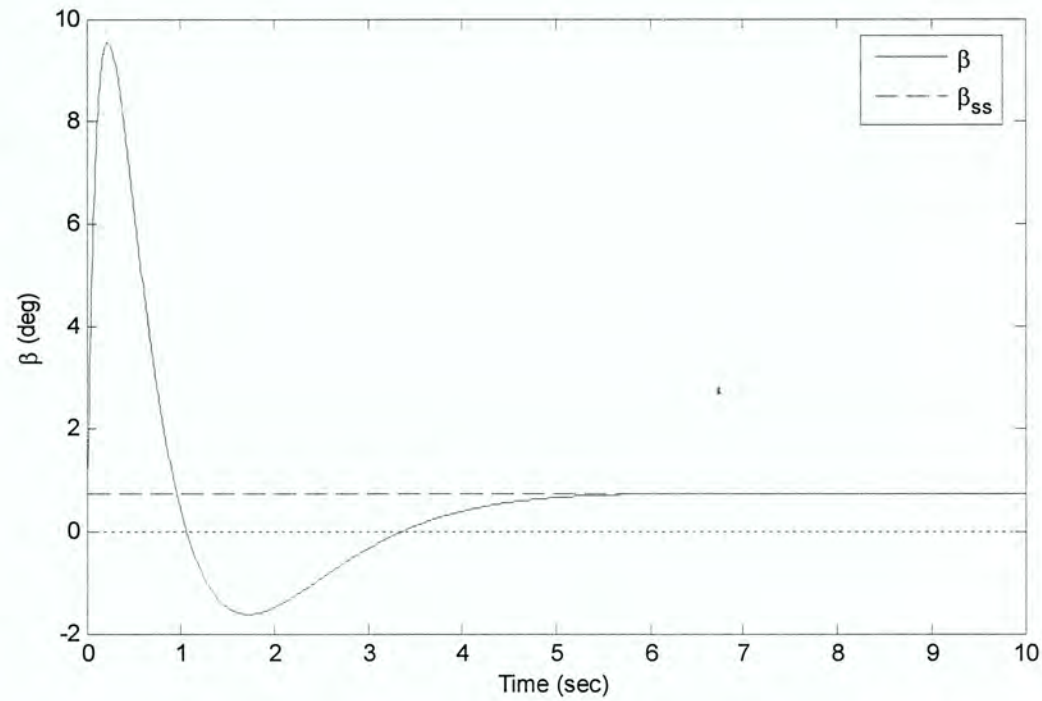
Section 3.3 derives the steady state solutions for yaw rate, lateral velocity, sideslip angle, and steer angle. One of the claims is that these steady state equations would enable immediate analysis of the steady state orientation of the vehicle without any simulation. This section verifies that claim by comparing the steady state equation with the simulated response. Figure 4.4 plots the response of two key steady state variables of interest, sideslip and steer angle, with the steady state value calculated from Eqns 3.34 and 3.40. The plot shows the steady state equations are effective predictors of the final vehicle orientation and steer angle. The steady state values are shown in Table 4.3 and indicate the vehicle has a small steady state sideslip angle. It is interesting to note that for constant cornering coefficients, the understeer gradient (K) goes to zero. This means no steady state steer angle. If the coefficients are not the same, the understeer gradient is non-zero, which results in steady state steer angle defined by Eqn. 3.33.

Table 4.3 Steady state values

r_{ss}	= 0.00 deg/sec
v_{ss}	= 0.19 ft/sec
β_{ss}	= 0.73 deg
$\delta_{R,ss}$	= 0.00 deg



(a)



(b)

Figure 4.4 Steady state response comparisons: (a) rear steer angle, (b) sideslip angle

4.4 System response to vehicle parameter changes

Section 4.2 shows the controller is effective for the typical operating parameters given in Table 4.1. This section now looks at the system response to changes in some of those parameters. The vehicle speed can vary depending on field and crop conditions and operator preference. The CG position changes as alternate headers and platforms are attached or as the grain tank fills with harvested crop. Likewise, the weight of the vehicle changes as grain enters the tank. The largest John Deere STS combine has a grain tank capacity of 300 bushels. At 56 – 60 lbs per bushel for corn, wheat, or beans, this results in an extra 18,000 lbs, about twice the weight of a combine with an empty tank. The tires can be replaced in the front or rear, and wheels can be added to the front, thus resulting in different cornering stiffness values and corresponding lateral forces. The controlled system must remain stable through these changes or be redesigned to achieve stability for different target parameters.

4.4.1 Effects of tire parameters

As discussed in the overview of the tire model in Chapter 2, the cornering stiffness has significant effect on vehicle dynamics. Figure 4.7 shows the closed-loop system roots with different values of $C_{\alpha f}$ and $C_{\alpha R}$ determined by the cornering coefficient, c_c. The plots are created by ranging one of the cornering coefficients from 0.00 to 0.20 deg^{-1} in increments of 0.005, while holding the other coefficient constant at 0.06 deg^{-1}. They are zoomed-in around the dominant roots and do not show some pairs further to the left on the negative real axis. The range gives cornering stiffness values of 0 – 5440 lbs/deg for each front tire and 0 – 1360 lbs/deg for each rear tire. The plots show that only for c_c values below approximately 0.005 for the front and 0.01 for the rear does the system become unstable. The full range of

cornering coefficients suggested by Metz ($c_c = 0.03 - 0.09$ deg^{-1}) results in a stable controlled system.

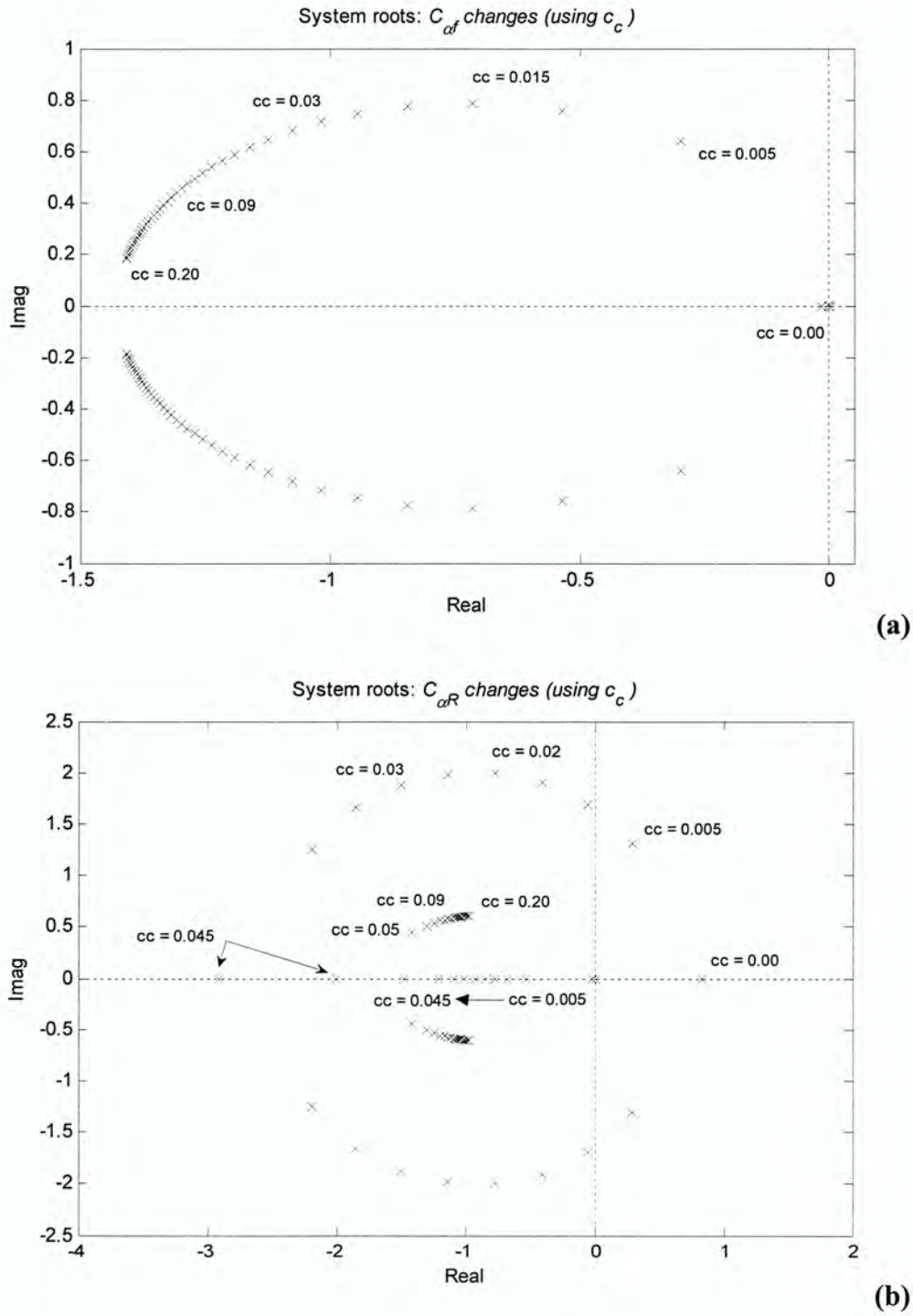


Figure 4.5 System roots for changes to $C_{\alpha f}$ (a) and $C_{\alpha R}$ (b) for $c_c = 0.00 - 0.20$ deg^{-1}

Figure 4.8 shows the overall system response to the Metz range of cornering coefficients. The front cornering stiffness range is 816 – 2448 lbs/deg and the rear cornering stiffness range is 204 – 612 lbs/deg for each tire. The plots are created by holding one coefficient constant while varying the other.

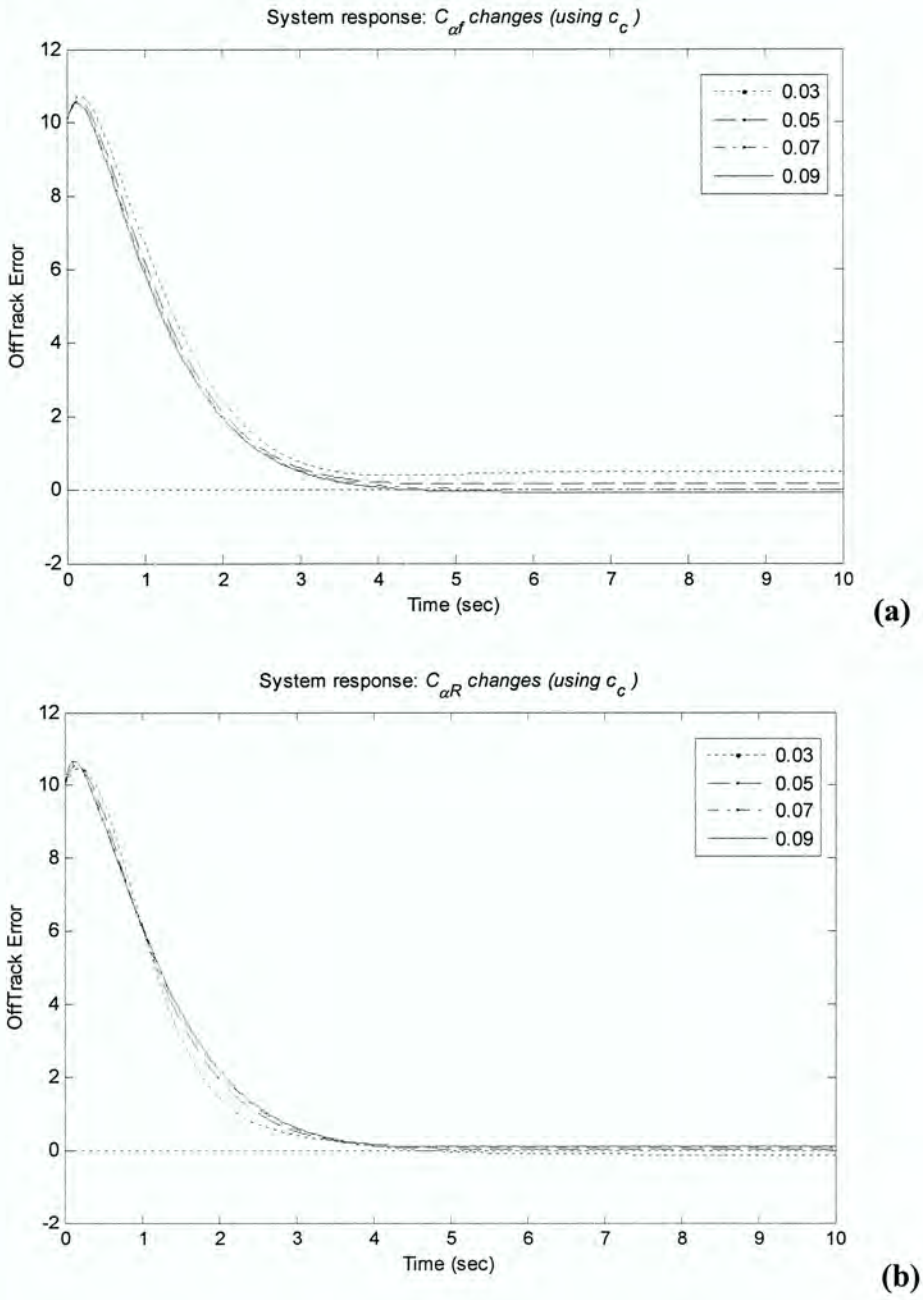


Figure 4.6 System response for changes to $C_{\alpha f}$ (a) and $C_{\alpha R}$ (b) for $c_c = 0.03 - 0.09$ deg^{-1}

For the most part, changes in tire parameters within the applicable range do not significantly affect the response. Examination of the steady state error after 10 seconds reveals a slight difference in the transient response, in particular the settling time. Steady state solutions for all runs do go to zero due to the integral gain, but only after time reaches upwards of 150+ seconds. The commanded steer angle at this time is around 0.7 degrees, which causes the response to be slow. The reason for this is because the system is not second order. This makes it difficult to predict the exact behavior because there may be contribution of several roots to the system response.

The following subsections include parameter changes that include the effect of the cornering stiffness. For example, the CG position determines the weight of the vehicle on the front and rear wheels. If the cornering stiffness is constant throughout the entire range of CG positions, the responses are slightly different than if allowed to change according to Eqn. 2.4. This will become clear in the following sections and is interesting to examine for two main reasons: 1) it shows to what extent the cornering stiffness affects the vehicle transient response, and 2) it demonstrates that the controller maintains stability and gives desirable response characteristics regardless of subtle differences in tire parameters. Unless otherwise stated, the front and rear cornering stiffness values for the simulation runs are as given in Table 4.1.

4.4.2 Changes in speed

The simulation looks at a speed range between 0 and 30 mph. Typical field speeds are somewhere between 0 and 12 mph. Higher speeds are included in the simulation to observe how the roots move and to see when the vehicle becomes unstable. In Figure 4.7, the

movement of the closed-loop system roots as the speed changes shows that for field speeds, the controlled vehicle is always stable. The trend suggests that at a speed of approximately 28 mph, the controlled system loses stability. This shows the controller is appropriate over the practical range of field speeds. In order to avoid any unstable speed, manufacturer specifications state the vehicle speed must be 12 mph or less to engage the controller [19]. The simulation shows that even in the event of a speed sensor malfunction, the controller can maintain vehicle stability.

Figure 4.8 shows the lateral error system response for each speed. As expected, low speeds result in slow response and high speeds result in more oscillatory behavior. Close to the target speed of 10 mph, the response exhibits desired damping and settling time. At higher speeds, the transient response is less desirable, but the system maintains stability. The controller gains seem effective in the appropriate field or harvest speed range.

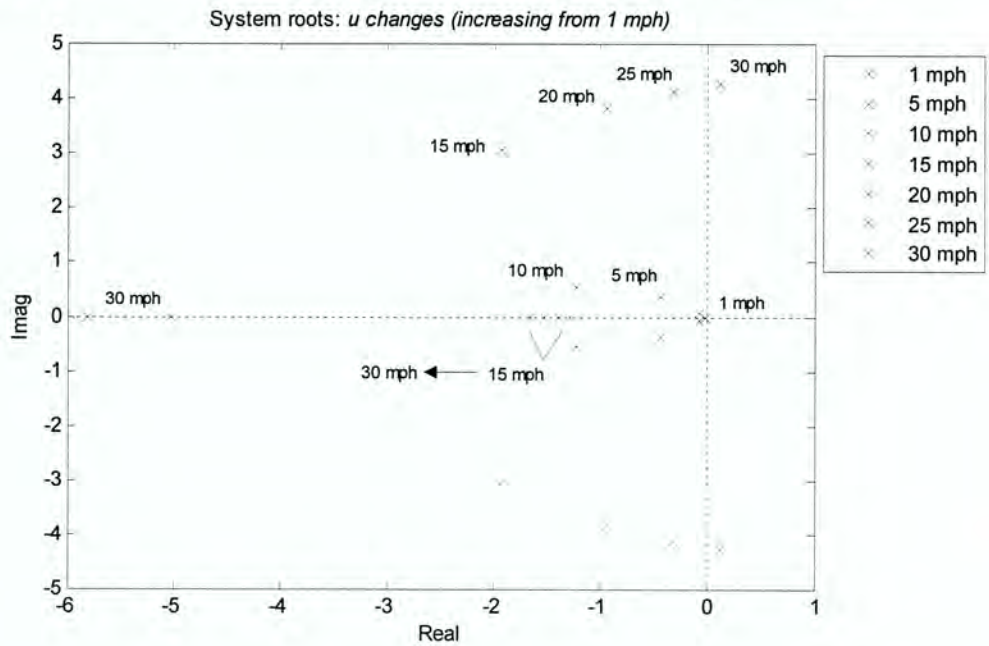


Figure 4.7 System roots as speed changes plotted on complex plane

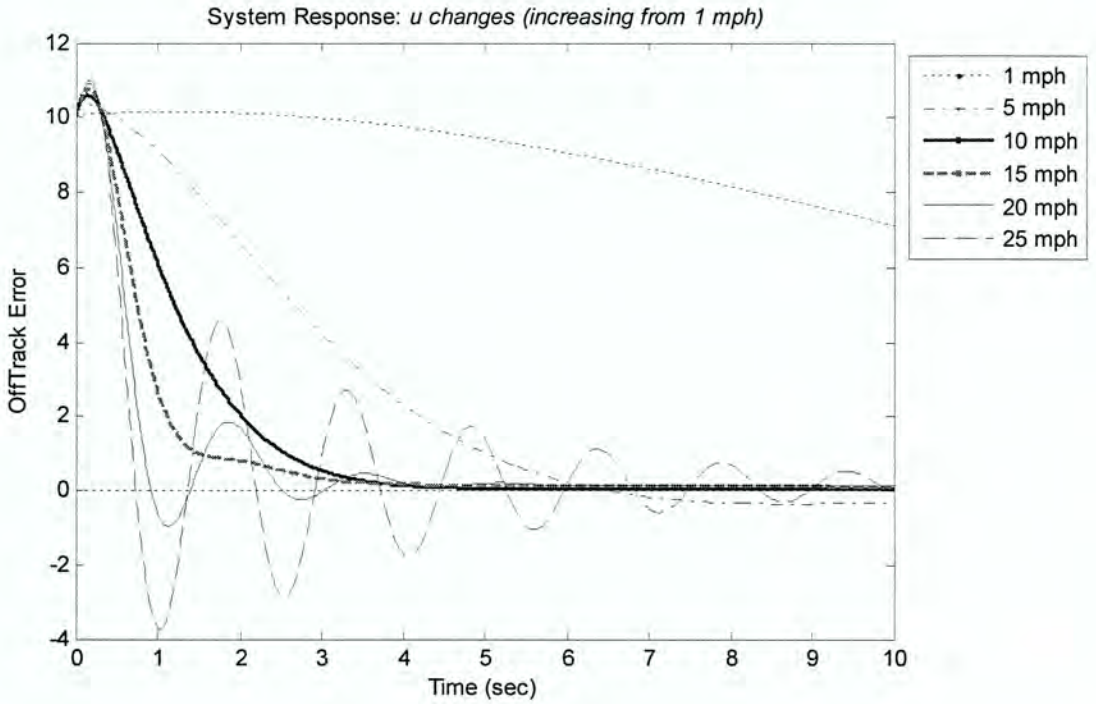


Figure 4.8 System response for speed range of $u = 1 - 25$ mph

4.4.3 Changes in CG position

The center of gravity position on a combine, or any other piece of agricultural equipment, changes with the addition of implements, headers or platforms, and with the addition of harvested crop entering the machine. Figure 4.9 shows the closed-loop roots of the vehicle system as the CG moves from rear to front (0% of total weight on front, W_f, to 100% of total weight on front) for both constant values $C_{\alpha f}$ and $C_{\alpha R}$ and values that change based on Eqn. 2.4. Results show that for constant values of cornering stiffness, CG locations near the rear are unstable. For values that change based on vertical load, only CG positions on the front and rear axles are unstable. This is because of positive real roots and dominant roots equal to zero. Overall, the results are satisfactory as the actual vehicle CG position would not realistically lie on one of the wheel axles.

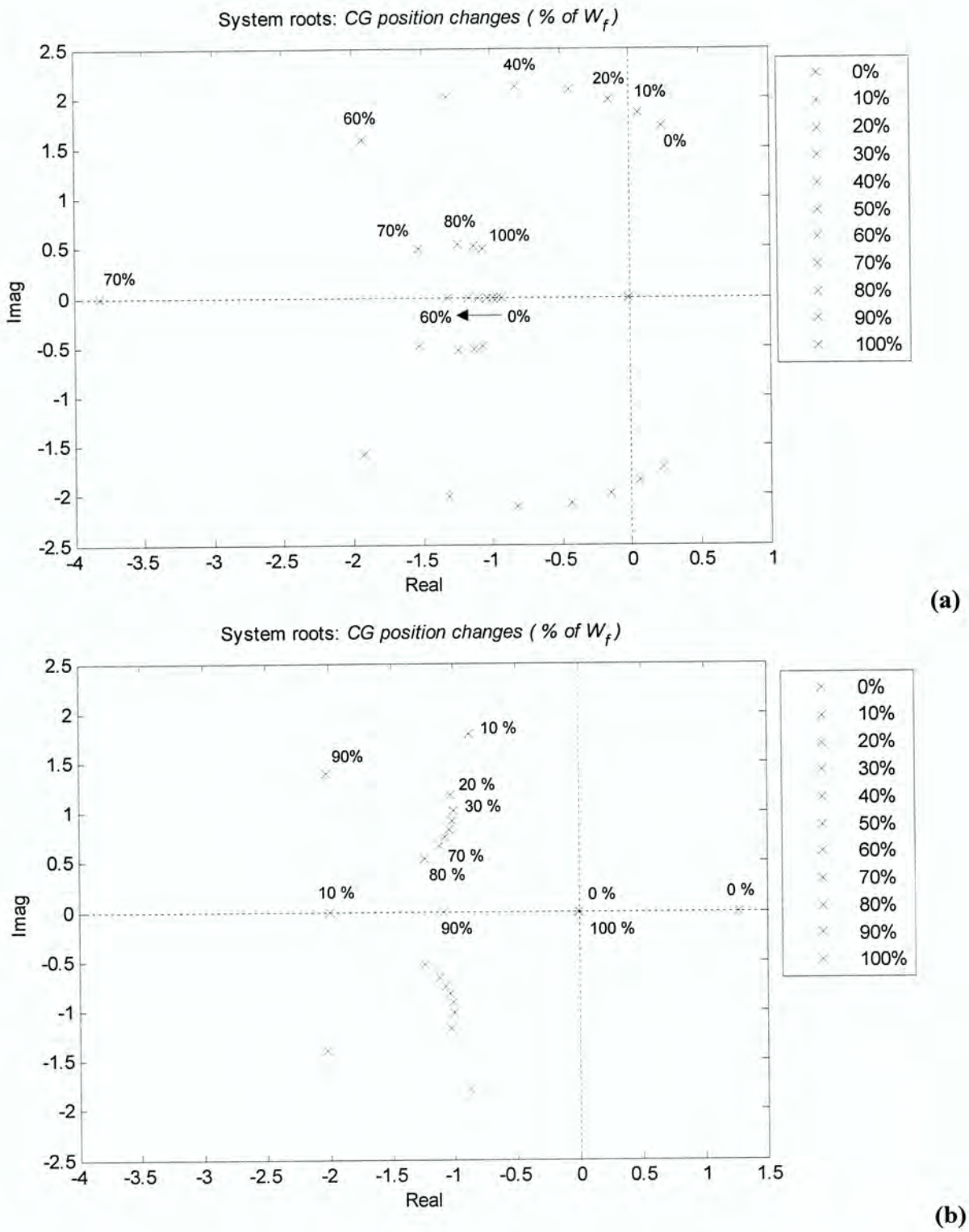
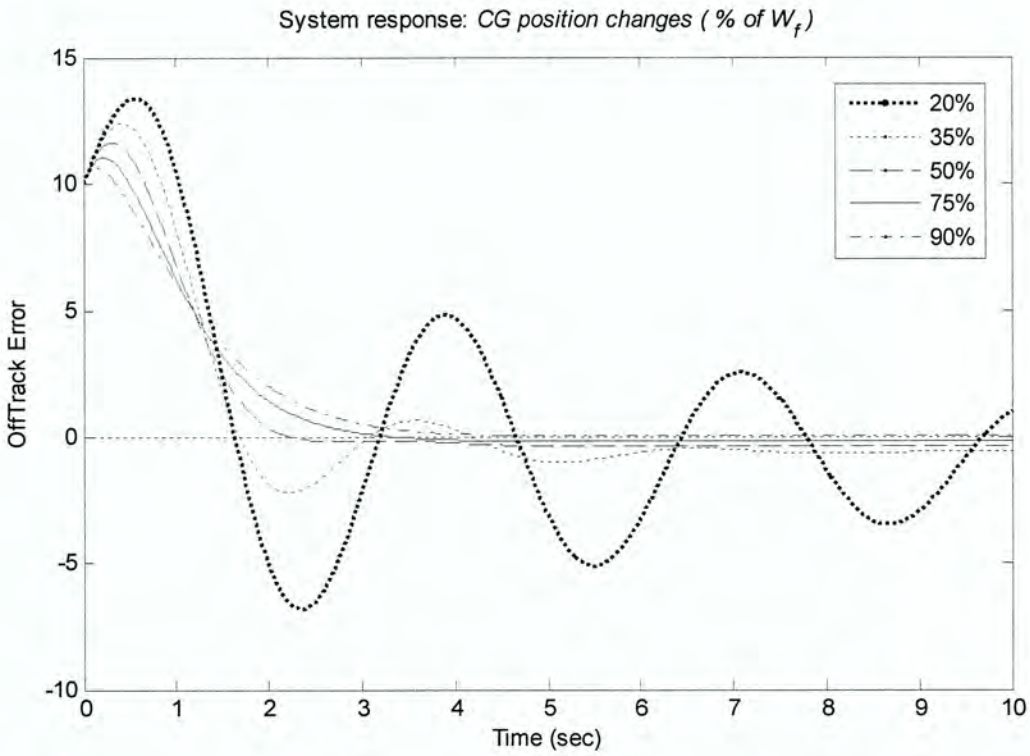


Figure 4.9 System roots as CG position changes plotted on complex plane at 10 mph: (a) constant $C_{\alpha f}$ and $C_{\alpha R}$, (b) $C_{\alpha f}$ and $C_{\alpha R}$ calculated by Eqn. 2.4

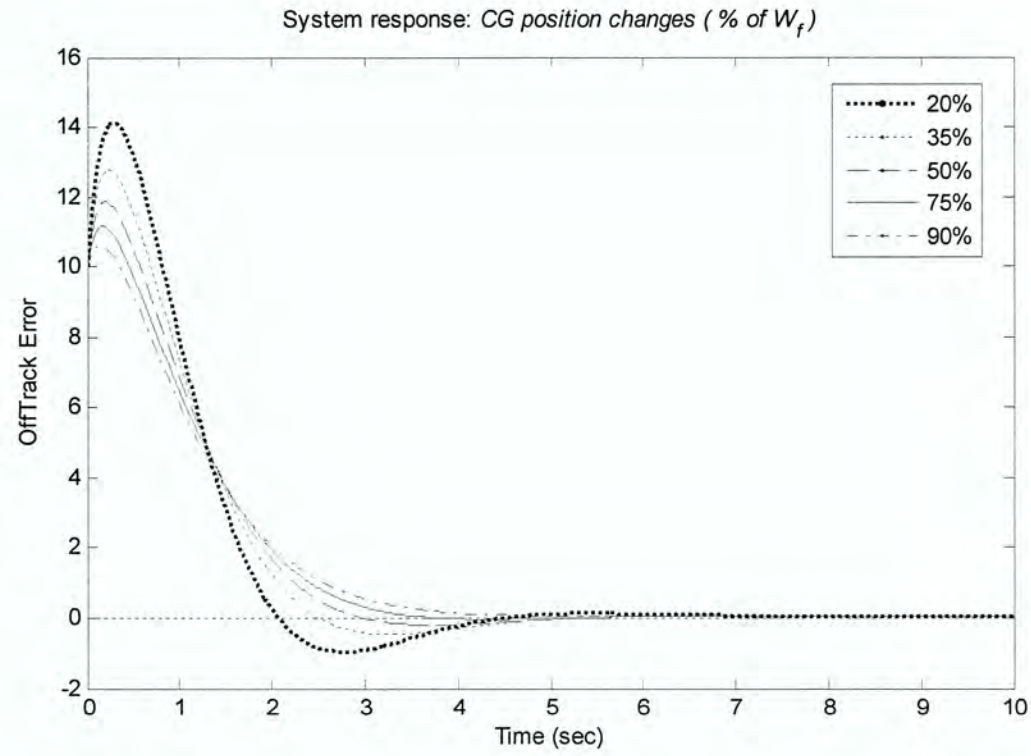
The system response for a realistic range of CG positions is given in Figure 4.10. The plot shows that CG positions further to the rear result in less damping, more overshoot, and an increase in yaw angle and lateral velocity. This correlates to an increase in yaw rate and lateral acceleration. As the CG position moves towards the front, damping increases and overshoot decreases, improving the transient response and reducing the lateral accelerations. This conclusion could be shown in additional plots, but can also be determined by observing the initial slope of the off track error curves. The derivative, or slope, of the off track position $\left(\dot{Y}\right)$ is a function of yaw and lateral velocity (see Eqn. 3.13). If the derivative of Y becomes large and does so quickly, the lateral velocity and/or yaw angle must also become large and have high accelerations.

This plot shows the dependence of system response on $C_{\alpha f}$ and $C_{\alpha R}$ values. Figure 4.10a shows as the CG moves towards the rear, the system begins to oscillate. This is because the rear lateral force generated using a constant $C_{\alpha R}$ does not create a sufficient moment arm to result in a fast response; thus, the vehicle position overshoots and slowly damps to zero. Figure 4.10b demonstrates that as cornering stiffness changes with load, damping is significantly increased and settling time is significantly decreased as the CG position moves rearward.

It is important to note the effect speed has on the CG position change response. Figure 4.11 shows the roots of the system at a speed of 20 mph. The vehicle becomes unstable as the CG position moves closer to the rear as speed increases for both constant and changing $C_{\alpha f}$ and $C_{\alpha R}$ values. At 20 mph, the controller maintains stability for realistic CG position shifts.

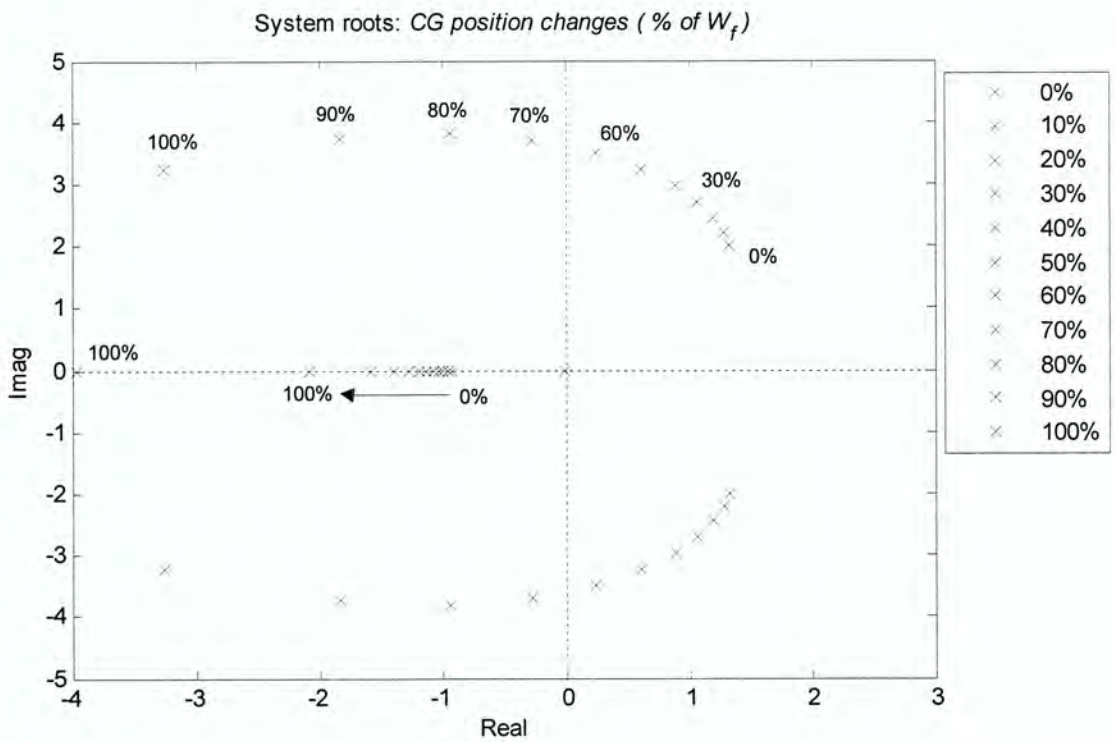


(a)

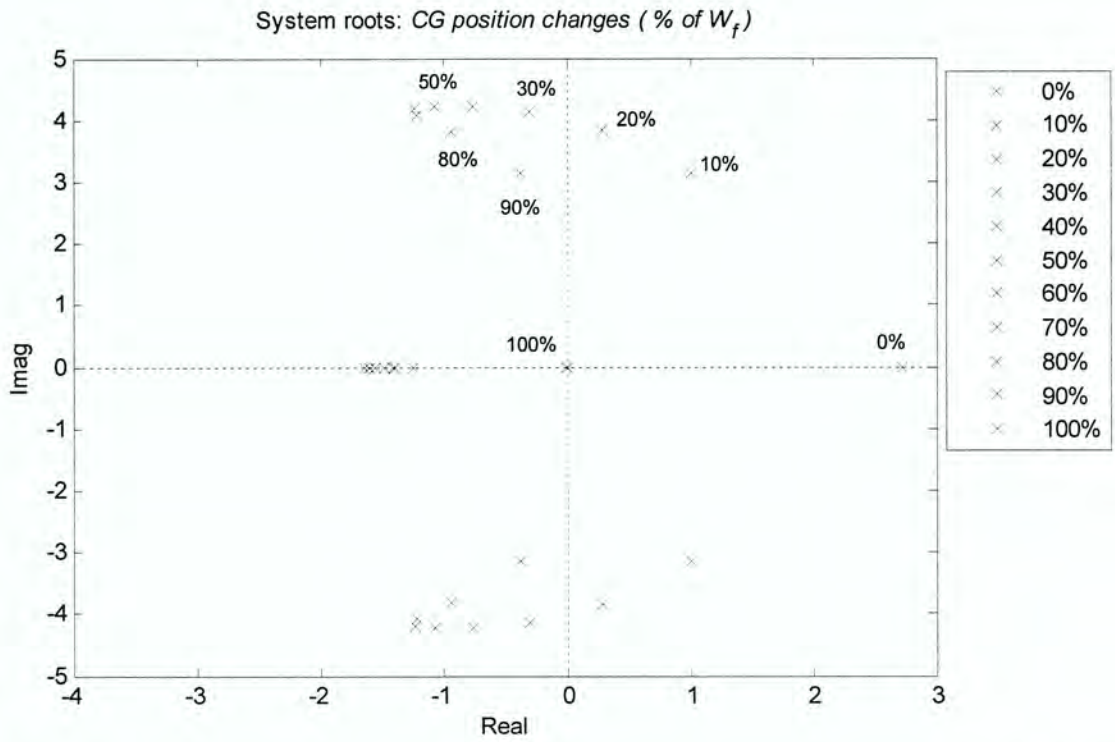


(b)

Figure 4.10 System response as CG position changes at 10 mph (a) constant $C_{\alpha f}$ and $C_{\alpha R}$, (b) $C_{\alpha f}$ and $C_{\alpha R}$ calculated by Eqn. 2.4



(a)



(b)

Figure 4.11 System roots as CG position changes plotted on complex plane at 20 mph (a): constant $C_{\alpha f}$ and $C_{\alpha R}$, (b) $C_{\alpha f}$ and $C_{\alpha R}$ calculated by Eqn. 2.4

4.4.4 Changes in vehicle weight

A third parameter of interest is the change in vehicle weight according to conditions described in the introduction to his chapter. The following plots show the response of the system to changes in weight from harvested crop entering the grain tank. The percentage given is the percent of full tank. These results assume constant cornering stiffness values and shift the CG position to the rear as the tank fills. According to Deere technical specifications for the STS combines, the CG position moves at most 4% with a full grain tank. Figure 4.12 shows the results, plotting the response to a percentage of the grain tank load.

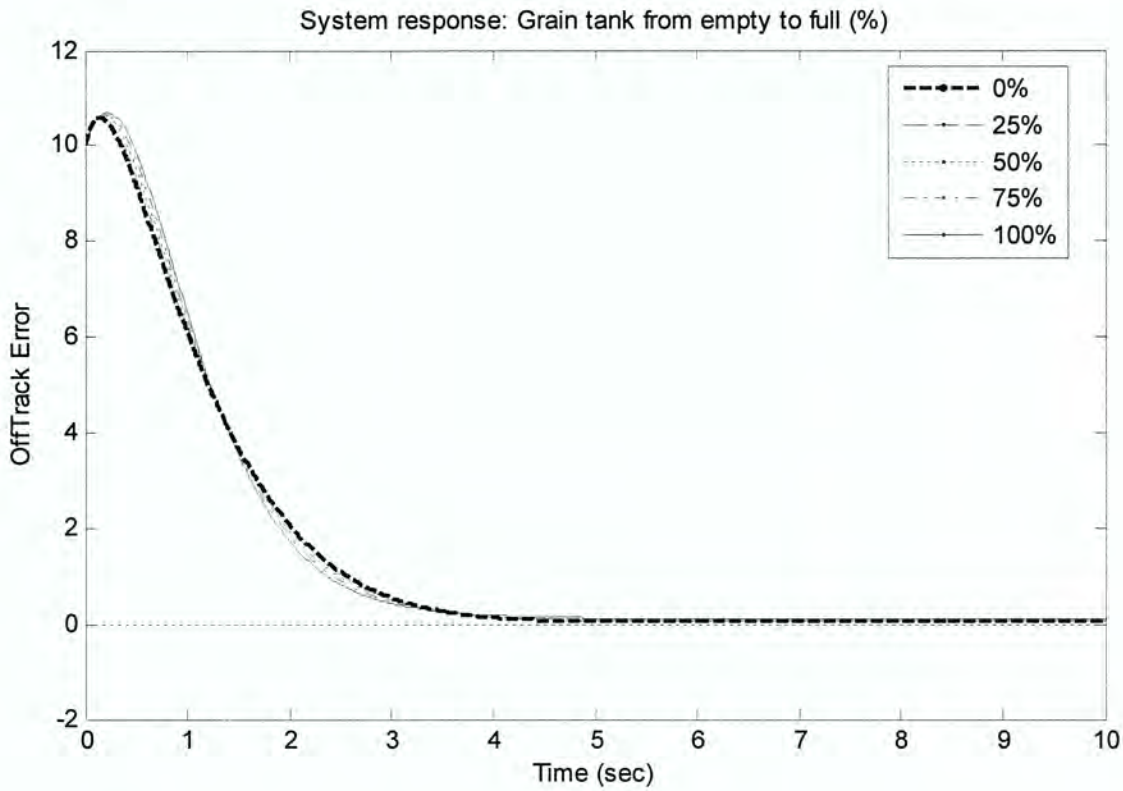


Figure 4.12 System response to weight change with static CG position

Results show that the system response is stable for all conditions and is nearly identical for the entire range. If the CG continues to move, the response begins to become unstable as it approaches the rear axle. This situation exhibits similar response to Figure 4.10a for the CG positions towards the front.

An interesting note to make, although not included in the plots, is the response of the system when using Eqn. 2.4 to calculate the cornering stiffness. The responses differ slightly, but are virtually identical. If the CG position moves further rearward, the response looks similar to Figure 4.10b.

4.5 Effects of sloped terrain

To fully complete the objective of this study, this section discusses the effects of terrain slope. The slope is the input to the system and does not affect system roots. In terms of the transfer function, changing the slope modifies the system zeros, which affect the amplitude of the response and not the nature of the response [4]. Therefore, in linear analysis, comparing different slopes is trivial, as a different slope value simply acts as a scalar. The interesting point to acknowledge from this study is the concept that the steady state sideslip and steer angles are a function of the slope (see Chapter 3, Section 3.3). The steady state sideslip and lateral velocity increase as the slope increases, but the controller always returns the vehicle to zero steady state lateral error for any grade of slope in linear analysis.

The previous sections all assumed an initial downhill lateral error of 10 feet. Figure 4.13 shows the comparison with an initial uphill lateral error of the same amount. The trajectory response differs slightly, as expected. The difference is due to the direction of the side slope force that the weight of the vehicle creates. For a downhill error, the force leads away from the desired path. For an uphill error, the force leads towards the path. The sum of the forces for the uphill situation ultimately leads to smaller initial lateral velocity and sideslip, which causes the vehicle to move towards the path more quickly and overshoot by a small margin. This shows another subtle effect of the slope; there is some difference in response if the vehicle is uphill or downhill of the desired path when the steering controller engages.

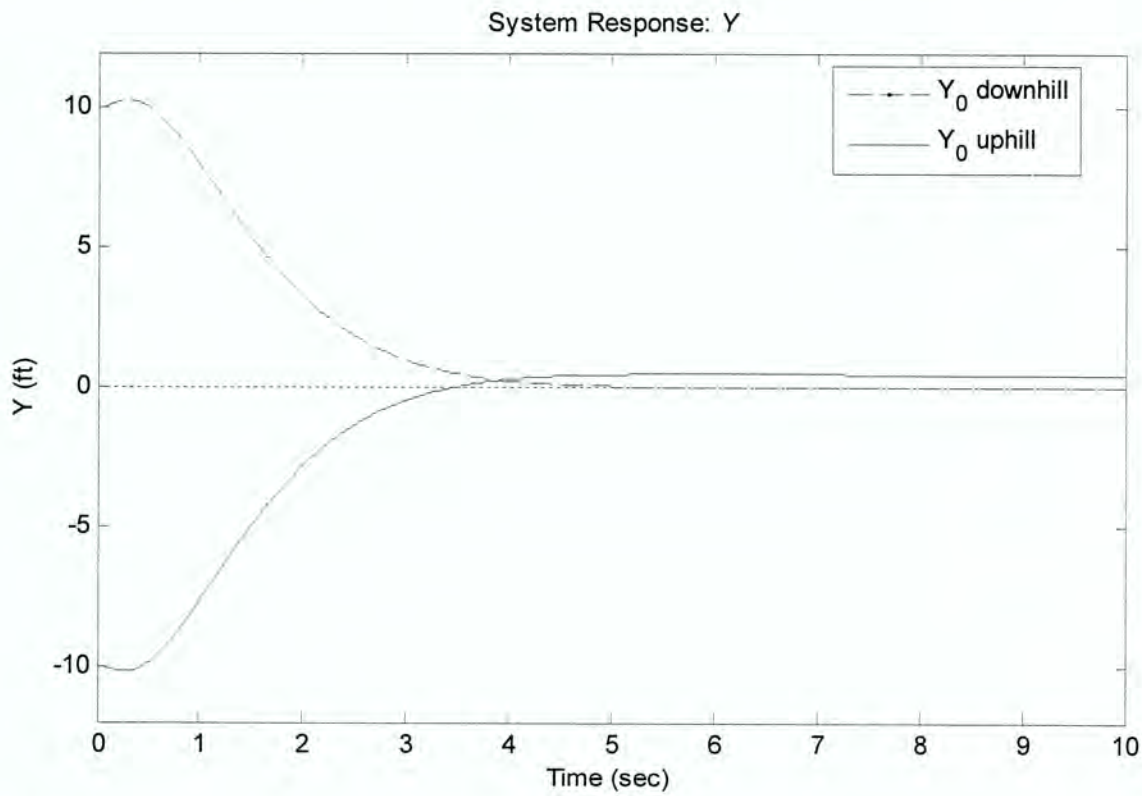


Figure 4.13 Comparison of response to downhill and uphill initial lateral errors of 10 ft

4.6 Additional comparisons and results

Appendix A, Section A.2 discusses a brief comparison to an additional model not included in the main body of this study. The results of the comparison do not necessarily add additional insight to the vehicle model or design; they simply show the difference between the linear model discussed here and a linear model including steering lag effects. The steering lag model does not assume an instantaneous steer angle, but rather incorporates a first order lag to make the steer command more realistic.

CHAPTER 5. CONCLUSIONS AND FUTURE WORK

The vehicle dynamics and steering controller model presented in this thesis provide a viable initial design and analysis tool to develop an automatic steering control guidance system for a John Deere STS combine. Simulations show the final design successfully handles vehicle parameter changes and maintains a stable response for realistic values.

The linear bicycle vehicle dynamics model uses lateral tire forces to govern the equations of motion. This is a key difference and advantage over the traditional kinematics model, as it creates the ability to model sloped terrain and observe the effects of parameter changes. One of the most interesting and conclusive observations from the results of the model is that steady state sideslip and steady state steer angle are independent of the controller and are functions of the slope of the terrain and the cornering coefficient of the tires. For a front steer only or a rear steer only vehicle, there will always exist a non-zero steady state sideslip.

The model and simulation are setup to be adaptable for a wide range of vehicle configurations. It could be expanded to analyze front steer and four-wheel steer vehicles or study the effects of tire pressure, soil conditions, or dual or triple wheel configurations by adjusting the tire parameters. A variety of terrain definitions could also be applied by simply replacing the step input with a lookup table of terrain data collected for a specific field.

A comparison to a non-linear model would demonstrate the significance and correctness of the linear assumptions. This would include no small angle approximations, a non-linear tire

model, and additional state variables, such as forward velocity, pitch, and roll. The effects of roll stability on sloped terrain would be interesting to examine. The addition of steering and tire lag would make the simulated vehicle more realistic.

Field data from a real combine would validate the simulation by comparing measured and simulated results. Results from the field study would provide greater insight to the validity of the approximations and accuracy of the tire model. The tire parameters of the vehicle model could be adjusted based on the measured data to yield more accurate simulations.

This model has also been used in a John Deere Immersive Combine Simulator, which was used for demonstration and training purposes at a large agricultural world trade show. Operators from around the world were able to gain exposure to navigation guidance systems and gain experience in automatically controlled vehicles. The project is ongoing and continues to be improved to meet the company needs.

APPENDIX A. MODEL COMPARISONS

A.1 Discrepancies in the three types of models

Two types of Simulink models are compared, one using the actual derivative and integral terms of the state variable as defined by the state equations, another using the software included PID controller block, which approximates an ideal derivative and integral Laplace transform. As seen in Figure A.1, there are differences in the response; it is interesting to note that the MATLAB and Simulink models that used the actual derivative and integral terms match in response as do the transfer function and Simulink model that uses the PID controller block. The settling time and steady state response is virtually identical, which suggests that either model still yields the desired final output.

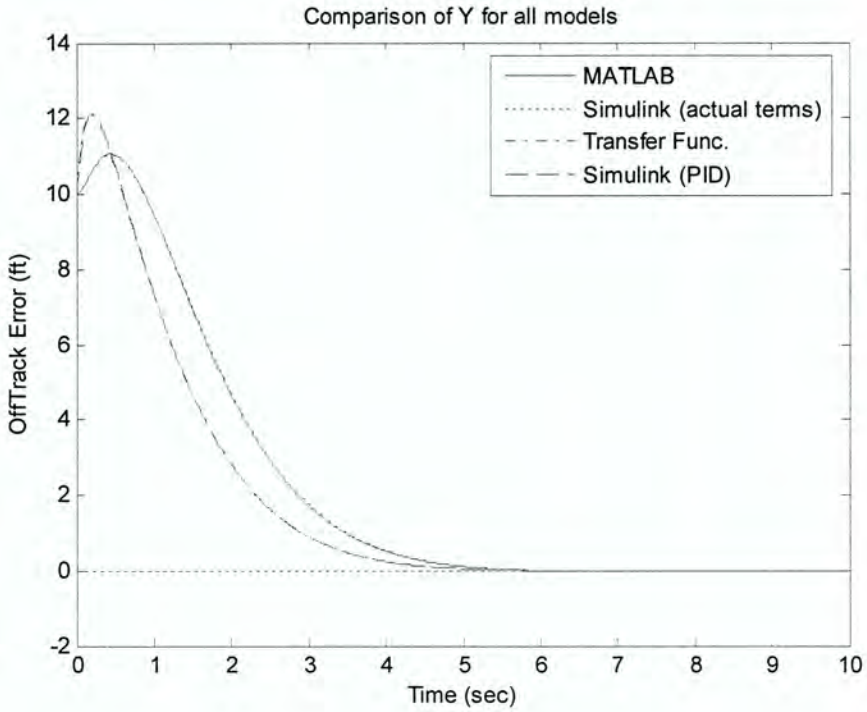


Figure A.1 Comparison of lateral error for all models with original setup

Upon further investigation, it was determined the reason for the discrepancies is because the MATLAB transfer function and the Simulink derivative block assume an initial derivative term of Y_0 / dt, which leads to a commanded steer angle of more than 1000 degrees. This is verified in Figure A.2, which shows the commanded steer angle with the modification made in the numerical integration code (referred to as MATLAB in plots). Because the model is linear, this unreasonable desired steer angle signal passes through the system and causes extremely high lateral forces, resulting in different transient response characteristics.

This conclusion is again verified by altering the numerical integration code to assume the same initial derivative term mentioned above for the first time step of the simulation and running the simulation again, as shown in Figure A.3. In summary, the models differ only because the difference in derivative term approximation, not because of difference in system dynamics; therefore, both models are used to analyze the system with confidence.

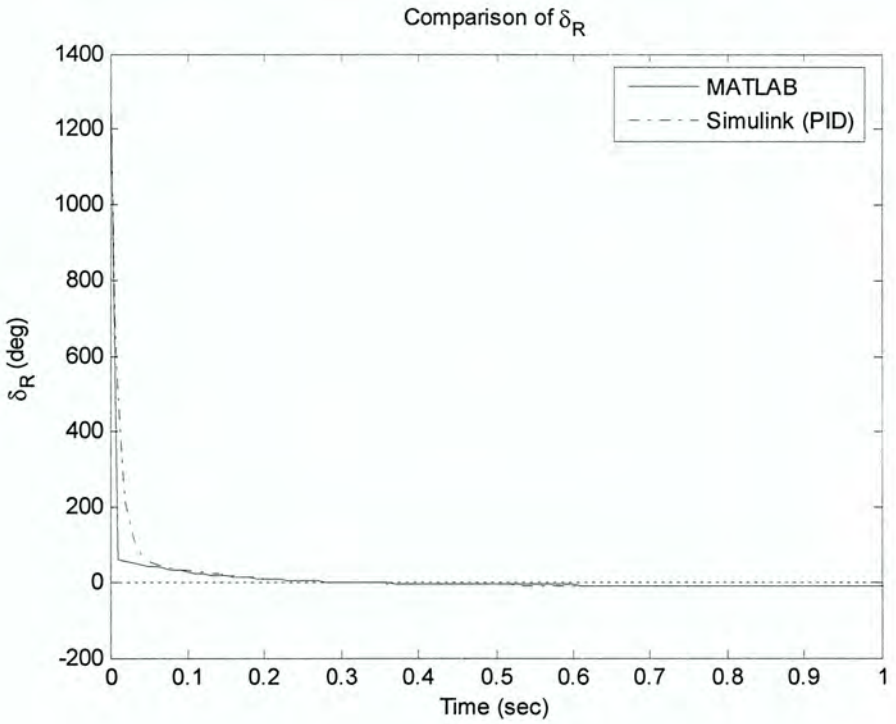


Figure A.2 Steer angle comparison with modification to the MATLAB model

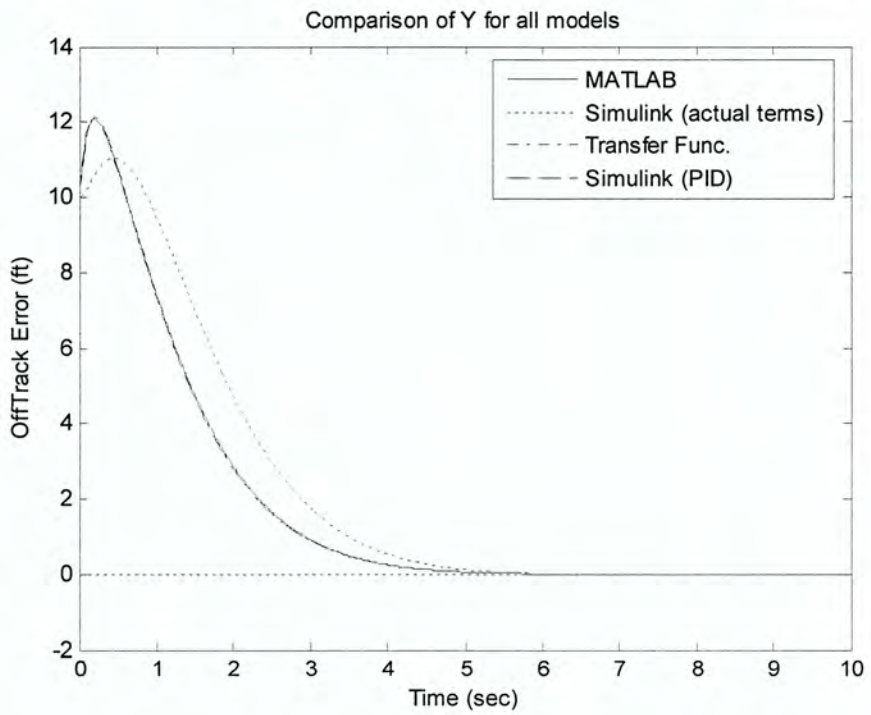


Figure A.3 Response comparison with modification to the MATLAB model

A.2 Additional model comparisons

A.2.1 Steering lag model

Introducing a steering lag make the simulation more realistic, as a commanded steer angle of 30 degrees cannot instantly be obtained. The MATLAB model was modified to include a first order lag by making δ a state variable. The equation for the derivative of the steer angle is given in Eqn. A.1.

$$\dot{\delta}_R = \left(\delta_{R,des} - \delta_R\right)\tau \tag{A.1}$$

where δ_R is the actual steer angle, $\delta_{R,des}$ is the commanded steer angle from the controller, and τ is the time constant for the for the first order lag. A time constant of 10, resulting in a rise time of 0.1 seconds was arbitrarily selected. Introducing a lag to the model eliminates the large initial derivatives of lateral velocity and yaw. This reduces the acceleration terms and provides a more realistic response. Figure A.4 shows the comparison of Y, ψ, and δ_R between the steering lag compensated model and the uncompensated model for the same vehicle parameters used in Section 4.2. Figure A.5 shows r, v, β, and a_y in g's defined by Eqn. A.2.

$$a_y = \frac{\left(\dot{v} + ur\right)}{g} \tag{A.2}$$

Results show that a steering lag compensated system has slightly different transient response characteristics for the heading and lateral error response. The biggest important difference is the acceleration terms. The max a_y in a non steering lag model is about 2.1 g's; a steering lag model only reaches 0.6 g's. Using a steering lag in the model allows the controller design to also account for operator comfort and vehicle structural integrity, both of which are compromised by high accelerations.

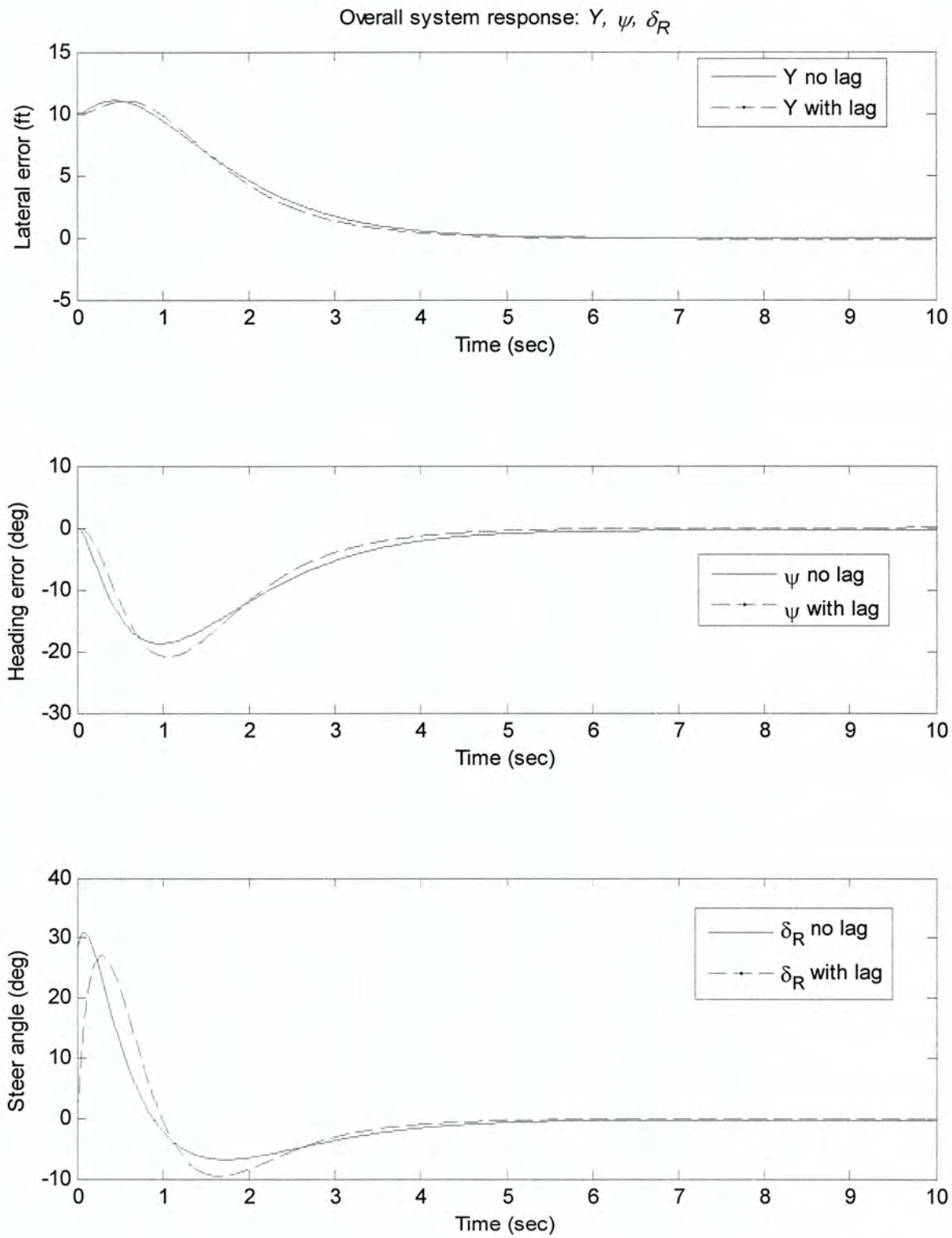


Figure A.4 System response comparison to steering lag model

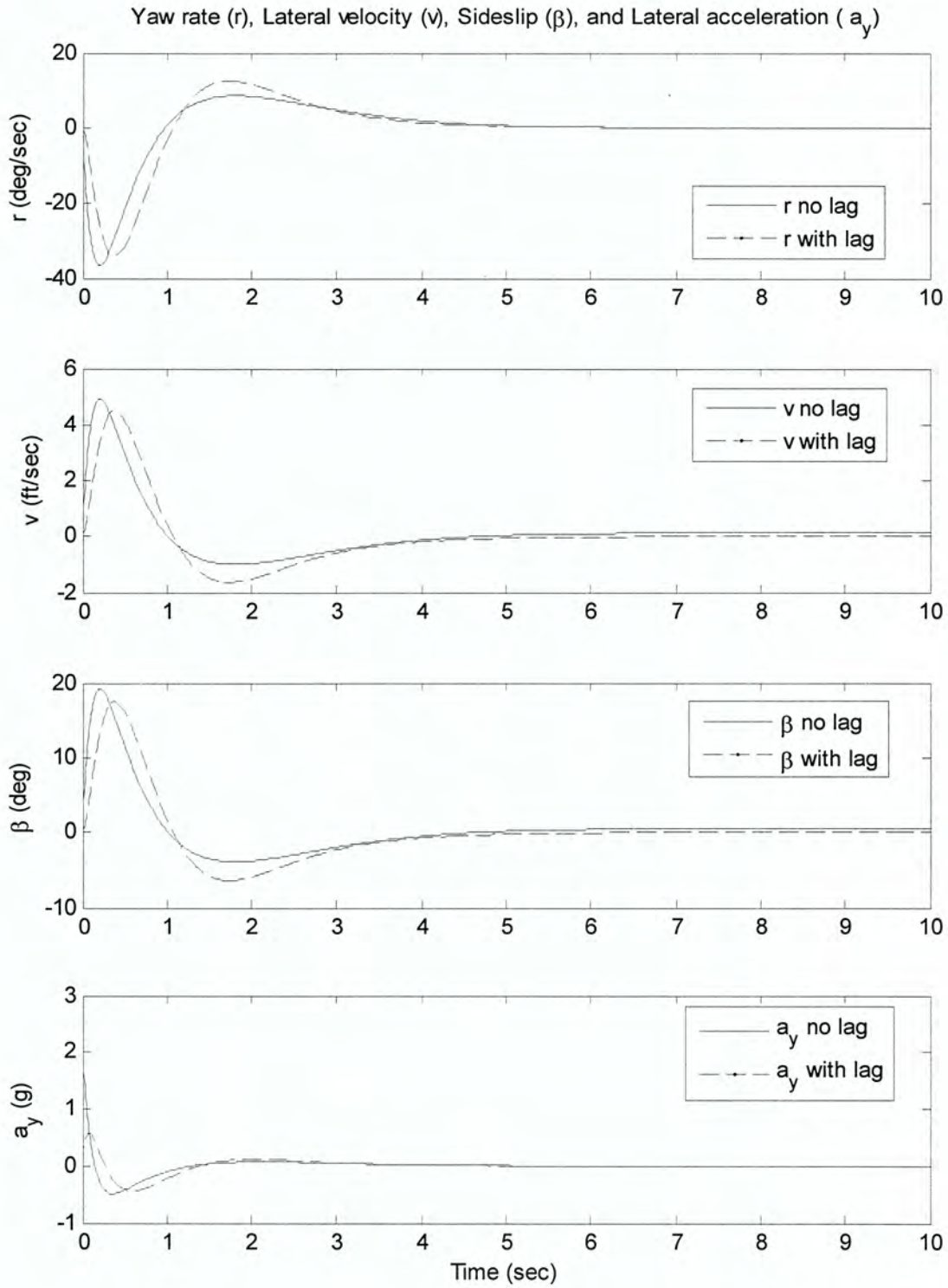


Figure A.5 Velocity and acceleration comparison to steering lag model

APPENDIX B. DETAILED DERIVATIONS

B.1 Free body diagram

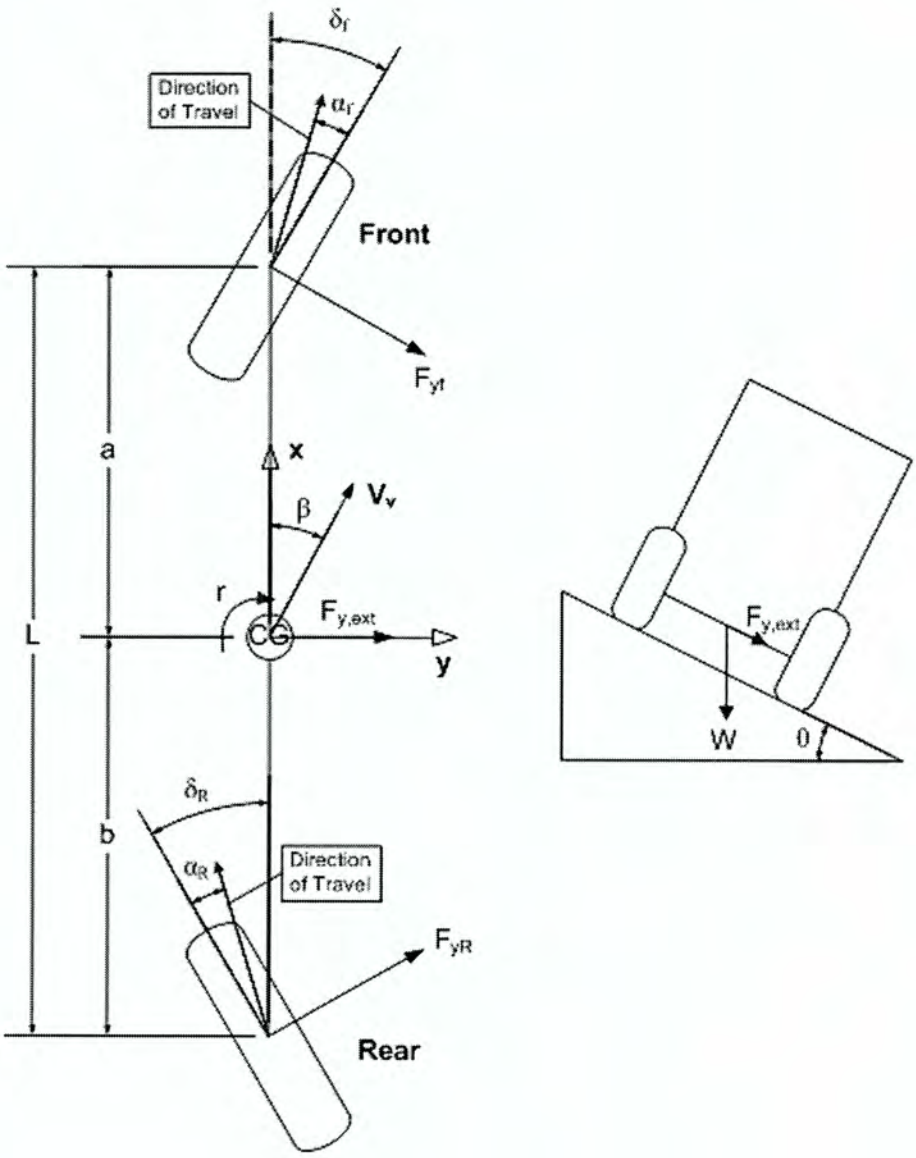


Figure B.1 Free body diagram of bicycle model and force due to sloped terrain

B.2 Equations of motion for *r* (yaw rate) and *v* (lateral velocity)

Two of the state equations needed to define the system are found by summing the forces in the *y*-direction and summing the moments about the center of gravity (CG) position. The acceleration in the *y*-direction is given by Eqn. 3.5 in the body of the thesis.

B.2.1 Derivation of yaw rate

$$\sum F_y = Ma_y$$

$$F_{yf} + F_{yR} + F_{y,ext} = M(\dot{v} + ur)$$

$$-C_{\alpha f}\alpha_f - C_{\alpha R}\alpha_R + W\sin(\theta) = M\dot{v} + Mur$$

$$-C_{\alpha f}\left(\frac{v+ar}{u} - \delta_f\right) - C_{\alpha R}\left(\frac{v-br}{u} - \delta_R\right) + W\sin(\theta) = M\dot{v} + Mur$$

$$M\dot{v} + \left(\frac{C_{\alpha f}}{u} + \frac{C_{\alpha R}}{u}\right)v + \left(Mu + \frac{C_{\alpha f}a}{u} - \frac{C_{\alpha R}b}{u}\right)r = C_{\alpha f}\delta_f + C_{\alpha R}\delta_R + W\sin(\theta)$$

$$\dot{v} + a_1 v + a_2 r = a_3\delta_f + a_4\delta_R + \frac{W}{M}\sin(\theta) \tag{B.1}$$

where

$$\begin{aligned} a_1 &= \frac{C_{\alpha f} + C_{\alpha R}}{Mu} \\ a_2 &= \frac{Mu^2 + C_{\alpha f}a - C_{\alpha R}b}{Mu} \\ a_3 &= \frac{C_{\alpha f}}{M} \\ a_4 &= \frac{C_{\alpha R}}{M} \end{aligned} \tag{B.2}$$

B.2.2 Derivation of lateral velocity

$$\sum M_{CG} = I\dot{r}$$

$$F_{yf}a - F_{yR}b = I\dot{r}$$

$$-C_{\alpha f}\left(\frac{v+ar}{u} - \delta_f\right)a - C_{\alpha R}\left(\frac{v-br}{u} - \delta_R\right)b = I\dot{r}$$

$$I\dot{r} + \left(\frac{C_{\alpha f}a^2 + C_{\alpha R}b^2}{u}\right)r + \left(\frac{C_{\alpha f}a - C_{\alpha R}b}{u}\right)v = C_{\alpha f}a\delta_f - C_{\alpha R}b\delta_R$$

$$\dot{r} + b_1 v + b_2 r = b_3\delta_f + b_4\delta_R \tag{B.3}$$

where

$$\begin{aligned} b_1 &= \frac{C_{\alpha f}a - C_{\alpha R}b}{Iu} \\ b_2 &= \frac{C_{\alpha f}a^2 + C_{\alpha R}b^2}{Iu} \\ b_3 &= \frac{C_{\alpha f}a}{I} \\ b_4 &= \frac{-C_{\alpha R}b}{I} \end{aligned} \tag{B.4}$$

B.3 Steady state solutions for *r*, *v*, δ, and β

The steady state solutions for yaw rate and lateral velocity are found using Cramer's Rule applied to the matrix representation shown below in Eqn. B.5. Sideslip and steer angle steady state relationships are algebraic manipulations of *r* and *v*.

$$\begin{bmatrix} a_1 & a_2 \\ b_1 & b_2 \end{bmatrix} \begin{bmatrix} v \\ r \end{bmatrix} = \begin{bmatrix} a_3\delta_f + a_4\delta_R + \dfrac{W}{M}\sin(\theta) \\ b_3\delta_f + b_4\delta_R \end{bmatrix} \tag{B.5}$$

B.3.1 Steady state yaw rate

Applying Cramer's Rule for *r* gives

$$r_{ss} = \frac{\begin{vmatrix} a_1 & a_3\delta_f + a_4\delta_R + \dfrac{W}{M}\sin(\theta) \\ b_1 & b_3\delta_f + b_4\delta_R \end{vmatrix}}{\begin{vmatrix} a_1 & a_2 \\ b_1 & b_2 \end{vmatrix}}$$

$$r_{ss} = \frac{a_1\left(b_3\delta_f + b_4\delta_R\right) - b_1\left(a_3\delta_f + a_4\delta_R + \dfrac{W}{M}\sin(\theta)\right)}{a_1b_2 - a_2b_1} \tag{B.6}$$

Breaking Eqn. B.6 into numerator and denominator portion allows for easier simplification. First, look at the numerator. Eqn. B.7 can be broken into three parts: front steer, rear steer, and slope.

Numerator

$$\left(a_1b_3 - b_1a_3\right)\delta_f + \left(a_1b_4 - b_1a_4\right)\delta_R - b_1\frac{W}{M}\sin(\theta) \tag{B.7}$$

Front steer, δ_f :

$$(a_1 b_3 - b_1 a_3) = \left(\frac{C_{\alpha f} + C_{\alpha R}}{Mu}\right)\frac{C_{\alpha f} a}{I} - \left(\frac{C_{\alpha f} a - C_{\alpha R} b}{Iu}\right)\frac{C_{\alpha f}}{M} \tag{B.8}$$

$$= \frac{{C_{\alpha f}}^2 a + C_{\alpha f} C_{\alpha R} a - {C_{\alpha f}}^2 a + C_{\alpha f} C_{\alpha R} b}{MuI}$$

$$= \frac{(a+b) C_{\alpha f} C_{\alpha R}}{MuI}$$

$$= \frac{L C_{\alpha f} C_{\alpha R}}{MuI} \tag{B.9}$$

Rear steer, δ_R :

$$(a_1 b_4 - b_1 a_4) = -\left(\frac{C_{\alpha f} + C_{\alpha R}}{Mu}\right)\frac{C_{\alpha R} b}{I} - \left(\frac{C_{\alpha f} a - C_{\alpha R} b}{Iu}\right)\frac{C_{\alpha R}}{M} \tag{B.10}$$

$$= \frac{{C_{\alpha R}}^2 b - C_{\alpha f} C_{\alpha R} a - {C_{\alpha R}}^2 b - C_{\alpha f} C_{\alpha R} b}{MuI}$$

$$= \frac{-(a+b) C_{\alpha f} C_{\alpha R}}{MuI}$$

$$= \frac{-L C_{\alpha f} C_{\alpha R}}{MuI} \tag{B.11}$$

Slope, θ :

$$b_1 \frac{W}{M} = \left(\frac{C_{\alpha f} a - C_{\alpha R} b}{Iu}\right)\frac{W}{M} \tag{B.12}$$

$$= \left(\frac{C_{\alpha f} a - C_{\alpha R} b}{Iu}\right)\frac{W}{M} \tag{B.13}$$

Combine the parts from each steer angle and the external slope force (Eqns. B.9, B.11, B.13),

$$= \frac{LC_{\alpha f}C_{\alpha R}}{MuI}\left(\delta_f - \delta_R\right) - \left(\frac{C_{\alpha f}a - C_{\alpha R}b}{Iu}\right)\frac{W}{M}\sin(\theta) \tag{B.14}$$

$$W_f = \frac{b}{L}W$$
$$W_R = \frac{a}{L}W \tag{B.15}$$

Substituting the relationships shown in Eqn. B.15, the numerator becomes

$$= \frac{LC_{\alpha f}C_{\alpha R}}{MuI}\left(\delta_f - \delta_R\right) - \left(\frac{LC_{\alpha f}W_R - LC_{\alpha R}W_f}{MuI}\right)\sin(\theta)$$

$$= \frac{L\left[C_{\alpha f}C_{\alpha R}\left(\delta_f - \delta_R\right) - \left(C_{\alpha f}W_R - C_{\alpha R}W_f\right)\sin(\theta)\right]}{MuI} \tag{B.16}$$

Now look at the denominator of Eqn. B.6.

Denominator

$$\left(a_1b_2 - a_2b_1\right) = \left(\frac{C_{\alpha f} + C_{\alpha R}}{Mu}\right)\left(\frac{C_{\alpha f}a^2 + C_{\alpha R}b^2}{Iu}\right) - \left(\frac{Mu^2 + C_{\alpha f}a - C_{\alpha R}b}{Mu}\right)\left(\frac{C_{\alpha f}a - C_{\alpha R}b}{Iu}\right) \tag{B.17}$$

$$= \frac{C_{\alpha f}{}^2a^2 + C_{\alpha f}C_{\alpha R}b^2 + C_{\alpha f}C_{\alpha R}a^2 + C_{\alpha R}{}^2b^2 - Mu^2C_{\alpha f}a + Mu^2C_{\alpha R}b - C_{\alpha f}{}^2a^2 + C_{\alpha f}C_{\alpha R}ab + C_{\alpha f}C_{\alpha R}ab - C_{\alpha R}{}^2b^2}{Mu^2I}$$

$$= \frac{(a+b)^2C_{\alpha f}C_{\alpha R} + Mu^2\left(C_{\alpha R}b - C_{\alpha f}a\right)}{Mu^2I}$$

$$= \frac{L^2C_{\alpha f}C_{\alpha R} + Mu^2\left(C_{\alpha R}b - C_{\alpha f}a\right)}{Mu^2I} \tag{B.18}$$

Next, combine numerator and denominator and simplify to obtain the final steady state solution for yaw rate, shown in Eqn. B.19.

$$= \frac{\dfrac{L\left[C_{\alpha f}C_{\alpha R}\left(\delta_f-\delta_R\right)-\left(C_{\alpha f}W_R-C_{\alpha R}W_f\right)\sin(\theta)\right]}{MuI}}{\dfrac{L^2C_{\alpha f}C_{\alpha R}+Mu^2\left(C_{\alpha R}b-C_{\alpha f}a\right)}{Mu^2I}}$$

$$= \frac{uL\left[C_{\alpha f}C_{\alpha R}\left(\delta_f-\delta_R\right)-\left(C_{\alpha f}W_R-C_{\alpha R}W_f\right)\sin(\theta)\right]}{L^2C_{\alpha f}C_{\alpha R}+Mu^2\left(C_{\alpha R}b-C_{\alpha f}a\right)}$$

$$= \frac{uL\left[C_{\alpha f}C_{\alpha R}\left(\delta_f-\delta_R\right)-\left(C_{\alpha f}W_R-C_{\alpha R}W_f\right)\sin(\theta)\right]}{L^2C_{\alpha f}C_{\alpha R}+Mu^2\left(C_{\alpha R}b-C_{\alpha f}a\right)}\left(\frac{1/L^2C_{\alpha f}C_{\alpha R}}{1/L^2C_{\alpha f}C_{\alpha R}}\right)$$

$$= \frac{u/L\left[\left(\delta_f-\delta_R\right)-\left(C_{\alpha f}W_R-C_{\alpha R}W_f\right)\dfrac{1}{C_{\alpha f}C_{\alpha R}}\sin(\theta)\right]}{1+\dfrac{Mu^2}{L^2C_{\alpha f}C_{\alpha R}}\left(C_{\alpha R}b-C_{\alpha f}a\right)}$$

$$= \frac{u/L\left[\left(\delta_f-\delta_R\right)-\left(\dfrac{W_R}{C_{\alpha R}}-\dfrac{W_f}{C_{\alpha f}}\right)\sin(\theta)\right]}{1+\dfrac{\dfrac{W}{g}u^2}{L^2}\left(\dfrac{b}{C_{\alpha f}}-\dfrac{a}{C_{\alpha R}}\right)}$$

$$= \frac{u/L\left[\left(\delta_f-\delta_R\right)+\left(\dfrac{W_f}{C_{\alpha f}}-\dfrac{W_R}{C_{\alpha R}}\right)\sin(\theta)\right]}{1+\dfrac{u^2}{Lg}\left(\dfrac{W_f}{C_{\alpha f}}-\dfrac{W_R}{C_{\alpha R}}\right)}$$

$$r_{ss} = \frac{u/L\left[\left(\delta_f-\delta_R\right)+K\sin(\theta)\right]}{1+\dfrac{Ku^2}{Lg}} \qquad \text{(B.19)}$$

B.3.2 Steady state steer angle

Rearranging the steady state yaw rate equation (see Eqn. B.19) and substituting the steady state relationship shown in Eqn. B.20 gives the steady state steer angle given by Eqn. B.21.

$$\frac{r}{u} = \frac{1}{R} \tag{B.20}$$

where u is velocity in the x-direction and R is the radius of turn,

$$\frac{rL}{u} = \frac{(\delta_f - \delta_R) + K\sin(\theta)}{1 + \frac{Ku^2}{Lg}}$$

$$\frac{L}{R}\left(1 + \frac{Ku^2}{Lg}\right) = (\delta_f - \delta_R) + K\sin(\theta)$$

$$\delta_f - \delta_R = \frac{L}{R} + \frac{Ku^2}{Rg} - K\sin(\theta)$$

$$\delta_f - \delta_R = \frac{L}{R} + Ka_y - K\sin(\theta)$$

$$\delta_f - \delta_R = \frac{L}{R} + K(a_y - \sin(\theta)) \tag{B.21}$$

where a_y is in g's.

B.3.3 Steady state lateral velocity

Applying Cramer's Rule to Eqn. B.5 for v yields

$$v_{ss} = \frac{\begin{vmatrix} a_3\delta_f + a_4\delta_R + \frac{W}{M}\sin(\theta) & a_2 \\ b_3\delta_f + b_4\delta_R & b_2 \end{vmatrix}}{\begin{vmatrix} a_1 & a_2 \\ b_1 & b_2 \end{vmatrix}}$$

$$v_{ss} = \frac{b_2\left(a_3\delta_f + a_4\delta_R + \frac{W}{M}\sin(\theta)\right) - a_2\left(b_3\delta_f + b_4\delta_R\right)}{a_1b_2 - a_2b_1} \tag{B.22}$$

Similar to steady state yaw rate, breaking Eqn. B.22 into numerator and denominator portion allows for easier simplification. The denominator will be the same as Eqn. B.18 and the numerator is broken into three parts. Eqn. B.15 is also used in the derivations.

Numerator

$$(a_3b_2 - a_2b_3)\delta_f + (a_4b_2 - a_2b_4)\delta_R + b_2\frac{W}{M}\sin(\theta) \tag{B.23}$$

Front steer, δ_f:

$$(a_3b_2 - a_2b_3) = \frac{C_{\alpha f}C_{\alpha R}(bL) - C_{\alpha f}aMu^2}{MuI} \tag{B.24}$$

$$= \frac{C_{\alpha f}C_{\alpha R}\left(bL - \frac{Mu^2a}{C_{\alpha R}}\right)}{MuI}$$

$$= \frac{C_{\alpha f}C_{\alpha R}\left(bL - \frac{W_Ru^2L}{gC_{\alpha R}}\right)}{MuI} \tag{B.25}$$

Rear steer, δ_R :

$$\left(a_4 b_2 - a_2 b_4\right) = \frac{C_{\alpha f} C_{\alpha R}(aL) + C_{\alpha R} b M u^2}{MuI} \tag{B.26}$$

$$= \frac{C_{\alpha f} C_{\alpha R}\left(aL + \frac{Mu^2 b}{C_{\alpha f}}\right)}{MuI}$$

$$= \frac{C_{\alpha f} C_{\alpha R}\left(aL + \frac{W_f u^2 L}{g C_{\alpha f}}\right)}{MuI} \tag{B.27}$$

Slope, θ :

$$b_2 \frac{W}{M} = \left(\frac{C_{\alpha f} a^2 + C_{\alpha R} b^2}{uI}\right)\frac{W}{M} \tag{B.28}$$

$$= \frac{W_R L a C_{\alpha f} + W_f L b C_{\alpha f}}{MuI}$$

$$= \frac{C_{\alpha f} C_{\alpha R}\left(\frac{W_R}{C_{\alpha R}} La + \frac{W_f}{C_{\alpha f}} Lb\right)}{MuI} \tag{B.29}$$

Combine the parts from each steer angle and the slope force (Eqns. B.25, B.27, B.29)

$$= \frac{C_{\alpha f} C_{\alpha R}\left(bL - \frac{W_R u^2 L}{g C_{\alpha R}}\right)\delta_f + C_{\alpha f} C_{\alpha R}\left(aL + \frac{W_f u^2 L}{g C_{\alpha f}}\right)\delta_R + C_{\alpha f} C_{\alpha R}\left(\frac{W_R}{C_{\alpha R}} La + \frac{W_f}{C_{\alpha f}} Lb\right)\sin(\theta)}{MuI}$$

$$= \frac{C_{\alpha f} C_{\alpha R}\left(\left(bL - \frac{W_R}{C_{\alpha R}}\frac{u^2 L}{g}\right)\delta_f + \left(aL + \frac{W_f}{C_{\alpha f}}\frac{u^2 L}{g}\right)\delta_R + \left(\frac{W_R}{C_{\alpha R}} La + \frac{W_f}{C_{\alpha f}} Lb\right)\sin(\theta)\right)}{MuI} \tag{B.30}$$

Next, combine numerator and denominator from and simplify to obtain the final steady state solution for lateral velocity, shown in Eqn. B.28.

$$= \frac{uC_{\alpha f}C_{\alpha R}\left(\left(bL - \frac{W_R}{C_{\alpha R}}\frac{u^2 L}{g}\right)\delta_f + \left(aL + \frac{W_f}{C_{\alpha f}}\frac{u^2 L}{g}\right)\delta_R + \left(\frac{W_R}{C_{\alpha R}}La + \frac{W_f}{C_{\alpha f}}Lb\right)\sin(\theta)\right)}{L^2 C_{\alpha f}C_{\alpha R} + \frac{u^2}{g}\left(W_f LC_{\alpha R} - W_R LC_{\alpha f}\right)} \quad \text{(B.31)}$$

Multiplying by

$$\frac{1/L^2 C_{\alpha f}C_{\alpha R}}{1/L^2 C_{\alpha f}C_{\alpha R}}$$

leads to the final steady state equation

$$v_{ss} = \frac{u\left(\left(\frac{b}{L} - \frac{W_R}{C_{\alpha R}}\frac{u^2}{Lg}\right)\delta_f + \left(\frac{a}{L} + \frac{W_f}{C_{\alpha f}}\frac{u^2}{Lg}\right)\delta_R + \left(\frac{W_f}{C_{\alpha f}}\frac{b}{L} + \frac{W_R}{C_{\alpha R}}\frac{a}{L}\right)\sin(\theta)\right)}{1 + \frac{Ku^2}{Lg}} \quad \text{(B.32)}$$

B.3.4 Steady state sideslip angle

The sideslip angle is defined by Eqn. 2.5, which is again shown here as Eqn. B.33. The simple relationship to *v* gives the steady state solution in Eqn. B.34.

$$\frac{v_{ss}}{u} = \tan(\beta_{ss}) = \beta_{ss} \quad \text{(B.33)}$$

$$\beta_{ss} = \frac{\left(\frac{b}{L} - \frac{W_R}{C_{\alpha R}}\frac{u^2}{Lg}\right)\delta_f + \left(\frac{a}{L} + \frac{W_f}{C_{\alpha f}}\frac{u^2}{Lg}\right)\delta_R + \left(\frac{W_f}{C_{\alpha f}}\frac{b}{L} + \frac{W_R}{C_{\alpha R}}\frac{a}{L}\right)\sin(\theta)}{1 + \frac{Ku^2}{Lg}} \quad \text{(B.34)}$$

Comparison to Gillespie [1] shows that this derivation is correct. Gillespie shows that steady state β is given by

$$\beta = 57.3\frac{c}{R} - \frac{W_r V^2}{C_{\alpha r} g R} \quad [1, \text{p208}] \tag{B.35}$$

where c is the distance from the rear to the CG (b), R is the radius of turn, and V is the forward speed (u). The 57.3 simply accounts for radians to degrees conversion. Setting δ_R and θ to zero in Eqn. B.34 gives

$$\beta_{ss} = \frac{\left(\frac{b}{L} - \frac{W_R}{C_{\alpha R}}\frac{u^2}{Lg}\right)\delta_f}{1 + \frac{Ku^2}{Lg}} \tag{B.36}$$

From the derivation of Eqn. B.21,

$$\delta_f = \frac{L}{R} + \frac{Ku^2}{Lg} \tag{B.37}$$

Substituting Eqn. B.37 into B.36 and rearranging gives the steady state sideslip equation shown in Eqn. B.38 ,which matches Eqn. B.35.

$$\beta_{ss} = \frac{\left(\frac{b}{L} - \frac{W_R}{C_{\alpha R}}\frac{u^2}{Lg}\right)\left(\frac{L}{R} + \frac{Ku^2}{Rg}\right)}{1 + \frac{Ku^2}{Lg}}$$

$$= \frac{\left(\frac{b}{L} - \frac{W_R}{C_{\alpha R}}\frac{u^2}{Lg}\right)\left(1 + \frac{Ku^2}{Lg}\right)\left(\frac{L}{R}\right)}{1 + \frac{Ku^2}{Lg}} = \left(\frac{b}{L} - \frac{W_R}{C_{\alpha R}}\frac{u^2}{Lg}\right)\left(\frac{L}{R}\right)$$

$$\beta_{ss} = \frac{b}{R} - \frac{W_R}{C_{\alpha R}}\frac{u^2}{Rg} \tag{B.38}$$

B.3.5 Alternate derivations for steady state velocity and sideslip

The sideslip is found by determining steady state lateral velocity and then applying Eqn. B.33. To find the steady state when tracking a straight line, begin by setting r_{ss} to zero in Eqn. B.5. Solving for steady state lateral sideslip in this manner results in

$$\beta_{ss} = \frac{a_3}{a_1}\delta_f + \frac{a_4}{a_1}\delta_R + \frac{W/M}{a_1}\theta$$

$$= \left(\frac{C_{\alpha f}/M}{(C_{\alpha f} + C_{\alpha R})/Mu} \right)\delta_f + \left(\frac{C_{\alpha R}/M}{(C_{\alpha f} + C_{\alpha R})/Mu} \right)\delta_R + \left(\frac{W/M}{(C_{\alpha f} + C_{\alpha R})/Mu} \right)\theta$$

which simplifies to

$$\beta_{ss} = \frac{C_{\alpha f}}{C_{\alpha f} + C_{\alpha R}}\delta_f + \frac{C_{\alpha R}}{C_{\alpha f} + C_{\alpha R}}\delta_R + \frac{W}{C_{\alpha f} + C_{\alpha R}}\theta \tag{B.39}$$

Another representation is

$$\beta_{ss} = \frac{b_3}{b_1}\delta_f + \frac{b_4}{b_1}\delta_R$$

$$= \left(\frac{C_{\alpha f} a/I}{(C_{\alpha f} a - C_{\alpha R} b)/Iu} \right)\delta_f - \left(\frac{C_{\alpha R} b/I}{(C_{\alpha f} a - C_{\alpha R} b)/Iu} \right)\delta_R$$

which simplifies to

$$\beta_{ss} = \frac{C_{\alpha f} a}{C_{\alpha f} a - C_{\alpha R} b}\delta_f - \frac{C_{\alpha R} b}{C_{\alpha f} a - C_{\alpha R} b}\delta_R \tag{B.40}$$

Note that Eqn. B.40 is not an appropriate expression when $K = 0$ because the denominator for both the δ_f and δ_R terms goes to zero.

For a rear steer vehicle, $\delta_f = 0$ and substituting the resulting expression for δ_R from Eqn. B.21 where $R = \infty$ and $a_y = 0$ gives

$$\beta_{ss} = \left(\frac{C_{\alpha R} K + W}{C_{\alpha f} + C_{\alpha R}} \right) \theta$$

With a little more algebraic manipulation, this becomes

$$\beta_{ss} = \frac{W_f}{C_{\alpha f}} \theta \tag{B.41}$$

or, using the definition of the cornering coefficient,

$$\beta_{ss} = \frac{1}{c_c} \theta \tag{B.42}$$

Note that units of must be in terms of radians and the c_c represents the cornering coefficient as the sum of both left and right tires in the bicycle model.

B.4 Roots and stability

Placing the state equations from above (Eqns. B.1 and B.3) into Laplace domain and setting the right and side of the equations equal to zero gives

$$(s + b_2) R + b_1 V = 0 \tag{B.39a}$$

$$(s + a_1) V + a_2 R = 0 \tag{B.39b}$$

Solving Eqn. B.39a for R and substituting into Eqn. B.39b results in

$$(s + a_1) V - a_2 \left(\frac{b_1}{s + b_2} \right) V = 0 \tag{B.40}$$

Solve for s to obtain the characteristic equation, shown in Eqn. B.41.

$$(s+a_1)-a_2\left(\frac{b_1}{s+b_2}\right)=0$$

$$(s+a_1)(s+b_2)-a_2b_1=0$$

$$s^2+(a_1+b_2)s+(a_1b_2-a_2b_1)=0 \tag{B.41}$$

Using the quadratic formula, the roots of the characteristic equation become

$$s=\frac{-(a_1+b_2)\pm\sqrt{(a_1+b_2)^2-4(a_1b_2-a_2b_1)}}{2} \tag{B.42}$$

System stability exists only for negative real roots, either strictly real values or complex conjugate pairs. The real values of the roots will always be negative since

$$a_1+b_2=\left(\frac{C_{\alpha f}+C_{\alpha R}}{Mu}+\frac{C_{\alpha f}a^2+C_{\alpha R}b^2}{Iu}\right)>0 \tag{B.43}$$

except when

$$(a_1+b_2)<\sqrt{(a_1+b_2)^2-4(a_1b_2-a_2b_1)} \tag{B.44}$$

After some simple algebra, this occurs when

$$(a_1b_2-a_2b_1)<0 \tag{B.45}$$

Next, substitute for a_1, a_2, b_1, and b_2 from Eqns. B.2 and B.4 and simplify

$$\left(a_1 b_2 - a_2 b_1\right) = \left(\frac{C_{\alpha f} + C_{\alpha R}}{Mu}\right)\left(\frac{C_{\alpha f} a^2 + C_{\alpha R} b^2}{Iu}\right) - \left(\frac{Mu^2 + C_{\alpha f} a - C_{\alpha R} b}{Mu}\right)\left(\frac{C_{\alpha f} a - C_{\alpha R} b}{Iu}\right)$$

$$= \frac{\left(C_{\alpha f} + C_{\alpha R}\right)\left(C_{\alpha f} a^2 + C_{\alpha R} b^2\right) - \left(Mu^2 + C_{\alpha f} a - C_{\alpha R} b\right)\left(C_{\alpha f} a - C_{\alpha R} b\right)}{Mu^2 I}$$

$$= \frac{C_{\alpha f} C_{\alpha R} a^2 + 2C_{\alpha f} C_{\alpha R} ab + C_{\alpha f} C_{\alpha R} b^2 + Mu^2\left(C_{\alpha R} b - C_{\alpha f} a\right)}{Mu^2 I}$$

$$= \frac{C_{\alpha f} C_{\alpha R}\left(a+b\right)^2 + Mu^2\left(C_{\alpha R} b - C_{\alpha f} a\right)}{Mu^2 I}$$

$$= \frac{C_{\alpha f} C_{\alpha R} L^2 + Mu^2\left(C_{\alpha R} b - C_{\alpha f} a\right)}{Mu^2 I} \cdot \frac{1/L^2 C_{\alpha f} C_{\alpha R}}{1/L^2 C_{\alpha f} C_{\alpha R}}$$

$$= \frac{1 + \frac{Mu^2}{L^2}\left(\frac{b}{C_{\alpha f}} - \frac{a}{C_{\alpha R}}\right)}{Mu^2 I / L^2 C_{\alpha f} C_{\alpha R}}$$

$$= \frac{1 + \frac{u^2}{Lg}\left(\frac{W_f}{C_{\alpha f}} - \frac{W_R}{C_{\alpha R}}\right)}{Mu^2 I / L^2 C_{\alpha f} C_{\alpha R}}$$

$$= \frac{L^2 C_{\alpha f} C_{\alpha R}\left(1 + \frac{Ku^2}{Lg}\right)}{Mu^2 I} \qquad \text{(B.46)}$$

When Eqn. B.46 replaces the left hand term in Eqn. B.45, this gives

$$\frac{L^2 C_{\alpha f} C_{\alpha R}\left(1+\frac{Ku^2}{Lg}\right)}{Mu^2 I} < 0$$

$$1+\frac{Ku^2}{Lg} < 0$$

$$\frac{Ku^2}{Lg} < -1 \tag{B.47}$$

where K is the understeer gradient defined as

$$K = \frac{W_f}{C_{\alpha f}} - \frac{W_R}{C_{\alpha R}} \tag{B.48}$$

Eqn. B.47 occurs when $K < 0$ since all other variables are positive; however, only negative values of K where,

$$K < \frac{-Lg}{u^2} \tag{B.49}$$

Another way to look at this is to see the speed, u, at which this occurs for a certain K. This is known as the critical speed (u_{crit}).

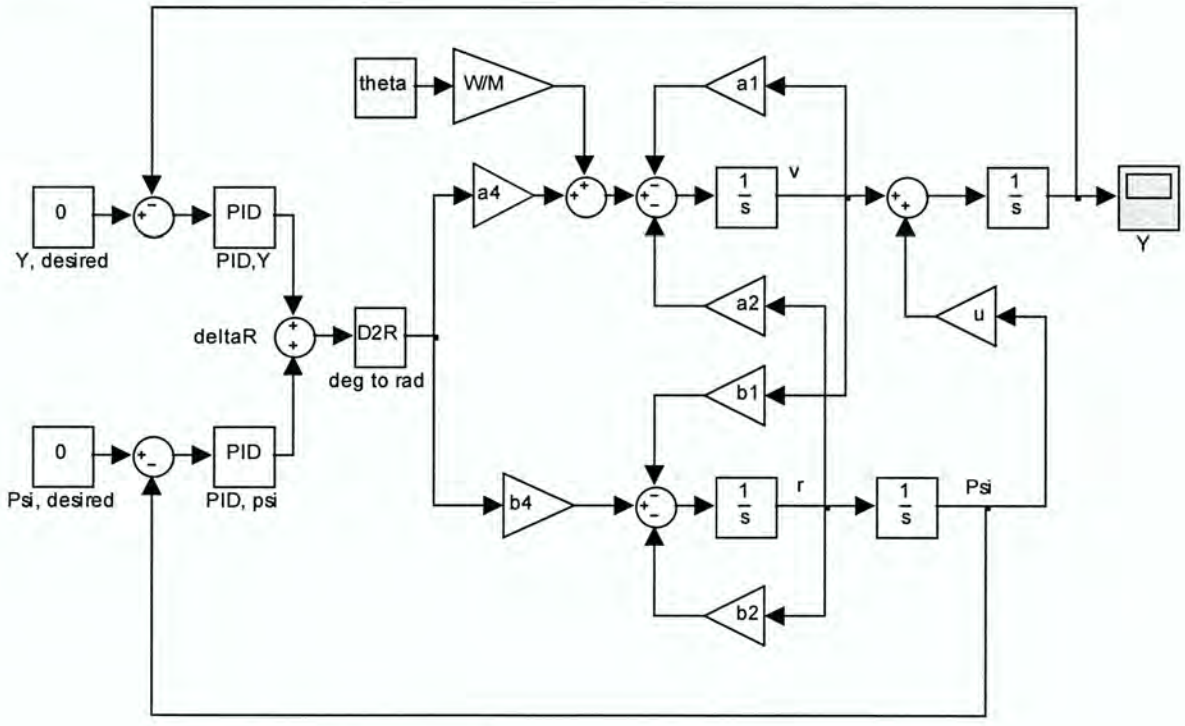
$$u = \sqrt{\frac{Lg}{|K|}} \tag{B.50}$$

B.5 Block diagram representation

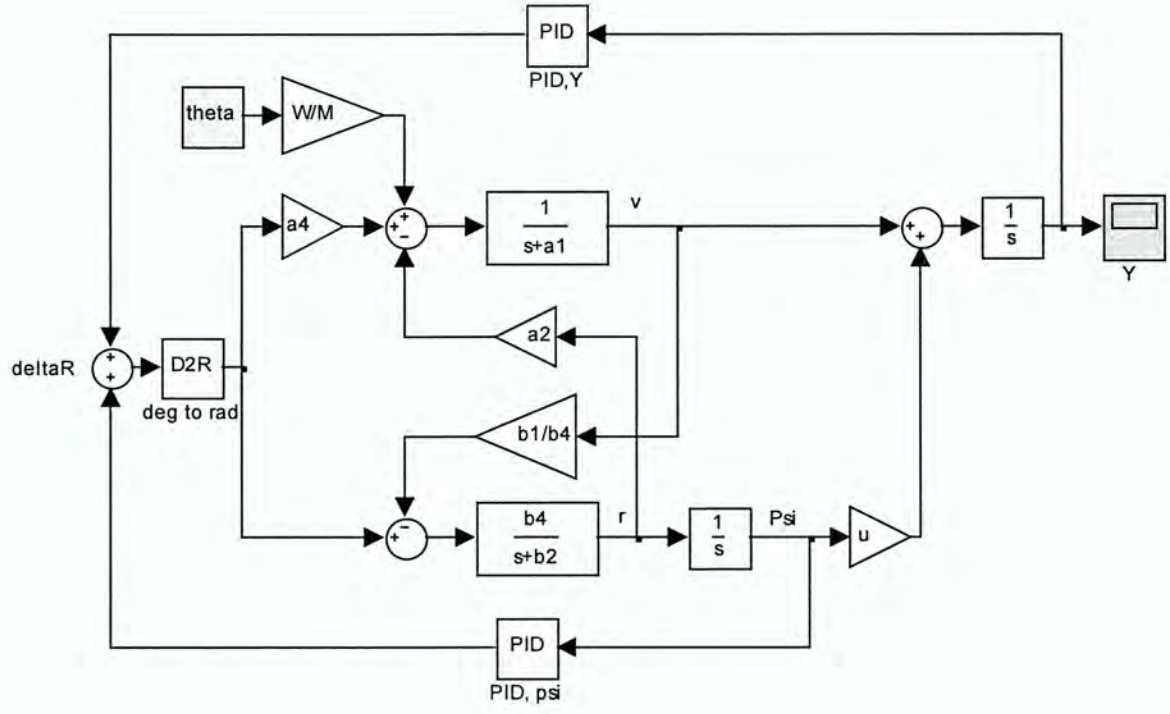
The following sections go through the step by step block diagram reduction of the system to a single transfer function. The diagrams were created using Simulink®, a product of The MathWorks, Inc., Natick, MA. Most figures are not named; rather, the step sequence is listed before each diagram. The state equations B.1 and B.3 are used to begin the model.

B.5.1 Block diagram reduction for slope input

Step 1:



Step 2:



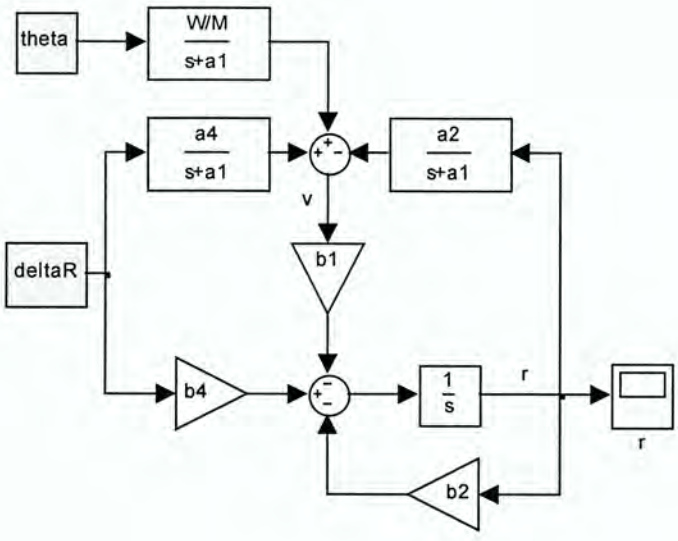
Step 3:

Use relationship discussed in Section 3.4.1 to reduce the block diagram by relating *v* to *r*, shown again here as Eqns. B.51 and B.52.

$$V = -\frac{a_2}{(s+a_1)}R + \frac{a_3}{(s+a_1)}\delta_f + \frac{a_4}{(s+a_1)}\delta_R + \frac{{}^{W}/_{M}}{(s+a_1)}\theta \tag{B.51}$$

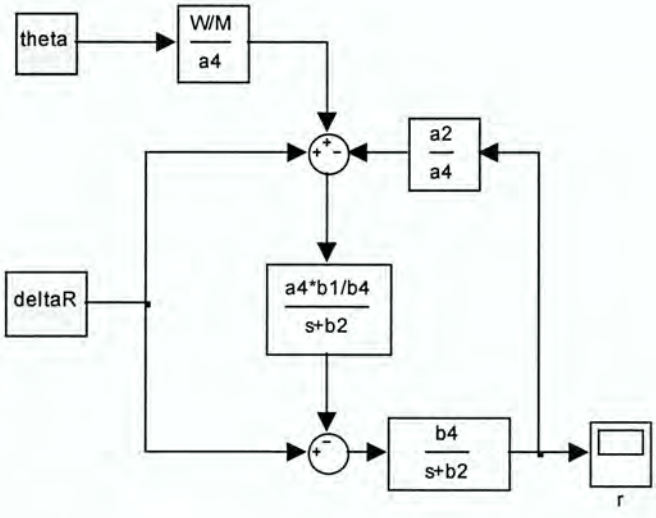
$$R = -\frac{b_1}{(s+b_2)}V + \frac{b_3}{(s+b_2)}\delta_f + \frac{b_4}{(s+b_2)}\delta_R \tag{B.52}$$

First, look only at the block diagram related up until the output of *r*. The diagram is shown of the following page.



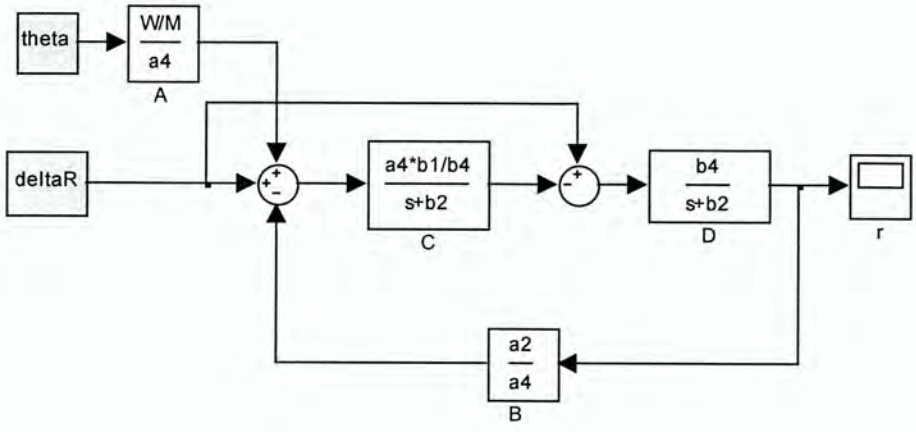
Step 4:

Reduce the feedback loop around r, move constant b_4 across the bottom summing junction, and move the transfer function block at the left of the top summing junction across the block.



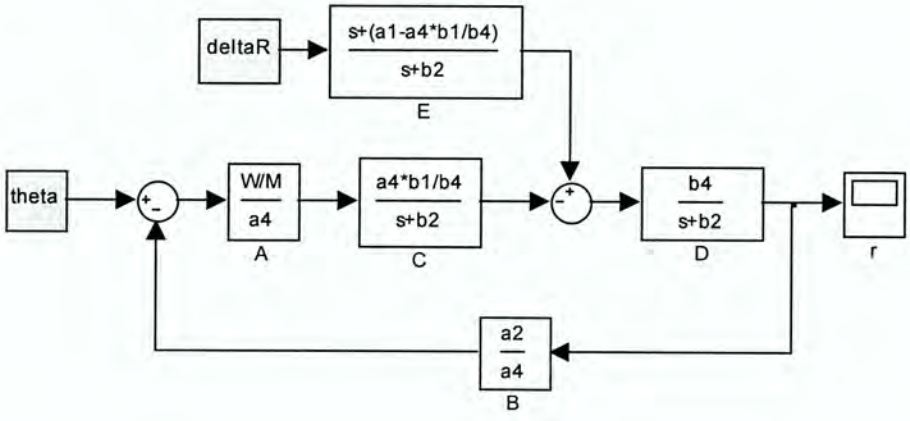
Step 5:

Rearrange diagram from Step 4.



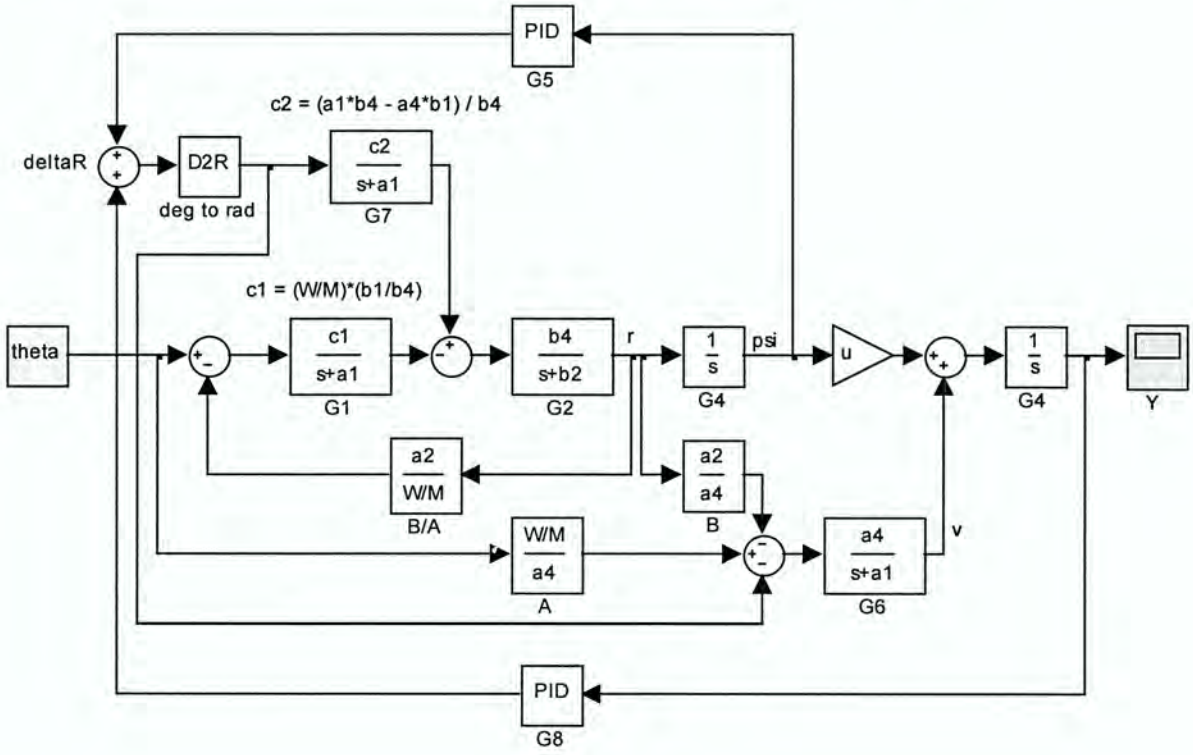
Step 6:

Rearrange Step 5 to make θ the main input.



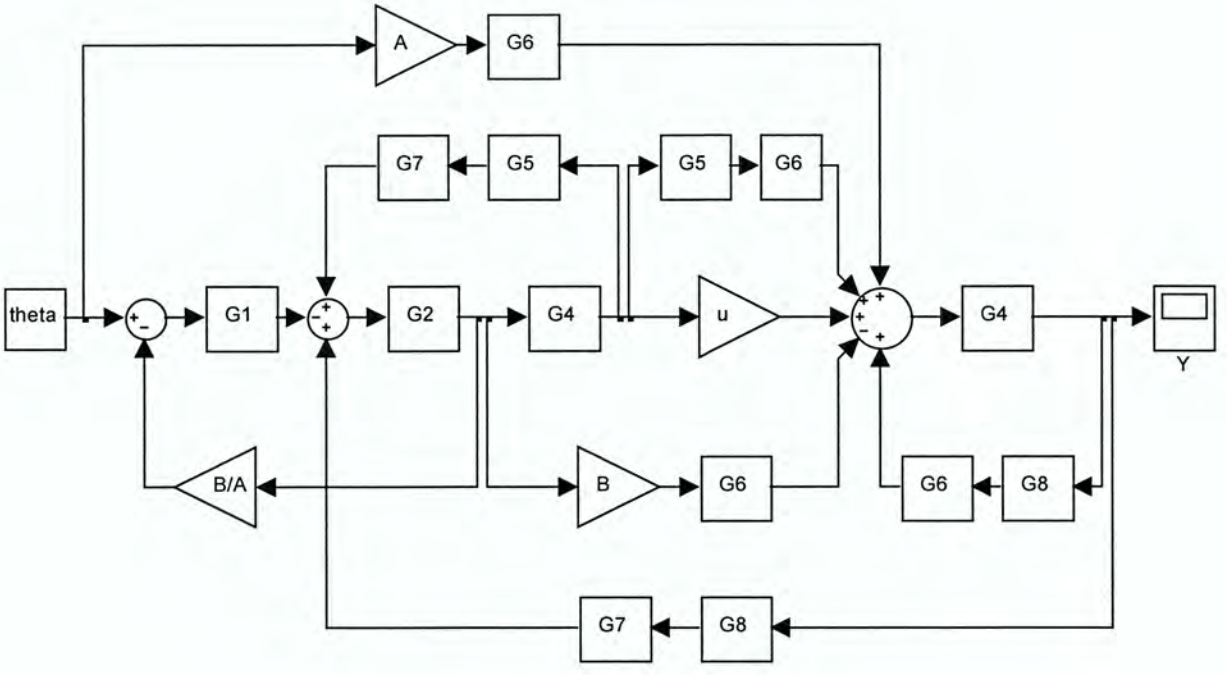
Step 7:

Substitute Step 6 back into the entire block diagram system using the relationship in Eqn. B.51 to recreate *v* to be summed at the last summing junction resulting in $\dot{Y}$.

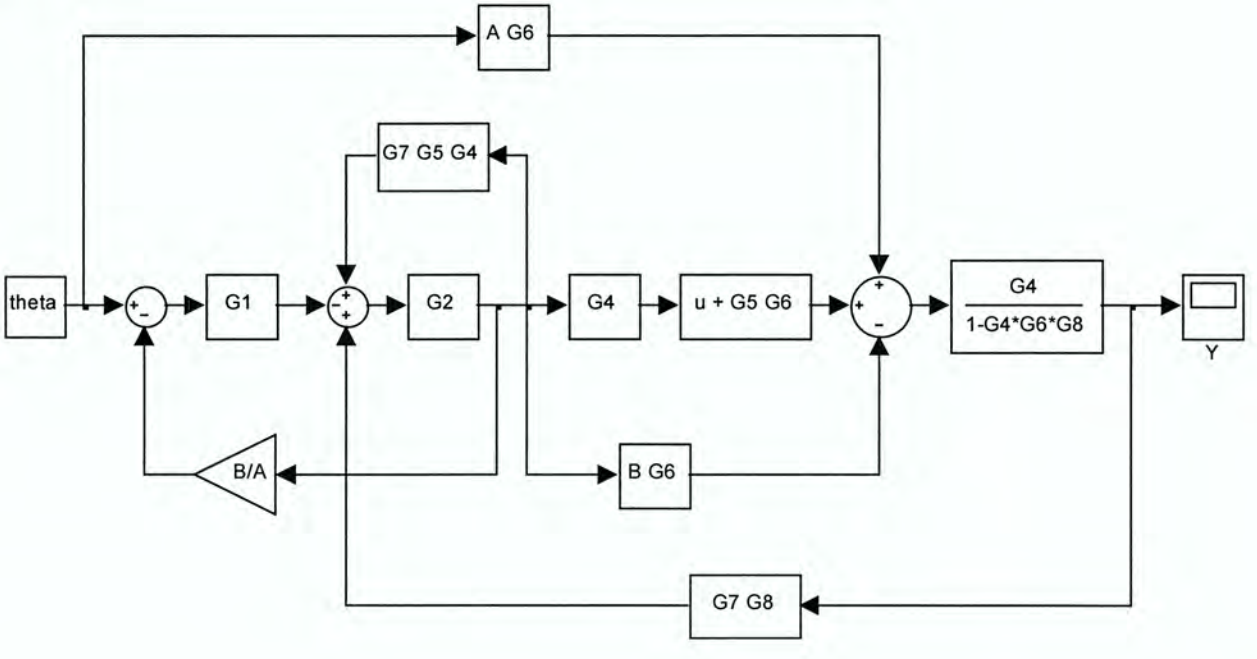


Step 8:

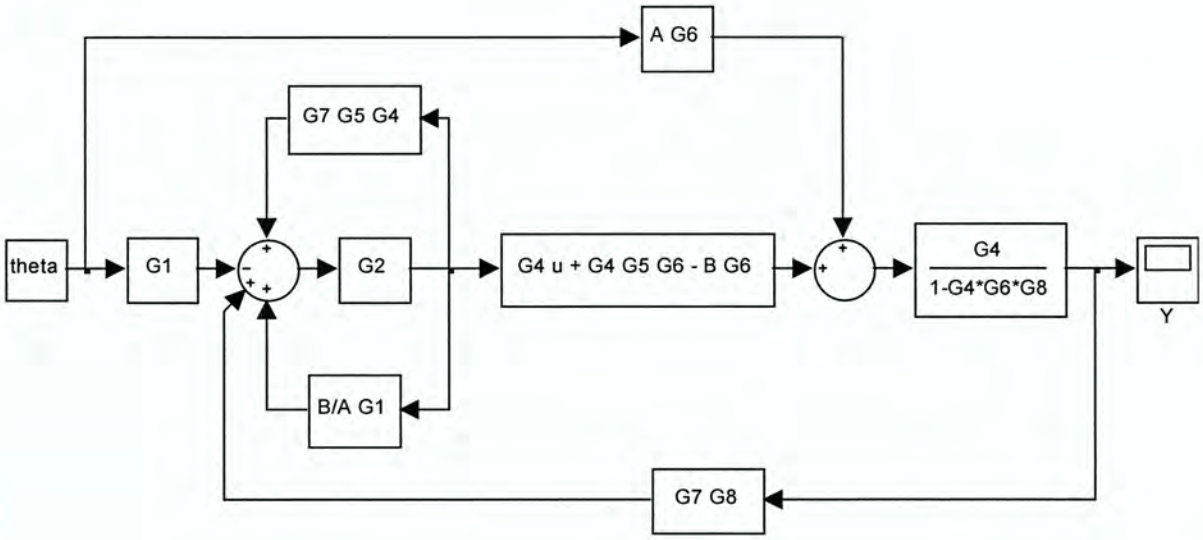
Use the labels underneath the blocks and the constants above them to make further simplification easier. The rear steer angle summing junction can be eliminated by feeding its output signal through to the other summing junctions. Likewise, the lateral velocity summing junction can be eliminated leaving the following block diagram representation. The D2R, degrees to radians conversion block has also been lumped into G5 and G8.


A
G6
G7
G5
G5
G6
theta
G1
G2
G4
u
G4
Y
B/A
B
G6
G6
G8
G7
G8

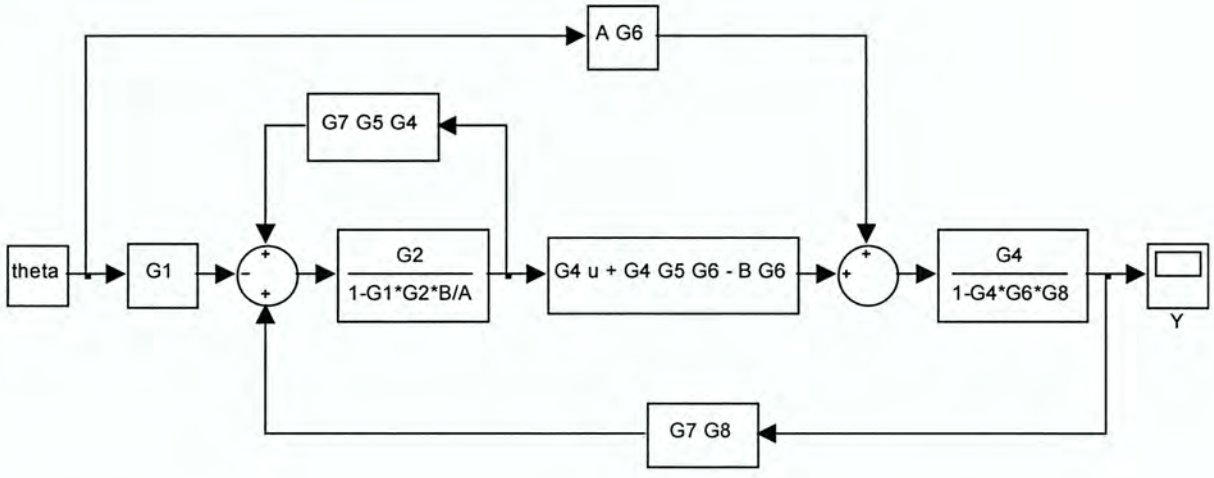
Step 9:


A G6
G7 G5 G4
theta
G1
G2
G4
u + G5 G6
G4
1-G4*G6*G8
Y
B/A
B G6
G7 G8

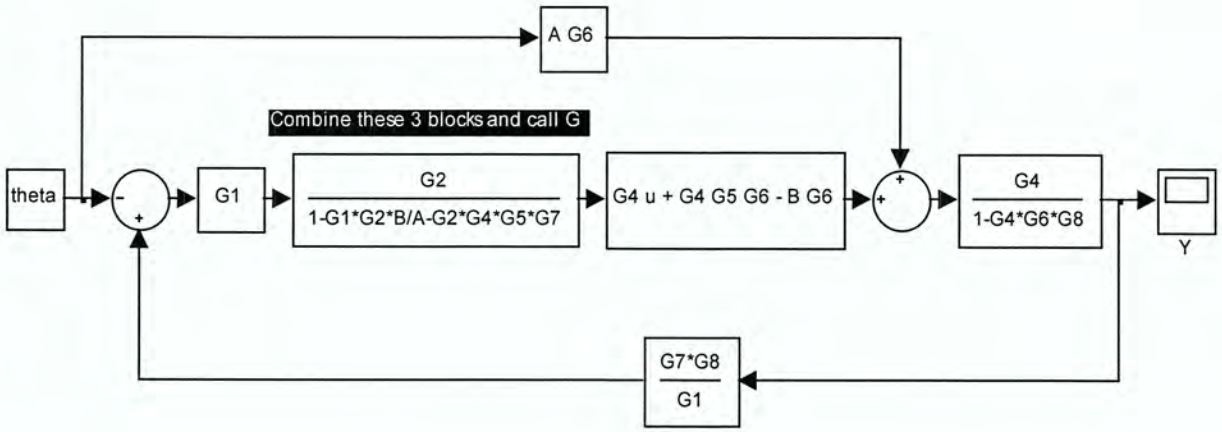
Step 10:



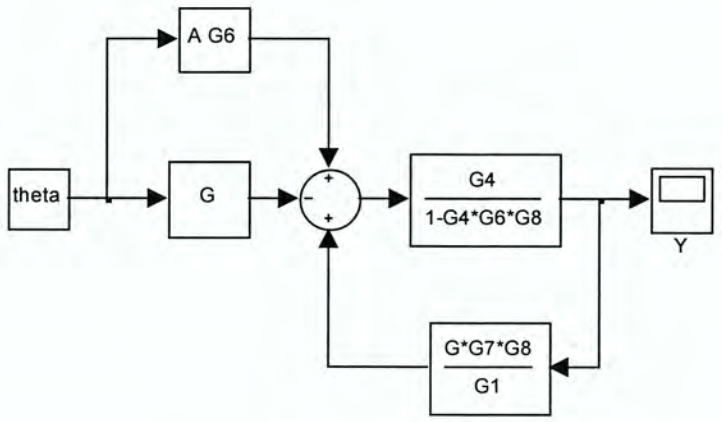
Step 11:



Step 12:

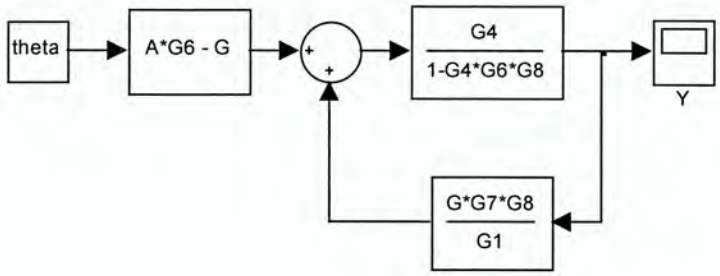


Step 13:

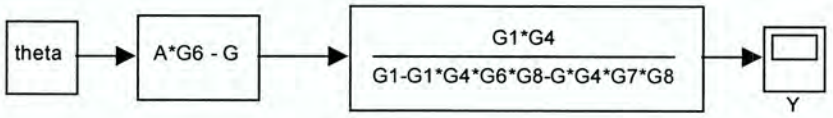


$$G = \frac{G_1 G_2 G_4 u - G_1 G_2 G_4 G_5 G_6 - B G_1 G_2 G_6}{1 - G_1 G_2 \dfrac{B}{A} - G_2 G_4 G_5 G_7} \qquad \text{(B.53)}$$

Step 14:



Step 15:

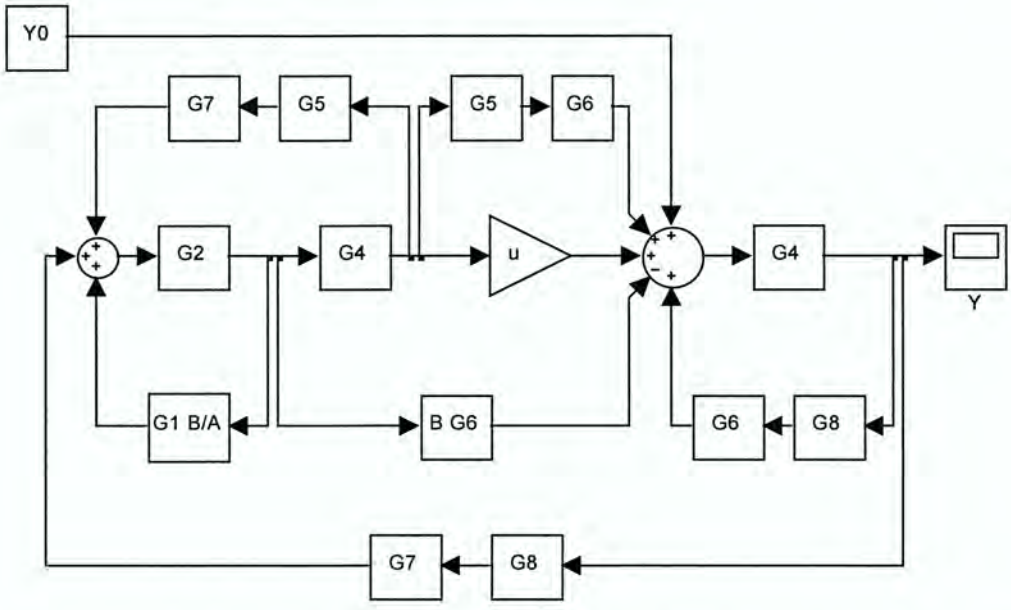


Combine the two transfer functions in Step 15 to achieve the final transfer function shown in Eqn. B.54 with G defined in Eqn. B.53.

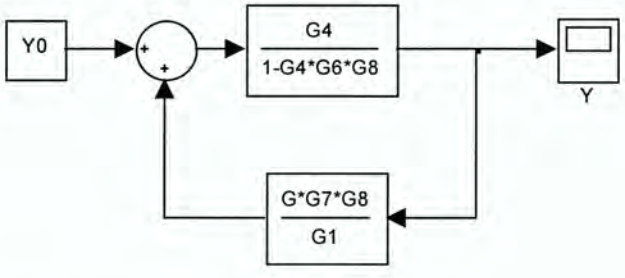
$$\frac{Y}{\theta} = \frac{AG_1G_4G_6 - GG_1G_4}{G_1 - G_1G_4G_6G_8 - GG_4G_7G_8} \tag{B.54}$$

B.5.2 Block diagram reduction for initial condition, Y_0

The reduction of the block diagram is almost identical to the reduction of the system with the slope as the input. To incorporate Y_0, θ is set to zero and therefore all signal paths with θ disappear. The resulting diagram is shown below.



This reduces in exactly the same manner as the slope input diagram, and produces the same step sequence with the only difference being the input is the initial condition rather than the slope. The difference can be seen by looking at Step 14 with Y_0 as the input. This reduces to the final transfer function given in Eqn. B.55.



$$\frac{Y}{Y_0} = \frac{G_1 G_4}{G_1 - G_1 G_4 G_6 G_8 - G G_4 G_7 G_8} \tag{B.55}$$

B.6 Transfer Function Derivations

This section substitutes values for the *G* terms in the previous section to derive the final transfer functions for the slope input and initial lateral error input. It also shows the generalized transfer functions for each of the controller gains. In order to simplify the main transfer function shown in Eqn. B.54, the numerator and denominator are separated and divided into parts. The parts are then combined and the overall transfer function is created.

B.6.1 Transfer function *G*

The transfer function *G* is used in both the numerator and denominator of Eqn. B.54. Its derivation is shown first in order to further derive the others. The expression for *G* is shown in Eqn. B.56.

$$G = \frac{G_1 G_2 G_4 u - G_1 G_2 G_4 G_5 G_6 - B G_1 G_2 G_6}{1 - G_1 G_2 \frac{B}{A} - G_2 G_4 G_5 G_7} \tag{B.56}$$

Numerator

$$G_1 G_2 G_4 u = \frac{g b_1}{b_4 (s + a_1)} \cdot \frac{b_4}{s + b_2} \cdot \frac{1}{s} \cdot u$$

$$= \frac{g b_1 u}{s(s + a_1)(s + b_2)} \tag{B.57}$$

$$G_1 G_2 G_4 G_5 G_6 = \frac{g b_1}{b_4 (s + a_1)} \cdot \frac{b_4}{s + b_2} \cdot \frac{1}{s} \cdot \frac{K_{d\psi} s^2 + K_{p\psi} s + K_{i\psi}}{s} \cdot x \cdot \frac{a_4}{s + a_1}$$

$$= \frac{g b_1 x a_4 \left(K_{d\psi} s^2 + K_{p\psi} s + K_{i\psi}\right)}{s^2 (s + a_1)^2 (s + b_2)} \tag{B.58}$$

$$BG_1G_2G_6 = \frac{a_2}{a_4}\cdot\frac{gb_1}{b_4(s+a_1)}\cdot\frac{b_4}{s+b_2}\cdot\frac{a_4}{s+a_1}$$

$$= \frac{a_2 gb_1}{(s+a_1)^2(s+b_2)} \tag{B.59}$$

Combine the parts of the numerator to get the overall numerator of *G*.

$$= \frac{gb_1\left(\left(u+xa_4K_{d\psi}-a_2\right)s^2+\left(ua_1+xa_4K_{p\psi}\right)s+xa_4K_{i\psi}\right)}{s^2(s+a_1)^2(s+b_2)} \tag{B.60}$$

Now look at the individual parts of the denominator.

Denominator

$$G_1G_2\frac{B}{A} = \frac{gb_1}{b_4(s+a_1)}\cdot\frac{b_4}{(s+b_2)}\cdot\frac{a_2}{a_4}\cdot\frac{a_4}{g}$$

$$= \frac{a_2b_1}{(s+a_1)(s+b_2)} \tag{B.61}$$

$$G_2G_4G_5G_7 = \frac{b_4}{s+b_2}\cdot\frac{1}{s}\cdot\frac{K_{d\psi}s^2+K_{p\psi}s+K_{i\psi}}{s}\cdot x\cdot\frac{s+(a_1b_4-a_4b_1)/b_4}{s+a_1}$$

$$= \frac{x\left(\left(K_{d\psi}b_4\right)s^3+\left(K_{d\psi}(a_1b_4-a_4b_1)+K_{p\psi}b_4\right)s^2+\left(K_{p\psi}(a_1b_4-a_4b_1)+K_{i\psi}b_4\right)s+K_{i\psi}(a_1b_4-a_4b_1)\right)}{s^2(s+a_1)(s+b_2)} \tag{B.62}$$

Combine the parts in the denominator and find a common denominator among those terms.

$$= \frac{s^2(s+a_1)(s+b_2)-a_2b_1s^2-x\left(\begin{array}{l}\left(K_{d\psi}b_4\right)s^3+\left(K_{d\psi}(a_1b_4-a_4b_1)+K_{p\psi}b_4\right)s^2\\ \quad+\left(K_{p\psi}(a_1b_4-a_4b_1)+K_{i\psi}b_4\right)s+K_{i\psi}(a_1b_4-a_4b_1)\end{array}\right)}{s^2(s+a_1)(s+b_2)}$$

Now simplify and collect the *s* terms to obtain the overall denominator of *G*.

$$= \frac{\begin{pmatrix} s^4 + (a_1 + b_2 - xb_4 K_{d\psi})s^3 + (a_1 b_2 - a_2 b_1 - K_{d\psi} x(a_1 b_4 - a_4 b_1) - K_{p\psi} xb_4)s^2 \\ + (-K_{p\psi} x(a_1 b_4 - a_4 b_1) - K_{i\psi} xb_4)s - K_{i\psi} x(a_1 b_4 - a_4 b_1) \end{pmatrix}}{s^2 (s + a_1)(s + b_2)}$$

(B.63)

Next, combine the numerator and denominator from Eqns. B.60 and B.63.

$$= \frac{gb_1((u - a_2 + K_{d\psi} xa_4)s^2 + (ua_1 + K_{p\psi} xa_4)s + K_{i\psi} xa_4)}{(s + a_1)\begin{pmatrix} s^4 + (a_1 + b_2 - K_{d\psi} xb_4)s^3 + (a_1 b_2 - a_2 b_1 + K_{d\psi} x(a_4 b_1 - a_1 b_4) - K_{p\psi} xb_4)s^2 \\ + (K_{p\psi} x(a_4 b_1 - a_1 b_4) - K_{i\psi} xb_4)s + K_{i\psi} x(a_4 b_1 - a_1 b_4) \end{pmatrix}}$$

(B.64)

Once again, break the equation into pieces to make simplification easier. First, expand the denominator and collect the *s* terms.

Denominator

$$\begin{aligned} &= s^5 + a_1 s^4 + (a_1 + b_2 - K_{d\psi} xb_4)(s^4 + a_1 s^3) + (a_1 b_2 - a_2 b_1 + K_{d\psi} x(a_4 b_1 - a_1 b_4) - K_{p\psi} xb_4)(s^3 + a_1 s^2) \\ &\quad + (K_{p\psi} x(a_4 b_1 - a_1 b_4) - K_{i\psi} xb_4)(s^2 + a_1 s) + K_{i\psi} x(a_4 b_1 - a_1 b_4)(s + a_1) \end{aligned}$$

$$\begin{aligned} &= s^5 + (2a_1 + b_2 - K_{d\psi} xb_4)s^4 \\ &\quad + (a_1^2 + a_1 b_2 - K_{d\psi} xa_1 b_4 + a_1 b_2 - a_2 b_1 + K_{d\psi} x(a_4 b_1 - a_1 b_4) - K_{p\psi} xb_4)s^3 \\ &\quad + (a_1^2 b_2 - a_1 a_2 b_1 - K_{d\psi} xa_1 (a_4 b_1 - a_1 b_4) - K_{p\psi} xa_1 b_4 + K_{p\psi} x(a_4 b_1 - a_1 b_4) - K_{i\psi} xb_4)s^2 \\ &\quad + (K_{p\psi} xa_1 (a_4 b_1 - a_1 b_4) - K_{i\psi} xa_1 b_4 + K_{i\psi} x(a_4 b_1 - a_1 b_4))s \\ &\quad + (K_{i\psi} xa_1 (a_4 b_1 - a_1 b_4)) \end{aligned}$$

(B.65)

Combine Eqn. B.65 with the numerator of Eqn. B.64 to obtain the final transfer function for *G* shown in two forms in Eqns. B.66 and B.67. The later is used when substituted into the expressions in following sections.

$$G = \frac{gb_1\left(\left(n_1 + K_{d\psi} n_2\right)s^2 + \left(n_3 + K_{p\psi} n^2\right)s + K_{i\psi} n_2\right)}{\left(s + a_1\right)\left(s^4 + \left(d_1 - K_{d\psi} d_2\right)s^3 + \left(d_3 + K_{d\psi} d_4 - K_{p\psi} d_2\right)s^2 + \left(K_{p\psi} d_4 - K_{i\psi} d_2\right)s + K_{i\psi} d_4\right)} \tag{B.66}$$

where

$$\begin{aligned} d_1 &= a_1 + b_2 \\ d_2 &= xb_4 \\ d_3 &= a_1 b_2 - a_2 b_1 \\ d_4 &= x\left(a_4 b_1 - a_1 b_4\right) \\ d_5 &= 2a_1 + b_2 \\ d_6 &= a_1^2 + 2a_1 b_2 - a_2 b_1 \\ d_7 &= a_1^2 - a_1 a_2 b_1 \\ d_8 &= xa_1\left(a_4 b_1 - a_1 b_4\right) \\ d_9 &= x\left(a_4 b_1 - 2a_1 b_4\right) \end{aligned} \tag{B.66a}$$

$$\begin{aligned} n_1 &= u - a_2 \\ n_2 &= xa_4 \\ n_3 &= ua_1 \end{aligned} \tag{B.66b}$$

$$G = \frac{gb_1\left(N_2 s^2 + N_1 s + N_0\right)}{\left(s + a_1\right)\left(s^4 + D_3 s^3 + D_2 s^2 + D_1 s + D_0\right)} \tag{B.67}$$

where

$$\begin{aligned} N_2 &= n_1 + K_{d\psi} n_2 \\ N_1 &= n_3 + K_{p\psi} n_2 \\ N_0 &= K_{i\psi} n_2 \end{aligned} \tag{B.67a}$$

$$\begin{aligned} D_3 &= d_1 - K_{d\psi} d_2 \\ D_2 &= d_3 + K_{d\psi} d_4 - K_{p\psi} d_2 \\ D_1 &= K_{p\psi} d_4 - K_{i\psi} d_2 \\ D_0 &= K_{i\psi} d_4 \end{aligned} \tag{B.67b}$$

B.6.2 Main transfer function T

$$\frac{Y}{\theta} = \frac{AG_1G_4G_6 - GG_1G_4}{G_1 - G_1G_4G_6G_8 - GG_4G_7G_8} \tag{B.68}$$

Numerator

$$AG_1G_4G_6 = \frac{g}{a_4} \cdot \frac{gb_1}{b_4(s+a_1)} \cdot \frac{1}{s} \cdot \frac{a_4}{s+a_1}$$

$$= \frac{g^2 b_1}{b_4 s(s+a_1)^2} \tag{B.69}$$

$$GG_1G_4 = \frac{gb_1(N_2s^2 + N_1s + N_0)}{(s+a_1)(s^4 + D_3s^3 + D_2s^2 + D_1s + D_0)} \cdot \frac{gb_1}{b_4(s+a_1)} \cdot \frac{1}{s}$$

$$= \frac{g^2 b_1^2 (N_2s^2 + N_1s + N_o)}{b_4 s(s+a_1)^2 (s^4 + D_3s^3 + D_2s^2 + D_1s + D_o)} \tag{B.70}$$

Subtracting Eqn. B.70 from B.69 and collecting *s* terms gives the overall numerator.

$$= \frac{g^2 b_1 (s^4 + D_3s^3 + (D_2 - b_1N_2)s^2 + (D_1 - b_1n_1)s + (D_0 - b_1N_0))}{b_4 s(s+a_1)^2 (s^4 + D_3s^3 + D_2s^2 + D_1s + D_0)} \tag{B.71}$$

Denominator

$$G_1 = \frac{gb_1}{b_4(s+a_1)} \tag{B.72}$$

$$G_1G_4G_6G_8 = \frac{gb_1}{b_4(s+a_1)} \cdot \frac{1}{s} \cdot \frac{a_4}{s+a_1} \cdot \frac{K_{dY}s^2 + K_{pY}s + K_{iY}}{s} x$$

$$= \frac{gb_1a_4x(K_{dY}s^2 + K_{pY}s + K_{iY})}{b_4s^2(s+a_1)^2} \tag{B.73}$$

$$GG_4G_7G_8 = \frac{gb_1\left(N_2s^2+N_1s+N_0\right)}{\left(s+a_1\right)\left(s^4+D_3s^3+D_2s^2+D_1s+D_0\right)}\cdot\frac{1}{s}\cdot\frac{b_4s+a_1b_4-a_4b_1}{b_4\left(s+a_1\right)}\cdot\frac{K_{dY}s^2+K_{pY}s+K_{iY}}{s}x$$

$$=\frac{gb_1x\left(N_2s^2+N_1s+N_0\right)\left(b_4s+a_1b_4-a_4b_1\right)\left(K_{dY}s^2+K_{pY}s+K_{iY}\right)}{b_4s^2\left(s+a_1\right)^2\left(s^4+D_3s^3+D_2s^2+D_1s+D_0\right)} \tag{B.74}$$

Combine the parts of the denominator to get

$$=\frac{gb_1\left(\begin{array}{l}s^2\left(s+a_1\right)\left(s^4+D_3s^3+D_2s^2+D_1s+D_0\right)-a_4x\left(K_{dY}s^2+K_{pY}s+K_{iY}\right)\left(s^4+D_3s^3+D_2s^2+D_1s+D_0\right)\\-x\left(N_2s^2+N_1s+N_0\right)\left(b_4s+a_1b_4-a_4b_1\right)\left(K_{dY}s^2+K_{pY}s+K_{iY}\right)\end{array}\right)}{b_4s^2\left(s+a_1\right)^2\left(s^4+D_3s^3+D_2s^2+D_1s+D_0\right)} \tag{B.75}$$

Now, combine numerator and the denominator (Eqns. B.71 and B.75) to obtain

$$=\frac{sg\left(\left(s^4+D_3s^3+D_2s^2+D_1s+D_0\right)-b_1\left(N_2s^2+N_1s+N_0\right)\right)}{s^2\left(s+a_1\right)\left(s^4+D_3s^3+D_1s+D_0\right)-\left(K_{dY}s^2+K_{pY}s+K_{iY}\right)\left(\begin{array}{l}a_4x\left(s^4+D_3s^3+D_2s^2+D_1s+D_0\right)\\+x\left(N_2s^2+N_1s+N_0\right)\left(b_4+s+a_1b_4-a_4b_1\right)\end{array}\right)} \tag{B.76}$$

Once again, separate the numerator and denominator to make simplification easier.

Denominator

Expand the equations and collect the *s* terms in two steps.

$$\begin{aligned}&=s^7+D_3s^6+D_2s^5+D_1s^4+D_0s^3+a_1s^6+a_1D_3s^5+a_1D_2s^4+a_1D_1s^3+a_1D_0s^2\\&\quad-\left(K_{dY}s^2+K_{pY}s+K_{iY}\right)\left(\begin{array}{l}a_4xs^4+\left(a_4xD_3+N_2b_4x\right)s^3+\left(a_4xD_2+N_2x\left(a_1b_4-a_4b_1\right)+N_1xb_4\right)s^2\\+\left(a_4xD_1+N_1x\left(a_1b_4-a_4b_1\right)+N_0xb_4\right)s+N_0x\left(a_1b_4-a_4b_1\right)+a_4xD_0\end{array}\right)\end{aligned}$$

$$
\begin{aligned}
&= s^7 + \left(D_3 + a_1 - K_{dY} n_2\right) s^6 \\
&\quad + \left(D_2 + a_1 D_3 - K_{dY}\left(n_2 D_3 + N_2 d_2\right) - K_{pY} n_2\right) s^5 \\
&\quad + \left(D_1 + a_1 D_2 - K_{dY}\left(n_2 D_2 - N_2 d_4 + N_1 d_2\right) - K_{pY}\left(n_2 D_3 + N_2 d_2\right) - K_{iY} n_2\right) s^4 \\
&\quad + \left(D_0 + a_1 D_1 - K_{dY}\left(n_2 D_1 - N_1 d_4 + N_0 d_2\right) - K_{pY}\left(n_2 D_2 - N_2 d_4 + N_1 d_2\right) - K_{iY}\left(n_2 D_3 + N_2 d_2\right)\right) s^3 \\
&\quad + \left(a_1 D_0 + K_{dY} N_0 d_4 - K_{pY}\left(n_2 D_1 - N_1 d_4 + N_0 d_2\right) - K_{iY}\left(n_2 D_2 - N_2 d_4 + N_1 d_2\right) - K_{dY} n_2 D_0\right) s^2 \\
&\quad + \left(K_{pY} N_0 d_4 - K_{iY}\left(n_2 D_1 - N_1 d_4 + N_0 d_2\right) - K_{pY} n_2 D_0\right) s \\
&\quad + \left(K_{iY} N_0 d_4 - K_{iY} n_2 D_0\right)
\end{aligned}
\tag{B.77}
$$

When substitution is made for variables, the last term goes to zero. This gives the resulting denominator for the overall transfer function.

$$= s^7 + DD_6 s^6 + DD_5 s^5 + DD_4 s^4 + DD_3 s^3 + DD_2 s^2 + DD_1 s \tag{B.78}$$

where $DD_{1\text{-}6}$ replace the corresponding expressions from Eqn. B.77. Substituting for the constants $D_{0\text{-}3}$, $N_{0\text{-}2}$, $d_{1\text{-}9}$, and $n_{1\text{-}3}$ defined in Eqns. B.66a through B.67b to get the $DD_{1\text{-}6}$ expressions in terms of Eqns. B.2 and B.4 results in the following expressions.

$$
\begin{aligned}
DD_6 &= K_{d\psi} m_1 + K_{dY} m_2 + m_3 \\
DD_5 &= K_{d\psi} m_4 + K_{p\psi} m_1 + K_{pY} m_2 + K_{dY} m_5 + m_6 \\
DD_4 &= K_{p\psi} m_4 + K_{i\psi} m_1 + K_{d\psi} m_7 + K_{iY} m_2 + K_{dY} m_8 + K_{pY} m_5 + m_9 \\
DD_3 &= K_{i\psi} m_4 + K_{p\psi} m_7 + K_{dY} m_{10} + K_{pY} m_8 + K_{iY} m_5 \\
DD_2 &= K_{i\psi} m_7 + K_{pY} m_{10} + K_{iY} m_8 \\
DD_1 &= K_{iY} m_{10}
\end{aligned}
\tag{B.78a}
$$

$$
\begin{aligned}
m_1 &= -xb_4 \\
m_2 &= -xa_4 \\
m_3 &= 2a_1 + b_2 \\
m_4 &= xa_4b_1 - 2xa_1b_4 \\
m_5 &= -xa_1a_4 - xa_4b_2 - xb_4u + xa_2b_4 \\
m_6 &= 2a_1b_2 - a_2b_1 + a_1^2 \\
m_7 &= xa_1a_4b_1 - xa_1^2b_4 \\
m_8 &= -xa_1a_4b_2 + xa_4b_1u - 2xa_1b_4u + xa_1a_2b_4 \\
m_9 &= a_1^2b_2 - a_1a_2b_1 \\
m_{10} &= xa_1a_4b_1u - xa_1^2b_4u
\end{aligned}
\tag{B.78b}
$$

Combine with the numerator to get and multiply both top and bottom by *1/s.*

$$
= \frac{g\left(s^4 + D_3s^3 + \left(D_2 - b_1N_2\right)s^2 + \left(D_1b_1N_1\right)s + \left(D_0 - b_1N_0\right)\right)}{s^6 + DD_6s^5 + DD_5s^4 + DD_4s^3 + DD_3s^2 + DD_2s + DD_1} \tag{B.79}
$$

Now, substitute for expressions in the numerator of Eqn. B.79 to obtain an expression with vehicle parameters as defined in Eqns. B.2 and B.4. This is shown in two steps.

$$
= g\begin{pmatrix} s^4 + \left(a_1 + b_2 - K_{d\psi}xb_4\right)s^3 + \left(a_1b_2 - a_2b_1 + K_{d\psi}x\left(a_4b_1 - a_1b_4\right) - K_{p\psi}xb_4 - b_1u + b_1a_2 - b_1K_{d\psi}xa_4\right)s^2 \\ + \left(K_{p\psi}x\left(a_4b_1 - a_1b_4\right) - K_{i\psi}xb_4 - b_1ua_1 - b_1K_{p\psi}xa_4\right)s + \left(K_{i\psi}x\left(a_4b_1 - a_1b_4\right) - b_1K_{i\psi}xa_4\right) \end{pmatrix}
$$

$$
= g\left(s^4 + \left(K_{d\psi}m_1 + z_1\right)s^3 + \left(K_{d\psi}z_2 + K_{p\psi}m_1 + z_3\right)s^2 + \left(K_{p\psi}z_2 + K_{i\psi}m_1 + z_4\right)s + K_{i\psi}z_2\right) \tag{B.80}
$$

where

$$
\begin{aligned}
z_1 &= a_1 + b_2 \\
z_2 &= -xa_1b_4 \\
z_3 &= a_1b_2 - b_1u \\
z_4 &= -b_1ua_1
\end{aligned}
\tag{B.80a}
$$

Finally, combine Eqn. B.80 and the denominator of Eqn. B.79 to obtain the final transfer function for a slope input shown in two forms.

$$=\frac{g\left(s^4+\left(K_{d\psi}m_1+z_1\right)s^3+\left(K_{p\psi}m_1+K_{d\psi}z_2+z^3\right)s^2+\left(K_{p\psi}z_2+K_{i\psi}m_1+z_4\right)s+K_{i\psi}z_2\right)}{\left(\begin{array}{l}s^6+\left(K_{d\psi}m_1+K_{dY}m_2+m_3\right)s^5+\left(K_{p\psi}m_1+K_{d\psi}m_4+K_{pY}m_2+K_{dY}m_5+m_6\right)s^4\\+\left(K_{p\psi}m_4+K_{i\psi}m_1+K_{d\psi}m_7+K_{pY}m_5+K_{iY}m_2+K_{dY}m_8+m_9\right)s^3\\+\left(K_{p\psi}m_7+K_{i\psi}m_4+K_{pY}m_8+K_{iY}m_5+K_{dY}m_{10}\right)s^2\\+\left(K_{i\psi}m_7+K_{pY}m_{10}+K_{iY}m_8\right)s+K_{iY}m_{10}\end{array}\right)} \quad \text{(B.81)}$$

$$\frac{Y}{\theta}=\frac{g\left(s^4+NN_3s^3+NN_2s^3+NN_2s^2+NN_1s+NN_0\right)}{s^6+DD_6s^5+DD_5s^4+DD_4s^3+DD_3s^2+DD_2s+DD_1} \quad \text{(B.82)}$$

where

$$\begin{aligned}NN_3&=K_{d\psi}m_1+z_1\\NN_2&=K_{p\psi}m_1+K_{d\psi}z_2+z_3\\NN_1&=K_{p\psi}z_2+K_{i\psi}m_1+z_4\\NN_0&=K_{i\psi}z_2\end{aligned} \quad \text{(B.82a)}$$

B.6.3 Transfer functions for generalized root locus of controller gains

The denominator in Eqn. B.81 can be rearranged to isolate each individual controller gain. The following derivations show this process in two steps. The first step collects all the gain terms, the last step divides by the right hand side and simplifies according to the expressions represented by Eqn. B.78a and B.78b.

$K_{p\psi}$ **:**

$$K_{p\psi}\left(m_1s^4+m_4s^3+m_7s^2\right)+\left(\begin{array}{l}s^6+\left(K_{d\psi}m_1+K_{dY}m_2+m_3\right)s^5\\+\left(K_{d\psi}m_4+K_{pY}m_2+K_{dY}m_5+m_6\right)s^4\\+\left(K_{i\psi}m_1+K_{d\psi}m_7+K_{pY}m_5+K_{iY}m_2+K_{dY}m_8+m_9\right)s^3\\+\left(K_{i\psi}m_4+K_{pY}m_8+K_{iY}m_5+K_{dY}m_{10}\right)s^2\\+\left(K_{i\psi}m_7+K_{pY}m_{10}+K_{iY}m_8\right)s+K_{iY}m_{10}\end{array}\right) \quad \text{(B.83)}$$

$$TK_{p\psi} = \frac{K_{p\psi}\left(m_1 s^4 + m_4 s^3 + m_7 s^2\right)}{s^6 + DD_6 s^5 + \left(DD_5 - K_{p\psi} m_1\right)s^4 + \left(DD_4 - K_{p\psi} m_4\right)s^3 + \left(DD_3 - K_{p\psi} m_7\right)s^2 + DD_2 s + DD_1}$$

(B.84)

$K_{i\psi}$:

$$K_{i\psi}\left(m_1 s^3 + m_4 s^2 + m_7 s\right) + \left(\begin{array}{l} s^6 + \left(K_{d\psi} m_1 + K_{dY} m_2 + m_3\right)s^5 \\ \quad + \left(K_{p\psi} m_1 + K_{d\psi} m_4 + K_{pY} m_2 + K_{dY} m_5 + m_6\right)s^4 \\ \quad + \left(K_{p\psi} m_4 + K_{d\psi} m_7 + K_{pY} M_5 + K_{iY} m_2 K_{dY} m_8 + m_9\right)s^3 \\ \quad + \left(K_{p\psi} m_7 + K_{pY} m_8 + K_{iY} m_5 + K_{dY} m_{10}\right)s^2 \\ \quad + \left(K_{pY} m_{10} + K_{iY} m_8\right)s + K_{iY} m_{10} \end{array}\right)$$

(B.85)

$$TK_{i\psi} = \frac{K_{i\psi}\left(m_1 s^3 + m_4 s^2 + m_7 s\right)}{s^6 + DD_6 s^5 + DD_5 s^4 + \left(DD_4 - K_{i\psi} m_1\right)s^3 + \left(DD_3 - K_{i\psi} m_4\right)s^2 + \left(DD_2 - K_{i\psi} m_7\right)s + DD_1}$$

(B.86)

$K_{d\psi}$:

$$K_{d\psi}\left(m_1 s^5 + m_4 s^4 + m_7 s^3\right) + \left(\begin{array}{l} s^6 + \left(K_{dY} m_2 + m_3\right)s^5 \\ \quad + \left(K_{p\psi} m_1 + K_{pY} m_2 + K_{dY} m_5 + m_6\right)s^4 \\ \quad + \left(K_{p\psi} m_4 + K_{i\psi} m_1 + K_{pY} m_5 + K_{iY} m_2 + K_{dY} m_8 + m_9\right)s^3 \\ \quad + \left(K_{p\psi} m_7 + K_{i\psi} m_4 + K_{pY} m_8 + K_{iY} m_5 + K_{dY} m_{10}\right)s^2 \\ \quad + \left(K_{i\psi} m_7 + K_{pY} m_{10} + K_{iY} m_8\right)s + K_{iY} m_{10} \end{array}\right)$$

(B.87)

$$TK_{d\psi} = \frac{K_{d\psi}\left(m_1 s^5 + m_4 s^4 + m_7 s^3\right)}{s^6 + \left(DD_6 - K_{d\psi} m_1\right)s^5 + \left(DD_5 - K_{d\psi} m_4\right)s^4 + \left(DD_4 - K_{d\psi} m_7\right)s^3 + DD_3 s^2 + DD_2 s + DD_1}$$

(B.88)

K_{pY} :

$$K_{pY}\left(m_2 s^4 + m_5 s^3 + m_8 s^2 + m_{10} s\right) + \left(\begin{array}{l} s^6 + \left(K_{d\psi} m_1 + K_{dY} m_2 + m_3\right) s^5 \\ \quad + \left(K_{p\psi} m_1 + K_{d\psi} m_4 + K_{dY} m_5 + m_6\right) s^4 \\ \quad + \left(K_{p\psi} m_4 + K_{i\psi} m_1 + K_{d\psi} m_7 + K_{iY} m_2 + K_{dY} m_8 + m_9\right) s^3 \\ \quad + \left(K_{p\psi} m_7 + K_{i\psi} m_4 + K_{iY} m_5 + K_{dY} m_{10}\right) s^2 \\ \quad + \left(K_{i\psi} m_7 + K_{io} m_8\right) s + K_{iY} m_{10} \end{array} \right)$$

(B.89)

$$TK_{pY} = \frac{K_{pY}\left(m_2 s^4 + m_5 s^3 + m_8 s^2 + m_{10} s\right)}{s^6 + DD_6 s^5 + \left(DD_5 - K_{pY} m_2\right) s^4 + \left(DD_4 - K_{pY} m_5\right) s^3 + \left(DD_5 - K_{pY} m_8\right) s^2 + \left(DD_2 - K_{pY} m_{10}\right) s + DD_1}$$

(B.90)

K_{iY} :

$$K_{iY}\left(m_2 s^3 + m_5 s^2 + m_8 s + m_{10}\right) + \left(\begin{array}{l} s^6 + \left(K_{d\psi} m_1 + K_{dY} m_2 + m_3\right) s^5 \\ \quad + \left(K_{p\psi} m_1 + K_{d\psi} m_4 + K_{pY} m_2 + K_{dY} m_5 + m_6\right) s^4 \\ \quad + \left(K_{p\psi} m_4 + K_{i\psi} m_1 + K_{d\psi} m_7 + K_{pY} m_5 + K_{dY} m_8 + m_9\right) s^3 \\ \quad + \left(K_{p\psi} m_7 + K_{i\psi} m_4 + K_{pY} m_8 + K_{dY} m_{10}\right) s^2 \\ \quad + \left(K_{i\psi} m_7 + K_{pY} m_{10}\right) s \end{array} \right)$$

(B.91)

$$TK_{iY} = \frac{K_{iY}\left(m_2 s^3 + m_5 s^2 + m_8 s + m_{10}\right)}{s^6 + DD_6 s^5 + DD_5 s^4 + \left(DD_4 - K_{iY} m_2\right) s^3 + \left(DD_3 - K_{iY} m_5\right) s^2 + \left(DD_2 - K_{iY} m_8\right) s}$$

(B.92)

K_{dY} :

$$K_{dY}\left(m_2 s^5 + m_5 s^4 + m_8 s^3 + m_{10} s^2\right) + \left(\begin{array}{l} s^6 + \left(K_{d\psi} m_1 + m_3\right) s^5 \\ \quad + \left(K_{p\psi} m_1 + K_{d\psi} m_4 + K_{pY} m_2 + m_6\right) s^4 \\ \quad + \left(K_{p\psi} m_4 + K_{i\psi} m_1 + K_{d\psi} m_7 + K_{pY} m_5 + K_{iY} m_2 + m_9\right) s^3 \\ \quad + \left(K_{p\psi} m_7 + K_{i\psi} m_4 + K_{pY} m_8 + K_{iY} m_5\right) s^2 \\ \quad + \left(K_{i\psi} m_7 + K_{pY} m_{10} + K_{iY} m_8\right) s + K_{iY} m_{10} \end{array} \right) \tag{B.93}$$

$$TK_{dY} = \frac{K_{dY}\left(m_2 s^5 + m_5 s^4 + m_8 s^3 + m_{10} s^2\right)}{s^6 + \left(DD_6 - K_{dY} m_2\right) s^5 + \left(DD_5 - K_{dY} m_5\right) s^4 + \left(DD_4 - K_{dY} m_8\right) s^3 + \left(DD_3 - K_{dY} m_{10}\right) s^2 + DD_2 s + DD_1} \tag{B.94}$$

B.6.4 Transfer function for initial lateral error

The denominator of the initial lateral error transfer function is the same derivation as shown in Section B.6.8. The numerator is much simpler to solve.

$$G_1 G_4 = \frac{g b_1}{b_4\left(s + a_1\right)} \cdot \frac{1}{s} = \frac{g b_1}{b_4 s\left(s + a_1\right)} \tag{B.95}$$

Combine the numerator and denominator to get

$$= \frac{s\left(s + a_1\right)\left(s^4 + D_3 s^3 + D_2 s^2 + D_1 s + D_0\right)}{\left(\begin{array}{l} s^2\left(s + a_1\right)\left(s^4 + D_3 s^3 + D_2 s^2 + D_1 s + D_0\right) - a_4 x\left(K_{dY} s^2 + K_{pY} s + K_{iY}\right)\left(s^4 + D_3 s^3 + D_2 s^2 + D_1 s + D_0\right) \\ \quad - x\left(N_2 s^2 + N_1 s + N_0\right)\left(b_4 s + a_1 b_4 - a_4 b_1\right)\left(K_{dY} s^2 + K_{pY} s + K_{iY}\right) \end{array} \right)}$$

The denominator is the same as Eqn. B.76, which reduces to Eqn. B.78, so multiply by 1/s.

$$= \frac{\left(s + a_1\right)\left(s^4 + D_3 s^3 + D_2 s^2 + D_1 s + D_0\right)}{s^6 + DD_6 s^5 + DD_5 s^4 + DD_4 s^3 + DD_3 s^2 + DD_2 s + DD_1} \tag{B.96}$$

Expand the numerator and collect like *s* terms.

$$= \frac{s^5 + (D_3 + a_1)s^4 + (D_2 + a_1 D_3)s^3 + (D_1 + a_1 D_2)s^2 + (D_0 + a_1 D_1)s + a_1 D_0}{s^6 + DD_6 s^5 + DD_5 s^4 + DD_4 s^3 + DD_3 s^2 + DD_2 s + DD_1} \tag{B.97}$$

Note that because the denominators are the same as before, the generalized root locus for the controller gains will also be the same. Next, substitute values for the constants in the numerator using the relationships given in Eqns. B.66a through B.67b. The process is shown in three steps.

$$\begin{aligned}
&= s^5 + (d_1 - K_{d\psi} d_2 + a_1)s^4 \\
&\quad + (d_3 + K_{d\psi} d_4 - K_{p\psi} d_2 + a_1 d_1 - a_1 K_{d\psi} d_2)s^3 \\
&\quad + (K_{p\psi} d_4 - K_{i\psi} d_2 + a_1 d_3 + a_1 K_{p\psi} d_4 - a_1 K_{p\psi} d_2)s^2 \\
&\quad + (K_{i\psi} d_4 + a_1 K_{p\psi} d_4 - a_1 K_{i\psi} d_2)s \\
&\quad + a_1 K_{i\psi} d_4
\end{aligned}$$

Additional substitution yields,

$$\begin{aligned}
&= s^5 + (a_1 + b_2 - K_{d\psi} x b_4 + a_1)s^4 \\
&\quad + (a_1 b_2 - a_2 b_1 + K_{d\psi} x(a_4 b_1 - a_1 b_4) - K_{p\psi} x b_4 + a_1^2 + a_1 b_2 - a_1 K_{d\psi} x b_4)s^3 \\
&\quad + (K_{p\psi} x(a_4 b_1 - a_1 b_4) - K_{i\psi} x b_4 + a_1^2 b_2 + a_1 a_2 b_1 + a_1 K_{p\psi} x(a_4 b_1 - a_1 b_4) - a_1 K_{p\psi} x b_4)s^2 \\
&\quad + (K_{i\psi} x(a_4 b_1 - a_1 b_4) + a_1 K_{p\psi} x(a_4 b_1 - a_1 b_4) - a_1 K_{i\psi} x b_4)s \\
&\quad + a_1 K_{i\psi} x(a_4 b_1 - a_1 b_4)
\end{aligned}$$

Collect like terms and simplify.

$$\begin{aligned}
&= s^5 + (K_{d\psi}(-x b_4) + 2a_1 + b_2)s^4 \\
&\quad + (K_{d\psi}(x a_4 b_1 - 2x a_1 b_4) + K_{p\psi}(-x b_4) + 2a_1 b_2 - a_2 b_1 + a_1^2)s^3 \\
&\quad + (K_{d\psi}(x a_1 a_4 b_1 - x a_1^2 b_4) + K_{p\psi}(x a_4 b_1 - 2x a_1 b_4) + K_{i\psi}(-x b_4) + a_1^2 b_2 - a_1 a_2 b_1)s^2 \\
&\quad + (K_{p\psi}(x a_1 a_4 b_1 - x a_1^2 b_4) + K_{i\psi}(x a_4 b_1 - 2x a_1 b_4))s \\
&\quad + K_{i\psi}(x a_1 a_4 b_1 - x a_1^2 b_4)
\end{aligned} \tag{B.98}$$

Using the variables $m_{1\text{-}10}$ as shown in Eqn. B.78b, the transfer function becomes

$$= \frac{s^5 + \left(K_{d\psi} m_1 + m_3\right)s^4 + \left(K_{d\psi} m_4 + K_{p\psi} m_1 + m_6\right)s^3 + \left(K_{d\psi} m_7 + K_{p\psi} m_4 + K_{i\psi} m_1 + m_9\right)s^2 + \left(K_{p\psi} m_7 + K_{i\psi} m_4\right)s + K_{i\psi} m_7}{s^6 + DD_6 s^5 + DD_5 s^4 + DD_4 s^3 + DD_3 s^2 + DD_2 s + DD_1}$$

$$\frac{Y}{Y_0} = \frac{s^5 + N_4 s^4 + N_3 s^3 + N_2 s^2 + N_1 s + N_0}{s^6 + DD_6 s^5 + DD_5 s^4 + DD_4 s^3 + DD_3 s^2 + DD_2 s + DD_1} \tag{B.99}$$

APPENDIX C. MATLAB CODE

This Appendix contains the computer code generated in MATLAB® (The MathWorks, Inc., Natick, MA). The entire code is included with the exception of the plotting routines, which makes the code much longer and detracts from the main objective of the code.

C.1 Linear simulation code: *combineSimBasicLinear.m*

```
% Entire simulation code
% Scott Evans

clear all
clc

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                          Vehicle & Speed Parameters                          %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

L = 11.5;                       % Wheel Base (ft)
g = 32.2;                       % gravity (ft/sec^2)
W = 34000;                      % weight (lbs)
M = W / g;                      % mass (slugs)

Wf_ratio = 0.80;                % Weight distribution (front)
Wr_ratio = 1 - Wf_ratio;        % Weight distribution (rear)
a = L * (1 - Wf_ratio);         % distance to CG from front (ft)
b = L * (1 - Wr_ratio);         % distance to CG from rear (ft)
Wf = W * Wf_ratio;              % weight distribution on front (lbs)
WR = W * Wr_ratio;              % weight distribution on back (lbs)
Izz = (M/4) * (a + b)^2;        % polar moment of inertia (slugs * ft^2)

u_mph = 10;                     % Speed (mph)
u = u_mph * 5280/3600;          % Speed (ft/sec)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                      Tire and friction (mu) parameters                       %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

cornerCoeffFront = 0.06;                % Cornering coefficient (Calphaf/Wf)
cornerCoeffRear = 0.06;                 % Cornering coefficient (CalphaR/WR)
Calphaf_deg = cornerCoeffFront * Wf;      % lbs/deg
CalphaR_deg = cornerCoeffRear * WR;       % lbs/deg

% Calphaf_deg = 1632;             % lbs/deg
% CalphaR_deg = 408;              % lbs/deg
```

```
CalphaR = CalphaR_deg * 180/pi * 2;      % lbs/rad (*2 -> 2 tires)
Calphaf = Calphaf_deg * 180/pi * 2;      % lbs/rad (*2 -> 2 tires)

K = Wf/Calphaf - WR/CalphaR;             % Understeer gradient
uCrit = sqrt(L*g/abs(K)) * 3600/5280;    % Critical speed (mph)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                    Steer input and contoller parameters                        %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

steerType = 1;                % 0: front steer, 1: rear steer
steerAngle = 0;               % Input angle (positive assumes right turn)

if steerType == 0
    deltaf_deg = steerAngle;     % Front steer (degs)
    deltaR_deg = 0;              % Rear steer (degs)
else
    deltaf_deg = 0;              % Front steer (degs)
    deltaR_deg = -steerAngle;    % Rear steer (degs)
end

deltaf0 = deltaf_deg * pi/180;  % Front steer (rads), 0 on end
distinguishes from deltaf in eom
deltaR0 = deltaR_deg * pi/180;  % Rear steer (rads), 0 on end
distinguishes from deltaR in eom

% Set the controller gain constants
KpOff = 3.7;
KdOff = 1.3;
KiOff = 0.05;

KpHead = 74.0;
KdHead = 0.0;
KiHead = 0.0;

controlOn = 1;  % Enter 1 to use controller

% Introduce a slope (side force)
thetaDeg = 5;                % Enter slope in degrees
theta = thetaDeg * pi/180;   % Slope in radians
Fy_ext = W*theta;            % Side force created by slope

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                       Time increment and time span                             %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

dt = 0.01;      % Time increment
t0 = 0.0;       % Time to start steering inputs - gives visual effect on
plots
tFinal = 10.0;          % End time

if controlOn
```

```
    tControl = t0;      % Time at which controller takes over.
else
    tControl = 1000;
end
tspan = 0:dt:tFinal;    % seconds

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                       Coefficients for EOM                                    %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

a1 = (Calphaf + CalphaR) / (M*u);
a2 = (M*u^2 + Calphaf*a - CalphaR*b) / (M*u);
a3 = Calphaf / M;
a4 = CalphaR / M;
b1 = (Calphaf*a - CalphaR*b) / (Izz*u);
b2 = (Calphaf*a^2 + CalphaR*b^2) / (Izz*u);
b3 = (Calphaf * a) / Izz;
b4 = -(CalphaR * b) / Izz;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                       ode45 setup and calculations                            %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

options = [];   % No options used in ode45

xinit = [0 0 0 0 10 0 0];  % Initial conditions

[t,x] = ode45('dynamicsBasicLinear', tspan, xinit, options, u, a, b, M,
Izz, W, deltaf0, ...
    deltaR0, Calphaf, CalphaR, t0, dt, tControl, Fy_ext, steerType, KpOff,
KdOff, ...
    KpHead, KdHead, KiOff, KiHead);

% Recaluclate variables since ode45 gets rid of some
i = 1;  % Initialize counter for each output variable

for time = tspan
    [dx(i,:), Fyf(i,:), FyR(i,:), alphaf(i,:), alphaR(i,:), beta(i,:), ...
     deltaf(i,:), deltaR(i,:), offError(i,:), headError(i,:), FyExt(i,:),
oErrDot(i,:)] = ...
        dynamicsBasicLinear(time, x(i,:), options, u, a, b, M, Izz, W,
deltaf0, ...
        deltaR0, Calphaf, CalphaR, t0, dt, tControl, Fy_ext, steerType,
KpOff, KdOff, ...
        KpHead, KdHead, KiOff, KiHead);

    i = i + 1;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%       Various calculations - Acceleration, angles, distances                  %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
ay = (dx(:,2) + u * x(:,1)) / g;    % Lateral acceleration: ay = vdot + u*r

% Checking steady state r and v according to my calculations
K = Wf/Calphaf - WR/CalphaR;

% vss parts
dfPart = (b/L) - (WR/CalphaR)*(u^2/(L*g));
dRPart = (a/L) + (Wf/Calphaf)*(u^2/(L*g));
thPart = (Wf/Calphaf)*(b/L) + (WR/CalphaR)*(a/L);

numv = u*(dfPart*deltaf + dRPart*deltaR + thPart*theta);

% rss parts
numr = (u/L)*((deltaf - deltaR) + K*theta);

den = 1 + K*u^2/(L*g);

% Steady state r, v, beta
rss = (numr / den) * 180/pi;
vss = numv / den;
beta_ss = (vss / u) * 180/pi;

% Steady state steer angle: df - dR = (L/R) + K(ay - sin(theta))
R = u ./ x(:,1);

for n=1:length(R)
    dss(n) = ((L/R(n)) + K*(ay(n) - theta)) * 180/pi;
end

if steerType == 1
    dss = -dss;
end

% betaSS and deltaRSS constant values
betaSS = (Wf/Calphaf)*theta;
betaSS = betaSS * 180/pi;

deltaRSS = K * theta;
deltaRSS = deltaRSS * 180/pi;

% Convert radians to degrees
r = x(:,1) * 180/pi;
beta = beta * 180/pi;
alphaf = alphaf * 180/pi;
alphaR = alphaR * 180/pi;
deltaf = deltaf * 180/pi;
deltaR = deltaR * 180/pi;
psi = x(:,3) * 180/pi;
headError = headError * 180/pi;

v = x(:,2);
X = x(:,4);
Y = x(:,5);
```

C.2 Linear Simulation dynamics code: *dynamicsBasicLinear.m*

```
% This function is the dynamics code for my thesis work.  It is a bicycle
% model, which will be built upon later.

function [dx, Fyf, FyR, alphaf, alphaR, beta, deltaf, deltaR, offError, ...
    headError, FyExt, oErrDot] ...
    = dynamicsBasicLinear(time, x, options, u, a, b, M, Izz, W,
deltaf0, ...
    deltaR0, Calphaf, CalphaR, t0, dt, tControl, Fy_ext, steerType, ...
    KpOff, KdOff, KpHead, KdHead, KiOff, KiHead)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                          State variables                                    %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%    x(1) = r,        x(4) = X,           x(7) = psi / s
%    x(2) = v,        x(5) = Y,
%    x(3) = psi,      x(6) = Y / s,
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                     Variable setup and calculations                         %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Calculate offTrack and heading error
offError = x(5);     % offTrack is simply Y for now
headError = x(3);    % heading is psi

% Calculate the integral of offTrack and heading
offInt = x(6);
headInt = x(7);

% Calculate the derivative terms
% Setup this way to show why transfer function doesn't match
% Change oErrDot to x(5)/dt to make happen.
if time < 0.01
    oErrDot = u * x(3) + x(2);  % Same as dx(5)
    hErrDot = x(1);             % Same as dx(3)
else
    oErrDot = u * x(3) + x(2);  % Same as dx(5)
    hErrDot = x(1);             % Same as dx(3)
end

% Setup steer angle to start after time, t0, for plotting purposes
if time < t0
    deltaf = 0;
    deltaR = 0;
    FyExt = 0;
elseif (time >= t0 & time < tControl)
    deltaf = deltaf0;
    deltaR = deltaR0;
    FyExt = Fy_ext;
else
```

```
    % Use controller function to calculate steer: 0 -> front, 1 -> rear
    if steerType == 0    % Front steer
        deltaf = controllerLinear(offError, oErrDot, offInt, headError,
hErrDot, headInt, KpOff, KdOff, KiOff, KpHead, KdHead, KiHead, steerType);
        deltaR = deltaR0;
    else    % Rear steer
        deltaf = deltaf0;
        deltaR = controllerLinear(offError, oErrDot, offInt, headError,
hErrDot, headInt, KpOff, KdOff, KiOff, KpHead, KdHead, KiHead, steerType);
    end
    FyExt = Fy_ext;
end

% Calculate slip angles and sideslip angle
alphaf = (x(2) + a * x(1)) / u - deltaf;
alphaR = (x(2) - b * x(1)) / u - deltaR;
beta = x(2) / u;

% Determine lateral forces
Fyf = -Calphaf * alphaf;
FyR = -CalphaR * alphaR;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                          State equations                                    %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   x(1) = r,        x(4) = X,           x(7) = psi / s
%   x(2) = v,        x(5) = Y,
%   x(3) = psi,      x(6) = Y / s,
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

dx(1) = (Fyf * a - FyR * b) / Izz;
dx(2) = (Fyf + FyR + FyExt) / M - u * x(1);
dx(3) = x(1);
dx(4) = u - x(2) * x(3);
dx(5) = u * x(3) + x(2);
dx(6) = x(5);
dx(7) = x(3);

dx = dx';   % Transpose the vector to use in ode45
```

C.3 Linear Controller PID controller code: *controllerLinear.m*

```
% This is the controller function that will be used to control the
steering
% input.  Pass in the offTrack error and the heading error, and output the
% steer angle needed to correct the vehicle.

function steerAngle = controllerLinear(offError, oErrDot, offInt,
headError, hErrDot, headInt, KpOff, KdOff, KiOff, KpHead, KdHead, KiHead,
type);

% Different depending on front (0) or rear steer (1)
switch (type)
    case 0
        steerAngle = -(KpOff*offError + KpHead*headError + KdOff*oErrDot +
KdHead*hErrDot + KiOff*offInt + KiHead*headInt);
    case 1
        steerAngle = (KpOff*offError + KpHead*headError + KdOff*oErrDot +
KdHead*hErrDot + KiOff*offInt + KiHead*headInt);
end

steerAngle = steerAngle * pi/180;
```

C.4 Transfer function model code: *transferFunctionFinal.m*

```
% Final transfer function code
% Scott Evans 3/20/06

clc
clear all

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                          Vehicle & Speed Parameters                         %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

L = 11.5;                       % Wheel Base (ft)
g = 32.2;                       % gravity (ft/sec^2)
W = 34000;                      % weight (lbs)
M = W / g;                      % mass (slugs)

Wf_ratio = 0.80;                % Weight distribution (front)
Wr_ratio = 1 - Wf_ratio;        % Weight distribution (rear)
a = L * (1 - Wf_ratio);         % distance to CG from front (ft)
b = L * (1 - Wr_ratio);         % distance to CG from rear (ft)
Wf = W * Wf_ratio;              % weight distribution on front (lbs)
WR = W * Wr_ratio;              % weight distribution on back (lbs)
Izz = (M/4) * (a + b)^2;        % polar moment of inertia (slugs * ft^2)

u_mph = 10;                     % Speed (mph)
u = u_mph * 5280/3600;          % Speed (ft/sec)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                      Tire and friction (mu) parameters                      %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

cornerCoeffFront = 0.06;                % Cornering coefficient (Calphaf/Wf)
cornerCoeffRear = 0.06;                 % Cornering coefficient (CalphaR/WR)
Calphaf_deg = cornerCoeffFront * Wf;    % lbs/deg
CalphaR_deg = cornerCoeffRear * WR;     % lbs/deg

% Calphaf_deg = 1195;           % lbs/deg
% CalphaR_deg = 1195;           % lbs/deg

CalphaR = CalphaR_deg * 180/pi * 2;     % lbs/rad (*2 -> 2 tires)
Calphaf = Calphaf_deg * 180/pi * 2;     % lbs/rad (*2 -> 2 tires)

K = Wf/Calphaf - WR/CalphaR;            % Understeer gradient
uCrit = sqrt(L*g/abs(K)) * 3600/5280;   % Critical speed (mph)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                    Steer input and contoller parameters                     %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

steerType = 1;                  % 0: front steer, 1: rear steer
steerAngle = 0;                 % Input angle (positive assumes right turn)
```

```
if steerType == 0
    deltaf_deg = steerAngle;     % Front steer (degs)
    deltaR_deg = 0;              % Rear steer (degs)
else
    deltaf_deg = 0;              % Front steer (degs)
    deltaR_deg = -steerAngle;    % Rear steer (degs)
end

deltaf0 = deltaf_deg * pi/180;  % Front steer (rads), 0 on end
distinguishes from deltaf in eom
deltaR0 = deltaR_deg * pi/180;  % Rear steer (rads), 0 on end
distinguishes from deltaR in eom

% Set the controller gain constants
KpOff = 3.7;
KdOff = 1.3;
KiOff = 0.05;

KpHead = 74.0;
KdHead = 0.0;
KiHead = 0.0;

t = 0.0:0.01:10.0;     % Time to run response

% Introduce a slope (side force)
thetaDeg = 5;                % Enter slope in degrees
theta = thetaDeg * pi/180;   % Slope in radians

% State initial condition for Y
Y0 = 10;    % Initial offTrack error (ft)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                              Coefficients for EOM                        %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
a1 = (Calphaf + CalphaR) / (M*u);
a2 = (M*u^2 + Calphaf*a - CalphaR*b) / (M*u);
a3 = Calphaf / M;
a4 = CalphaR / M;
b1 = (Calphaf*a - CalphaR*b) / (Izz*u);
b2 = (Calphaf*a^2 + CalphaR*b^2) / (Izz*u);
b3 = (Calphaf * a) / Izz;
b4 = -(CalphaR * b) / Izz;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                    Hand calculations (refer to paper)                    %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Checking the DD values after substitutions are made
x = pi/180;

m1 = -x*b4;
m2 = -x*a4;
```

```
m3 = 2*a1 + b2;
m4 = x*a4*b1 - 2*x*a1*b4;
m5 = -x*a1*a4 - x*a4*b2 - x*b4*u + x*a2*b4;
m6 = 2*a1*b2 - a2*b1 + a1^2;
m7 = x*a1*a4*b1 - x*a1^2*b4;
m8 = -x*a1*a4*b2 + x*a4*b1*u - 2*x*a1*b4*u + x*a1*a2*b4;
m9 = a1^2*b2 - a1*a2*b1;
m10 = x*a1*a4*b1*u - x*a1^2*b4*u;

DD6 = KdHead*m1 + KdOff*m2 + m3;
DD5 = KdHead*m4 + KpHead*m1 + KpOff*m2 + KdOff*m5 + m6;
DD4 = KpHead*m4 + KiHead*m1 + KdHead*m7 + KiOff*m2 + KdOff*m8
+ KpOff*m5 + m9;
DD3 = KiHead*m4 + KpHead*m7 + KdOff*m10 + KpOff*m8 + KiOff*m5;
DD2 = KiHead*m7 + KpOff*m10 + KiOff*m8;
DD1 = KiOff*m10;

% Checking numT values after sustitution
z1 = a1 + b2;
z2 = -x*a1*b4;
z3 = a1*b2 - b1*u;
z4 = -b1*u*a1;

NN3 = KdHead*m1 + z1;
NN2 = KdHead*z2 + KpHead*m1 + z3;
NN1 = KpHead*z2 + KiHead*m1 + z4;
NN0 = KiHead*z2;

% Transfer function Y / theta
numTh = g*[1 NN3 NN2 NN1 NN0];
denTh = [1 DD6 DD5 DD4 DD3 DD2 DD1];

Th = tf(numTh, denTh);
rootsDen = roots(denTh);
[YTh, tTh] = step(theta*Th, t);

% Transfer function Y / Y0
N4 = KdHead*m1 + m3;
N3 = KdHead*m4 + KpHead*m1 + m6;
N2 = KdHead*m7 + KpHead*m4 + KiHead*m1 + m9;
N1 = KpHead*m7 + KiHead*m4;
N0 = KiHead*m7;

numTh2 = [1 N4 N3 N2 N1 N0];
denTh2 = [1 DD6 DD5 DD4 DD3 DD2 DD1];

Th2 = tf(numTh2, denTh2);
rootsDen2 = roots(denTh2);
[YTh2, tTh2] = impulse(Y0*Th2, t);


% Combination transfer function result
YThAll = YTh + YTh2;
```

```
% Generalized Root Locus
TKpHead = tf([m1 m4 m7 0 0], ...
    [1 DD6 (DD5-KpHead*m1) (DD4-KpHead*m4) (DD3-KpHead*m7) DD2 DD1]);
TKiHead = tf([m1 m4 m7 0], ...
    [1 DD6 DD5 (DD4-KiHead*m1) (DD3-KiHead*m4) (DD2-KiHead*m7) DD1]);
TKdHead = tf([m1 m4 m7 0 0 0], ...
    [1 (DD6-KdHead*m1) (DD5-KdHead*m4) (DD4-KdHead*m7) DD3 DD2 DD1]);

TKpOff = tf([m2 m5 m8 m10 0], ...
    [1 DD6 (DD5-KpOff*m2) (DD4-KpOff*m5) (DD3-KpOff*m8) (DD2-KpOff*m10)
DD1]);
TKiOff = tf([m2 m5 m8 m10], ...
    [1 DD6 DD5 (DD4-KiOff*m2) (DD3-KiOff*m5) (DD2-KiOff*m8) 0]);
TKdOff = tf([m2 m5 m8 m10 0 0], ...
    [1 (DD6-KdOff*m2) (DD5-KdOff*m5) (DD4-KdOff*m8) (DD3-KdOff*m10) DD2
DD1]);
```

C.5 Transfer function codes for vehicle parameter changes

The main difference between the code presented in Section C.4 and the code for the vehicle parameter changes was the entire code was put in a *for* loop to run through the desired parameter range specified outside the loop. Some variables were indexed in order to create an array containing the results for the whole range of parameter values. Because very little changes to the code were made, it is unnecessary to show the code for each individual parameter change. The code for the varying the weight of the combine is shown below as it contained slightly more modifications. In this example, the plotting routine is included and the basic setup for all parameter change modifications can be seen.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                          Vehicle & Speed Parameters                          %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

L = 11.5;                       % Wheel Base (ft)
g = 32.2;                       % gravity (ft/sec^2)

WBase = 34000;                          % Base unloaded weight of vehicle
WUnitCrop = 60;                         % lbs/bushel;
tankCap = 300;                          % Grain tank capacity (bushels)
tankRange = 0.0:0.25:1.00;              % Range of tank capacity to cycle (%)

WCrop = WUnitCrop*tankRange*tankCap; % Weight of crop for range of tank
weightRange = WBase + WCrop;            % Range of weights to look at (lbs)

CGShift = 0.00:0.01:0.04;           % CG location shift as grain enters (%)

m = 1;                          % Counter
for n = weightRange             % Cycling through CG positions

W = n;                          % weight (lbs)
M = W / g;                      % mass (slugs)

% Wf_ratio = 0.80;
% Wf_ratio = 0.80 + CGShift(m);   % Weight distribution (front)
Wf_ratio = 0.80 - CGShift(m);   % Weight distribution (front)
Wr_ratio = 1 - Wf_ratio;        % Weight distribution (rear)
a = L * (1 - Wf_ratio);         % distance to CG from front (ft)
b = L * (1 - Wr_ratio);         % distance to CG from rear (ft)
```

```
Wf = W * Wf_ratio;                  % weight distribution on front (lbs)
WR = W * Wr_ratio;                  % weight distribution on back (lbs)
Izz = (M/4) * (a + b)^2;            % polar moment of inertia (slugs * ft^2)

u_mph = 10;                     % Speed (mph)
u = u_mph * 5280/3600;          % Speed (ft/sec)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                          Tire and friction (mu) parameters                    %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

cornerCoeffFront = 0.06;                % Cornering coefficient (Calphaf/Wf)
cornerCoeffRear = 0.06;                 % Cornering coefficient (CalphaR/WR)
Calphaf_deg = cornerCoeffFront * Wf;    % lbs/deg
CalphaR_deg = cornerCoeffRear * WR;     % lbs/deg

Calphaf_deg = 1632;          % lbs/deg
CalphaR_deg = 408;           % lbs/deg

CalphaR = CalphaR_deg * 180/pi * 2;     % lbs/rad (*2 -> 2 tires)
Calphaf = Calphaf_deg * 180/pi * 2;     % lbs/rad (*2 -> 2 tires)

K = Wf/Calphaf - WR/CalphaR;    % Understeer gradient
uCrit = sqrt(L*g./abs(K));       % critical speed

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                        Steer input and contoller parameters                   %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

steerType = 1;              % 0: front steer, 1: rear steer
steerAngle = 0;             % Input angle (positive assumes right turn)

if steerType == 0
    deltaf_deg = steerAngle;    % Front steer (degs)
    deltaR_deg = 0;             % Rear steer (degs)
else
    deltaf_deg = 0;             % Front steer (degs)
    deltaR_deg = -steerAngle;    % Rear steer (degs)
end

deltaf0 = deltaf_deg * pi/180;  % Front steer (rads), 0 on end
distinguishes from deltaf in eom
deltaR0 = deltaR_deg * pi/180;  % Rear steer (rads), 0 on end
distinguishes from deltaR in eom

% Set the controller gain constants
KpOff = 3.7;
KdOff = 1.3;
KiOff = 0.05;

KpHead = 74.0;
KdHead = 0.0;
KiHead = 0.0;
```

```
t = 0.0:0.01:10;     % Time to run the response for

% Introduce a slope (side force)
thetaDeg = 5;                 % Enter slope in degrees
theta = thetaDeg * pi/180;    % Slope in radians
Fy_ext = W*sin(theta);        % Side force created by slope

% State initial condition for Y (ft)
Y0 = 10;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                     Turn on and off plotting routines                          %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

plots = 1;              % Enter 1 if I want to plot
stepResp = 0;           % Enter 1 to see step to slope only
poleZero = 0;           % Enter 1 to see pole zero map
rootLocus = 0;          % Plot generalized root locus
impResp = 0;            % Plot impulse response with IC: Y0
stepImpResp = 0;        % Plot step and impulse on subplots
TotalResp = 1;          % Plot overall response
sysRoots = 0;           % Plot the system roots

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                              Coefficients for EOM                              %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

a1 = (Calphaf + CalphaR) / (M*u);
a2 = (M*u^2 + Calphaf*a - CalphaR*b) / (M*u);
a3 = Calphaf / M;
a4 = CalphaR / M;
b1 = (Calphaf*a - CalphaR*b) / (Izz*u);
b2 = (Calphaf*a^2 + CalphaR*b^2) / (Izz*u);
b3 = (Calphaf * a) / Izz;
b4 = -(CalphaR * b) / Izz;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                  Hand calculations (refer to paper)                            %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Checking the DD values after substitutions are made
x = pi/180;

m1 = -x*b4;
m2 = -x*a4;
m3 = 2*a1 + b2;
m4 = x*a4*b1 - 2*x*a1*b4;
m5 = -x*a1*a4 - x*a4*b2 - x*b4*u + x*a2*b4;
m6 = 2*a1*b2 - a2*b1 + a1^2;
m7 = x*a1*a4*b1 - x*a1^2*b4;
m8 = -x*a1*a4*b2 + x*a4*b1*u - 2*x*a1*b4*u + x*a1*a2*b4;
m9 = a1^2*b2 - a1*a2*b1;
```

```
m10 = x*a1*a4*b1*u - x*a1^2*b4*u;

DD6 = KdHead*m1 + KdOff*m2 + m3;
DD5 = KdHead*m4 + KpHead*m1 + KpOff*m2 + KdOff*m5 + m6;
DD4 = KpHead*m4 + KiHead*m1 + KdHead*m7 + KiOff*m2 + KdOff*m8
+ KpOff*m5 + m9;
DD3 = KiHead*m4 + KpHead*m7 + KdOff*m10 + KpOff*m8 + KiOff*m5;
DD2 = KiHead*m7 + KpOff*m10 + KiOff*m8;
DD1 = KiOff*m10;

% Checking numT values after sustitution
z1 = a1 + b2;
z2 = -x*a1*b4;
z3 = a1*b2 - b1*u;
z4 = -b1*u*a1;

NN3 = KdHead*m1 + z1;
NN2 = KdHead*z2 + KpHead*m1 + z3;
NN1 = KpHead*z2 + KiHead*m1 + z4;
NN0 = KiHead*z2;

% Transfer function Y / theta
numTh = g*[1 NN3 NN2 NN1 NN0];
denTh = [1 DD6 DD5 DD4 DD3 DD2 DD1];

Th = tf(numTh, denTh);
rootsDen(:,m) = roots(denTh);
[YTh(:,m), tTh(:,m)] = step(theta*Th, t);

% Transfer function Y / Y0
N4 = KdHead*m1 + m3;
N3 = KdHead*m4 + KpHead*m1 + m6;
N2 = KdHead*m7 + KpHead*m4 + KiHead*m1 + m9;
N1 = KpHead*m7 + KiHead*m4;
N0 = KiHead*m7;

numTh2 = [1 N4 N3 N2 N1 N0];
denTh2 = [1 DD6 DD5 DD4 DD3 DD2 DD1];

Th2 = tf(numTh2, denTh2);
rootsDen2(:,m) = roots(denTh2);
[YTh2(:,m), tTh2(:,m)] = impulse(Y0*Th2, t);

% Combination transfer function result
YThAll(:,m) = YTh(:,m) + YTh2(:,m);

% Generalized Root Locus
TKpHead = tf([m1 m4 m7 0 0], ...
    [1 DD6 (DD5-KpHead*m1) (DD4-KpHead*m4) (DD3-KpHead*m7) DD2 DD1]);
TKiHead = tf([m1 m4 m7 0], ...
    [1 DD6 DD5 (DD4-KiHead*m1) (DD3-KiHead*m4) (DD2-KiHead*m7) DD1]);
TKdHead = tf([m1 m4 m7 0 0 0], ...
    [1 (DD6-KdHead*m1) (DD5-KdHead*m4) (DD4-KdHead*m7) DD3 DD2 DD1]);
```

```
TKpOff = tf([m2 m5 m8 m10 0], ...
    [1 DD6 (DD5-KpOff*m2) (DD4-KpOff*m5) (DD3-KpOff*m8) (DD2-KpOff*m10)
DD1]);
TKiOff = tf([m2 m5 m8 m10], ...
    [1 DD6 DD5 (DD4-KiOff*m2) (DD3-KiOff*m5) (DD2-KiOff*m8) 0]);
TKdOff = tf([m2 m5 m8 m10 0 0], ...
    [1 (DD6-KdOff*m2) (DD5-KdOff*m5) (DD4-KdOff*m8) (DD3-KdOff*m10) DD2
DD1]);


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                    Plots                                     %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if plots

if poleZero

figure(1)
pzmap(theta*Th)
legend([num2str(tankRange(m)*100), ' %'], 'location', 'northwest')

end

if TotalResp

figure(2)
plot(tTh, YThAll, [min(t) max(t)], [0 0], 'k:')
title('System response: Grain tank from empty to full (%)')
xlabel('Time (sec)')
ylabel('OffTrack Error')
legend('0%', '25%', '50%', '75%', '100%', 'location', 'best')
% ylim([-2 16])
end

if rootLocus

figure(3)
subplot(311)
rlocus(TKpHead)
title('Generalized Root Locus for KpHead')
legend([num2str(tankRange(m)*100), ' %'], 'location', 'northwest')

subplot(312)
rlocus(TKiHead)
title('Generalized Root Locus for KiHead')
legend([num2str(tankRange(m)*100), ' %'], 'location', 'northwest')

subplot(313)
rlocus(TKdHead)
title('Generalized Root Locus for KdHead')
legend([num2str(tankRange(m)*100), ' %'], 'location', 'northwest')
```

```
figure(4)
subplot(311)
rlocus(TKpOff)
title('Generalized Root Locus for KpOff')
legend([num2str(tankRange(m)*100), ' %'], 'location', 'northwest')

subplot(312)
rlocus(TKiOff)
title('Generalized Root Locus for KiOff')
legend([num2str(tankRange(m)*100), ' %'], 'location', 'northwest')

subplot(313)
rlocus(TKdOff)
title('Generalized Root Locus for KdOff')
legend([num2str(tankRange(m)*100), ' %'], 'location', 'northwest')

end

if sysRoots

figure(5)
plot(real(rootsDen), imag(rootsDen), 'x', [-200 2], [0 0], 'k:', ...
    [0 0], [-10 10], 'k:', 'MarkerSize', 7')
title('System roots: Grain tank from empty to full (%)')
xlabel('Real')
axis([-7 0 -2.5 2.5])
ylabel('Imag')
legend('0%', '25%', '50%', '75%', '100%', 'location', 'best')

end

end

m = m + 1;
end
```

REFERENCES

1. Gillespie, T. D., *Fundamentals of Vehicle Dynamics*, Warrendale, PA: Society of Automotive Engineers, Inc., 1992.
2. E.J. Baack, "A zero slip vehicle dynamics model for virtual reality applications", MS Thesis, Iowa State University, 2003.
3. J.T. Hummel, "A drive train model for use with heavy-duty agricultural and construction vehicles," MS Thesis, Iowa State University, 2005.
4. Nise, Norman S., *Control Systems Engineering*, John Wiley & Sons, 2004
5. Bell, T. 2000. Automatic tractor guidance using carrier-phase differential GPS. *Computers and Electronics in Agriculture* 25:53-66.
6. Ashraf, M.A., Torisu, R., Takeda, J. 2002. Autonomous Traveling of Off-road Vehicles along Rectangular Path on Sloped Terrain. In: Proceedings of the Automation Technology for Off-Road Equipment, July 26-27, 2002, pg. 412-421.
7. Wu, D., Q. Zhang, J.F. Reid, H. Qiu, E.R. Benson. 1998. Model Recognition and Simulation of an E/H Steering Controller on Off-Road Equipment. Fluid Power Systems and Technology 1998. New York, NY: ASME 55-60.
8. Reid, J.F., Zhang, Q., Noguchi, N., Dickson, M. 2000. Agricultural automatic guidance research in North America. *Computers and Electronics in Agriculture* 25:155-167.
9. O'Connor, M., T. Bell, G. Elkaim, B. Parkinson, 1996. Automatic steering of farm vehicles using GPS. *Paper presented at the 3rd International Conference on Precision Agriculture*. Minneapolis, MN, June 23-26.
10. Metz, L. D. 1993. Dynamics of Four-Wheel-Steer Off-Highway Vehicles. SAE Paper No. 930765.
11. Bernard, J. E. ME 549: Vehicle Dynamics class notes. Iowa State University, Fall Semester, 2004.
12. Steward, B. L. Four-wheel Steer Modeling and Simulation Notes. January, 2006.
13. Feng, L., He, Y., Zhang, Q. 2004. Dynamic Trajectory of a Tractor-Implement System for Automated Navigation Applications. In: Proceedings of Automation Technology for Off-Road Equipment, Kyoto, Japan, October 7-8, 2004.

14. Feng, L., He, Y. 2005. Study on dynamic model of tractor system for automated navigation applications. Journal of Zhejiang University SCIENCE, 2005 6A(4):270-275.
15. Miller, M. A., Steward, B. L. Control and Evaluation Methods for Multi-Mode Steering. In: Proceedings of the Automation Technology for Off-Road Equipment, Chicago, IL, USA, July 26-27, 2002, pg. 357-366.
16. Huh, K., Kim, J. 2001. Active Steering Control Based on the Estimated Tire Forces. ASME Journal of Dynamic Systems, Measurement, and Control, Vol. 123, pg. 505-511.
17. Norris, W. R., Zhang, Q., Sreenivas, R., Lopez-Dominguez, J. C. 2003. A Design Tool for Operator – Adaptive Steering Controllers. Transactions of the ASAE, Vol. 46(3): 883-891.
18. O'Connor, M., Elkaim, G., Parkinson, B. 1995. Kinematic GPS for Closed-Loop Control of Farm and Construction Vehicles. Presented at ION GPS-95, Palm Springs, CA, September 12-15, 1995.
19. Deere & Company. *GREENSTAR Guidance Parallel Tracking and AutoTrac Assisted Steering Systems Operator's Manual*. Moline, IL, 2005.
20. Owen, R. H., Bernard, J.E. 1982. Directional Dynamics of a Tractor-Loader-Backhoe. Vehicle System Dynamics 11 (1982), pp. 251-265.

ACKNOWLEDGEMENTS

I would like to first thank and acknowledge my Heavenly Father who has given me the opportunities and abilities to learn. His help and guidance have been an inspiration, a blessing, and a strength. He has truly guided me in my life as I have made educational, family, and career decisions.

My appreciation extends to all the members of my Program of Study Committee who have helped me learn and develop as an engineer and a person. To Dr. Jim Bernard, who has given me the freedom to explore and helped find ways to ensure the needs of me and my family were met, both educationally and financially. He has willingly listened to my requests and given me sound advice on all aspects of life. To Dr. Greg Luecke for guiding me through the fundamentals of controls and for helping me gain experience in real world engineering applications through our John Deere real-time immersive combine simulator. The project has helped pave the way towards my next career step and has given me an opportunity to become acquainted with and learn from a great professor and person. To Dr. Brian Steward, who has helped me hone my skills through his classes and given me exposure to the agricultural side of engineering. I appreciate his willingness to meet and help me understand both simple and difficult concepts regardless of the subject.

I would also like to thank all those at Brigham Young University and Iowa State University and the Virtual Reality Applications Center who have had anything and everything to do with my success and enjoyment while pursuing my engineering education and degrees.

Administrators, professors, staff members, and students have all played an integral role in helping me obtain my goals. The names are too numerous to list, but nonetheless, they all deserve recognition and my appreciation.

Special thanks also needs to be given to the several John Deere employees at various divisions who so graciously took time out of their busy work schedule to answer questions and provide assistance.

Last, but certainly not least, I would like to thank my family for the support and love they've given to me. To my wife, Lindsay, who always instilled confidence and who spent countless hours making lunches, typing equations, and picking up the slack when I was not home. She has never doubted and has always been there to care for my every need. To my kids, Kamiah and Wesley, who always managed to put a smile on my face no matter what the circumstance. I wouldn't trade them for the world. And to my extended family: parents, siblings, in-laws, and grandparents, who were always there to give support in many ways. They helped sustain us when times were difficult and helped keep life in perspective.

To everyone everywhere, I will always be grateful.