2007

Investigation of roll angle feed-forward control and look-ahead functions to improve automatic guidance on sloped terrain

James Addison Bosserd II
Iowa State University

Investigation of roll angle feed-forward control and look-ahead functions to improve automatic guidance on sloped terrain

by

James Addison Bosserd II

A thesis submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Major: Agricultural Engineering (Agricultural Power & Machinery)

Program of Study Committee:
Brian L. Steward, Major Professor
Stuart J. Birrell
Greg R. Luecke
Shufeng Han

Iowa State University

Ames, Iowa

2007

ProQuest Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to take this opportunity to briefly thank all of the people that have helped contribute to my success, whether it was mentally or physically. I first want to extend my thanks to Iowa State University, and specifically the Department of Agricultural and Biosystems Engineering. They provided me the opportunity to continue my education in Agricultural Engineering and supported that learning process with countless resources, which are too numerous to mention. Those at Iowa State who deserve special credit are the members of my Program of Study Committee: Dr. Stuart Birrell, Dr. Greg Luecke and Dr. Brian Steward. Dr. Birrell was an excellent source of knowledge and support throughout the process. Dr. Luecke helped to develop my engineering and professional skills, both in and out of the classroom. Lastly, Dr. Steward provided me with countless hours of support and guidance in addition to his roll as my teacher and mentor. Without his knowledge and personal experience none of this would have been possible

I would also like to thank John Deere and Dr. Shufeng Han. John Deere provided the financial support and resources that made this project possible. Dr. Han provided me with any information I required, helped to contribute his specialized knowledge to this area of study and also served on my committee.

Lastly, I would like to thank my family (Jim, Jane, Steve and Becky Bosserd, plus the rest of the clan) and friends (especially all those I met along the way: Brian Gelder, Jeremiah Davis, Kelly Thorp, Adam Van De Walle and too many others to mention) for providing support, knowledge and their ability to have fun during this process as well as helping me to remember the things that are most important in life.

ABSTRACT

As agricultural automatic guidance systems are further developed, greater attention is being given to their performance under all reasonable conditions including the side slopes of hills. To investigate the potential of improving auto-guidance performance using feed-forward control with expected vehicle roll angle acquired from a slope path profile, a PID navigation controller was simulated along with tractor and tire models. There were four main steps to the simulation process: (1) The development of the gains for a roll angle feed-forward controller, (2) Evaluation of the controller as it responded to both step and sinusoidal slope profiles, (3) Investigation of the effects of different vehicle velocities and changes in cornering stiffness compared to nominal values, and (4) Determination if looking-ahead into the slope increases roll angle performance. Use of a roll angle feed-forward controller reduced the maximum off-tracking distance by up to 87 percent. For the sinusoidal slope profiles, increased velocity had a negligible effect on off-tracking error. Using a look-ahead distance of 2 m for future estimates of roll angle reduced the off-tracking distance by up to 50%. When compared to the nominal values, increasing or decreasing the tire cornering stiffness adversely affected the off-tracking error for the given vehicle model.

CHAPTER 1. INTRODUCTION

1.1 Motivation

For much of the previous century, advances in the development of agricultural equipment, along with improvements in genetics, pest control, and fertilization have allowed producers to achieve higher levels of productivity. For instance, in 1900 the average American farmer produced enough food for 7 people; today in 2006 that number has risen to 135 (Minnesota Beef Council, 2006). Agricultural mechanization is ranked number 7 in the National Academy of Engineering's list of the 20 greatest engineering achievements of the 20th century (Constable, et al., 2003). Recently, precision agriculture has come to the forefront with a goal to improve the precision with which plants are managed through the increasing use of sensing, automation, and control – information technologies, rather than simply increases in the power and size of power units and implements (Cox, 2002).

Precision agriculture has led to the development of several new agricultural and off-road vehicle automation technologies. A key enabler for precise vehicle localization has been Global Positioning Systems (GPS) technology. When the potential of GPS technology in agriculture was first considered, it brought about the concept of making georeferenced measurements to document and manage spatial variability that occurs in agricultural fields. Thus one of the first uses of GPS in agriculture was georeferencing yield measurements from harvest yield monitors for the generation of yield maps. Taking this further, with prescription maps, producers could vary their chemical input application rates using GPS on chemical application vehicles, such as sprayers and spreaders.

The potential of GPS paired with motorized vehicles was quickly utilized for precision agriculture. Vehicle guidance assistance first provided visual cues to the driver through a display such as a light bar mounted in front of the driver. Automatic guidance was the next precision agriculture technology using GPS, where the operator's steering input is removed for the majority of the time in the field. Automatic guidance works by using GPS signals to pinpoint the vehicle location in the field. The operator drives across the field creating an initial path, called an A-B line, to follow. Once the location is known, a controller is then used to send a signal to the hydraulic steering valve based on a distance from the first path. The vehicle proceeds across the field following parallel tracks laid out by the controller set by a predetermined implement width. An operator was still required to monitor the field operations, control the implement or other machine functions, and to turn the vehicle at the end of each path. With GPS-based automatic guidance systems, producers have been able to cut down on inputs by minimizing overlapping passes through fields and reducing fuel consumption (Pettygrove, et al., 2000; Lambert, et al., 2000).

Automatic guidance systems are a rapidly growing market and are being adopted for reasons such as increased efficiency and cost reductions. Early adoption of guidance systems occurred in the fertile valleys of California, where the region is well known for its produce production. Produce crops are high value crops, but also have high cost inputs, so operational and input efficiency is important. It is also common practice to use drip-tape irrigation; producers found that the guidance technology allowed them to precisely place tape in the ground and return to the same tapelines year after year for planting (Abidine, et al., 2004). Word spread in the agricultural community and soon adoption of this technology started to move in the direction of the row-crop corn and soybean fields of the Midwest

(Finck, 2004). Farmers in the Midwest have found that with automatic guidance systems they can work more ground with less equipment than they could without the new technology (Watson, et al., 2002). There is also an increasing trend of utilizing automatic guidance for controlled traffic systems where the vehicles travel in the same desired paths in an effort to reduce compaction across a field.

Before spending the money on the expensive, high accuracy automatic steering units, producers adopted lower cost systems to confirm the technology worked. It was soon clear, however, that a higher level of accuracy was desirable, especially with the ability to use automatic guidance technology to enable highly specialized practices such as controlled traffic farming and strip tilling (Smith, 2003). Higher localization accuracies are achieved by utilizing a localized base station receiver, and are known as Real Time Kinematic GPS (RTKGPS) which can provide a 2 centimeter accuracy level. This was an improvement over the previous systems which were based on differential signals covering larger areas and were not referenced to a local base station. The less accurate technology is known as Differential GPS (DGPS), which can only provide accuracy of 10 centimeters. This was still an improvement over GPS receivers with no differential correction signals that typically only have an accuracy of greater than 3 meters. The RTKGPS technology has provided producers a higher level of accuracy that they now require for improved efficiencies. However, there was still one problem facing producers. The new automatic guidance systems only operated at a high level of accuracy in straight paths, on relatively flat terrain; otherwise they incurred errors that would negate the improved efficiency.

1.2 Background

Agricultural vehicles must operate on sloping terrain, given that much agricultural land has hills and valleys and undulating topography. To operate effectively on sloping terrains, several potential problems must be overcome. Safety on sloping terrain can be a concern and extreme slopes can lead to vehicle overturn. To improve the safety in these situations a Roll-over Protection System (ROPS) standard has been developed and adopted by agricultural tractor manufacturers. The purpose of a ROPS system is in the event of a tractor overturning, a two or four post system built around the operator station will provide the operator with a safety envelope, avoiding a crushing fatality (ASAE, 2004). The narrow front wheel configuration on tractors, popular from the 1930s through the 1950s, caused tractors to be prone to roll-over and have since been eliminated. On combines, not only are vehicle overturns possible, but also slopes cause grain inside the machine to flow to one side, which causes the fluidized bed on the cleaning shoe to break down, leading to increased grain losses. To overcome this problem, some modern combines have self-leveling cleaning systems that allow for normal operation of the grain cleaning systems on extreme slopes. Now with the advancement of automatic guidance systems, slopes are an issue once again.

The first noticeable problem with GPS-based automatic guidance is that on a slope the GPS receiver is no longer directly over the center of the vehicle. Instead, the receiver location is projected directly to a point on the ground along the gravity vector (Figure 1.1). This effect results in the tractor constantly shifted off-track towards the uphill side of the slope. By installing roll sensors along with their receivers, most manufacturers have built in adjustments that eliminated the constant offset due to the roll angle.

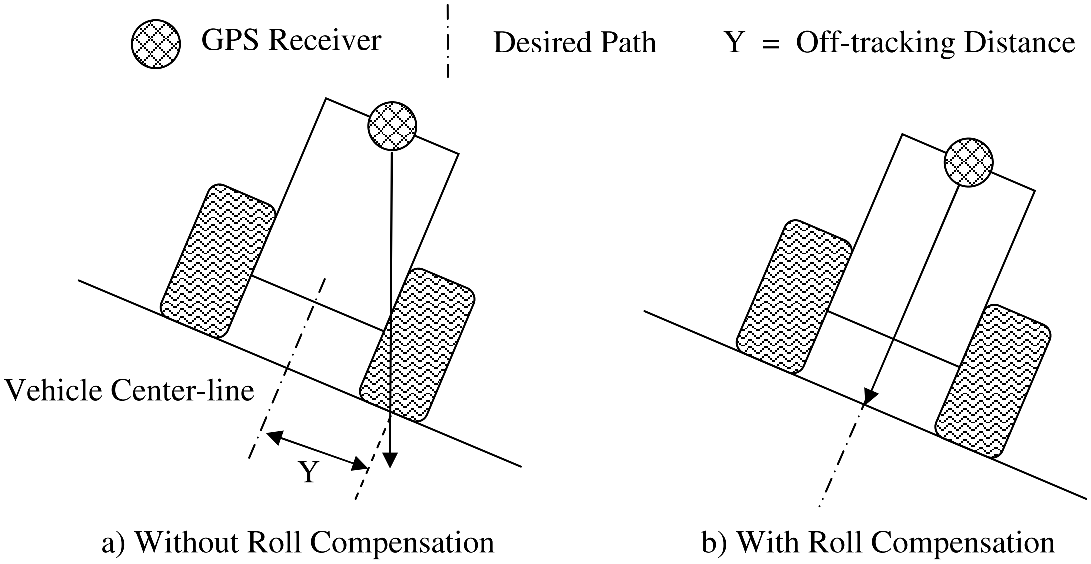

GPS Receiver
Desired Path
Y = Off-tracking Distance
Vehicle Center-line
Y
a) Without Roll Compensation
b) With Roll Compensation

Figure 1.1 Tractors not-equipped with roll compensation (a) track to a line directly below the GPS receiver leading to systematic off-tracking distances. Tractor with roll compensation (b) minimizes this error source.

Another effect of operating on a side slope is the gravity vector contains not only a normal force component opposed by the ground pushing up against the tires, but also a lateral force component that must be counteracted by an opposing lateral tire force (Figure 1.2). This additional lateral force acts as a disturbance to the navigation controller system by changing the balance of forces on the vehicle. The component of the weight parallel to the slope must be balanced by equal lateral tire forces. Without a controller, uncompensated lateral force leads to a constant off-tracking distance. As the vehicle attempts to follow a pre-determined path, an off-tracking distance is measured.

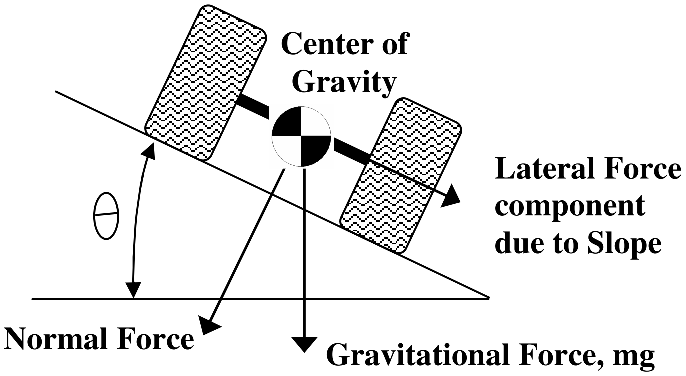


Figure 1.2 On a side slope, the weight is divided into a component normal to the slope and another component parallel to the slope.

Before describing any further the effects of the disturbances caused by slope, it is important to systematically set up the variables and nomenclature to describe this problem. The off-tracking error (E_{OT}) is the perpendicular distance between the vehicle's center of gravity (or some other control point) and the desired path. The heading error (E_H) is the angle created between the desired path's vector and the vehicle's velocity vector. When oriented correctly with the world coordinate system (WCS), the heading error is the yaw angle. The vehicle operates within the WCS, and it is this coordinate system that automatic guidance is based upon; that is, the vehicle is being navigated to track a line which is fixed or stationary in the world. To simplify the analysis, the X-axis of the WCS was set so that it coincided with the straight-line path being tracked. This convention led to E_{OT} being the distance along the Y coordinate that the vehicle was displaced from the x axis and the heading error to be equal to the yaw angle of the vehicle, ψ. Whenever E_{OT} and E_H appear in the analysis, these variables can be immediately replaced with Y and ψ.

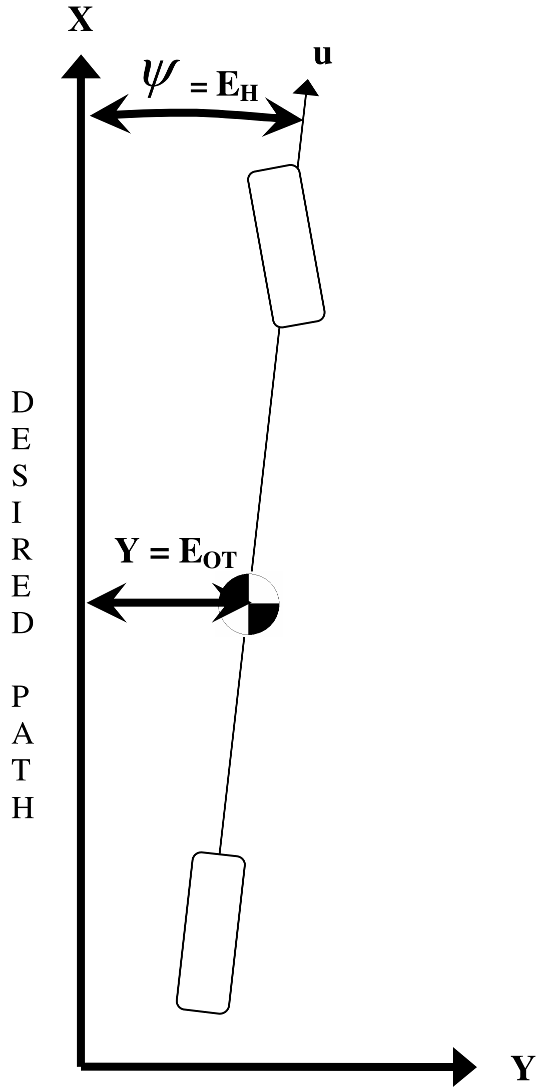


Figure 1.3 Vehicle Off-tracking error (E_{OT}) and Heading error (E_H) are equivalent to Y and ψ when the world coordinate system is orientated with the WCS X-axis aligned with the desired straight-line path.

On a front wheel steered vehicle, such as a tractor, it is impossible to eliminate the heading error due to the effects of gravity, when operating on a slope. Only an understeer gradient of zero, for a rear steered vehicle, would result in a steering angle of zero, or a four-wheel steer vehicle with all four wheels steered in the uphill direction would remove the heading error when operating on a slope (Evans, 2006). When a tire is steered, the direction vector of the tire is made up of lateral (Y) and longitudinal (X) force vectors. It is the lateral force vector that applies the forces required to steer the vehicle. When operating on a slope, a four wheel steer vehicle would be evenly applying lateral forces, front to back, to compensate for the slope. With only one set of wheels steering, one end of the vehicle is creating sufficient lateral forces to track to the path. Since the opposite end of the vehicle is not generating the same lateral forces, this end of the vehicle will have a greater off-tracking error than the other, creating a constant heading error. Besides heading error, the only factor that remains to affect the guidance is the off-tracking error. The controller will use the off-tracking and heading errors in order to make corrections toward the path to reduce these errors. However without controller compensation for the slope, the steering angle commanded will only be that required to obtain zero off-tracking on level ground, thus creating a constant off-tracking error in slope applications. The exception to this is utilizing an integral control to drive the off-tracking error to zero.

Evans (2006) developed a yaw plane bicycle model of a rear steered vehicle to investigate the performance of an automatic guidance system for a combine on side-slopes. Evans found the steady state relationship between slope, turning radius and steering angles for a four wheel steered vehicle to be:

$$\delta_f - \delta_r = \frac{L}{R} + K\left(a_y - \theta\right) \tag{1-1}$$

where δ_f and δ_r are the front and rear steering angles in radians, L is the vehicle length in meters, R is the radius for a steady state turn in meters, K is the understeer gradient, a_y is the lateral acceleration in g's and θ is the roll angle in radians. When tracking a straight line, the lateral acceleration would be equal to zero and the radius would be infinite. Thus on a side slope, tracking a straight line would require the steering angle to be equal to the understeer gradient times the roll angle.

Furthermore, Evans calculated the steady state heading angle when equation 1-1 is substituted back into δ_r ($\delta_f = 0$ for rear wheel steer), resulting in the relationship,

$$\beta_{ss} = \frac{W_f}{C_{\alpha f}}\theta \tag{1-2}$$

where, β_{ss} is the steady state sideslip angle, W_f is the weight on the front of the vehicle and $C_{\alpha f}$ is the cornering stiffness of the front tires. This equation shows that if there is a slope present, then a two wheel, rear steered vehicle will experience some degree of side-slip in order to track the desired path.

Evans modeled a proportional-integral-derivative (PID) automatic guidance navigation controller which used off-tracking and heading error feed-back signals. The major differences between Evans' work and that of this thesis are (1) He did not use feed-forward control, (2) He also did not investigate the use of a look-ahead function, and (3) He did not include the effects of steering delay on the system, except in an appendix. Evans showed that integral and derivative heading error feedback terms could be dropped since they had little impact on performance. He derived the transfer function relationship between off-tracking error and slope angle. The automatic guidance controller and the vehicle system were simulated with a five degree step slope input. His results showed that in most cases the off-track error was reduced to an acceptable value within five seconds. Evans investigated the effect of vehicle parameters such as changes in center of gravity location (fore and aft), vehicle weight, vehicle velocity and tire cornering stiffness. He observed from simulation and root locus analysis that only the slope and the tire cornering coefficients had an effect on the system response. Evans included some steering delay work in his appendices, from which initial results showed differences from his other work. As a result, steering lag was included along with feed-forward control in this thesis, which is more realistic since vehicle steering systems are hydraulically powered and changes in steering angle occur through finite fluid flow. Although Evans' work showed that vehicle forward velocity had a negligible effect on the controller performance, our use of feed-forward simulations included steering lag and also different types of slopes, so controller performance over multiple velocities were simulated in our work. The work accomplished in this thesis also took the slope simulations a step farther and utilized different slope types to test the controller (Evans, 2006).

1.3 Research Objectives

The main goal of this research was to investigate the potential of improving auto-guidance off-tracking performance on side-slopes by feeding forward vehicle roll angles. Supporting this main research goal were the following specific objectives:

1. To investigate the effect of a roll angle feed-forward signal on controller performance.
2. To quantify the response of the controller on slopes.
3. To investigate the effects of changing parameters on vehicle controller performance.
4. To determine if any improvements can be made using a look-ahead function.

Each of these action items were considered a high priority in order to prove or disprove any beneficial effects that the controller hoped to gain. In addition to these four items, analytical work was required to determine a gain for the roll angle feed-forward term.

One main way this controller differs from those in previous work is the use of a feed-forward controller. This controller utilizes a proportional gain on the feed-forward signal to compensate for the force of gravity on the vehicle. This compensation is accomplished by the gain effectively "forcing" the output of the controller to be larger than it would be otherwise. The feed-forward aspect of the simulation was expected to have the single greatest effect on the way the controller affected the automatic guidance of the vehicle. The remainders of the simulations were secondary to these tests and were believed to only slightly enhance the functionality of the controller that is already improved upon by using feed-forward control.

Once the controller was designed, its effectiveness had to be verified. This was accomplished though the use of two different test slope profiles. Both utilized a straight path with varying slopes. The first slope profile investigated was a sinusoidal slope with amplitude of 5 degrees. The second slope profile used was a 5 degree step slope. These two slopes were then used to determine the effectiveness of the controller to track a given line.

Changes in vehicle parameters were important to show the robustness of a controller to operate effectively in many situations and using different vehicle parameters. The feed-forward model was constructed with this idea in mind and allowed for the ease of changing variables to simulate different vehicle makes and models. In his work on controller development in sloped terrain, Evans (2006) found that changes in individual vehicle parameters such as center of gravity and mass had little effect on controller performance. Due to these findings, the parameter changes in this simulation do not deal with center of gravity or vehicle mass. Two parameters were of particular importance: vehicle velocity and tire cornering stiffness. In a situation where response delays can cause poor controller performance, velocity has the potential to have a negative effect on the system. Tire cornering stiffness is the factor that deals with tire to soil interaction. For most of the simulations, one set of nominal tire cornering stiffness values were chosen. A separate simulation was run to examine how other cornering stiffness values, when compared to the nominal values, affected controller performance. The off-tracking integral controller gain was also investigated due to its small value. Integral off-tracking error gain uses a value selected as described in chapter three.

The last objective is determining the effectiveness of utilizing a look-ahead function. The basis behind this function is the use of a Digital Elevation Model (DEM) of a specific terrain. A DEM provides an elevation element to a standard surface area map. With GPS technologies, farmers can make their own terrain maps (DEMs) of their fields (Westphalen, 2004). In the field, roll angles can be measured using roll angle sensors; however with the availability of a DEM it is possible to look-ahead into the terrain to estimate upcoming roll angles. As a result, instead of basing the controller off of the roll angle at the vehicle's current location, looking-ahead into the DEM can pull out a point a certain distance in front of the vehicle and feed-forward the upcoming expected roll-angle of the vehicle. With this additional function, off-tracking error for automatic guidance could be further reduced. For testing purposes, the idea behind a DEM was captured. However, instead of using an actual DEM, we were able to look-ahead into our predetermined slope path to utilize upcoming roll angles.

1.4 Summary of work

Chapter 1 provided the background for the research by laying out the motivation for this study. Chapter 2 introduces the vehicle model, describes in detail the development of the controller model, its gain values, the simulations used in conjunction with them and corresponding controller development. Chapter 3 describes the methodology of the simulations. Chapter 4 presents the simulation results and some discussion. Chapter 5 outlines the conclusions derived from the results and offers suggestions in areas of continuing research. The Appendices contain a detailed list of the vehicle parameters used as well as full, detailed derivations of the steady state equations of the automatic guidance controlled vehicle system, as well as the MATLAB and SIMULINK models used to run the simulations (ver. 7, Mathworks, Natick, MA).

CHAPTER 2. SIMULATION AND CONTROLLER DEVELOPMENT

2.1 Vehicle Model

A vehicle model was developed that served as the basis for all simulations in the project. In previous work, Westphalen (2004) developed a similar vehicle model in his work on a rear steering controller, which was then updated for this specific simulation. The vehicle model developed was a three degree of freedom; yaw-plane bicycle model which is a standard model used among vehicle dynamics experts (Figure 2.1). Two main areas where the new model differs from the one shown in Figure 2.1 are the earlier model included a rear wheel steering angle for a four wheel steer vehicle, and it did not include lateral forces due to gravity. This section contains an abridged derivation of the defining equations for the model. Table 2.1 provides a list of all the variables used in the vehicle model. The complete model development is documented in Appendix B.

The switch from four to two wheel steering involved simply setting the rear steering angle to zero. The primary reason for this change was that an agricultural tractor (model 8320, Deere and Co., Moline, IL) was being simulated. Front wheel steering tractors dominate the market in the Midwest, specifically those using an automatic guidance system. Naturally this type of steering set-up best suited the research being done for this market area. The forces generated on the model are a result of the tire slip angle. A lateral force is applied perpendicular to each tire, which causes the vehicle to yaw, and is larger when the tires are turned.

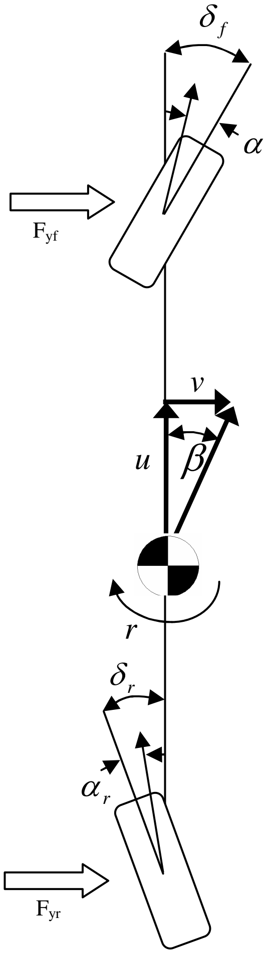


Figure 2.1 Four-wheel steer, three degree of freedom, yaw-plane bicycle model based on Westphalen (2004).

The addition of a force due to slope went beyond a simple parameter change. This lateral force is a factor of the vehicle's mass times gravity times the sine of the slope angle. This force was then added into the force balance equation of the dynamic model, where it was applied to the center of gravity (Figure 2.2).

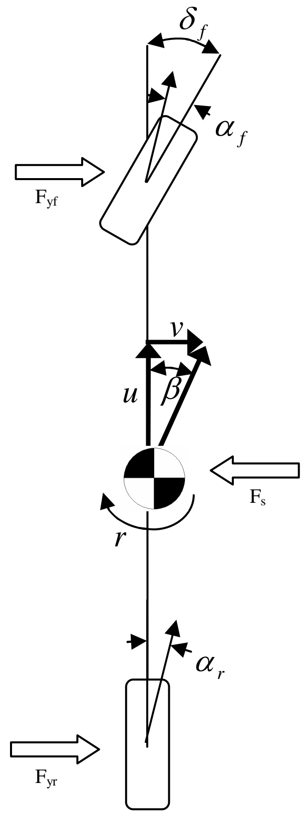


Figure 2.2 Modified yaw-plane bicycle model for front wheel steer vehicle operating on slopes with an additional lateral force due to gravity.

As stated earlier, this model is based on a 3-degree of freedom, yaw plane bicycle model. This allows for both lateral and horizontal movement as well as rotation about the yaw axis. The model is limiting in some respects by only including three degrees of freedom and thus not including pitch, roll and vertical movement. Vertical movement is not important from an automatic guidance perspective and thus it was not necessary to include it in the model. Weight transfer due to pitch and roll is important for vehicle dynamics, but in this case the addition of those components would make the model too complex. Part of the reason these were omitted was also due to the vehicle operating at relatively slow speeds.

Operation on a slope does cause weight transfer regardless of speed, but the complexity of the simulation required is beyond the scope of the problem addressed in this research.

Unlike traditional yaw-plane bicycle models, this model had the addition of a gravity force vector. The roll degree of freedom was omitted from the model by rotating the yaw plane about the vehicle's centerline. The additional force was then applied via a lateral force located on the uphill side of the vehicle creating a force vector expressed mathematically as:

$$F_s = mg\sin(\theta) \tag{2-1}$$

where F_s is the lateral force, θ is the roll angle, and m is the vehicle mass. Positive roll angles are defined as a counterclockwise rotation of the vehicle about an axis pointed toward the front of the vehicle. Thus, a positive roll angle will result in a vehicle weight component towards the left through the vehicle CG.

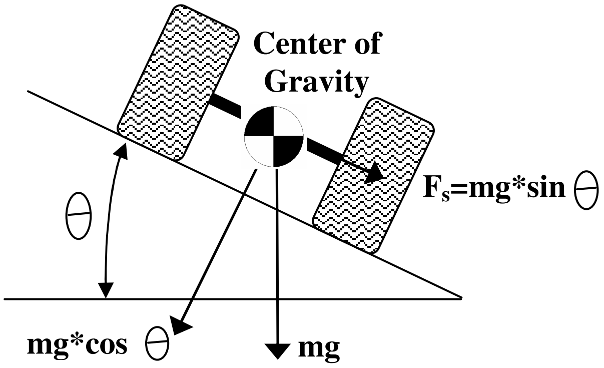


Figure 2.3 Lateral force vector generated from the gravitational vector acting upon a vehicle as viewed from the front.

Table 2.1 List of variables that are used in this simulation, and their descriptions.

Variable	Description	Units
α_f	Front wheel slip angle	rad
α_r	Rear wheel slip angle	rad
a	Distance from CG to front axle	m
β	Sideslip angle	rad
b	Distance from CG to rear axle	m
C_c	Cornering Coefficient	rad^{-1}
$C_{\alpha f}$	Cornering Stiffness for front tire	N/rad
$C_{\alpha r}$	Cornering Stiffness for rear tire	N/rad
d	Look-ahead distance	m
δ_f	Front steering angle	rad
$\delta_{f,c}$	Commanded front steering angle	rad
δ_r	Rear steering angle	rad
F_s	Lateral force due to gravity	N
F_{yf}	Lateral force for front wheels	N
F_{yr}	Lateral force for rear wheels	N
g	Gravitational acceleration	m/s^2
I	Moment of Inertia	kg-m^2
K	Understeer gradient	rad
K_H	Heading Gain	unit less
K_{OT}	Off tracking Gain	rad/m
L	Length of wheel base	m
m	Vehicle mass	kg
r	Yaw Rate	rad/s
u	Longitudinal velocity	m/s
v	Lateral velocity	m/s
W	Vehicle weight	N
W_f	Vehicle weight for front tires	N
W_r	Vehicle weight for rear tires	N
X	Vehicle position in WCS (longitudinal)	m
Y	Vehicle position in WCS (lateral)	m
θ	Roll angle	rad
ψ	Vehicle heading angle	rad

The equations of motion for the vehicle were developed based on Westphalen (2004) with some changes made to better represent the vehicle system. The summations of the lateral forces acting on the vehicle give the first equation of motion:

$$F_{yf} + F_{yr} - F_s = m\left(\dot{v} + ur\right) \tag{2-2}$$

The next equation was derived by summing the moments about the center of gravity:

$$F_{yf}a - F_{yr}b = I_{zz}\dot{r} \tag{2-3}$$

When the automatically guided vehicle was tracking the straight line, only relatively small steering angles were required for accurate control, thus allowing the use of a small angle assumption in the front and rear lateral tire force equations:

$$F_{yf} = -C_{\alpha f}\alpha_f \tag{2-4}$$

$$F_{yr} = -C_{\alpha r}\alpha_r \tag{2-5}$$

Substituting these equations into 2-2, and then simplifying results in:

$$\dot{v} = \frac{-C_{\alpha f}\alpha_f - C_{\alpha r}\alpha_r - W\sin(\theta) - mur}{m} \tag{2-6}$$

The front and rear slip angles, α_f and α_r respectively, are functions of the slip angle at the center of gravity, steering angle, forward velocity, raw rate and distance to the CG:

$$\alpha_f = \beta + \frac{ar}{u} - \delta_f \tag{2-7}$$

$$\alpha_r = \beta - \frac{br}{u} - \delta_r \tag{2-8}$$

Where β:

$$\beta = \tan^{-1}\left(\frac{v}{u}\right) \tag{2-9}$$

Although δ_r will eventually be set to zero since we are simulating a front-steer vehicle, it is being kept in the equations of motion in order to create a more generic solution. Using the small angle assumption for β ($\beta = v/u$) and then substituting 2-7, 2-8 into 2-6 results in our first state equation:

$$\overset{\bullet}{v} = \frac{-C_{\alpha f}\left(\frac{v+ar}{u} - \delta_f\right) - C_{\alpha r}\left(\frac{v-br}{u} - \delta_r\right) - W\sin(\theta) - mur}{m} \tag{2-10}$$

Now taking the same steps with equation 2-3, finalizing the second state equation:

$$\overset{\bullet}{r} = \frac{-C_{\alpha f}\left(\frac{v+ar}{u} - \delta_f\right)a + C_{\alpha r}\left(\frac{v-br}{u} - \delta_r\right)b}{I_{zz}} \tag{2-11}$$

Finally, because we were controlling the position of the vehicle relative to a straight-line path, two more state equations for the yaw angle and lateral position relative to the path are required.

$$\overset{\bullet}{Y} = v + u\psi \tag{2-12}$$

Where Y is measured with respect to the center of gravity.

$$\overset{\bullet}{\psi} = r \tag{2-13}$$

The steering system was modeled as a first-order delay with a time constant, τ. The steering delay is based on $\overset{\bullet}{\delta}_f$ and shown next.

$$\dot{\delta}_f = -\delta_f \frac{1}{\tau} + \delta_{f,c} \frac{1}{\tau} \tag{2-14}$$

The steering system has additional non-linearities, which were not included here but were included in the simulations. These non-linearities include steering angle and steering rate saturation limits, and also hydraulic non-linearity within the steering system. These were important to the system as they have a direct affect on how the system responds regardless of the commanded values. These prevent the model from generating an unrealistic steering signal that would not be attainable with a vehicle. Rewriting the equations 2-10 through 2-14 in matrix form gives:

$$\begin{bmatrix} \dot{v} \\ \dot{r} \\ \dot{Y} \\ \dot{\psi} \\ \dot{\delta}_f \end{bmatrix} = \begin{bmatrix} A & B & 0 & 0 & E \\ C & D & 0 & 0 & G \\ 1 & 0 & 0 & u & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{1}{\tau} \end{bmatrix} \begin{bmatrix} v \\ r \\ Y \\ \psi \\ \delta_f \end{bmatrix} + \begin{bmatrix} F & -\frac{W}{m} & 0 \\ H & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{1}{\tau} \end{bmatrix} \begin{bmatrix} \delta_r \\ \sin(\theta) \\ \delta_{f,c} \end{bmatrix} \tag{2-15}$$

Where:

$$A = \frac{-C_{\alpha f} - C_{\alpha r}}{mu} \tag{2-16}$$

$$B = \frac{-C_{\alpha f} a + C_{\alpha r} b - mu^2}{mu} \tag{2-17}$$

$$C = \frac{-C_{\alpha f} a + C_{\alpha r} b}{I_{zz} u} \tag{2-18}$$

$$D = \frac{-C_{\alpha f} a^2 - C_{\alpha r} b^2}{I_{zz} u} \tag{2-19}$$

$$E = \frac{C_{\alpha f}}{m} \tag{2-20}$$

$$F = -\frac{C_{\alpha r}}{m} \tag{2-21}$$

$$G = \frac{C_{\alpha f}\, a}{I_{zz}} \tag{2-22}$$

$$H = \frac{C_{\alpha r} b}{I_{zz}} \tag{2-23}$$

Then simplifying for front wheel steer alone is accomplished by setting δ_r equal to zero.

2.2 Steady State Solutions

2.2.1 Non-Feed Forward Steady State

Once the equations of motion for the vehicle were developed, then analysis was performed to calculate the steady state solutions based on these equations. For automatic guidance, the three steady state variables of most importance are heading error (yaw angle, ψ), off-tracking error (Y), and steering angle (δ_f). These three variables define the tracking and motion of the vehicle during simulation. Determining the steady-state values for each variable provided a baseline which was then compared to future calculations. A brief outline of each steady state derivation is shown below. More details are provided in Appendix B.

This analysis started with an automatic guidance navigation controller with proportional off-tracking and heading error feedback. The control law for this prototype controller was:

$$\overline{\delta}_f = \overline{\delta}_{f,c} = -K_{OT}\left(\overline{Y} + d \sin\overline{\psi}\right) - K_H \overline{\psi} \tag{2-24}$$

where d is the look-ahead distance.

Then to find the steady-state relationships, all of the state variable in the vehicle model were replaced with steady state variables as indicated with the bar. Then, since all of the steady state variables were constant for particular input sets, the first derivatives go to zero. This reduced the first two state equations to:

$$\dot{\bar{v}} = 0 = A\bar{v} + E\bar{\delta}_f - g\sin\left(\bar{\theta}\right) \qquad (2\text{-}25)$$

And

$$\dot{\bar{r}} = 0 = C\bar{v} + G\bar{\delta}_f \qquad (2\text{-}26)$$

Since $\dot{\bar{Y}}$ also goes to zero, the third state equation reduced to:

$$\bar{\psi} = -\frac{\bar{v}}{u} \Rightarrow \bar{v} = -\bar{\psi}u \qquad (2\text{-}27)$$

Also from the fourth state equation:

$$\bar{r} = 0 \qquad (2\text{-}28)$$

Now substituting the control law (2-24) into equations 2-25 and 2-26, and substituting for $\bar{v}$ using (2-27), then solving for $\bar{Y}$, resulted in:

$$\bar{Y} = \frac{-A\bar{\psi}u - EK_{OT}d\sin\bar{\psi} - EK_H\psi - g\sin\left(\bar{\theta}\right)}{EK_{OT}} \qquad (2\text{-}29)$$

And

$$\bar{Y} = \frac{-C\bar{\psi}u - GK_{OT}d\sin\bar{\psi} - GK_H\bar{\psi}}{GK_{OT}} \qquad (2\text{-}30)$$

Setting these two equations equal to each other then factoring:

$$\frac{\left(-Au - EK_H\right)G}{EK_{OT}G}\bar{\psi} - \frac{\left(-Cu - GK_H\right)E}{GK_{OT}E}\bar{\psi} = \frac{g\sin\bar{\theta}}{EK_{OT}} \qquad (2\text{-}31)$$

Pulling $\overline{\psi}$ out and simplifying again:

$$\overline{\psi} = \frac{Gg \sin \overline{\theta}}{(-GA + EC)u} \tag{2-32}$$

Substituting back in for A, C, E and G along with more simplification, yielded the steady state solution for heading error.

$$\overline{\psi} = \frac{W \sin \overline{\theta}}{C_{\alpha r}\left(1 + \frac{b}{a}\right)} \tag{2-33}$$

And with further manipulation:

$$\overline{\psi} = \frac{W_r \sin \overline{\theta}}{C_{\alpha r}} \tag{2-34}$$

The solution for steady state off-tracking error begins using equations 2-30 and 2-34 above. Solving for $\overline{Y}$ this time and substituting right away gives:

$$\overline{Y} = \frac{\left(-\left(\frac{-C_{\alpha f}a + C_{\alpha r}b}{I_{zz}u}\right)u - \left(\frac{C_{\alpha f}a}{I_{zz}}\right)K_{OT}d - \left(\frac{C_{\alpha f}a}{I_{zz}}\right)K_H\right)}{\left(\frac{C_{\alpha f}a}{I_{zz}}\right)K_{OT}}\left(\frac{W \sin \overline{\theta}}{C_{\alpha r}\left(1 + \frac{b}{a}\right)}\right) \tag{2-35}$$

Simplifying and using the equality from equation 2-34:

$$\overline{Y} = \frac{\left[(1 - K_{OT}d - K_H)\frac{W_r}{C_{\alpha r}} - \frac{W_r b}{C_{\alpha f}a}\right]\sin \overline{\theta}}{K_{OT}} \tag{2-36}$$

Using the definition of front and rear normal loads in the following equation:

$$W_f = W_r\left(\frac{b}{a}\right) \tag{2-37}$$

Finally substituting this variable into the $\overline{Y}$ equation and simplifying yields:

$$\overline{Y} = \frac{-\left[\left(K_{OT} d + K_H\right)\frac{W_r}{C_{\alpha r}} - \frac{W_r}{C_{\alpha r}} + \frac{W_f}{C_{\alpha f}}\right]\sin\overline{\theta}}{K_{OT}} \tag{2-38}$$

The understeer gradient, K, which is a factor of the vehicle's ability to steer, is used to finalize the steady state solution (Gillespie, 1992).

$$K = \frac{W_f}{C_{\alpha f}} - \frac{W_r}{C_{\alpha r}} \tag{2-39}$$

$$\overline{Y} = \frac{-\left[\left(K_{OT} d + K_H\right)\frac{W_r}{C_{\alpha r}} + K\right]\sin\overline{\theta}}{K_{OT}} \tag{2-40}$$

Most of the analytical work was already completed for the steady state steering angle. Combining the steady state equations 2-24, 2-35, and 2-36 and simplifying results in the equation of the steering angle, results in the expression:

$$\overline{\delta}_f = K\sin\overline{\theta} \tag{2-41}$$

Which matches the results of Evans (2006).

From this analysis, it was observed that the steady state steering angle was directly related to the slope and the understeer gradient. If the understeer gradient is non-zero, and the vehicle is operating on a side slope, there would always be a steady state non-zero steering angle. This angle was necessary to track the desired path in the steady state. The steady state off-track error ($\overline{Y}$) is the constant distance the vehicle remains from the path after the system has reached steady state. To improve automatic guidance performance, this term must be driven towards zero. However, the expression for $\overline{Y}$ shows that with a steady-state heading error greater than zero, the off-track error can be different, depending on from which part of the vehicle it was measured. This steady-state error was a function of the slope, rear weight, rear cornering stiffness, heading and off-tracking error gains and also the understeer gradient. The steady state heading angle error ($\overline{\psi}$) provides the lateral tire forces at the rear of the vehicle to offset the lateral gravitational force. For a two wheel steer vehicle, this error will always be greater than zero due to unequal generation of lateral forces from front to rear. The steady state heading error was directly related to the weight on the rear of the vehicle, the rear tire cornering stiffness and the slope. This analysis showed that as long as the vehicle was operating on a slope the heading error was non-zero. At this point the vehicle was orientated at an angle to the desired path and different parts of the vehicle would have different off-tracking errors.

2.2.2 Feed-Forward Steady State

One initial question that needed to be answered for the development of a roll angle feed-forward controller was what roll angle feed-forward gain should be used. Selecting the feed-forward gain was non-trivial, however. While the lateral gravitational force component only disturbs the system through the lateral acceleration state equation, the actuation through the steering angle was applied to both lateral acceleration and yaw acceleration equations. Thus a simple cancellation of the disturbance signal with feed-forward control does not work in this situation. The first step in determining the feed-forward gain was to try adding K*sin(θ) to the control law to counteract the additional disturbance force due to gravity. Here steady-state variables are used because of analysis that follows, but for actual application, we will use general variables.

$$\overline{\delta}_f = -K_{OT}\left(\overline{Y} + d\overline{\psi}\right) - K_H\overline{\psi} + K\sin\overline{\theta} \tag{2-42}$$

Substituting the control law in (2-29 and 2-30) results in:

$$0 = A\left(-\overline{\psi}u\right) + E\left(-K_{OT}\left(\overline{Y} + d\overline{\psi}\right) - K_H\overline{\psi} + K\sin\overline{\theta}\right) - g\sin\overline{\theta} \tag{2-43}$$

$$0 = C\left(-\overline{\psi}u\right) + G\left(-K_{OT}\left(\overline{Y} + d\overline{\psi}\right) - K_H\overline{\psi} + K\sin\overline{\theta}\right) \tag{2-44}$$

Further substitutions similar to those done previously and solving for $\overline{Y}$ yields:

$$\overline{Y} = \frac{-\left(Cu + GK_{CON}\right)\overline{\psi} + GK\sin\theta}{GK_{OT}} \tag{2-45}$$

Where:

$$K_{CON} = K_{OT}d + K_H \tag{2-46}$$

Now substituting into 2-43 and simplifying gives the same steady state heading angle as before.

$$\overline{\psi} = \frac{W_r}{C_{\alpha r}}\sin\theta \tag{2-47}$$

For the steady state off-tracking distance, equation 2-44 is solved for $\overline{\psi}$.

$$\overline{\psi} = \frac{GK_{OT}\overline{Y} - GK\sin\theta}{-\left(Cu + GK_{CON}\right)} \tag{2-48}$$

Inserting $\overline{\psi}$ into equation 2-43.

$$\left(Au + EK_{CON}\right)\left[\frac{GK_{OT}\overline{Y} - GK\sin\theta}{-\left(Cu + GK_{CON}\right)}\right] + EK_{OT}\overline{Y} = \left(EK - g\right)\sin\theta \tag{2-49}$$

Once this is simplified and the correct values of A, C, E, G are also inserted, the final off-tracking error equation was found to be:

$$\overline{Y} = \left[\frac{K}{K_{OT}} - \frac{1}{K_{OT}} \left[\left(K_{OT} d + K_H \right) \frac{W_r}{C_{\alpha r}} + K \right] \right] \sin\theta \tag{2-50}$$

The K/K$_{OT}$ terms cancel out which yields:

$$\overline{Y} = \left[-\frac{1}{K_{OT}} \left[\left(K_{OT} d + K_H \right) \frac{W_r}{C_{\alpha r}} \right] \right] \sin\theta \tag{2-51}$$

Now that the off-tracking error is derived, the steady state steering angle can be found based on:

$$\overline{\delta}_f = -K_{OT}\overline{Y} - \left(K_{OT} d + K_H \right)\overline{\psi} + K\sin\theta \tag{2-52}$$

Substituting 2-51 and simplifying gives the steady state steering angle, which ended up being the same steady state steering angle derived in 2-41.

$$\overline{\delta_f} = K\sin\theta \tag{2-53}$$

Like the non-feed-forward control law, this revised control law with roll angle feed forward control did not eliminate steady state errors. The next step then in determining the feed forward gain began with the previous feed forward steady state solution and applied an arbitrary gain C instead of K, the understeer gradient, to $\sin\theta$, resulting in the control law:

$$\overline{\delta}_f = -K_{OT}\left(\left(\overline{Y} + d\overline{\psi} \right) - K_H\overline{\psi} \right) + C\sin\theta \tag{2-54}$$

Since C is a constant, the steady-state off-tracking error in equation 2-50 changes to:

$$\overline{Y} = \left[\frac{C}{K_{OT}} - \frac{1}{K_{OT}} \left[\left(K_{OT} d + K_H \right) \frac{W_r}{C_{\alpha r}} + K \right] \right] \sin\theta \tag{2-55}$$

To determine the value of C that resulted in a zero steady-state off-track error, $\overline{Y}$ was set to zero and then the equation was solved for C, resulting in:

$$C = \left(K_{OT} d + K_H\right)\frac{W_r}{C_{\alpha r}} + K \tag{2-56}$$

This gave a steady state steering angle once substituted back into 2-54.

$$\bar{\delta}_f = \left(\left(K_{OT} d + K_H\right)\frac{W_r}{C_{\alpha r}} + K\right)\sin\theta \tag{2-57}$$

Although the steady state steering angle had changed, the heading angle ($\bar{\psi}$) remained the same. The steering angle represented the final feed-forward term used for the controller. The equation was used to improve the vehicle guidance over the non-feed-forward steering command shown in 2-41. The addition of the off-tracking and heading terms in the feed-forward gain drove the steady state error to zero.

2.3 Model Controller

In section 2.2 above, the steady state equations were derived for the vehicle controller. The design utilized feed-forward control to send adjusted error signals into the model and ideally, eliminating the steady-state errors caused by gravity. In this section, we added integral and derivative off-track error terms. This provided for potentially better control of the system than what was proposed analytically in the previous section. This updated controller is composed of proportional, integral and derivative controllers. These are more commonly referred to as PID controllers. Each gain in the PID controller contributes their own error signal adjustment. All of these signals were based on the off-tracking and heading error signals received from the simulation. In total, there could have been six controller terms; however, previous work by Evans (2006) showed that the integral and derivative heading controllers were best set to zero and thus can be omitted. This leaves the following four controller gains: K_HP, $K_{OT}P$, $K_{OT}I$ and $K_{OT}D$. The following equation describes how the desired signal to the controller was formed using the PID gains where d is the look-ahead distance.

$$\delta_f = K_H P * \psi + K_{OT} P * Y + d * K_{OT} D * \dot{Y} + K_{OT} I * \int Y dt \qquad (2\text{-}58)$$

The controller gain values were determined by using a combination of kinematic and dynamic vehicle models. When designing the controller we knew that the tracking dynamics (Y and ψ) dominate the system's response, and the steering delay would be a secondary effect of the dynamics. Adjustments were made by increasing the integral off-tracking gain ($K_{OT}I$) to reduce the steady-state error, then the derivative gain ($K_{OT}d$) was increased to improve system dampening, however the proportional controller can affect both types of response (Nise, 2004). The kinematic model with proportional feedback control will have the following characteristic equation.

$$s^2 + \left(\frac{v\left(K_H P + K_{OT} D\right)}{L} \right) s + \frac{v^2}{L} K_{OT} = 0 \qquad (2\text{-}59)$$

To design our controller, two equations were used for the damping ratio and natural frequency of the system.

$$\zeta = \frac{K_H P + K_{OT} D}{2\sqrt{K_{OT} P}\sqrt{L}} \qquad (2\text{-}60)$$

$$\omega_n = \frac{v}{\sqrt{L}} \sqrt{K_{OT} P} \qquad (2\text{-}61)$$

Where ζ is the damping ratio and ω_n is the natural frequency (Steward, 2006).

Next we chose our desired distance to peak, d_p, for the second order response. For this, 2 meters was chosen. The time to peak, t_p, could then be related to d_p through equation 2-62.

$$t_p = \frac{d_p}{V} \tag{2-62}$$

The next step was to then choose a desired damping ratio, based on an average value with limited overshoot, but not overly damped, for which ζ was set to 0.5. Using a plot of standard unit step response curves for second order equations found in any standard dynamic systems textbook, the value of ω_n could be found based upon t_p (Shearer, et al., 1997). Using equation 2-61 we were then able to solve for K_{OT}. Once K_{OT} was determined, equation 2-60 was then used to find $K_{H,}$ with the selected value for ζ .

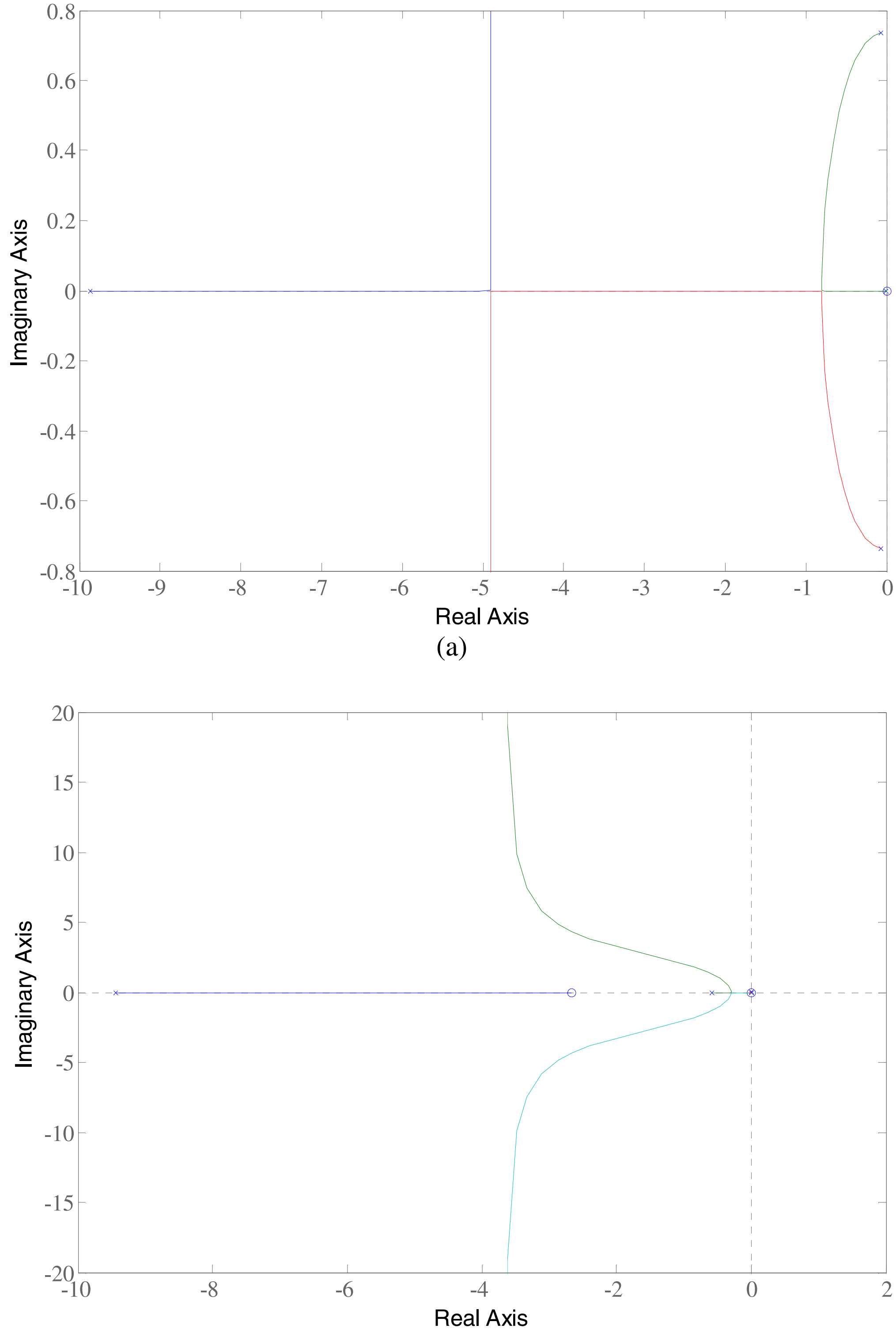
A kinematic model was used first to obtain generalized root locus plots of the gains. The above analysis uses the second order model as a starting point. We started with a second order model because it is actually tractable to calculate gains.

$$s^3 + \left(\frac{K_H P + K_{OT} D \cdot v}{K_H d + L/v} \right) s^2 + \left(\frac{K_{OT} P \cdot v}{K_H d + L/v} \right) s + \frac{K_{OT} I \cdot v}{K_H d + L/v} = 0 \tag{2-63}$$

Generalized root locus show the response of one variable on a system changing while the rest are held constant (Nise, 2004). A generalized root locus is a plot of the possible closed loop pole locations with changes in individual gains. Depending on their location, closed loop poles may be stable, marginally stable or unstable. These plots allowed gains to be found that provided the best settling times, and least amount of overshoot for each individual controller gain.

Figures 2.4 and 2.5 below show the generalized root-locus for each of the gain variables. For all controller gains, except for one, the systems are always stable. K_{OT}I is only in the stable region briefly, before going unstable when K_{OT}I is about 0.0825. Thus it required the selection of a very small value from a limited range for this term. For the K_HP term we found a number of values that would work well based on the root locus, but when used in the dynamic model, a value of 0.4 was found to work the best. Both $K_{OT}P$ and $K_{OT}D$ have a zero and two poles right at the origin. Although the location of their other poles and zeros are different, when running them in the dynamic model they ended up utilizing the same value to gain favorable results.

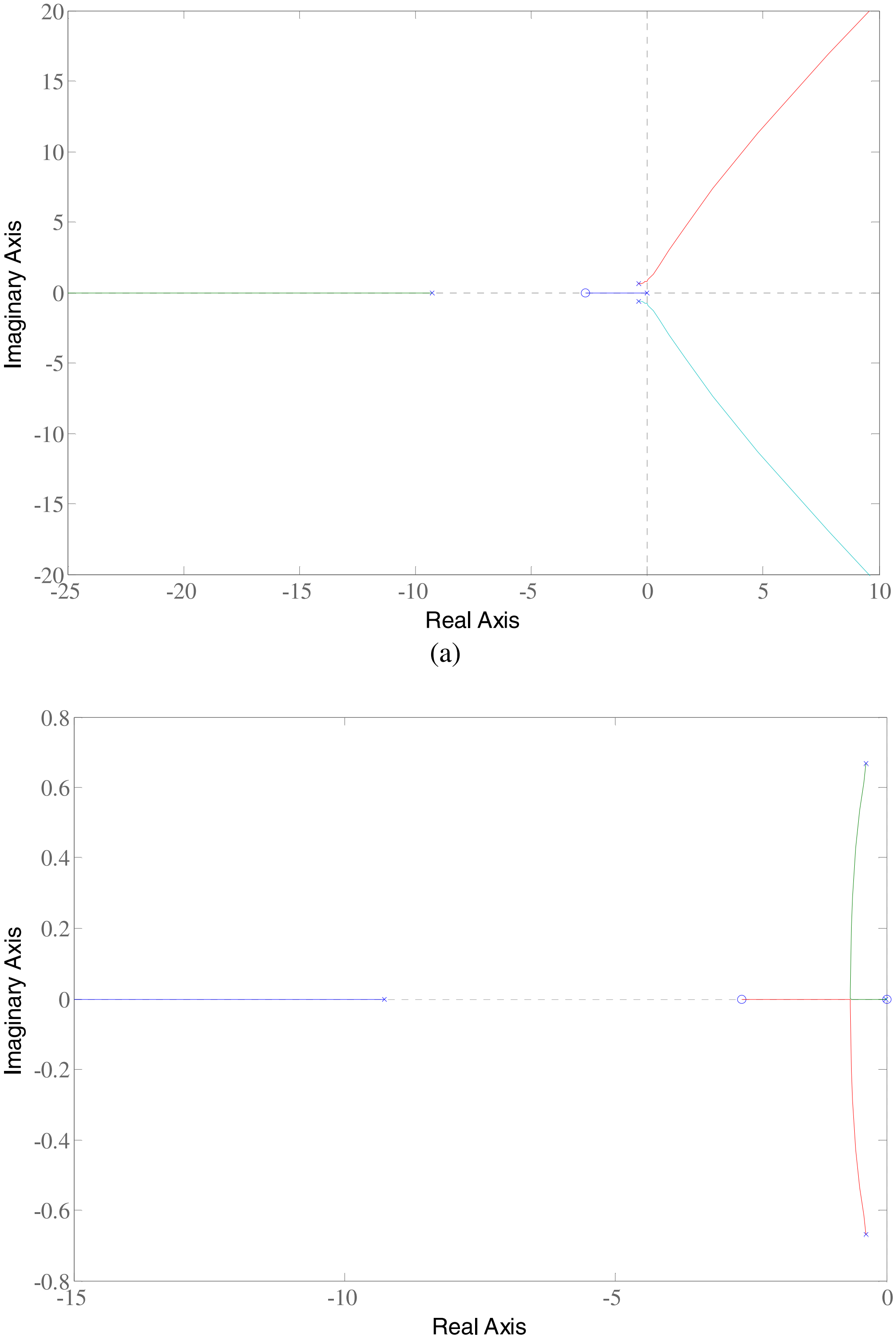
The next step involved testing the gains in the dynamic model to check the system response to the controller. A system response was desired that would have the fastest damping time without sacrificing excessive system overshoot. A step function returned the system response for four different velocities in graphical form (Figure 2.6). One of the first things evident from this graph was the strong correspondence between velocity and amplitude. A higher velocity led to higher peak amplitude, which would correlate to a maximum off-tracking distance. Due to the higher speeds do causing greater overshoot, they also take longer distances to settle. Except for the fastest speed, it appeared all the responses get to their steady state values around 60 meters. From these observations it is obvious to see that vehicle velocity dominates the response.



(a)

(b)

Figure 2.4 Root Loci varying each gain value used in the controller. a) K_HP varying with $K_{OT}P = 0.1$, $K_{OT}I = 0.0001$ and $K_{OT}D = 0.001$ b) $K_{OT}P$ varying with $K_HP = 0.4$, $K_{OT}I = 0.0001$ and $K_{OT}D = 0.001$.



(a)

(b)

Figure 2.5 Root Loci varying each gain value used in the controller. a) $K_{OT}I$ varying with $K_HP = 0.4$, $K_{OT}P = 0.01$ and $K_{OT}D = 0.001$ b) $K_{OT}D$ = varying with $K_HP = 0.4$, $K_{OT}P = 0.01$ and $K_{OT}I = 0.0001$.

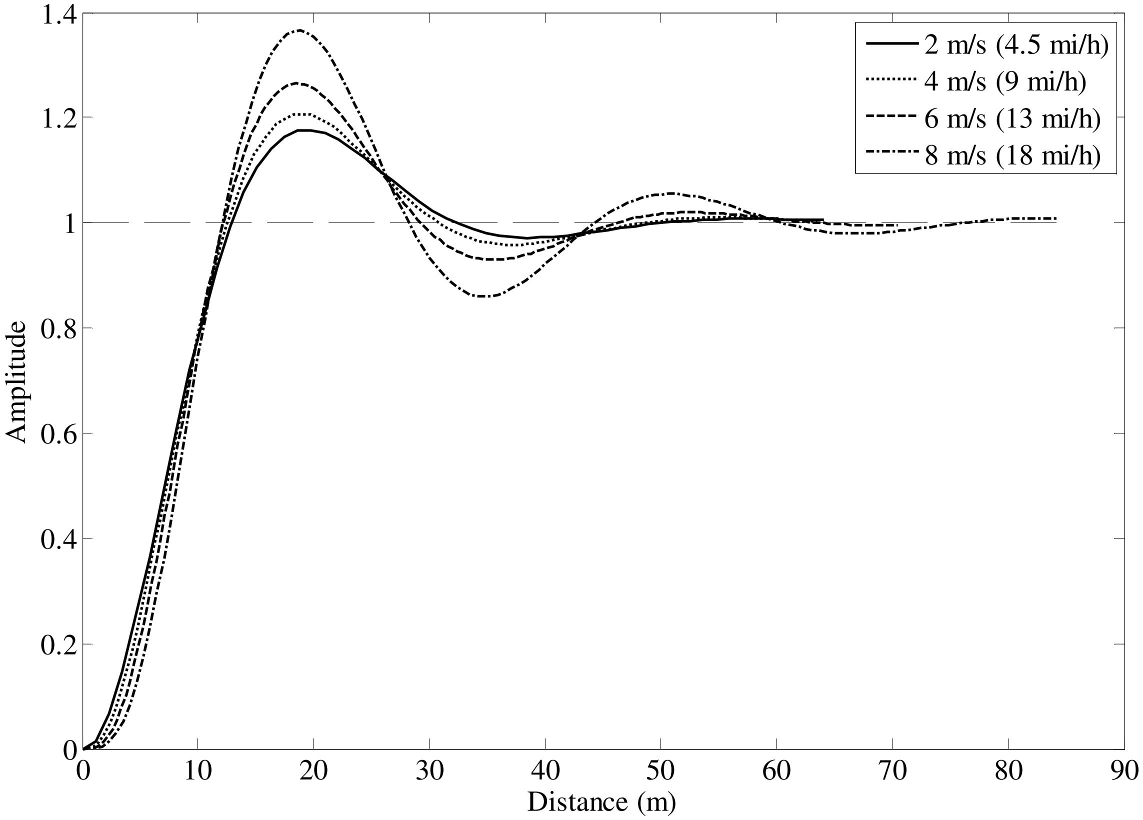


Figure 2.6 Comparisons in the step response to dynamic vehicle model using final gain values from Table 2.2 at four different velocities.

Table 2.2 Final Proportional, Integral, and Derivative controller gain values, where OT terms are for off-tracking error and H terms are heading error.

Gain	Numerical Value
$K_{OT}P$	0.1
$K_{OT}I$	0.0001
$K_{OT}D$	0.001
$K_{H}P$	0.4
$K_{H}I$	0
$K_{H}D$	0

Utilizing the root locus and system response simulations, final values for the gains were determined, which are highlighted in Table 2.2. These gain values were substantially lower than those derived by Evans (2006), even though both had similar applications. One factor that was likely to cause some differences between this work and Evan's was that our vehicle model included steering system dynamics. The steering system was modeled as a first order delay with a 0.1 second time constant. There were also saturation limits on the steering angle rate and the overall steering angle. These limits were only applied to the model used in the vehicle simulations, and not in the analytical work. An additional, possibly substantial, difference between Evans work and that presented in this thesis was that his work was based on a combine with rear steering which weighed more and had substantially different vehicle parameters.

2.4 Modeling Tire-Soil Interactions

A major factor in any vehicle modeling simulation is the interaction between the vehicle and the soil surface. Vehicle modeling has been around for a long time; however, there is still not a consensus on how to best model tire-soil interaction as related to slip angle and lateral tire force in the agricultural field. The main problem is the high uncertainty in the lateral force relationship between the tire and the soil.

The difference between the actual direction of travel and the tire angle is called the slip angle (α). The lateral force (F_y) developed on the tire, causes the vehicle to turn and is heavily influenced by the tire to soil interaction. This issue was magnified when operating on hillsides. This was a result of gravity causing additional slippage and thus the vehicle was less apt to go in the direction the tires are pointed. The force on the tires in the y direction was calculated based on this slip angle, multiplied by the cornering stiffness ($C_{\alpha f}$) of the tire. The cornering stiffness was defined by:

$$C_{\alpha} \equiv \left. \frac{dF_y}{d\alpha} \right|_{\alpha=0} \tag{2-64}$$

In the automobile industry, this interaction is well known and documented. This is due to roads having a known coefficient of friction between the surface and the tire. In addition, for automobiles, the road surface does not deform compared to off-road applications where soil deformation occurs as a vehicle tire passes over it. Furthermore, tire design is drastically different for agricultural operations. Both on and off-road tires are deigned for their own particular environments. The type of contact patch each tire type has for its own environment is completely different. In some scenarios, such as a car traveling at low speeds on dry pavement, it is acceptable to assume that the direction the tires are pointed is also the same as the velocity vector (Bernard, 2005). Although tractors travel at low speeds when automatic guidance systems are engaged, a simple kinematic model that does not include lateral wheel slip for the direction of travel would not be robust enough for neither the higher speeds nor the slopes encountered during these simulations. Baack (2003), and others, running off-road equipment simulations were able to avoid some of these issues by assuming that there was no tire slip.

For the automobile industry, extensive data is collected by the tire manufacturers to obtain cornering stiffness values to be used in simulations. Values are displayed in what is called a carpet plot (Figure 2.7), which is used to calculate the cornering stiffness of tires.

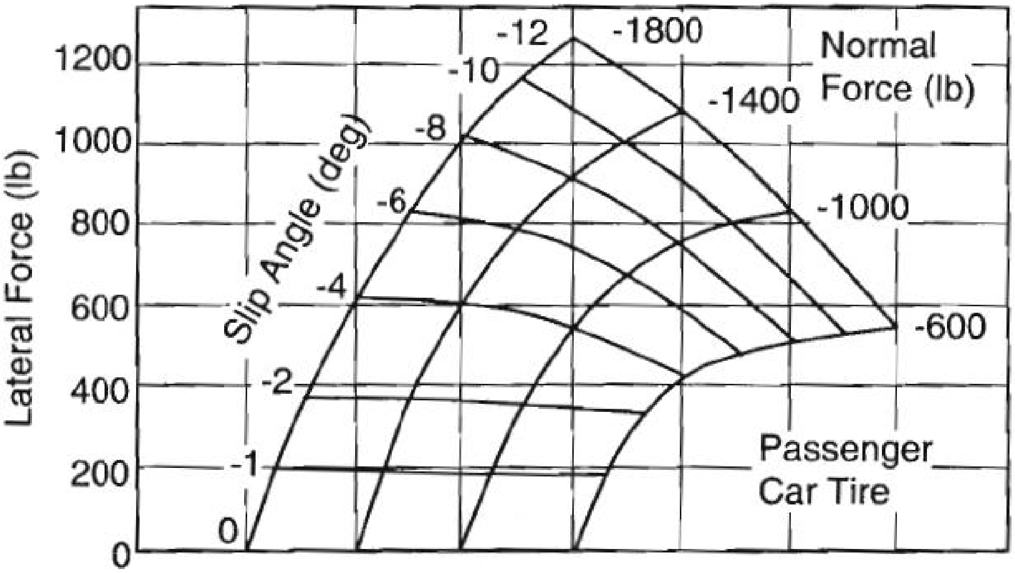


Figure 2.7 Carpet Plot for a passenger car tire (Gillespie 1992).

For the agricultural tire industry, carpet plots were not readily available. Instead, a cornering coefficient was used. The equation for this value was based upon the normal load and cornering stiffness for any particular tire on the vehicle.

$$c_c = \frac{C_\alpha}{F_v} \qquad (2\text{-}65)$$

From this equation the cornering stiffness was determined for the tires and used in the vehicle model.

When engaged in automatic guidance, tractors are not operating on dry pavement, but rather on soil whose physical properties are affected by many different variables such as soil shear strength, moisture content and vegetation to name a few. Several different tire models have been developed relating tire slip angle and the vertical tire force to the tire lateral force. These models are the linear model, the bi-linear model, the polynomial model, a Pacejka modified sinusoidal model and an exponential model. The linear model applies a linear lateral load until tire saturation, the bi-linear model is representative of the cornering force curve with two linear regions, the exponential model follows the relationship of shear stress to soil deformation and the Pacejka model fits tire data to a modified sine function. The linear model was not used since it only worked well for low speeds and had no vertical load effect. A bi-linear model was disregarded for the same issues with vertical load, but also was inaccurate compared to others as the lateral force reached the maximum value. The polynomial model was more accurate in representing tire functionality than the previous two models, but it did not include a camber effect nor did the coefficients relate to any real values. A Pacejka modified sinusoidal model seemed to have all the benefits the others lacked, however it required extensive data for curve fitting (Metz, 1993). Thus Metz settled on an exponential model that didn't have a camber effect, but included vertical load effects as well as the coefficients corresponding to real data values. This exponential tire model was represented mathematically by:

$$LFC \equiv \frac{F_y}{F_V} = A\left[1 - e^{-B\alpha}\right] \qquad \text{(2-66)}$$

Where LFC is the lateral force coefficient, F_y is the lateral tire force,

F_V is the normal tire force on an individual tire and α is the tire side slip angle. A and B are exponential tire model parameters and are estimated from empirical data.

B changes as a function of normal tire force according to the expression:

$$B = \left[C / A\right]\left[F_{VREF} / F_V\right]^m + D / A \qquad \text{(2-67)}$$

Where C, D, and m are model parameters estimated from empirical data, and

F_{VREF} is a reference normal tire force.

Based on the definition of the cornering coefficient in equation 2-65 and that of cornering stiffness in 2-64, it can be shown that the cornering stiffness is:

C_{α} = ABF$_V$ (2-68)

Since

$$c_c = \frac{C_{\alpha f}}{W_f} = \frac{\left.\frac{dF_L}{d\alpha}\right|_{\alpha=0}}{W_f} = \left.\frac{dLFC}{d\alpha}\right|_{\alpha=0} = \frac{d}{d\alpha}\left\{A\left[1-e^{-B\alpha}\right]\right\}\Big|_{\alpha=0} = ABe^{-B\alpha}\Big|_{\alpha=0} = AB \qquad (2\text{-}69)$$

Metz also presented a list of cornering coefficients ranging from low plowed field values up to high meadow values based on off-road tires. The type of soil, moisture content and ground coverings all create different situations in which a tractor controller must be able to operate (Metz, 1993).

Using normal forces on each of the tires for the agricultural tractor being modeled (model 8320, John Deere, Moline, IL.), cornering stiffness was calculated for individual tires using the exponential tire model parameters found from Metz and were based on the work of Swanghart and Rott (1984). Swanghart and Rott (1984) experimentally determined the cornering stiffness of tires for different, broad classes of soil/field conditions. Cornering stiffness was determined by pulling a rolling tire in a straight line at particular slip angle and measuring the force developed, cornering stiffness values were created for different types of tires in different types of soil (Swanghart, et al., 1984). From Metz's model the cornering stiffness were used for a single tire with the static normal forces on the individual tires for the John Deere 8320 tractor in a plowed field, cornfield and in a meadow. These values were then multiplied by two for front tires and by four for rear tires in a dual rear wheel configuration.

CHAPTER 3. EXPERIMENTAL DESIGN

3.1 Methods

This chapter discusses the methodology used to investigate controller performance. All simulations were based on mathematical models executed using both MATLAB and SIMULINK (ver. 7, Mathworks, Natick, MA). There were four main steps in the methodology: (1) designing the gains for the roll-angle feed-forward controller for the automatic guidance simulation, (2) controller simulation as it responded to straight paths, both step and sinusoidal test slope profiles, (3) investigation of the effects of different vehicle velocities and changes in cornering stiffness from the nominal values used to design the feed-forward controller gains and (4) determining if feeding-forward expected roll angle at some look-ahead distance from the current vehicle location could improve the controller performance. The purpose of each of these steps was to assess the potential of roll-angle feed-forward control in improving automatic guidance controller performance on side-slopes under varying operating conditions.

A John Deere 8320 tractor with front wheel assist and dual rear wheels (Deere and Co., Moline, IL) was used in the simulations and was modeled as a SIMULINK block diagram. The difference in the model used for the simulations and that described in Chapter 2 was that it also included steering angle saturation and steering rate saturation. The controller still functioned as the analytical model, but the saturation values limited the steering angle and rate so they would not be outside the realistic limits for an off-road vehicle. The controller was then implemented as a discrete time controller in a MATLAB function with a 0.2 second sampling time. The SIMULINK model was called to simulate the vehicle dynamics over 0.2 time horizons. At the end of each time horizon, the vehicle states were used to calculate the next controller output and were used as the initial conditions for simulating vehicle dynamics over the next 0.2 time horizon. The MATLAB code for these simulations is located in Appendix E and the SIMULINK block diagram is shown in Appendix F.

3.1.1 Feed-Forward and Non-Feed-Forward Controller Comparisons

To meet the first and fourth research objectives, and to assess improvements in vehicle tracking under automatic guidance with a feed-forward controller four different controller simulations were performed over two prototypical test slope profiles which are described further below. These tests were: (1) a navigation controller test with no feed-forward control, (2) a navigation controller test with roll-angle feed-forward control using the roll angle at the current vehicle location, (3) a navigation controller test with estimated roll angle feed-forward control at 2 meters ahead of the current vehicle location and at (4) 10 meters ahead of the current vehicle location. The control law for the steering controllers used in these tests was equation 2-58.

From each simulation, the off-track errors were calculated and stored at each 0.2 second interval over the duration of the simulation. Off-track error was defined as the perpendicular distance of a point 1.83 m in front of the center of the rear vehicle axle from the A-B line that was being tracked. Based on the off-track error vector for each simulation case, two performance metrics were calculated: the maximum off-tracking distance and the off-tracking percentage. The maximum off-tracking distance was the largest magnitude off-tracking value observed during the simulation. In the field, this metric represents the single largest overlapping or gap distance between paths. The off-track percentage was the percentage of the total path length where the vehicle experienced an off-track distance greater than 0.025 meters (2.5 cm). The percentage was calculated by dividing the number of off-track error values along the path greater than 0.025 meters by the total number of error values calculated along the path. The off-track percentage provided an indication of the proportion of the off-track errors that were above an acceptable level. According to John Deere, the highest StarFire RTK GPS receiver accuracy was sub-inch, or less than 2.54 cm (DE, 2006). This GPS position measurement mode is used in the most demanding automatic guidance situations, such as laying drip tape and control traffic farming, which also require the most precise automatic guidance controller performance. While this research was estimating automatic guidance tracking performance, using the sub-inch threshold to calculate this performance metric made sense and would indicate a level of performance that ensures the navigation controller performance would not adversely affect the accuracy of even the most precise systems.

3.1.2 Test Slope Profiles

During simulation, two different test slope profiles were used. The first test slope profile started with a horizontal plane. Then after 100 meters in the direction of travel of the tractor, the slope underwent a step change to a positive, or counter clockwise about the longitudinal vector of the vehicle body pointed in the forward travel direction, five degree slope (Figure 3.1). The slope change resulted in a vehicle roll angle step that caused a gravitational lateral force component to act on the vehicle towards the left hand side of the vehicle. After 200 meters along the vehicle travel direction, the slope profile returned to horizontal through another step change in slope. Simulation continued along this slope profile for another 150 meters.

The stepped slope profile enabled simulation of controller step responses and enabled observation of standard time response parameters such as rise time and percent overshoot for the system. The stepped slope change of five degrees was based on that value being an upper threshold for slopes found in many agricultural areas. In addition, this value provided enough change to observe the effect of side-slope on automatic guidance controller performance, but was not too large to demand that the controller to perform well under unrealistic slope conditions. Although not impossible in nature, it was not expected that this stepped slope would ever be encountered in reality by an operator running an automatic guidance system in their vehicle. Nonetheless, this profile provided important insight into controller performance.

The second slope profile started with a zero degree slope, but then immediately picked up a sinusoidal slope profile. The amplitude of this slope was five degrees, similar to the step slope, and had a period of 200 meters. As in the step slope simulation, the slope returned to zero degrees after 200 meters. At this point the simulation continued at zero degrees for another 150 meters.

In addition to the step slope, the sinusoidal profile was chosen for a number of reasons. Not only is a sine wave a common mathematical function, but it also represented a situation that would be more likely to be found under normal operating conditions. The smooth transitions generated by the sine wave also provide a more realistic situation that would be expected in the field, rather than a sharp slope change found in the step simulation. The sinusoidal slope also allowed the system response to be shown over both a positive and a negative slope profile. This sign change in the slope allowed the controller to work with the lateral force being exerted from two different sides (Figure 3.1). In the transition between the positive and negative slopes, the look-ahead simulation contains a two meter section, where the vehicle is experiencing a positive roll angle while the controller is looking ahead and processing a negative roll angle. This transition period is important to ensure that changes in slope sign do not cause excessive errors.

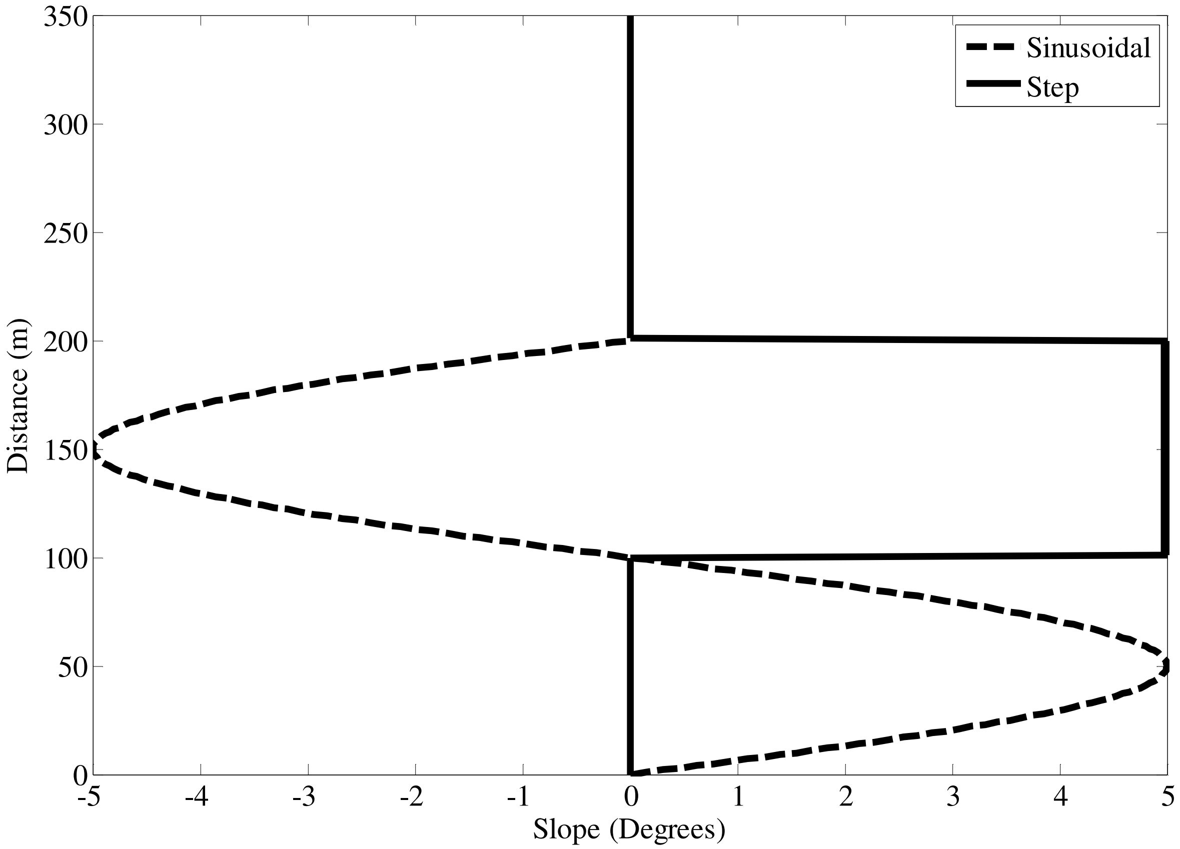


Figure 3.1 Slope profile for step and sinusoidal slopes.

3.1.3 Vehicle Velocity

For each controller-type and test slope profile, simulations were run at four vehicle velocities, 2, 4, 6 and 8 meters per second (4.5, 9, 13.4 and 18 miles per hour). This set of velocities covered most of the expected working range of the vehicle under automatic guidance control plus some additional higher velocities. Most current automatic guidance systems on the market today disable themselves at speeds near or above 12 miles per hour, thereby creating a limit to the vehicle velocity where the controller would be expected to operate (Deere & Company, 2006). However, new applications are constantly being developed, and it would not be unlikely for automatic guidance systems to increase their upper speed limit in the future.

Automatic guidance must produce an acceptable level of performance over a range of vehicle velocities. As speed increases, the controller has less time to react, and depending on the sampling rate, it may not be able to control the vehicle. At each sampling point in time the controller is receiving a measurement of vehicle location and heading, calculates off-track and heading errors based on the A-B line, and outputs a command to the steering controller based on the control law. At the higher vehicle speeds, the steering dynamics may lead to the steering angle lagging behind the steering angle command. In addition, larger lateral steering forces are required to accelerate the vehicle laterally and to rotate the vehicle about the yaw axis. These factors can lead to the vehicle exhibiting greater overshoot and larger magnitude errors at higher speeds. Evans (2006) showed initially that at higher speeds the system response improved; however, eventually these speed increases reached a threshold and caused the response to become more underdamped.

3.1.4 Changes in Cornering Stiffness Compared to Nominal Values

Because of uncertainly in tire-soil interaction and variation across changes in field conditions, it was also important to investigate controller performance as the tire cornering stiffness coefficients change relative to those used in the design of the roll angle feed-forward controller gain. The roll-angle feed-forward gain, C, was calculated using equation 2-56 based on the nominal cornering stiffness coefficient for the tires supporting the weight of the tractor on a maize field: $C_{\alpha f}$ = 373432 N/rad and $C_{\alpha r}$ = 633421 N/rad (Swanghart, et al., 1984) and was fixed as a constant in the controller based on the current design. It was important to explore what happens as the actual cornering stiffness changes due to soil/field conditions while C is fixed in the controller. Ideally, the gain selected would be very robust and work well for all applications. By analyzing different cornering stiffness coefficient values relative to the nominal value used to fix C, the effects of changes in soil conditions on controller performance can be studied.

For this comparison, three simulations were run. Each set had a different cornering stiffness based on Metz's data from a plowed field, a corn field and a meadow. For each field condition, simulations were run using the sinusoidal path and roll angle feed-forward control, but without look-ahead capabilities. These simulations were accomplished by substituting the cornering stiffnesses from a plowed field and a meadow into the vehicle model. The additional values represented the low and high ends, respectively, of the cornering stiffness spectrum based on Swanghart's data (1984) that an off-road tire would likely encounter in an automatic guidance situation.

3.1.5 DEM Look-Ahead

The Digital Elevation Model (DEM) look-ahead function is based upon the idea of having prior knowledge of the terrain over which the vehicle is traveling. Using elevation data taken from an onboard DEM, the controller would then use this data as part of the function along with feed-forward control. The idea was based on using the current location and heading, from which upcoming roll angles could be estimated at expected future locations. The motivation behind this strategy was that highly accurate DEMs are becoming available and could be loaded into vehicles that will be used in the field. Our hypothesis was that feeding-forward estimated upcoming roll angles would result in improved performance.

When the look-ahead command was activated, the slope at the vehicle's current location was not used for estimating the roll angle for feed-forward control. Instead, the controller looked-ahead along the desired path and estimated the slope it would encounter in the near future. In this case, the controller looked ahead into the sinusoidal slope function and pulled out the slope based on one meter segments of the curve. For testing purposes, both 2 and 10 meter look-ahead distances were selected. With a 10 m look-ahead distance, and our average tested speed of 5 m/s, the controller will be looking for a different slope two seconds before it arrives at the location. The 2 meter distance was selected since it is the peak distance, d_p, used to design the controller gains. For the simulations, real DEMs were not used, instead roll angles were found in the test slope profile at the look-ahead distance in front of the current location.

CHAPTER 4. RESULTS

4.1 Changes in Feed-Forward and Look-Ahead Control

Two factors were under investigation that had the potential to have the greatest effect on vehicle control. These factors were the following: (1) the addition of roll angle feed-forward control as compared with non-feed-forward control and (2) the use of estimated roll angles at specific look-ahead distances from the current vehicle location as compared with the use of roll angles at the current location. From the roll angle feed-forward control comparison, the results showed a substantial advantage with the use of feed-forward control. The simulations also showed that the addition of the look-ahead function, when combined with roll angle feed-forward control, proved to further improve upon the control of the vehicle. Sections 4.1.1 and 4.1.2 detail the results of these simulations. Throughout the remainder of this chapter, all tables will use the nomenclature of FF for roll angle feed-forward control, and LA for roll angle feed-forward control using estimates of the roll angles at look-ahead distances ahead of the current vehicle location. The number in parenthesis after the label, LA, represents the look-ahead distance used in any particular function.

4.1.1 Feed-Forward Control

Our hypothesis was and our results confirmed that the addition of a feed-forward input to the controller would have the greatest impact on automatic guidance performance on hill side-slopes. For clarity, this section will only discuss the differences between the non-feed-forward and roll angle feed-forward simulations.

In the sinusoidal slope profile simulations, the controller with roll angle feed-forward control performed well with a maximum off-track error ranging from 0.010 m to 0.013 m with the maximum off-track error occurring at a speed of 8 m/s. When the controller without roll angle feed-forward was simulated, the maximum off-track error was almost six times larger at 0.076 m for the lower three vehicle velocities and 0.077 m at the top speed (Table 4.1). The controller without feed-forward control was only able to obtain 44.40 percent path accuracy; however, the controller simulated using feed-forward control was able to produce results with zero percent of the path beyond the 0.025 meter boundary (Figure 4.1 and 4.2). For the accompanying figures in this section, there will be two error boundary lines at +/- 0.025 meters, which represent our prescribed off-tracking distance tolerances. In the metrics any values outside these lines contributed to the off-tracking error percentage.

Table 4.1 Performance metrics calculated from simulations with the sinusoidal slope profiles using controller models with and without roll angle feed-forward control.

Speed	Maximum Off-tracking Distance (m)		Off-Tracking Percentage	
(m/s)	No FF	FF, No LA	No FF	FF, No LA
2	0.076	0.010	44.40	0.00
4	0.076	0.011	44.46	0.00
6	0.076	0.011	44.50	0.00
8	0.077	0.013	44.52	0.00

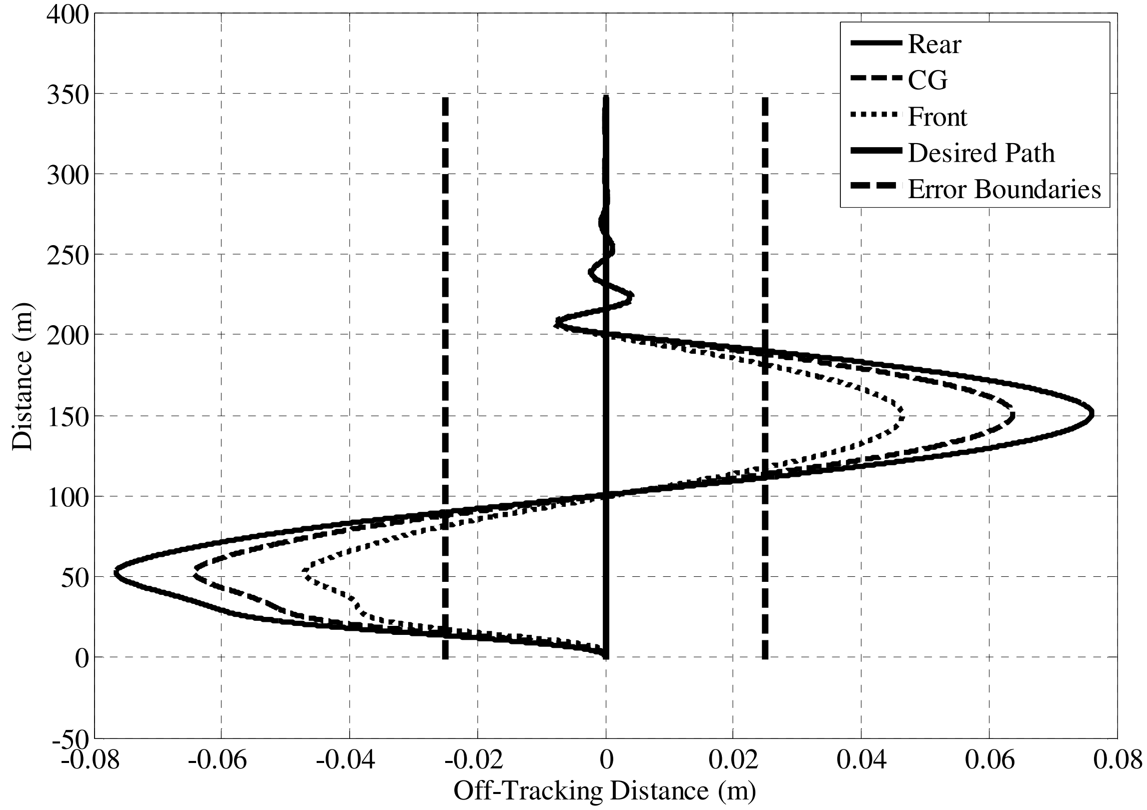


Figure 4.1 Trajectories of the tractor front axle center, vehicle center of gravity, and rear axle center for sinusoidal slope profile at eight m/s for automatic guidance navigation controller tracking the x = 0 line without feed-forward control.

A substantial difference in the performance metrics was observed from the simulations of automatic guidance navigation controllers with and without roll angle feed-forward control with estimates of the roll angle at the vehicle location (Table 4.1) In every simulation, the performance metrics, where feed-forward control was used, were substantially less than those without feed-forward control. These results were consistently observed across all speed ranges and in both test slope profiles. In the sinusoidal slope profiles, the off-tracking distance differences between the feed-forward and non-feed-forward simulations were similar for slow speeds and high speeds, with the average off-tracking distance difference being 0.065 meters.

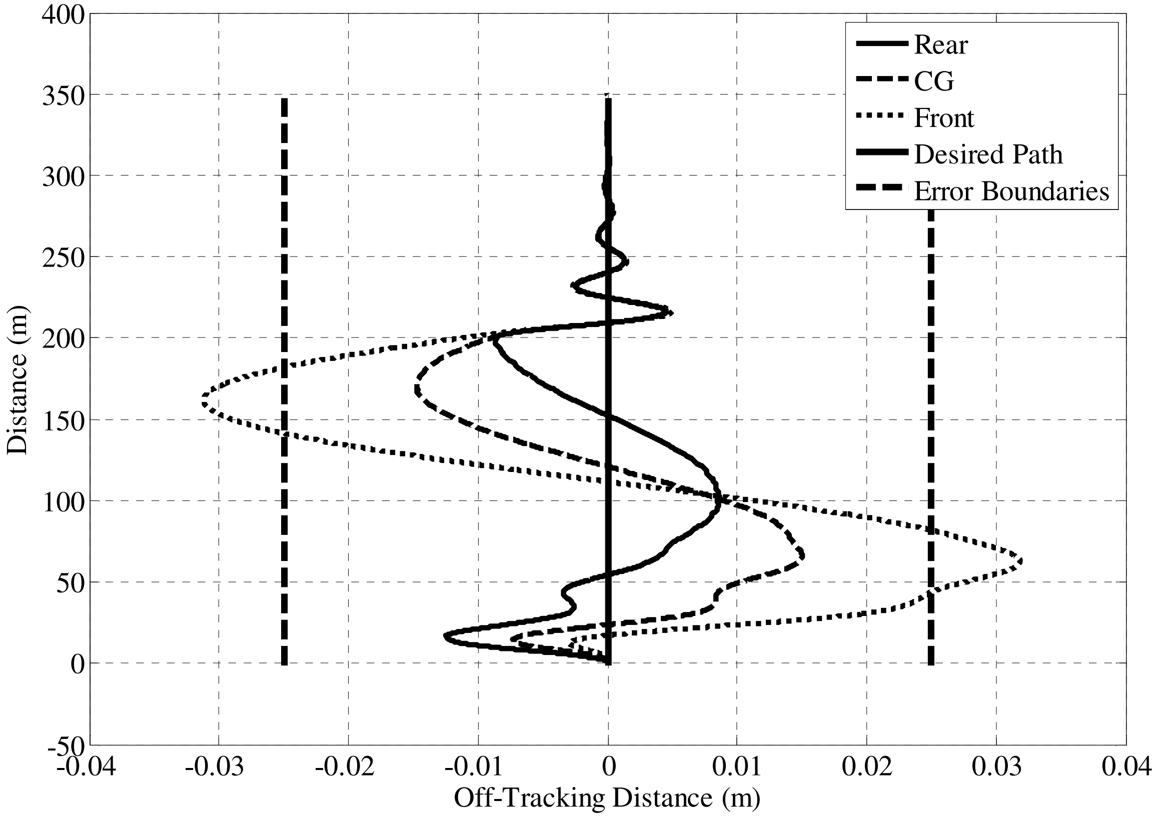


Figure 4.2 Trajectories of the tractor front axle center, vehicle center of gravity, and rear axle center for sinusoidal slope profile at eight m/s for automatic guidance navigation controller tracking the x = 0 line with roll angle feed-forward control.

In the step slope profile simulations, the largest improvement in performance based on the difference in maximum off-track error comparing feed-forward with non-feed-forward control schemes was 0.069 m and occurred at 8 m/s. Similarly, when looking at the off-tracking percentage, the biggest difference between the two controllers also was at the highest velocity. With roll angle feed-forward control, the step slope simulation was able to reduce the off-tracking distance to values between 0.031 and 0.042 meters (Table 4.2). At the same time, the off-tracking error percentage was held to an average value of less than 4.5%. These values were a large improvement over those of the non-feed-forward control which averaged an off-tracking error percentage of just under 30%.

In both step slope profile simulations, the results were noticeably more underdamped than that of the sinusoidal slope profiles as the step input excited the dominant second order dynamic, which from the plots had a damping ratio of 0.0368 at 2 m/s and 0.0374 at 8 m/s. These dominant dynamics are consistent with the simulation results in Section 2.3. Although the performance metrics for the feed-forward and non-feed-forward responses differ greatly, both types of vehicle control show similar settling distances of just over 300 meters (Figures 4.3 and 4.4).

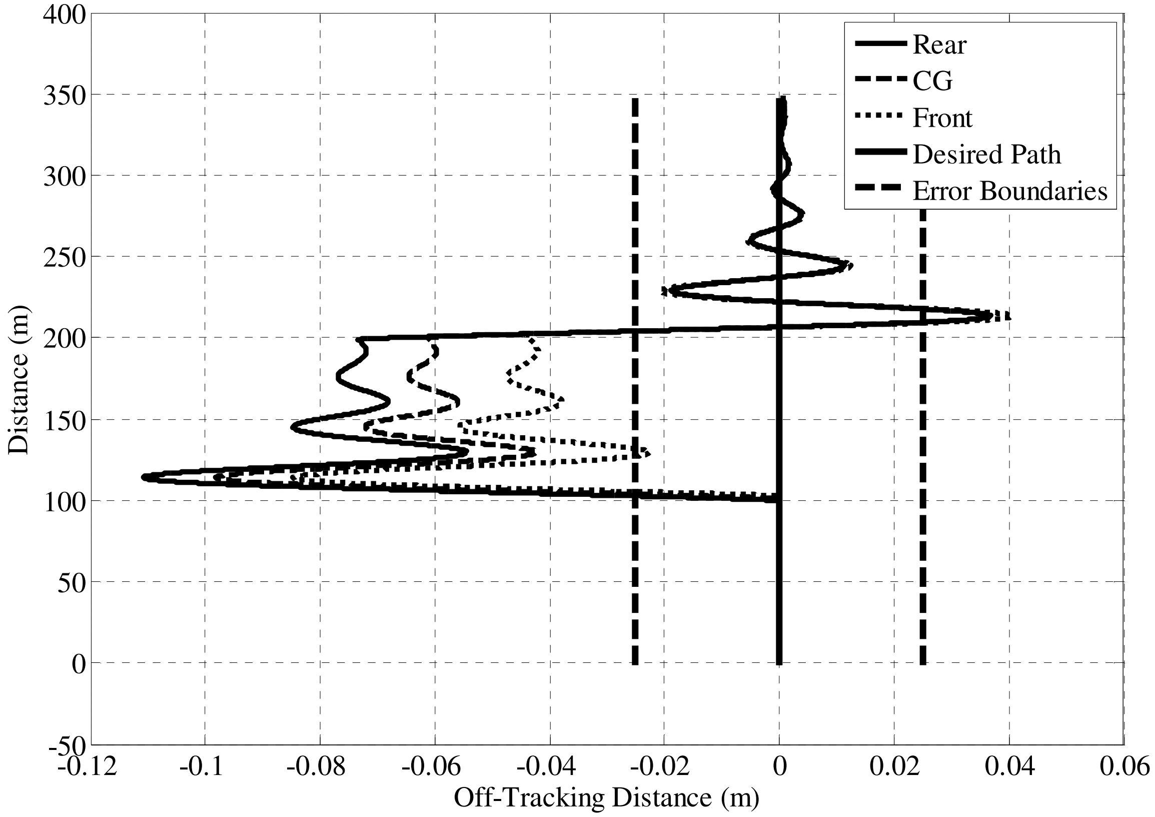


Figure 4.3 Trajectories at the center of the vehicle's front and rear axles, as well as at the center of gravity for the step slope profiles for eight m/s with the controller tracking a path at zero m over a distance of 350 m without feed-forward control.

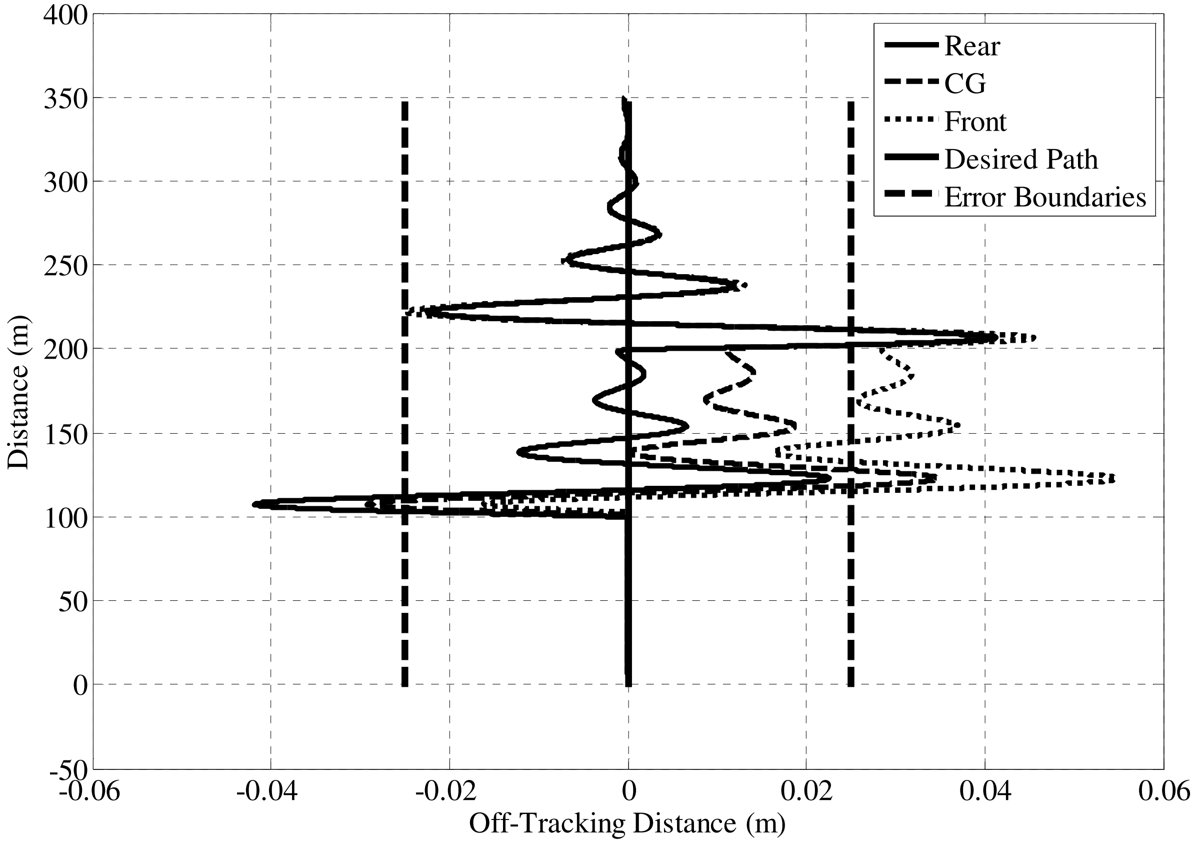


Figure 4.4 Trajectories at the center of the vehicle's front and rear axles, as well as at the center of gravity for the step slope profiles for eight m/s with the controller tracking a path at zero meters over a distance of 350 m with roll angle feed-forward control.

Table 4.2 Performance metrics calculated from simulations with the step slope profiles using controller models with and without roll angle feed-forward control.

Speed	Maximum Off-tracking Distance (m)		Off-Tracking Percentage	
(m/s)	No FF	FF, No LA	No FF	FF, No LA
2	0.093	0.031	28.68	3.85
4	0.096	0.034	28.80	4.22
6	0.102	0.037	29.98	4.66
8	0.111	0.042	31.34	5.18

It is important to observe the location of the front and rear of the vehicle, as indicated by the trajectories of the front and rear axle center locations, for each vehicle controller case and the effect that these locations had on the simulation responses. For the non-feed-forward control simulations, the rear of the vehicle followed the front, but the rear of the vehicle had a greater deviation from the line being tracked than the front in the downhill direction of the slope (Figure 4.5).

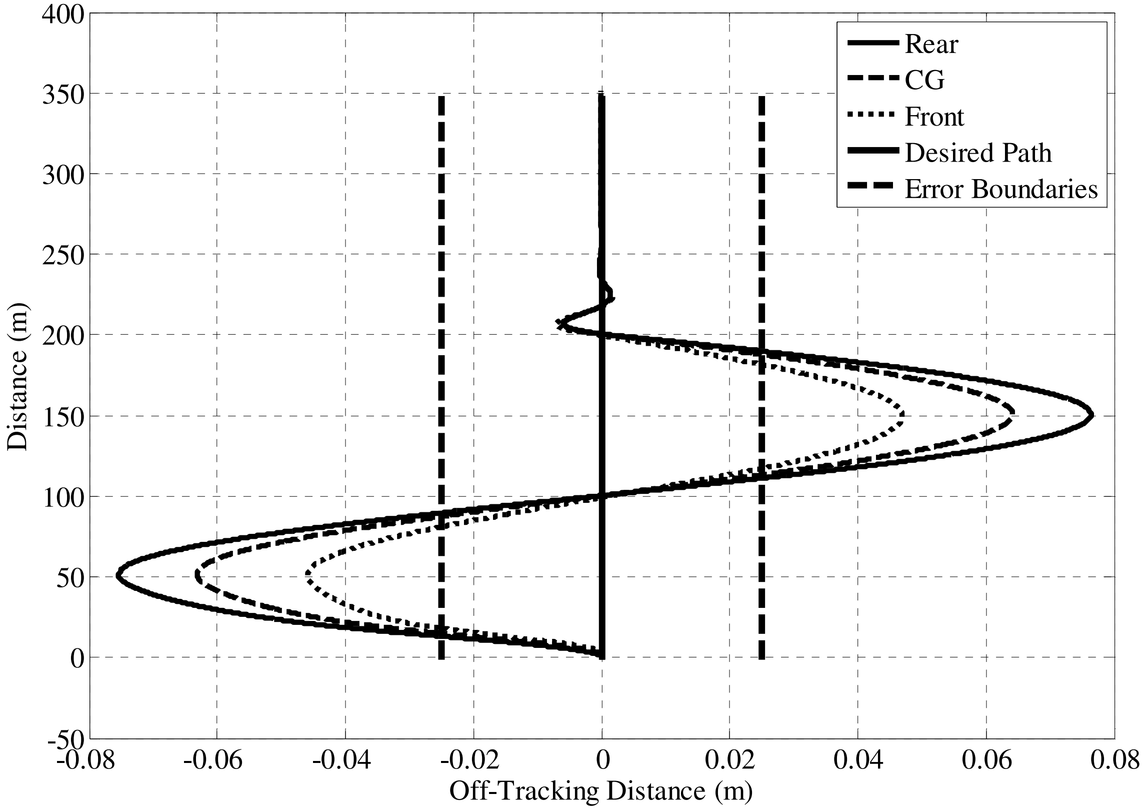


Figure 4.5 Rear axle off-tracking distance relationship to the position of the front axle for non-feed-forward using the sinusoidal slope profile at a speed of four m/s.

In the corresponding roll angle feed-forward control simulation case, the rear traced a trajectory that was still downhill relative to the front, but was a smaller deviation from the line being tracked because of the additional steering correction component from the feed-forward term, which shifted the entire vehicle up the slope (Figure 4.6). These observations were consistent with the analytical results for both non-feed forward and feed forward controller schemes (Eqn. 2-33 and 2-46) indicating that the vehicle yaw will increase (clockwise looking down on the top of the vehicle) for positive roll angles.

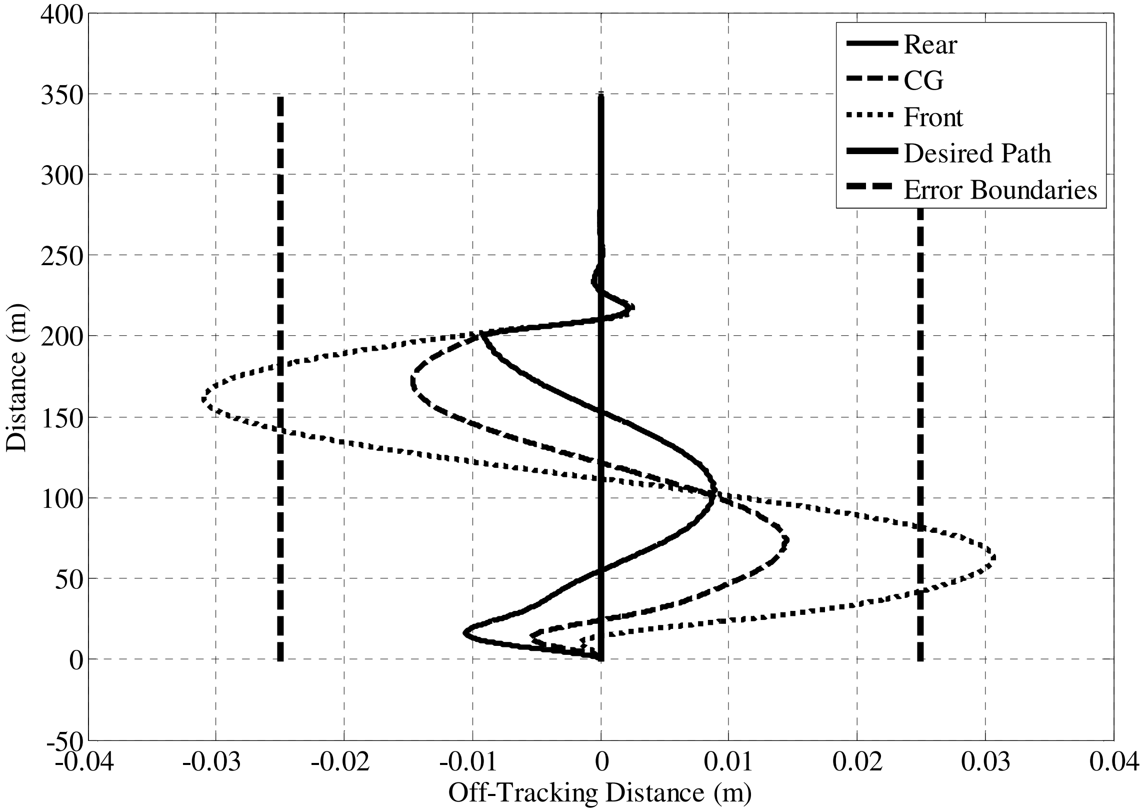


Figure 4.6 Rear axle off-tracking distance relationship to the position of the front axle for feed-forward control using the sinusoidal slope profile at a speed of four m/s.

4.1.2 Slope Look-Ahead

After observing that roll angle feed-forward control improved the tracking performance, feed-forward controller performance was next investigated using estimates of near future roll angles instead of roll angles measured at the current vehicle location. This was done to compare the case where a roll angle sensor is mounted on a vehicle and measures current vehicle roll angle with the case where a Digital Elevation Model (DEM) could be used for roll angle estimates based on future vehicle postures according to the current vehicle location and direction. Our hypothesis was that the controller could look ahead into a field-level DEM and use the predicted upcoming roll angle to improve automatic guidance performance. Instead of using an actual DEM, the controller used pre-determined slopes along the step and sinusoidal slope profiles.

Although the improvements are not as great as what was previously seen with the comparison between feed-forward and non-feed-forward control schemes, the performance metrics showed that there is a significant increase when using an appropriate look-ahead distance in conjunction with the roll angle feed-forward control. For the 2 m/s simulation and a look-ahead distance of two meters, the maximum off-tracking distance was cut nearly in half to 0.016 m from 0.031 m without using look-ahead control (Table 4.3). Similar results show the off-tracking error percentage was reduced from 4.22% without using look-ahead control to 0%, using look-ahead control at 4 m/s. The look-ahead distance of 2 m at a speed of 8 m/s on the step slope profile reduced the off-tracking error the greatest out of both look-ahead distance and all four velocities with a reduction of 0.017 m. However, as much as the 2 meter distance reduced the error, the 10 meter distance doubled the error, reaching an error point as high as 0.075 m. Clearly the look-ahead distance must be selected to correspond with the system dynamics.

By looking at more data from the step slope profiles, other differences between the two types of control were also evident. When the simulation stepped to the five degree slope, both controllers drove to the same steady state responses, however the look-ahead had less overshoot (Figures 4.7 and 4.8).

Table 4.3 Performance metrics comparing the look-ahead distances of 0, 2 and 10 meters for simulations of roll angle feed-forward control using a step slope profile at 2, 4, 6 and 8 m/s.

Speed (m/s)	Maximum Off-tracking Distance (m)			Off-Tracking Percentage		
	FF, No LA	FF, LA (2 m)	FF, LA, (10 m)	FF, No LA	FF, LA (2 m)	FF, LA, (10 m)
2	0.031	0.016	0.059	3.85	0.00	6.02
4	0.034	0.020	0.065	4.22	0.00	6.17
6	0.037	0.026	0.060	4.66	0.69	6.20
8	0.042	0.025	0.075	5.18	0.42	11.17

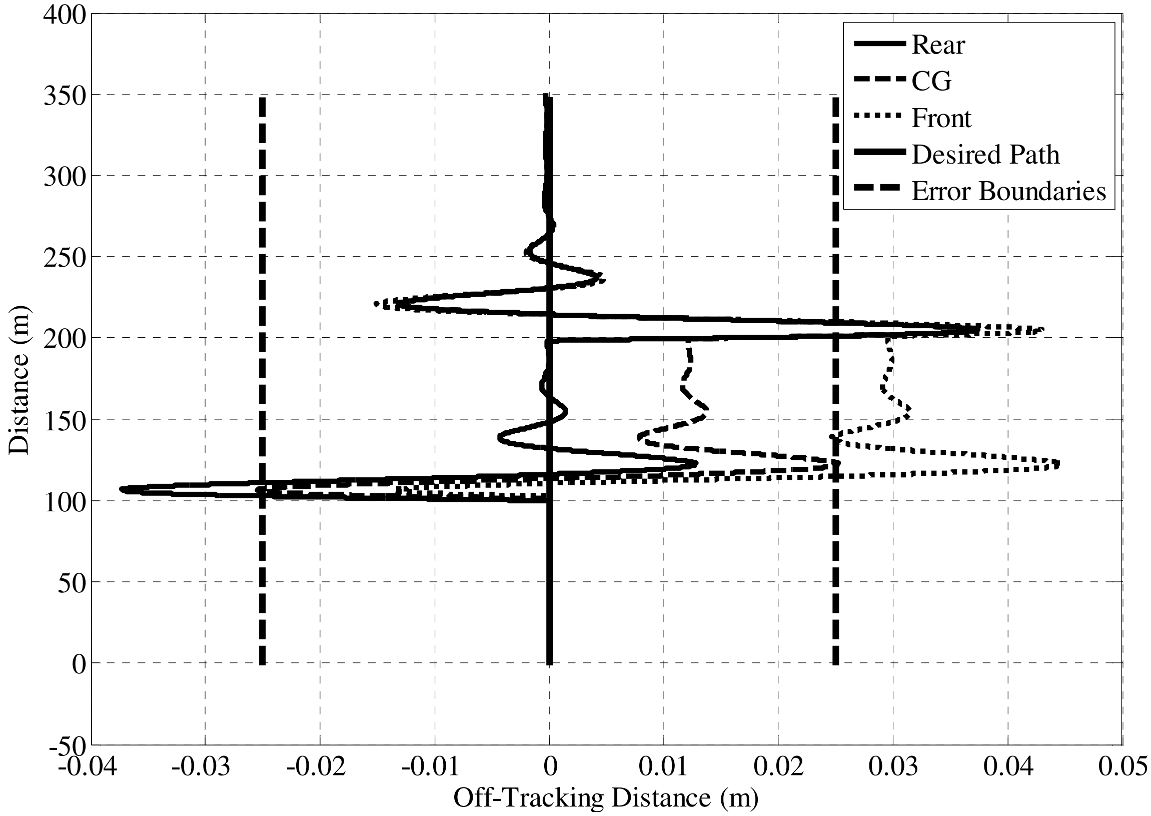


Figure 4.7 Visual simulation response cues of a step slope profile at 6 m/s comparing roll angle feed-forward control without a look-ahead distance.

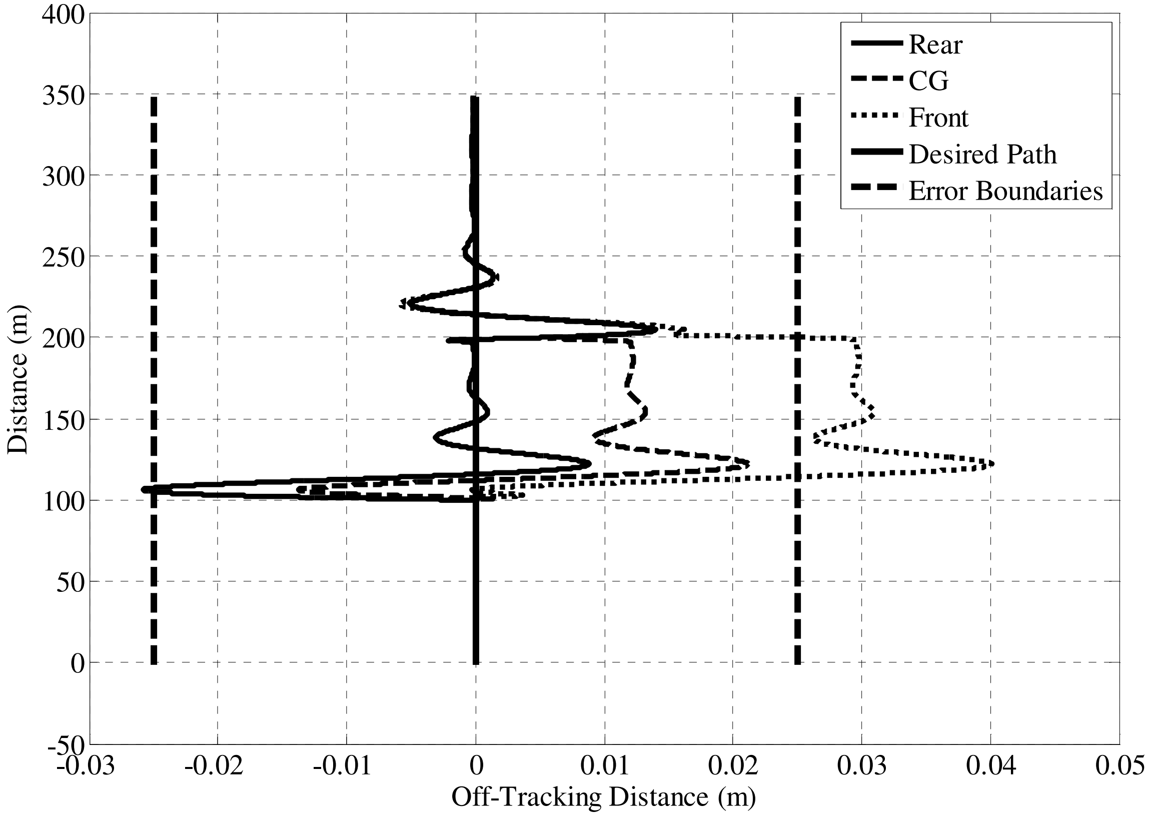


Figure 4.8 Visual simulation response cues of a step slope profile at 6 m/s comparing roll angle feed-forward control using a look-ahead distance of 2m.

In addition to the step slope profile, simulations were also performed with the sinusoidal slope profile. Although the sinusoidal slope responses did not reduce the off-tracking distance as much as the step slope profile, when comparing them to the non-look-ahead simulations, they did have the lowest overall off-tracking distance of 0.005 meters (Table 4.4). When compared to the non-look-ahead data, the sinusoidal slope profiles using look-ahead control cut the off-tracking distance by 50%.

Table 4.4 Performance metrics comparing the look-ahead distances of 0, 2 and 10 meters for simulations of roll angle feed-forward control using a sinusoidal slope profile at 2, 4, 6 and 8 m/s.

Speed (m/s)	Maximum Off-tracking Distance (m)			Off-Tracking Percentage		
	FF, No LA	FF, LA (2 m)	FF, LA, (10 m)	FF, No LA	FF, LA (2 m)	FF, LA, (10 m)
2	0.010	0.005	0.015	0.00	0.00	0.00
4	0.011	0.005	0.015	0.00	0.00	0.00
6	0.011	0.006	0.017	0.00	0.00	0.00
8	0.013	0.006	0.020	0.00	0.00	0.00

Unlike the difference in look-ahead (2m) and non-look-ahead cases, going from 2 to a 10 meter look-ahead value did not show an increase in performance. For both the step and sinusoidal slope profiles, the 10 meter look-ahead distance was worse than not using any look-ahead distance. For the sinusoidal slopes, where the 2 meter look-ahead distance effectively cut the off-tracking distance errors in half, the 10 meter look-ahead distance increased the off-tracking errors by 50%.

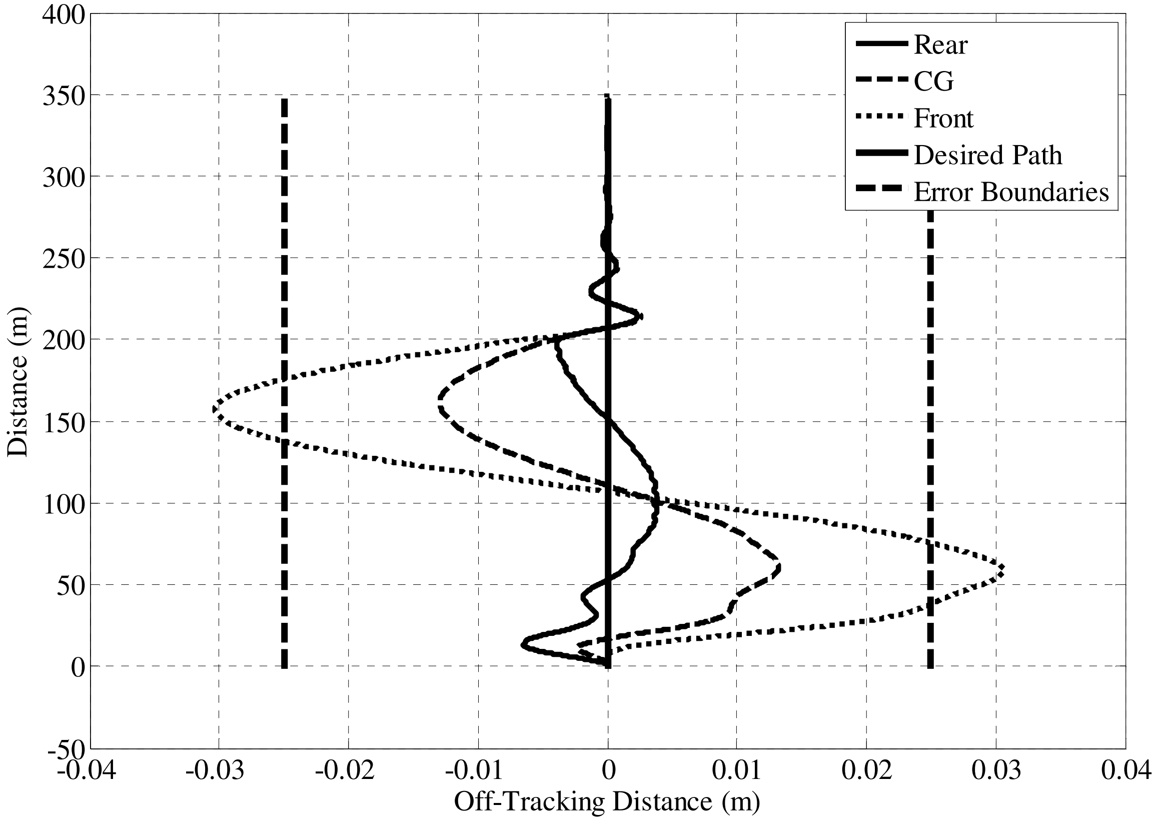


Figure 4.9 Simulation responses comparing look-ahead distances of 2 meters for a sinusoidal slope profile at a speed of 8 m/s and using roll angle feed-forward control.

In the sinusoidal performance metrics, it was noticeable that in both simulations, the off-tracking distances were below the off-tracking error distance threshold (0.025m) for the length of the run. However, from additional data, it was also evident that the 2 meter look-ahead distance simulation did not have as large amplitudes as the 10 meter distance. This results in the 2 meter look-ahead distance controller tracking the desired path better, especially when comparing the path with the rear axle center trajectory (Figures 4.9 and 4.10).

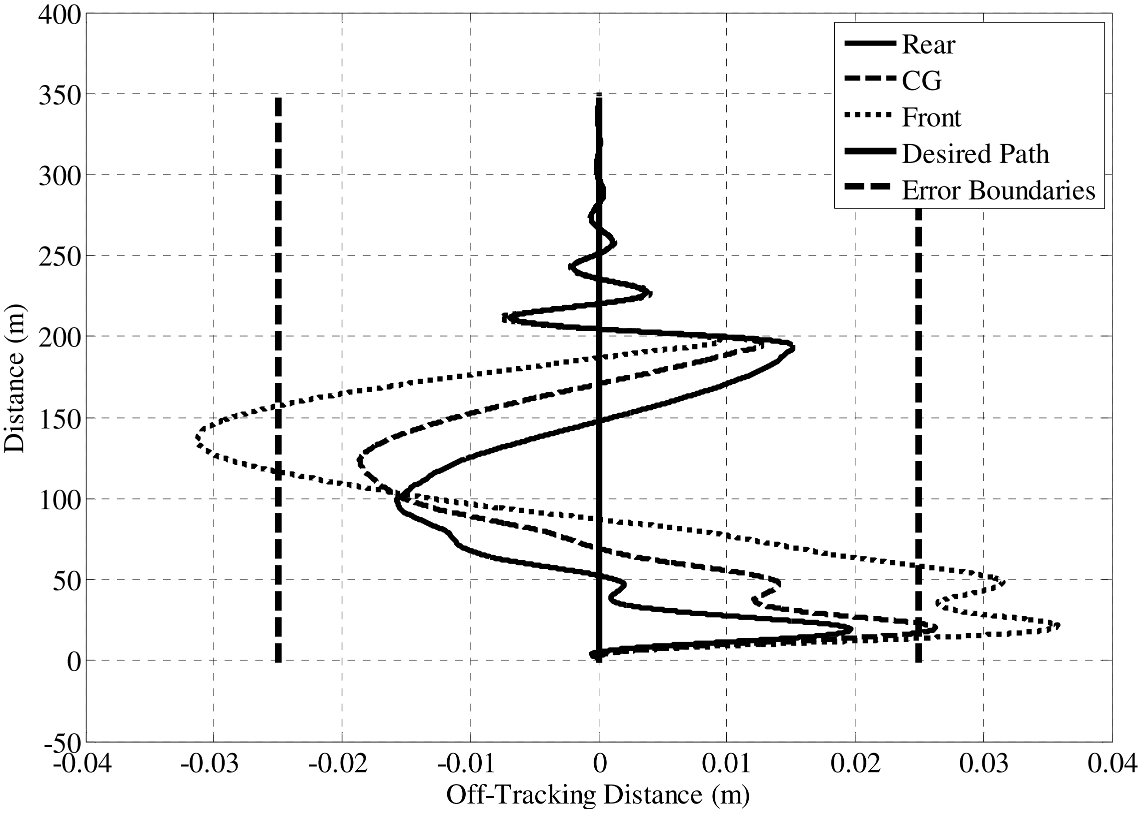


Figure 4.10 Simulation responses comparing look-ahead distances of 10 meters for a sinusoidal slope profile at a speed of 8 m/s and using roll angle feed-forward control.

4.2 Parameter Changes

By using a specific vehicle for all of the simulations, the majority of the vehicle parameters such as wheelbase and weight, were clearly defined and held constant throughout. However, three variables were chosen to further investigate controller performance. The three values under investigation were: the vehicle velocity, the integral off-tracking error feedback term, and the cornering stiffnesses for different field conditions. To determine if there was any change in performance with vehicle velocity, simulations of 2, 4, 6 and 8 m/s were conducted. The roll angle feed-forward controller development resulted in an integral off-tracking term which was very small. Further simulations were conducted to see if an increase in this value would improve the vehicle response. Lastly, the tire cornering stiffness, which dealt with tire to ground interactions, had a large uncertainty associated with it since it is influenced by many independent factors. As stated in section 2.4, much work has been published in this area due to this uncertainty, thus it was thought to be beneficial to explore how these changes would affect the roll angle feed-forward controller.

4.2.1 Changes in Velocity

For each of the four controller simulations (non-feed-forward, feed-forward and both types of look-ahead functions) and the sinusoidal slope profile, the off-tracking distance did not vary significantly with increased velocities. From the data, the largest change in off-tracking distance when comparing a 2 m/s velocity with an 8 m/s velocity was five millimeters, which was for the roll angle feed-forward control with a look-ahead distance of ten meters (Table 4.1). Although each of the controllers did show an increase in off-tracking distance when comparing the 2 and 8 m/s simulations, this change only averaged an increase of 2.5 mm. Similarly, for the simulations that show any off-tracking percentage, mainly the non-feed-forward simulation, the off-tracking percentage between any two speeds did not change by one percent.

Additional data allowed for the further investigation of the vehicle response beyond the performance metrics of off-tracking distance and off-tracking percentage. A visual comparison showed that two simulations, both using non-feed-forward controllers on a sinusoidal slope profile, initially look nearly identical in their maximum off-tracking distance and period for speeds of 2 and 8 m/s (Figures 4.11 and 4.12).

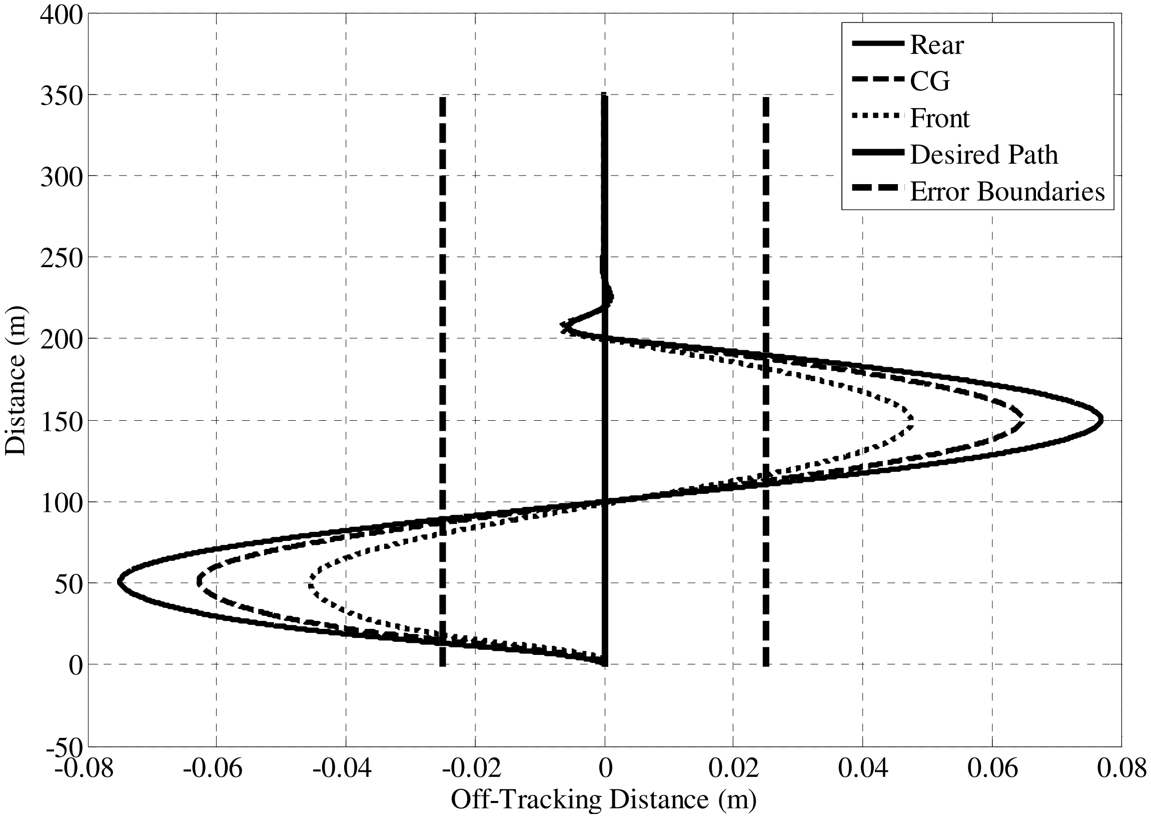


Figure 4.11 The effects on vehicle trajectories for non-feed-forward control simulations comparing different vehicle velocities using a sinusoidal slope profile at 2 m/s.

However, upon closer inspection it is apparent that the 8 m/s simulation takes close to 25 meters longer to reach steady state. Overall, the simulations did not uncover any significant relationship between the vehicle velocity and the off-track error. This confirms what was published by Evans (2006) and shows that changes due to speed are negligible.

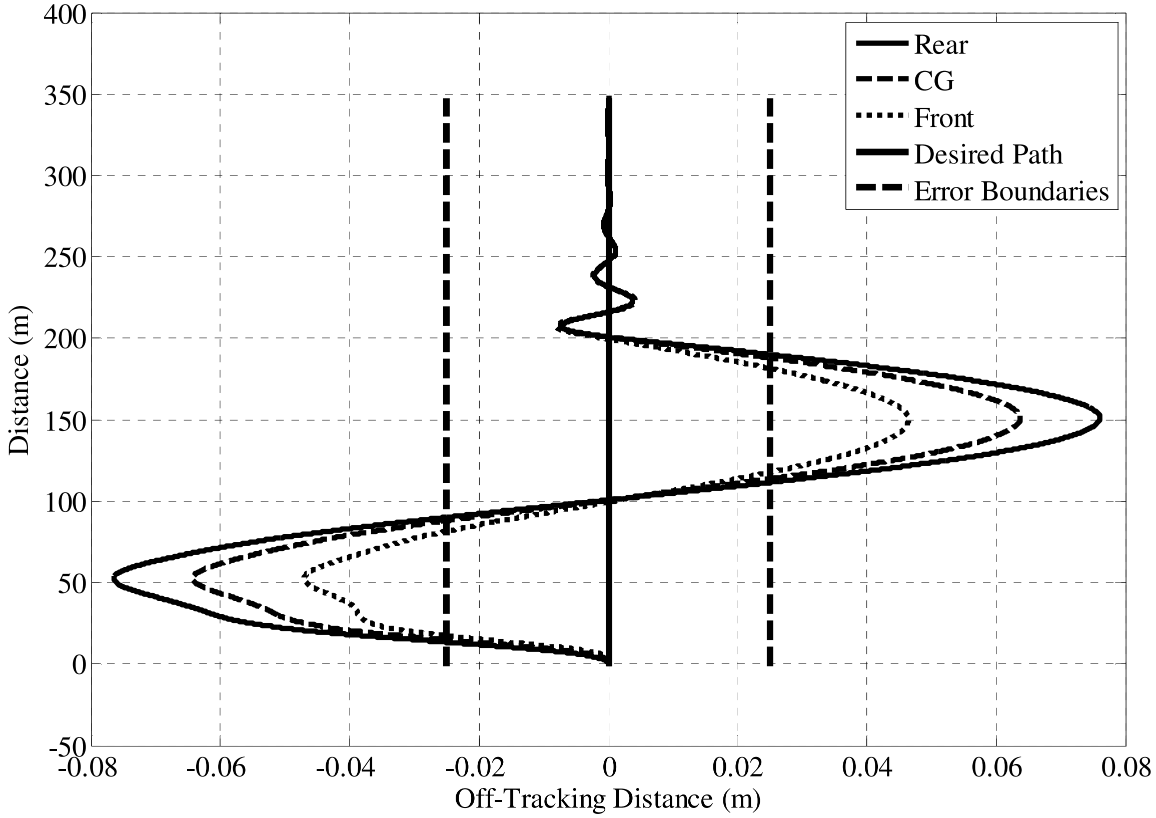


Figure 4.12 The effects on vehicle trajectories for non-feed-forward control simulations comparing different vehicle velocities using a sinusoidal slope profile at 8 m/s.

As in the sinusoidal slope profiles, further data analysis reveals the responses of the vehicle beyond what was described with performance metrics. With the increased speed, the sinusoidal profiles did not have a large change in response; however, from the step slope profiles simulations, the vehicle dynamics had lower damping at the highest speed level (Figures 4.13 and 4.14). This lower damping corresponds to the increase in velocity also increased the settling distance from 250 to 300 meters when going from 2 to 8 m/s. Once again the results are different than what was seen with the sinusoidal slope profiles. However, despite the changes in the system responses, the overall effect was minor, especially when considering drastic change in slope that was a result of the step slope profile.

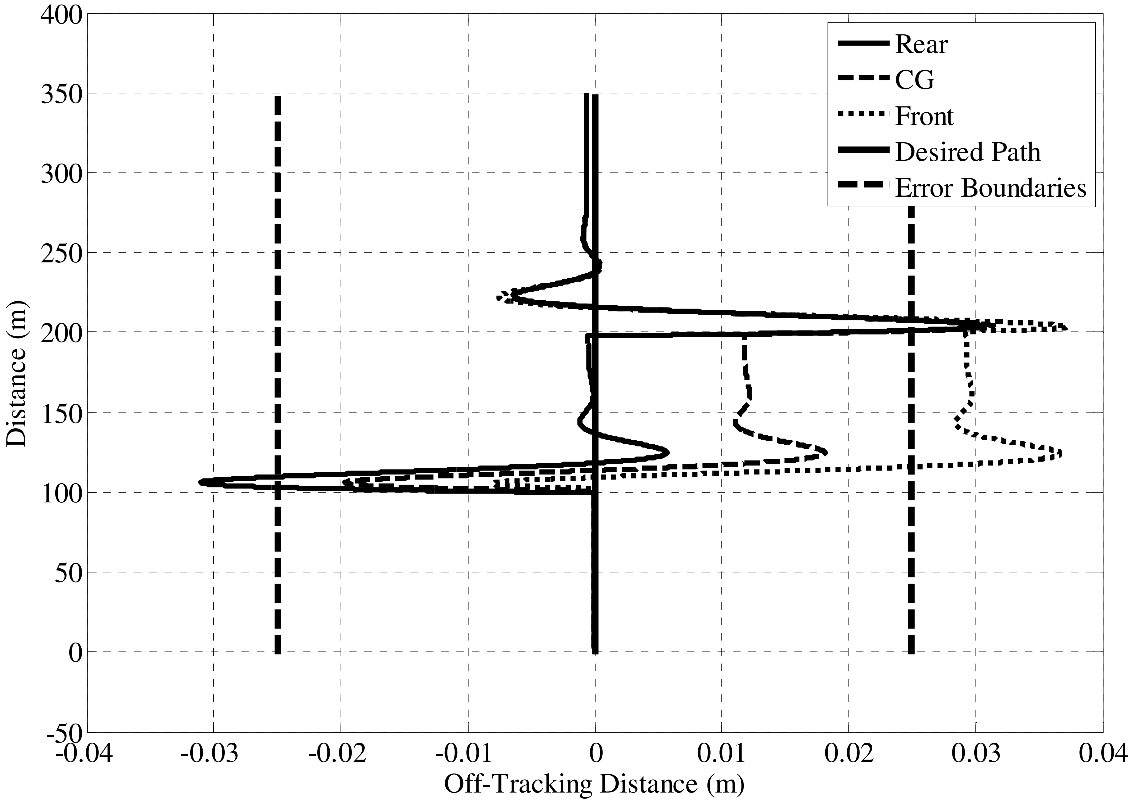


Figure 4.13 Comparison of off-tracking trajectory from a path at x=0 for a 2 m/s velocity using a roll angle feed-forward controller, without the look-ahead function on a step slope profile.

Results from the sinusoidal slope profiles showed very little change with increased velocities and when they did change it was in a predictable manner. The four step slope profile controller simulation results did not mimic those of the sinusoidal slopes. For the non-feed-forward control simulation the results showed similar characteristics to the sinusoidal paths in that the off-tracking distance increased with velocity (Table 4.2). However, in the remaining three simulations, the off-tracking distances were not all greater than they were at the previous speed for the same simulation. For instance, the roll angle feed-forward controller using a look-ahead distance of ten meters had the second highest off-tracking distance for the simulation occur at a speed of 4 m/s.

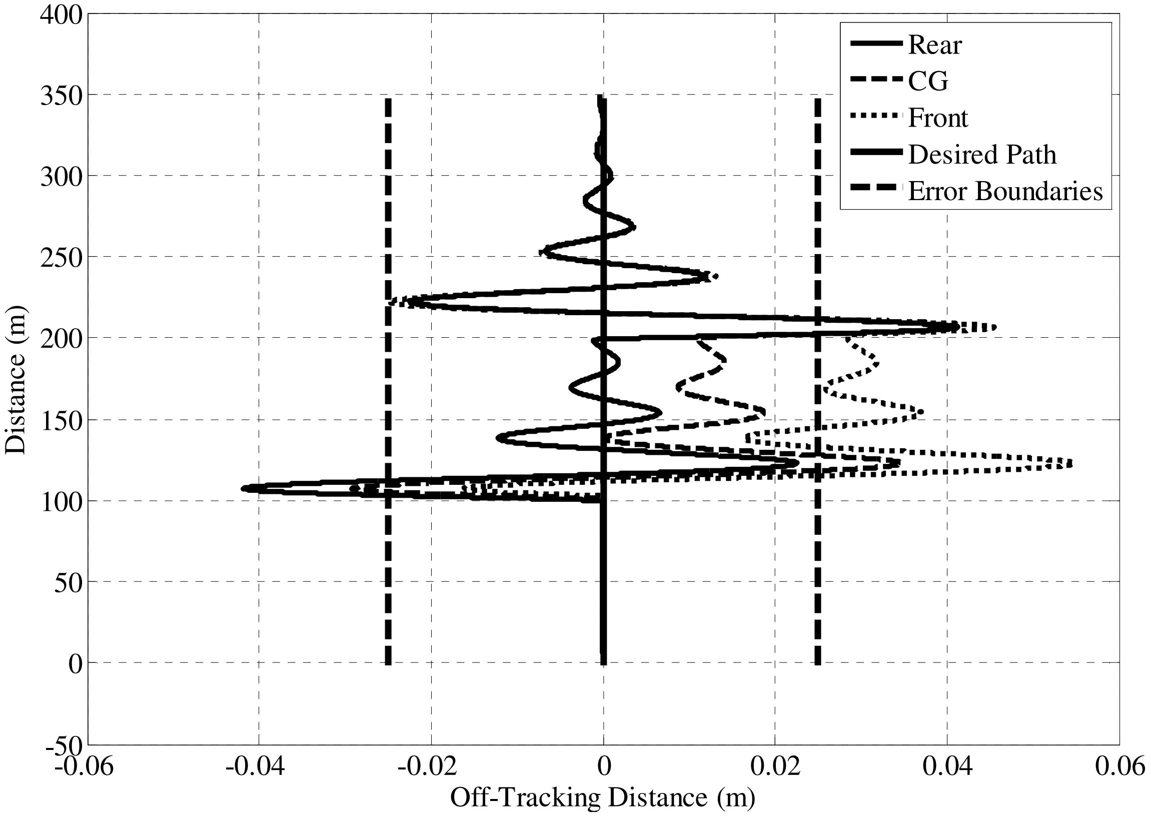


Figure 4.14 Comparison of off-tracking trajectory from a path at x=0 for an 8 m/s velocity using a roll angle feed-forward controller, without the look-ahead function on a step slope profile.

Another difference for the step slope profile simulations was the magnitude of the off-tracking distances. Out of all of the different velocity simulations the most any one off-tracking distance increased by was 0.018 meters, which was the difference between the 2 and 8 m/s velocities using non-feed-forward control. This value appeared to be small, but when calculated, it made up 72% of our allowable error (25 mm). This was not the same result as found in the sinusoidal simulations. The results showed that for the step slope profiles, increased velocities do negatively affect the controller performance.

4.2.2 Changes in $K_{OT}I$

From the controller development in chapter three, one value was chosen for the integral off-tracking gain ($K_{OT}I = 0.0001$). This number was so small that the integral component of the feedback controller was practically negligible. Thus, since integral control will drive steady-state error to zero, further simulations were run using a larger integral off-tracking gain to evaluate whether it would have a significant impact upon the off-tracking results. In order to provide a small sample set for simplified and direct comparisons, simulations of the sinusoidal slope profile alone were used with the controllers that have a larger integral error term.

When the feedback error term, $K_{OT}I$, was increased to 0.01, the non-feed-forward controller performance improved (Tables 4.5 and 4.6). This was most notable at the 2 m/s velocity level where the maximum off-tracking distance with the increased $K_{OT}I$ was over three centimeters less than that of the tests with $K_{OT}I$ being equal to 0.0001. However, as the speed increased the advantages of the larger value for $K_{OT}I$ were less as they both were within one centimeter off-tracking distance at 8 m/s. Also in the simulation for an 8 m/s vehicle velocity, the original value for the off-tracking gain had an off-tracking percentage of 44.52% while the larger gain only had 42.91% off-tracking. Although it was expected that the values would not change with velocity similar to what was seen previously, in the case of a larger $K_{OT}I$, an increasing velocity created larger off-tracking errors and percentages. The changes were not negligible either; the off-tracking distance increased by over 70% while the error percentage increased by 30%. This was a very interesting result as was substantially different than previous results.

Table 4.5 Performance metrics for an increased integral off-tracking error ($K_{OT}I = 0.01$) using a sinusoidal slope profile for different vehicle controllers.

Performance Metric	Speed (m/s)	No FF	FF, No LA	FF, LA (2m)	FF, LA, (10m)
Maximum Off-tracking Distance (m)	2	0.043	0.011	0.011	0.021
	4	0.063	0.011	0.008	0.022
	6	0.072	0.011	0.008	0.021
	8	0.076	0.012	0.007	0.021
Off-Tracking Percentage	2	32.66	0.00	0.00	0.00
	4	41.76	0.00	0.00	0.00
	6	43.25	0.00	0.00	0.00
	8	42.91	0.00	0.00	0.00

Table 4.6 Performance metrics for the original integral off-tracking error ($K_{OT}I = 0.0001$) using a sinusoidal slope profile for different vehicle controllers.

Performance Metric	Speed (m/s)	No FF	FF, No LA	FF, LA (2m)	FF, LA, (10m)
Maximum Off-tracking Distance (m)	2	0.076	0.010	0.005	0.015
	4	0.076	0.011	0.005	0.015
	6	0.076	0.011	0.006	0.017
	8	0.077	0.013	0.006	0.020
Off-Tracking Percentage	2	44.40	0.00	0.00	0.00
	4	44.46	0.00	0.00	0.00
	6	44.50	0.00	0.00	0.00
	8	44.52	0.00	0.00	0.00

In the roll angle feed-forward simulations where the look-ahead functions were not used, the controller performance did not change with the larger integral gain term. When comparing the 4 and 6 m/s velocity simulations for both sets of gains used the off-tracking distance values remained the same at 0.011 meters. At the remaining two velocity levels the new and old data were within 1 mm. Thus far it seemed that a value 100 times larger than the gain chosen could have been substituted for the original value.

Further data shows, however, that the two graphical responses are not identical (Figures 4.15 and 4.16). Both simulations appear to have matching responses, for the front and the center of gravity trajectories, up until the 200 meter mark. At the same time, the rear trajectory for the larger gain stayed closer to the desired path. However, at this point the gain value of 0.0001 quickly settled into the path, while the gain of 0.01 takes over 100 meters longer to reach the same off-tracking distance from the path.

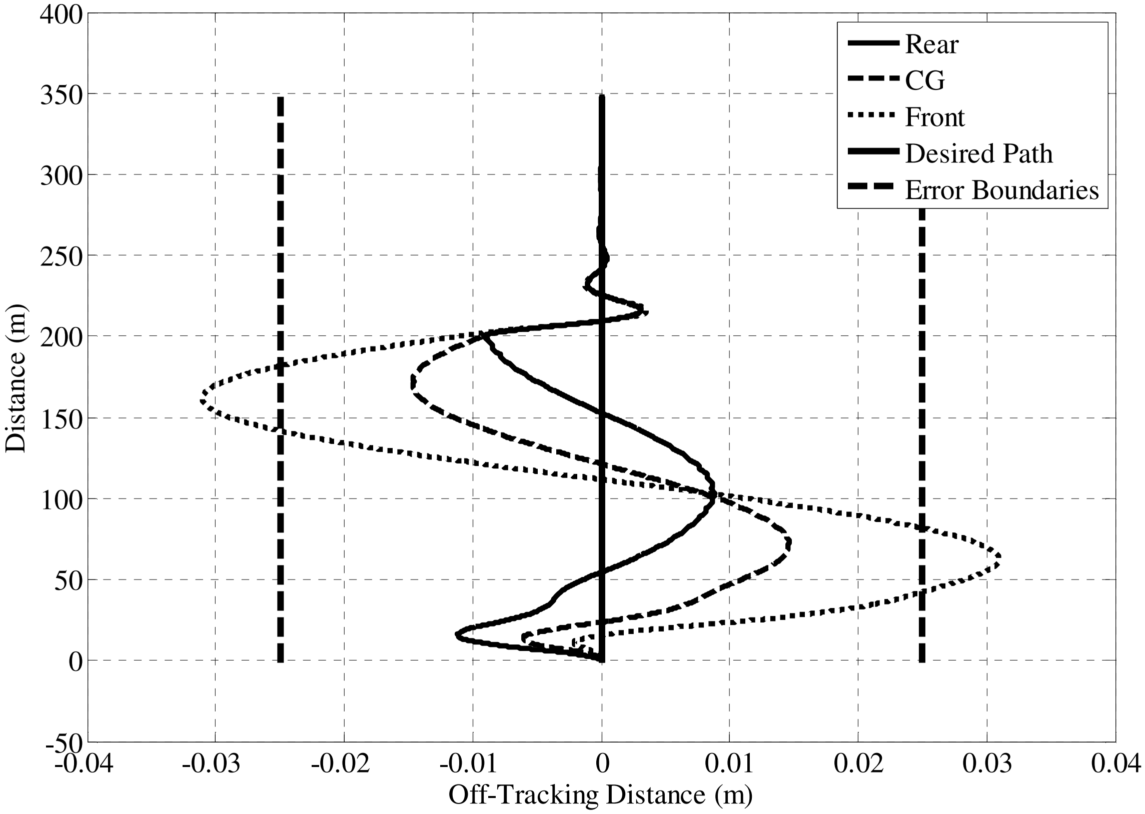


Figure 4.15 Controller response to a sinusoidal slope profile using roll angle feed-forward control at 6 m/s and comparing the effects of the integral off-tracking gains of $K_{OT}I = 0.0001$ on vehicle response.

By the performance metrics, the original integral off-tracking gain performed better than the new gain with roll angle estimation at look-ahead distances. At the 2 m/s velocity, a gain of 0.01 resulted in an off-tracking distance of 0.011 meters while a gain of 0.0001 had a much lower off-tracking distance of 0.005 meters. In both the 2 and 10 meter look-ahead functions, the original gain outperformed the larger gain. Since the gain of 0.0001 performed equal to or better than the gain of 0.01 in the cases where the smallest off-tracking distances were found, the controller was not changed and continued to use the original gain values. One point of interest in the results is that for the 2 meter look-ahead function and the gain of 0.01, the off-tracking distance decreased as the vehicle velocity increased. This was opposite from what was observed and discussed previously with the non-feed-forward control for the larger gain.

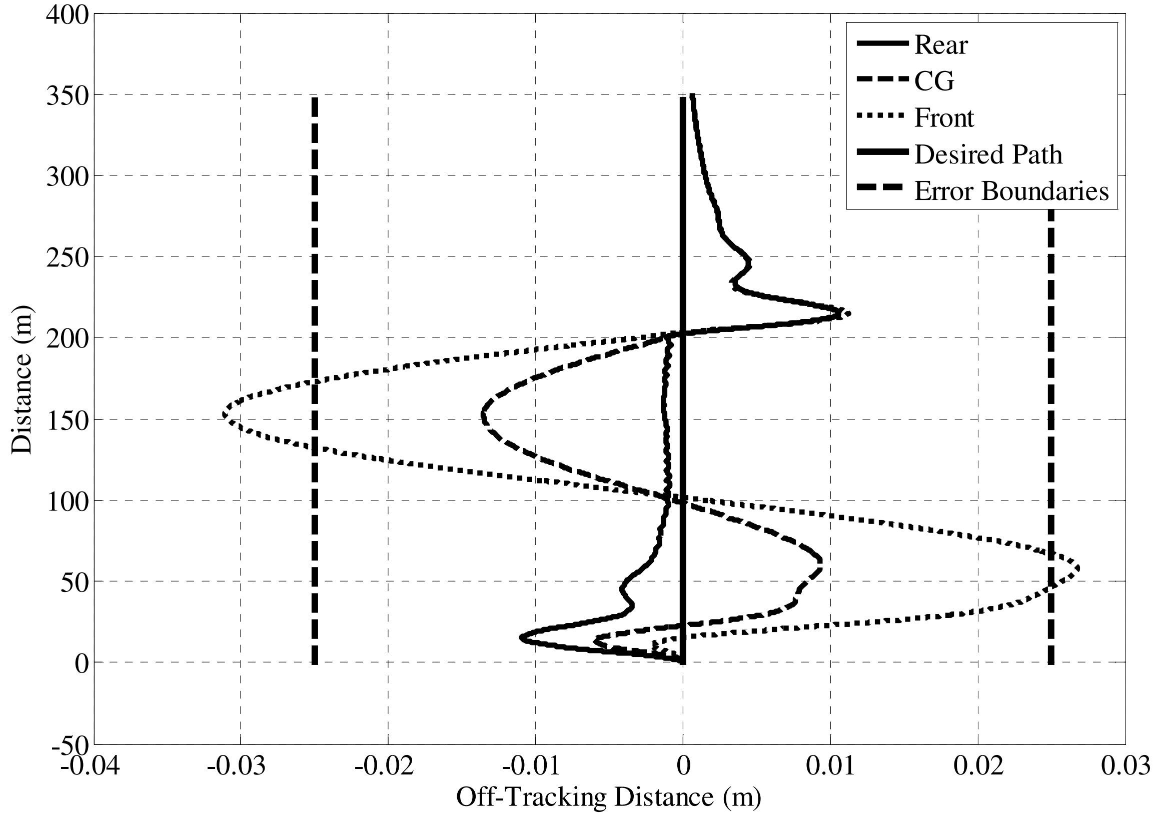


Figure 4.16 Controller response to a sinusoidal slope profile using roll angle feed-forward control at 6 m/s and comparing the effects of the integral off-tracking gain of $K_{OT}I = 0.01$ on vehicle response.

4.2.3 Changing Tire Conditions for Fixed K

One issue encountered when using this controller was that the gains used in the controller had to be set before starting the simulations. However, in real field conditions, the cornering stiffness may be changing based on differences in soil conditions. In order to optimize the controller, it was thought that the cornering stiffness used in the controller needed to remain close to the actual cornering stiffness in the field.

The controller was based around Metz's cornering stiffness for a corn field, so this data served as the baseline for subsequent simulations using different cornering stiffness values (Figure 4.17). To provide a close comparison, only simulations on a sine slope profile using feed-forward control were compared.

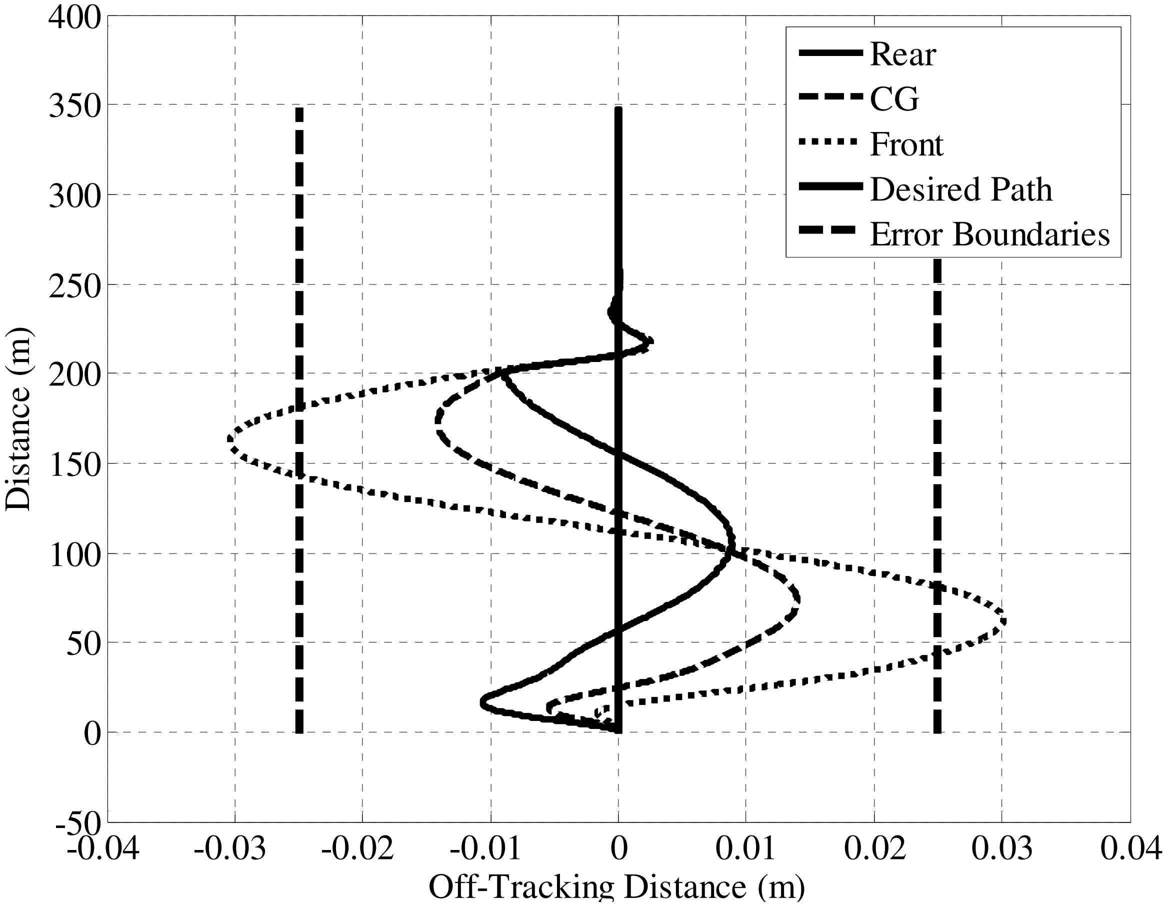


Figure 4.17 Vehicle trajectories for a sinusoidal slope profile, using feed-forward control, without the look-ahead function at a speed of 4 m/s while using the corn field cornering stiffnesses.

The plowed field had a smaller cornering stiffness, so it was not surprising to see the off-tracking distance to be worse than that of the corn field. The performance metrics showed that the plowed field was substantially worse, producing off-tracking distances that were over five times higher than that of the baseline data (Table 4.7). This result was observed across all four speeds. Comparing the plowed field against the corn field data showed a definite increase in error; however, a conclusion could not yet be made as to whether the cause of the increased off-tracking distance was due solely to the lower cornering stiffness of the plowed field, or whether the difference was a result of the actual cornering stiffness not matching the simulated cornering stiffness (Figure 4.18).

Table 4.7 Cornering stiffness in different field conditions for a sine slope using feed-forward control without the look-ahead function.

Field Conditions/ Cornering Stiffness	Speed (m/s)	Maximum Off-Tracking Distance (m)	Off-Tracking Percentage
Plowed field $C_{\alpha f}$ = 226615 N/rad $C_{\alpha r}$ = 384476 N/rad	2	0.052	38.23
	4	0.052	38.32
	6	0.051	38.32
	8	0.054	38.37
Corn field (Baseline) $C_{\alpha f}$ = 373432 N/rad $C_{\alpha r}$ = 633421 N/rad	2	0.010	0.00
	4	0.011	0.00
	6	0.011	0.00
	8	0.013	0.00
Meadow (Grassy field) $C_{\alpha f}$ = 665837 N/rad $C_{\alpha r}$ = 1129470 N/rad	2	0.034	25.96
	4	0.033	26.06
	6	0.033	26.14
	8	0.033	26.37

Taking a closer look at the trajectories from the case using the plowed field cornering stiffnesses, an interesting result was found when comparing the front axle off-tracking distances. In the corn field baseline data, the front axle had the greatest off-tracking errors while the rear axle remained close to the desired path. However, for the plowed field the opposite occurred. The front axle remained close to the desired path while the rear axle distances were much greater. This could have been a result of the low cornering stiffnesses of the plowed field in combination with the dual rear tires on the test vehicle.

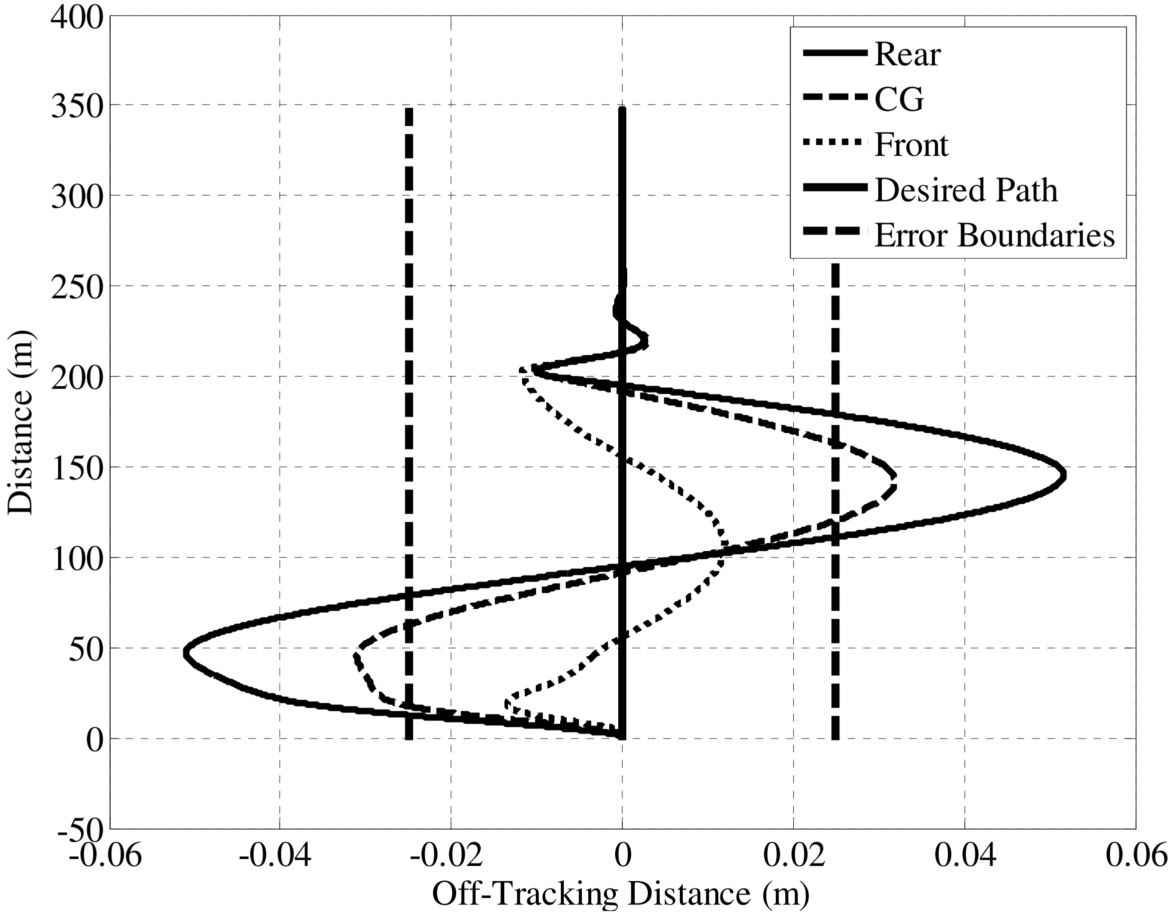


Figure 4.18 Vehicle trajectories for a sinusoidal slope profile, using feed-forward control, without the look-ahead function at a speed of 4 m/s while using the plowed field cornering stiffnesses.

The meadow field had a larger cornering coefficient than the corn field; however the data once again showed higher off-tracking distances than that of the baseline data (Figure 4.19). This result was similar to the plowed field, except in this case the off-tracking distances were three times higher than the baseline. With this data it was clear that the errors being generated were not solely based on a high or low cornering stiffness, but instead at least partially due to their differences from the baseline gains. However, the plowed field used cornering stiffnesses that were closer to the corn field values, yet had larger off-tracking distances than the meadow. Without further evidence the errors from the plowed field cannot be fully rationalized, but they are likely due to the low cornering stiffness and not being able to generate as high lateral forces.

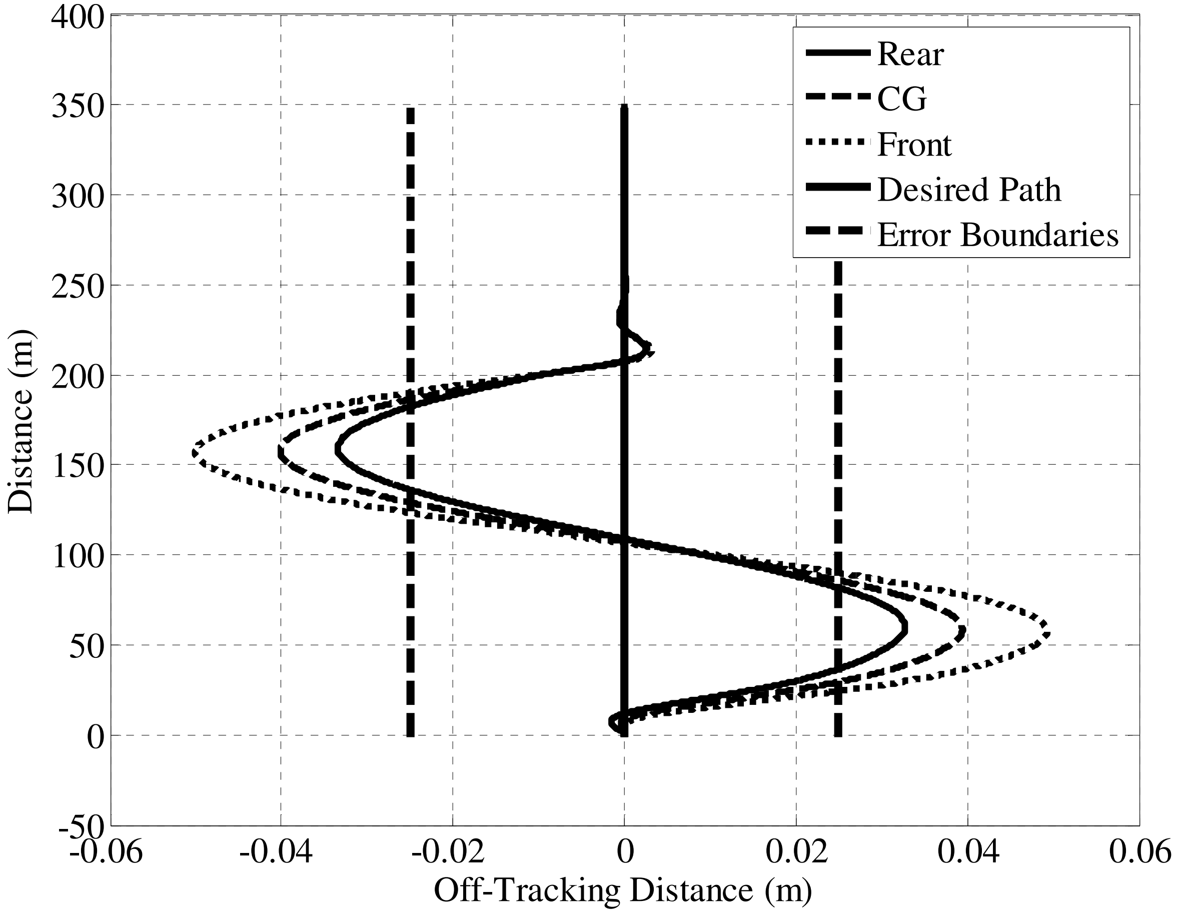


Figure 4.19 Vehicle trajectories for a sinusoidal slope profile, using feed-forward control, without the look-ahead function at a speed of 4 m/s while using the meadow cornering stiffnesses.

It is also important to note that in all three cases, the results represented an improvement over the non-feed-forward simulations. These results aligned with the theory, stated in the opening paragraph of this section, that in order to optimize the controller, the theoretical cornering stiffness must remain close to the actual coefficient.

CHAPTER 5. CONCLUSIONS AND SUGGESTIONS FOR FURTHER WORK

5.1 Conclusions

From this research, we derived the following conclusions:

- The roll angle feed-forward controller was the single biggest improvement to reduce off-tracking errors of a vehicle. In all cases and situations it outperformed the non-feed forward controller.

- The roll angle feed-forward controller gains its off-tracking advantage by over-driving the vehicle to the uphill side of the desired path and allows the gravitational force to reduce the off-tracking error.

- Increasing velocities in the sinusoidal slope paths do not significantly change the maximum off-tracking error; this was most likely a result of lateral forces generated by the tires increasing as the vehicle velocity increases, even though the tendency to overshoot would be greater. Meanwhile the gravity force we were trying to overcome remains constant, regardless of vehicle speed. As velocity increases, there was only a negligible difference in maximum and percentage off-tracking values. This confirms what Evans found and extends Evan's work to the roll angle feed-forward controller case.

- In the step slope simulations, an increase in velocity led to larger off-track errors and percentages. This result was due to the instantaneous change in slope that the step causes, fully exciting the closed loop system dynamics. Damping decreases as the speed increases, leading to larger errors.

- Estimating the roll angle 2 meters ahead of the current vehicle location cut the off-tracking distance by up to 50%. However, using a larger distance of 10 meters made the controller perform worse than the case where the current vehicle roll angle was used.
- By increasing the integral off-tracking gain, our performance metrics were not adversely affected. However, in some simulations the larger integral gain did show an ill effect in the settling time and system damping.
- The cornering stiffness used in the controller should be matched to the cornering stiffness of the field in order to have the greatest reduction in off-tracking distance. Derivations from the nominal value did result in increased metrics, but not as large as the case with no feed-forward control.

5.2 Suggestions for Further work

- All the work achieved for this project was simulated in MATLAB. It seems natural that the next step would be to test the controller in a real environment to see how it holds up to a different set of demands. This also would provide an interesting comparison to the simulated results of the controller. We have already integrated the controller into an automatic guidance equipped vehicle to test in real field conditions. However, for this type of testing, simulation and field testing offer very difficult challenges. One obstacle to field testing is an accurate method for measuring the off-tracking distance for the desired path. The A-B line must be set using vehicle mounted GPS receivers so there is a limitation on how accurate a path can be physically measured when dealing with such small off-tracking distances. Using the same GPS receivers for tracking a path and calculating off-tracking distances does not allow for an independent source of measurement to confirm that errors are not solely due to the receivers.

- The controller should be tested across a wide range of tires, different ground cover and varying degrees of soil moisture. For these tests, the controller was only set-up using the tire cornering stiffness for a corn field. This was due to the fact that the goal of this project was to study the controller performance and not to resolve any tire-to-soil contact issues. We know from section 4.2.3, that changing the actual cornering coefficient relative to what the controller used increases off-tracking errors. In this area there is a vast amount of variability and relatively little work was done compared to what is being done in other areas to improve automatic guidance. Different environmental conditions also can play a major roll in the effectiveness of a steering controller. Thus all of these issues need further exploration either in simulation or in field tests.
- The controller should be tested on slopes less than five degrees. In order to get a more direct comparison, only one maximum slope value was used. As a result, further studies may show the controller reacts differently for different grades and may not truly be a linear system.
- Apply the controller to different vehicles using alternate tractor parameters as well as non-tractor vehicle platforms. For instance, Evan's work involved the automatic guidance of a combine. An interesting comparison could be made between the two controllers if the vehicle parameters were changed to reflect those of a combine.
- Instead of using a three degree of freedom bicycle model, simulate a 6-degree dynamic vehicle model. Currently the yaw-plane bicycle model only contains half the

degrees of freedom needed to exactly replicate a real vehicle. A six degree of freedom model would also enhance the dynamic changes going on within the vehicle while operating on a slope, thus may provide a more accurate representation of a real vehicle. This would allow for the changes in normal forces due to vehicle pitch and roll. With so much of the controller relying on the tire to ground interaction, a change in the normal force could potentially yield some very beneficial results.

In conclusion, the feed-forward and look-ahead functions performed as expected reducing the off-tracking errors of the system. With the look-ahead distance set to two meters, the combination of these two control systems dominated the system responses compared to the other test cases. Further testing of the controller on a real vehicle platform would help verify our simulations and provide better feedback on the system. This would be very beneficial if the vehicle were navigated over differing terrain and soil conditions.

APPENDIX A. VEHICLE PARAMETERS FROM A JOHN DEERE 8320 TRACTOR

Variable	Value	Units
a	1.745	m
b	1.225	m
L	2.97	m
$C_{\alpha f}$	87.764	N/rad
$C_{\alpha r}$	135.86	N/rad
u	2, 4, 6 and 8	m/s
I_{zz}	50.64	kg*m^2
m	12660	kg
g	9.81	m/s^2
W	124192.6	N
W_f	51224.2	N
W_r	72968.4	N
θ	0 to 0.0873	radians

APPENDIX B. STEADY STATE EQUATIONS

B.1 Equations of Motion

The first equation of motion is found by summing the forces about the model in the y-direction.

$$F_s = W\sin(\theta) \tag{B-1}$$

$$\sum F = m\left(\dot{v} + ur\right) \tag{B-2}$$

$$F_{yf} + F_{yr} - F_s = m\left(\dot{v} + ur\right) \tag{B-3}$$

$$F_{yf} = -C_{\alpha f}\alpha_f \tag{B-4}$$

$$F_{yr} = -C_{\alpha r}\alpha_r \tag{B-5}$$

Substituting B-4 and B-5 into B-3

$$-C_{\alpha f}\alpha_f - C_{\alpha r}\alpha_r - W\sin(\theta) + m\dot{v} + mur \tag{B-6}$$

Then solving for $\dot{v}$

$$\dot{v} = \frac{-C_{\alpha f}\alpha_f - C_{\alpha r}\alpha_r - W\sin(\theta) - mur}{m} \tag{B-7}$$

Next the front and rear slip angles are calculated based on equations 2-7, 2-8 and 2-9.

$$\alpha_f = \beta + \frac{ar}{u} - \delta_f = \tan^{-1}\left(\frac{v}{u}\right) + \frac{ar}{u} - \delta_f = \frac{v + ar}{u} - \delta_f \tag{B-8}$$

$$\alpha_r = \beta - \frac{br}{u} - \delta_r = \tan^{-1}\left(\frac{v}{u}\right) - \frac{br}{u} - \delta_r = \frac{v - br}{u} - \delta_r \tag{B-9}$$

Then substituting these variables into B-7 yields

$$\overset{\bullet}{v} = \frac{-C_{\alpha f}\left(\dfrac{v+ar}{u} - \delta_f\right) - C_{\alpha r}\left(\dfrac{v-br}{u} - \delta_r\right) - W\sin(\theta) - mur}{m} \tag{B-10}$$

And

$$\overset{\bullet}{v} = A\bar{v} + B\bar{r} + E\bar{\delta}_f + F\bar{\delta}_r - \frac{W\sin(\theta)}{m} \tag{B-11}$$

Where:

$$A = \frac{-C_{\alpha f} - C_{\alpha r}}{mu} \tag{B-12}$$

$$B = \frac{-C_{\alpha f}a + C_{\alpha r}b - mu^2}{mu} \tag{B-13}$$

$$E = \frac{C_{\alpha f}}{m} \tag{B-14}$$

$$F = \frac{C_{\alpha r}}{m} \tag{B-15}$$

The second equation of motion is derived by summing the moments about the center of gravity.

$$\sum M + I_{zz}\overset{\bullet}{r} \tag{B-16}$$

$$F_{yf}a - F_{yr}b = I_{zz}\overset{\bullet}{r} \tag{B-17}$$

Then again substituting B-4 and B-5 variables, this time into B-17

$$-C_{\alpha f}\alpha_f a + C_{\alpha r}\alpha_r b = I_{zz}\overset{\bullet}{r} \tag{B-18}$$

Then solving for $\overset{\bullet}{r}$

$$\overset{\bullet}{r} = \frac{-C_{\alpha f}\alpha_f a + C_{\alpha r}\alpha_r b}{I_{zz}} \tag{B-19}$$

And substituting the front and rear slip angles (B-8 and B-9) yields

$$\overset{\bullet}{r} = \frac{-C_{\alpha f}\left(\frac{v+ar}{u} - \delta_f\right)a + C_{\alpha r}\left(\frac{v-br}{u} - \delta_r\right)b}{I_{zz}} \tag{B-20}$$

And

$$\overset{\bullet}{r} = C\bar{v} + D\bar{r} + G\bar{\delta}_f + H\bar{\delta}_r \tag{B-21}$$

Where:

$$C = \frac{-C_{\alpha f}a + C_{\alpha r}b}{I_{zz}u} \tag{B-22}$$

$$D = \frac{-C_{\alpha f}a^2 + C_{\alpha r}b^2}{I_{zz}u} \tag{B-23}$$

$$G = \frac{C_{\alpha f}a}{I_{zz}} \tag{B-24}$$

$$H = \frac{-C_{\alpha r}b}{I_{zz}} \tag{B-25}$$

Lastly since we are controlling the position of the vehicle relative to a straight-line path, two more state equations for the yaw angle and lateral position relative to the path are required.

$$\overline{\overset{\bullet}{Y}} = \bar{v} - b\bar{r} + u\bar{\psi} \Rightarrow \bar{\psi} = -\frac{\bar{v}}{u} \tag{B-26}$$

$$\overline{\overset{\bullet}{\psi}} = 0 = \bar{r} \tag{B-27}$$

Rewriting the equations in matrix form

$$\begin{bmatrix} \dot{v} \\ \dot{r} \\ \dot{Y} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} A & B & 0 & 0 \\ C & D & 0 & 0 \\ 1 & 0 & 0 & u \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} v \\ r \\ Y \\ \psi \end{bmatrix} + \begin{bmatrix} E & F & -\frac{W}{m} \\ G & H & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_f \\ \delta_r \\ \sin(\theta) \end{bmatrix} \tag{B-28}$$

B.2 Steady State Heading Error

In the steady state, the first derivatives go to zero.

$$\dot{v} = 0 = A\bar{v} + E\bar{\delta}_f - g\sin(\theta) \tag{B-29}$$

$$\dot{r} = 0 = C\bar{v} + G\bar{\delta}_f \tag{B-30}$$

Since B-26 will also go to zero, this reduces to:

$$\bar{\psi} = -\frac{\bar{v}}{u} \Rightarrow \bar{v} = -\bar{\psi}u \tag{B-31}$$

From 2-24 we get the control law for the controller.

$$\bar{\delta}_f = \bar{\delta}_{f,c} = -K_{OT}\left(\bar{Y} + d\sin\bar{\psi}\right) - K_H\psi \tag{B-32}$$

Then substituting B-31 and B-32 into B-29 and B-30.

$$0 = A\left(-\bar{\psi}u\right) + E\left(-K_{OT}\left(\bar{Y} + d\sin\bar{\psi}\right) - K_H\psi\right) - g\sin(\theta) \tag{B-33}$$

$$0 = C\left(-\bar{\psi}u\right) + G\left(-K_{OT}\left(\bar{Y} + d\sin\bar{\psi}\right) - K_H\psi\right) \tag{B-34}$$

Solving for $\bar{Y}$

$$\bar{Y} = \frac{-A\bar{\psi}u - EK_{OT}d\sin\bar{\psi} - EK_H\psi - g\sin(\theta)}{EK_{OT}} \tag{B-35}$$

$$\bar{Y} = \frac{-C\bar{\psi}u - GK_{OT}d\sin\bar{\psi} - GK_H\psi}{GK_{OT}} \tag{B-36}$$

Setting equations B-35 and B-36 equal to one another and simplifying:.

$$\frac{-A\bar{\psi}u - EK_{OT}d\sin\bar{\psi} - EK_{H}\psi - g\sin(\theta)}{EK_{OT}} = \frac{-C\bar{\psi}u - GK_{OT}d\sin\bar{\psi} - GK_{H}\psi}{GK_{OT}} \qquad \text{(B-37)}$$

$$\frac{-Au - EK_{H}}{EK_{OT}}\bar{\psi} - d\sin\bar{\psi} - \frac{g\sin\theta}{EK_{OT}} = \frac{-Cu - GK_{H}}{GK_{OT}}\bar{\psi} - d\sin\bar{\psi} \qquad \text{(B-38)}$$

$$\frac{(-Au - EK_{H})G}{EK_{OT}G}\bar{\psi} - \frac{(-Cu - GK_{H})E}{GK_{OT}E}\bar{\psi} = \frac{g\sin\theta}{EK_{OT}} \qquad \text{(B-39)}$$

$$\frac{(-Au - EK_{H})G - (-Cu - GK_{H})E}{GEK_{OT}}\bar{\psi} = \frac{g\sin\theta}{EK_{OT}} \qquad \text{(B-40)}$$

Solving for $\bar{\psi}$ and simplifying:

$$\bar{\psi} = \frac{g\sin\theta}{EK_{OT}}\left(\frac{GEK_{OT}}{(-Au - EK_{H})G - (-Cu - GK_{H})E}\right) = \frac{Gg\sin\theta}{(-Au - EK_{H})G - (-Cu - GK_{H})E}$$

(B-41)

$$\bar{\psi} = \frac{Gg\sin\theta}{(-GA + EC)u + (-GE - GE)K_{H}} = \frac{Gg\sin\theta}{(-GA + EC)u} \qquad \text{(B-42)}$$

Substituting A, C, E and G back into the equations:

$$\bar{\psi} = \frac{\left(\frac{C_{\alpha f}a}{I_{zz}}\right)g\sin\theta}{-\frac{C_{\alpha f}a}{I_{zz}}\left(\frac{-C_{\alpha f} - C_{\alpha r}}{mu}\right)u + \frac{C_{\alpha f}}{m}\left(\frac{-C_{\alpha f}a + C_{\alpha r}b}{I_{zz}u}\right)u} = \frac{\left(\frac{C_{\alpha f}a}{I_{zz}}\right)g\sin\theta}{-\frac{C_{\alpha f}a}{I_{zz}}\left(\frac{-C_{\alpha f} - C_{\alpha r}}{m}\right) + \frac{C_{\alpha f}}{m}\left(\frac{-C_{\alpha f}a + C_{\alpha r}b}{I_{zz}}\right)}$$

(B-43)

$$\bar{\psi} = \frac{\left(\frac{C_{\alpha f}a}{I_{zz}}\right)g\sin\theta}{\left(\frac{C_{\alpha f}{}^{2}a + C_{\alpha f}C_{\alpha r}a - C_{\alpha f}{}^{2}a + C_{\alpha f}C_{\alpha r}b}{I_{zz}m}\right)} = \frac{\left(\frac{C_{\alpha f}a}{I_{zz}}\right)g\sin\theta}{\left(\frac{C_{\alpha f}C_{\alpha r}(a+b)}{I_{zz}m}\right)} \qquad \text{(B-44)}$$

$$\bar{\psi} = \frac{ag\sin\theta}{\left(\frac{C_{\alpha r}\left(a+b\right)}{m}\right)} = \frac{aW\sin\theta}{C_{\alpha r}\left(a+b\right)} \tag{B-45}$$

Finally yields the steady state solution for heading error.

$$\bar{\psi} = \frac{W\sin\theta}{C_{\alpha r}\left(1+\frac{b}{a}\right)} \tag{B-46}$$

B.3 Steady State Off-Tracking Error

Starting with equation B-36 for the steady state off-tracking error and simplifying:

$$\bar{Y} = \frac{-C\bar{\psi}u - GK_{OT}d\sin\bar{\psi} - GK_{H}\psi}{GK_{OT}} \tag{B-47}$$

$$\bar{Y} = \frac{-Cu - GK_{OT}d - GK_{H}}{GK_{OT}}\bar{\psi} \tag{B-48}$$

Also using the steady state heading error (B-46)

$$\bar{\psi} = \frac{W\sin\theta}{C_{\alpha r}\left(1+\frac{b}{a}\right)} \tag{B-49}$$

Substituting in the equations for C, G and B-49, then reducing:

$$\bar{Y} = \frac{\left(-\left(\frac{-C_{\alpha f}a + C_{\alpha r}b}{I_{zz}u}\right)u - \left(\frac{C_{\alpha f}a}{I_{zz}}\right)K_{OT}d - \left(\frac{C_{\alpha f}a}{I_{zz}}\right)K_{H}\right)}{\left(\frac{C_{\alpha f}a}{I_{zz}}\right)K_{OT}}\left(\frac{W\sin\theta}{C_{\alpha r}\left(1+\frac{b}{a}\right)}\right) \tag{B-50}$$

$$\bar{Y} = \frac{\frac{C_{\alpha f}a}{I_{zz}}\left(1 - \frac{C_{\alpha r}b}{C_{\alpha f}a} - K_{OT}d - K_{H}\right)W\sin\theta}{\left(\frac{C_{\alpha f}a}{I_{zz}}\right)K_{OT}C_{\alpha r}\left(1+\frac{b}{a}\right)} = \frac{\left(1 - \frac{C_{\alpha r}b}{C_{\alpha f}a} - K_{OT}d - K_{H}\right)W\sin\theta}{K_{OT}C_{\alpha r}\left(1+\frac{b}{a}\right)} \tag{B-51}$$

Further reducing B-49

$$\frac{W\sin\theta}{C_{\alpha r}\left(1+\frac{b}{a}\right)}=\frac{W_r\sin\theta}{C_{\alpha r}} \tag{B-52}$$

Then changing this in B-51

$$\overline{Y}=\frac{\left(1-\frac{C_{\alpha r}b}{C_{\alpha f}a}-K_{OT}d-K_H\right)\frac{W_r}{C_{\alpha r}}\sin\theta}{K_{OT}} \tag{B-53}$$

$$\overline{Y}=\frac{\left[\left(1-K_{OT}d-K_H\right)\frac{W_r}{C_{\alpha r}}-\frac{W_rb}{C_{\alpha f}a}\right]\sin\theta}{K_{OT}} \tag{B-54}$$

Using the definition of front and rear normal loads:

$$W_f=W_r\left(\frac{b}{a}\right) \tag{B-55}$$

Using B-55 to simplify further:

$$\overline{Y}=\frac{\left[-\left(-1+K_{OT}d+K_H\right)\frac{W_r}{C_{\alpha r}}+\frac{W_f}{C_{\alpha f}}\right]\sin\theta}{K_{OT}} \tag{B-56}$$

$$\overline{Y}=\frac{-\left[\left(K_{OT}d+K_H\right)\frac{W_r}{C_{\alpha r}}-\frac{W_r}{C_{\alpha r}}+\frac{W_f}{C_{\alpha f}}\right]\sin\theta}{K_{OT}} \tag{B-57}$$

Using the understeer gradient, K, form Gillespie (1992)

$$K=\frac{W_f}{C_{\alpha f}}-\frac{W_r}{C_{\alpha r}} \tag{B-58}$$

Yields the steady state off-tracking error.

$$\overline{Y} = \frac{-\left[\left(K_{OT}d + K_H\right)\dfrac{W_r}{C_{\alpha r}} + K\right]\sin\theta}{K_{OT}} \tag{B-59}$$

B.4 Steady State Steering Angle

For the steady state steering angle B-32 is used as a starting point.

$$\overline{\delta}_f = -K_{OT}\left(\overline{Y} + d\sin\overline{\psi}\right) - K_H\overline{\psi} \tag{B-60}$$

$$\overline{\delta}_f = -K_{OT}\overline{Y} - K_{OT}d\sin\overline{\psi} - K_H\overline{\psi} \tag{B-61}$$

From section B.3 we use B-59 to substitute for $\overline{Y}$.

$$\overline{\delta}_f = -K_{OT}\left(\frac{-\left(K_{OT}d + K_H\right)\dfrac{W_r}{C_{\alpha r}}\sin\theta - K\sin\theta}{K_{OT}}\right) - K_{OT}d\sin\overline{\psi} - K_H\overline{\psi} \tag{B-62}$$

$$\overline{\delta}_f = -\left(-K_{OT}d\frac{W_r}{C_{\alpha r}}\sin\theta - K_H\frac{W_r}{C_{\alpha r}}\sin\theta - K\sin\theta\right) - K_{OT}d\sin\overline{\psi} - K_H\overline{\psi} \tag{B-63}$$

From B-52

$$\overline{\psi} = \frac{W_r}{C_{\alpha r}}\sin\theta \tag{B-64}$$

And substituting into B-63

$$\overline{\delta}_f = \left(K_{OT}d\frac{W_r}{C_{\alpha r}}\sin\theta + K_H\frac{W_r}{C_{\alpha r}}\sin\theta + K\sin\theta\right) - K_{OT}d\left(\frac{W_r}{C_{\alpha r}}\sin\theta\right) - K_H\left(\frac{W_r}{C_{\alpha r}}\sin\theta\right)$$

(B-65)

Finally reduces to the steady state steering angle

$$\overline{\delta}_f = K\sin\theta \tag{B-66}$$

B.5 Feed-Forward Steady State Heading Error

Once again starting with B-32

$$\bar{\delta}_f = -K_{OT}\left(\overline{Y} + d\overline{\psi}\right) - K_H\overline{\psi} + K\sin\theta \qquad \text{(B-67)}$$

Also starting with B-33 and B-34

$$0 = A\left(-\overline{\psi}u\right) + E\left(-K_{OT}\left(\overline{Y} + d\overline{\psi}\right) - K_H\overline{\psi} + K\sin\theta\right) - g\sin\theta \qquad \text{(B-68)}$$

$$0 = C\left(-\overline{\psi}u\right) + G\left(-K_{OT}\left(\overline{Y} + d\overline{\psi}\right) - K_H\overline{\psi} + K\sin\theta\right) \qquad \text{(B-69)}$$

From B-31

$$\bar{v} = -\overline{\psi}u \qquad \text{(B-70)}$$

And rearranging

$$0 = -A\overline{\psi}u - EK_{OT}\overline{Y} - E\left(K_{OT}d + K_H\right)\overline{\psi} + \left(EK - g\right)\sin\theta \qquad \text{(B-71)}$$

$$0 = -C\overline{\psi}u - GK_{OT}\overline{Y} - G\left(K_{OT}d + K_H\right)\overline{\psi} + GK\sin\theta \qquad \text{(B-72)}$$

From 2-45 we get

$$K_{CON} = K_{OT}d + K_H \qquad \text{(B-73)}$$

B-71 can be reduced to

$$\overline{Y} = \frac{-\left(Cu + GK_{CON}\right)\overline{\psi} + GK\sin\theta}{GK_{OT}} \qquad \text{(B-74)}$$

Setting $\overline{Y}$ equal to zero and combining with B-72 yields

$$0 = -A\overline{\psi}u - EK_{OT}\left[\frac{-\left(Cu + GK_{CON}\right)\overline{\psi} + GK\sin\theta}{GK_{OT}}\right] - EK_{CON}\overline{\psi} + \left(EK - g\right)\sin\theta \qquad \text{(B-75)}$$

Finally solving for $\overline{\psi}$

$$\overline{\psi} = \frac{W_r}{C_{\alpha r}}\sin\theta \tag{B-76}$$

B.6 Feed-Forward Steady State Off-Tracking Error

Equation 2-47 is used as a starting point

$$\overline{\psi} = \frac{GK_{OT}\overline{Y} - GK\sin\theta}{-(Cu + GK_{CON})} \tag{B-77}$$

Inserting this into B-68

$$(Au + EK_{CON})\overline{\psi} + EK_{OT}\overline{Y} = (EK - g)\sin\theta \tag{B-78}$$

$$(Au + EK_{CON})\left[\frac{GK_{OT}\overline{Y} - GK\sin\theta}{-(Cu + GK_{CON})}\right] + EK_{OT}\overline{Y} = (EK - g)\sin\theta \tag{B-79}$$

$$\left[-\frac{(Au + EK_{CON})}{(Cu + GK_{CON})}GK_{OT} + EK_{OT}\right]\overline{Y} = \left[-\frac{(Au + EK_{CON})}{(Cu + GK_{CON})}GK + EK - g\right]\sin\theta \tag{B-80}$$

After simplifying, equations for A, C, E and G are inserted

$$\frac{C_{\alpha f}C_{\alpha r}(a + b)K_{OT}}{I_{zz}m}\overline{Y} = \left[\frac{C_{\alpha f}C_{\alpha r}(a + b)K}{I_{zz}m} - g\left[\frac{-C_{\alpha f}a + C_{\alpha r}b}{I_{zz}} + \frac{C_{\alpha f}a}{I_{zz}}K_{CON}\right]\right]\sin\theta \tag{B-81}$$

Then solving for $\overline{Y}$

$$\overline{Y} = \left[\frac{K}{K_{OT}} - \frac{1}{K_{OT}}\left[(K_{OT}d + K_H)\frac{W_r}{C_{\alpha r}} + K\right]\right]\sin\theta \tag{B-82}$$

B.7 Feed-Forward Steady State Steering Angle

From B-66

$$\overline{\delta}_f = K\sin\theta \tag{B-83}$$

Where C is an arbitrary gain used in place of K

$$\overline{\delta}_f = C \sin\theta \tag{B-84}$$

Also using B-76

$$\overline{\psi} = \frac{W_r}{C_{\alpha r}} \sin\theta \tag{B-85}$$

Using C in B-82

$$\overline{Y} = \left[\frac{C}{K_{OT}} - \frac{1}{K_{OT}} \left[\left(K_{OT} d + K_H \right) \frac{W_r}{C_{\alpha r}} + K \right] \right] \sin\theta \tag{B-86}$$

For zero off-track ($\overline{Y} = 0$) and solving for C

$$C = \left(K_{OT} d + K_H \right) \frac{W_r}{C_{\alpha r}} + K \tag{B-87}$$

Finally yields the steady state steering angle

$$\overline{\delta}_f = \left(\left(K_{OT} d + K_H \right) \frac{W_r}{C_{\alpha r}} + K \right) \sin\theta \tag{B-88}$$

APPENDIX C. PERFORMANCE METRICS

Sinusoidal Path			Sinusoidal Path, $K_{OT}I = 0.01$			Step Path		
No Feed-Forward, No Look Ahead			No Feed-Forward, No Look Ahead			No Feed-Forward, No Look Ahead		
Speed (m/s)	Max OT	OT Percentage	Speed (m/s)	Max OT	OT Percentage	Speed (m/s)	Max OT	OT Percentage
2	0.076	44.40	2	0.043	32.66	2	0.093	28.68
4	0.076	44.46	4	0.063	41.76	4	0.096	28.80
6	0.076	44.50	6	0.072	43.25	6	0.102	29.98
8	0.077	44.52	8	0.076	42.91	8	0.111	31.34
Feed-Forward, No Look-Ahead			Feed-Forward, No Look-Ahead			Feed-Forward, No Look-Ahead		
Speed (m/s)	Max OT	OT Percentage	Speed (m/s)	Max OT	OT Percentage	Speed (m/s)	Max OT	OT Percentage
2	0.010	0.00	2	0.011	0.00	2	0.031	3.85
4	0.011	0.00	4	0.011	0.00	4	0.034	4.22
6	0.011	0.00	6	0.011	0.00	6	0.037	4.66
8	0.013	0.00	8	0.012	0.00	8	0.042	5.18
Feed-Forward, Look-Ahead (2m)			Feed-Forward, Look-Ahead (2m)			Feed-Forward, Look-Ahead (2m)		
Speed (m/s)	Max OT	OT Percentage	Speed (m/s)	Max OT	OT Percentage	Speed (m/s)	Max OT	OT Percentage
2	0.005	0.00	2	0.011	0.00	2	0.016	0.00
4	0.005	0.00	4	0.008	0.00	4	0.020	0.00
6	0.006	0.00	6	0.008	0.00	6	0.026	0.69
8	0.006	0.00	8	0.007	0.00	8	0.025	0.42
Feed-Forward, Look-Ahead (10m)			Feed-Forward, Look-Ahead (10m)			Feed-Forward, Look-Ahead (10m)		
Speed (m/s)	Max OT	OT Percentage	Speed (m/s)	Max OT	OT Percentage	Speed (m/s)	Max OT	OT Percentage
2	0.015	0.00	2	0.021	0.00	2	0.059	6.02
4	0.015	0.00	4	0.022	0.00	4	0.065	6.17
6	0.017	0.00	6	0.021	0.00	6	0.060	6.20
8	0.020	0.00	8	0.021	0.00	8	0.075	11.17

APPENDIX D. TIRE PARAMETERS

	On- Highway			Plowed field	
Normal Load =	25,500	N			
A =	0.67	unit less	A =	0.65	unit less
B =	0.29305	1/degrees	B =	0.11931	1/degrees
m =	0.14		m =	0.14	
C/A=	1.01		C/A=	0.411	
D/A=	0.8401		D/A=	0.3418	
Calpha=	5006.7	N/degree	Calpha=	1977.6	N/degree
	286861.7	N/rad		113307.6	N/rad
Cornering Coefficient =	0.19634	1/degree	Cornering Coefficient =	0.07755	1/degree

	Corn Field			Meadow	
A =	0.53	unit less	A =	0.88	unit less
B =	0.24113	1/degrees	B =	0.25894	1/degrees
m =	0.14		m =	0.14	
C/A=	0.8294		C/A=	0.891	
D/A=	0.6894		D/A=	0.7407	
Calpha=	3258.8	N/degree	Calpha=	5810.5	N/degree
	186716.2	N/rad		332918.8	N/rad
Cornering Coefficient =	0.12780	1/degree	Cornering Coefficient =	0.22786	1/degree

	FRONT		REAR		
Soil Conditions	Caf (N/rad)	Cafall (N/rad)	Car (N/rad)	Carall (N/rad)	K
On-Highway	286861.7	573723.5	243365.4	973461.8	0.01390
Plowed Field	113307.6	226615.2	96119.1	384476.5	0.03518
Corn Field	186716.2	373432.4	158355.5	633421.8	0.02132
Meadow	332918.8	665837.6	282367.7	1129470.7	0.01196
K +Wr/Car*(Kot*d+Kh)					
0.06752	Kot =	0.01		Wf (N)	51000
0.17094	d =	1.5		Wr (N)	73000
0.07753	Kh =	0.7			
0.10373					
0.05818					

APPENDIX E. MATLAB CODE

This section contains the complete MATLAB (The MathWorks, Inc., Natick, MA) code used in the simulation of the vehicle and controller. This code has been edited to take that which was used for plotting to reduce its length and only include the pertinent information.

E.1 Vehicle Simulation Code: *vehiclesim8.m*

```
function[Time,Y,Delf,Delr]=vehiclesim8(speed,SLOPEPATH,simtime,PATHTYPE,LASLOPE,KpHEadi
ng,KpOTrack,KiOTrack,FF);
%  Name: VehicleSim8
%  Purpose: To simulate a vehicle which is being navigated down a path
%  and has ARS 4WS control being applied to minimize the off-tracking
%  error between the front and rear axle centers
%  vehiclesim was uses model vehicle.m
%  vehiclesim2 was developed using the a simulink model vehicle4WS.mdl
%  vehiclesim3 includes capability to select the model
%  vehiclesim4 multiple-line-segment paths capability
%  vehiclesim5 Added side-slope input 2/18/06
%  vehiclesim6 Using 2WD only and John Deere 8320 parameters 5/18/05
%  vehiclesim7 Added integral controller 5/22/06, variable slope path 5/24/06
%  vehilcesim8 Made FF control corrections and LA slope corrections
close all;
% MODEL = 1 vehicle.m
% MODEL = 2 vehicle4WS.mdl
MODEL = 2;
%PATHTYPE = 1 90 degree turns
%PATHTYPE = 2 Lateral shifts
%PATHTYPE = 3 Straight

%SLOPEPATH = 1 5 deg step slope
%SLOPEPATH = 2 sinusoidal slope (-5 deg to +5 deg, 1 period over 200m)

%LASLOPE = 1 look-ahead into SLOPEPATH turned off
%LASLOPE = 2 looks ahead into SLOPEPATH by distance "dla" for upcoming slope
dla = 2;  %Look ahead distance into SLOPEPATH

TURN_RADIUS_LIMIT = 1;
Min_Rad = 30.49; %Westphalen used 15.24m and 30.49 m
global A B C D E F G H delfc delrc a b tau u L lambda theta %KpH KpOT KdOT KiOT

% Path is defined in by using the end points of line segments to specify
% the A-B points of the path
if PATHTYPE == 2
   PATH = [  0 0; 0 27; 3.81 36.1; 3.81 66.1; 0 75.2; 0 102.2];
```

```
elseif PATHTYPE == 1
   PATH = generatePath(2);
else
%    PATH = [  0 0; 0 300]; % Horizontal Line on y-axis
   PATH = [  0 0; 300 0]; % Vertical Line on x-axis
end

% Slope is defined by using the end points of line segments to specify
% the slope at any point along the path
if SLOPEPATH == 1
   dist = 0:1:350;
   SLOPE = [zeros(1,101) 0.0873*ones(1,250)];
   step = 351;
%     dist = 0:9000:9000; %max dist, one increment
%     SLOPE = 0.0873; %5 deg step
%     step = 2; %total number of steps
elseif SLOPEPATH == 2
   dist = 0:1:350; %max dist, 1 m increments
   SLOPE = 0.0873*sin(dist*(2*pi/200)); %5 deg sin wave, period = 200m
   step = 351; %total number of steps
end

% Defining the State Variables
% y(1): lateral velocity in the vehicle coordinate system
% y(2): yaw rate
% y(3): yaw or heading angle
% y(4): rear steering angle
% y(5): x-coordinate of rear axle center in the world coordinate system
% y(6): y-coordinate of rear axle center in the world coordinate system
% y(7): x-coordinate of CG
% y(8): y-coordinate of CG
% y(9): front steering angle
% y(10): x-coordinate of front axle center in the world coordinate system
% y(11): y-coordinate of front axle center in the world coordinate system

% Caf = 226615; % (N/rad) based on Metz's plowed field
% Car = 384476; % (N/rad) based on Metz's plowed field

% Caf = 665837; % (N/rad) based on Metz's meadow field
% Car = 1129470; % (N/rad) based on Metz's meadow field

Caf = 373432; % (N/rad) based on Metz's corn field
Car = 633421; % (N/rad) based on Metz's corn field

m = 12660; % kg
Iz = 27998; % kg-m^2
a = 1.745; % m
b = 1.225; % m

L = a + b;

u = speed; %u = 2; % m/s
tau = 0.1; % s
```

```
lambda = 1.52; % m distance behind front wheel where heading deviation is measured

Wf = m*9.8*b/L;
Wr = m*9.8*a/L;

K = Wf/Caf - Wr/Car;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Set up the path
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
path_ind = 2;
ydif = PATH(path_ind,2) - PATH(path_ind-1,2); %B_Y - A_Y;
xdif = PATH(path_ind,1) - PATH(path_ind-1,1); % B_X - A_X;
[slope,vertical] = CalcSlope(ydif,xdif);
intercept = CalcIntercept(PATH(path_ind-1,1),PATH(path_ind-1,1),slope,vertical);
PathHeading = pi/2 - atan2(xdif,ydif); % Heading angle, psi, is defined as CW
                        % relative to the positive WCS X-axis;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Vehicle Model Parameters:
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
A = (Caf+Car)*(-1/(m*u));
%B = (Caf*a-Car*b-u^2)*(-1/(m*u));
B = (Caf*a-Car*b+u^2*m)*(-1/(m*u)); % Corrected from Mark's Model  122905 See Notes p. (31)
C = (Caf*a-Car*b)*(-1/(Iz*u));
%D = (Caf*a^2-Car*b^2)*(-1/(Iz*u));
D = (Caf*a^2+Car*b^2)*(-1/(Iz*u)); % Corrected from Mark's Model  122905 See Notes p. (31)
E = Caf/m;
F = -Car/m; % Changed sign from Mark model  122805 See Westphalen p. 38-39
G = Caf*a/Iz;
H = Car*b/Iz; % Changed sign from Mark model  122805 See Westphalen p. 38-39
fsa = 0;
rsa = 0;
delfc = fsa*pi/180; % radians
delrc = rsa*pi/180; % radians

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Simulation Initialization
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
rear_steer_off = 1;
SampTime = 0.2; % sec
Steps = simtime/SampTime;
Time = [];
Y = [];
Delf = [];
Delr = [];
Error = [];
RError = [];
PATHSEG = [];
HEading = [];
RError_sum = 0;
RE = [];
lookahead = [];
```

```
LA = [];
OffTrackError = 0;
intOffTrackError = 0;

Front_Look_ahead_distance = 1.5;  %1.5 is original value
Rear_Look_ahead_distance = 1.83; %m Westphalen p. 53
q=round(L/(u*SampTime)+10); %Number of path points to store

% DEFINE SLOPE
theta = SLOPE(1);
if MODEL==1
   if PATHTYPE == 1
      yinit = [0 0 0 0 -b 0 0 0 0 a 0];
   elseif PATHTYPE == 2
      yinit = [0 0 pi/2 0 0 -b 0 0 0 0 a];  % Facing to the Right
   else
      yinit = [0 0 0 0 -b 0 0 0 0 a 0];
   end
else
   if PATHTYPE == 1
      yinit = [0 0 0 0 -b 0 0 0 0 a 0 delfc delrc u theta];
   elseif PATHTYPE == 2
      yinit = [0 0 pi/2 0 0 -b 0 0 0 0 a delfc delrc u theta]; % Facing to the Right
   else
      yinit = [0 0 0 0 -b 0 0 0 0 a 0 delfc delrc u theta]; % Facing up
   end
   options = simset('MaxStep',0.001,'SrcWorkspace','current');
end


h = waitbar(0,'Please wait...');
for i = 1:Steps
   if MODEL == 1
      [t,y] = ode45('vehiclewslopeinput',[0 SampTime],yinit);
   else
      [t,x,y] = sim('vehicle4WSwslopeinput',[0 SampTime],options,[0 yinit]);
   end
   Y = [Y; y];
   Time = [ Time; t+(i-1)*SampTime];
   last = size(y,1);
   % Update which line segment should be used for the path
   LA_Point = [Front_Look_ahead_distance*sin(pi/2-y(last,3)) + y(last,10)...
            Front_Look_ahead_distance*cos(pi/2-y(last,3)) + y(last,11)];

   if (path_ind < size(PATH,1))
     next_segment = checkfornextsegment(path_ind,PATH,LA_Point);
   else
     next_segment = 0;
   end
   if (next_segment)
      %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
      % Update the path to new line segment
      %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
      path_ind = path_ind+1;
      ydif = PATH(path_ind,2) - PATH(path_ind-1,2);%B_Y - A_Y;
```

```
    xdif = PATH(path_ind,1) - PATH(path_ind-1,1);% B_X - A_X;
    [slope,vertical] = CalcSlope(ydif,xdif);
    intercept = CalcIntercept(PATH(path_ind-1,2),PATH(path_ind-1,1),slope,vertical);
    PathHeading = pi/2 - atan2(xdif,ydif); % Heading angle, psi, is defined as CW
                                           % relative to the positive WCS X-axis;
end

if LASLOPE == 1
    cur_ind = ceil(step*(y(last,7)/350)); %selects slope segment
    if cur_ind < 200
        theta = SLOPE(cur_ind); %setting theta for the slope
        theta_control= SLOPE(cur_ind); %setting theta that the FF controller uses
    elseif cur_ind >= 200,
        theta = 0;
        theta_control= 0; %setting theta that the FF controller uses
    end
elseif LASLOPE == 2
    cur_ind = ceil(step*(y(last,7)/350)); %selects slope segment
    if cur_ind < (200 - dla),
        theta_control = SLOPE(cur_ind + dla); %setting theta that the FF controller uses
    elseif cur_ind >= (200 - dla),
        theta_control = 0;
    end
    if cur_ind < 200
        theta = SLOPE(cur_ind); %setting theta for the slope
    elseif cur_ind >= 200,
        theta = 0; %setting theta for the slope
    end
end

% Implement navigation control
CurrentVehicleHeading=OutputCorrecter(y(last,3));
HeadingError = GetHeadingError(PathHeading,CurrentVehicleHeading);
[distance,pt_Cur_LA_x,pt_Cur_LA_z]  = GetPerpenDistance(slope, intercept,...
    PATH(path_ind-1,2),y(last,6),y(last,5),pi/2-y(last,3),...
    vertical,Front_Look_ahead_distance);
OldOffTrackError = OffTrackError;
OffTrackError  = GetOffTrackError(distance,PATH(path_ind-1,2),...
    PATH(path_ind-1,1),PATH(path_ind,2),PATH(path_ind,1),y(last,6),...
    y(last,5),pi/2-y(last,3),Front_Look_ahead_distance);
[distance,pt_Cur_LA_x,pt_Cur_LA_z]  = GetPerpenDistance(slope, intercept,...
    PATH(path_ind-1,2),y(last,6),y(last,5),pi/2-y(last,3),...
    vertical,0);
rearOffTrackError = GetOffTrackError(distance,PATH(path_ind-1,2),...
    PATH(path_ind-1,1),PATH(path_ind,2),PATH(path_ind,1),y(last,6),...
    y(last,5),pi/2-y(last,3),0);
intOffTrackError = intOffTrackError+OffTrackError*SampTime;
delOT=(OffTrackError-OldOffTrackError)/SampTime;
fsa=SSU(HeadingError,OffTrackError,intOffTrackError,delOT);

KpH = 0.4;
KpOT = 0.1;
KdOT = 0.001;
KiOT = 0.0001;
```

```
%  Feed-forward Control****************************************************
%%% Correcting the error of non-zero SS off-track error as seen in James' thesis
      delfc = fsa*pi/180+(K +Wr/Car*(KpOT*Front_Look_ahead_distance+KpH))*sin(theta_control);
% convert from degrees to radians

%  No Feed-forward
%     delfc = fsa*pi/180; % convert from degrees to radians

      delrc = 0; % radians

    % Store states as initial conditions to start out the next simulation
    if MODEL==1
       yinit = y(last,:);
    else
       yinit = [y(last,:) delfc delrc u theta];
    end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [slope,vertical] = CalcSlope(ydif,xdif)
% This block supports an embeddable subset of the MATLAB language.
% See the help menu for details.

% Check to see if vertical
if (ydif == 0)
    slope = 0;
    vertical = 1;
else
    slope = xdif/ydif;
    vertical = 0;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function intercept = CalcIntercept(pt_A_x,pt_A_z,slope,vertical)
% This block supports an embeddable subset of the MATLAB language.
% See the help menu for details.

% Check to see if vertical
if (vertical~=1)
   intercept = pt_A_z - (slope*pt_A_x);
else
   intercept = 0;
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function distance1  =
GetOffTrackError(distance,pt_A_y,pt_A_x,pt_B_y,pt_B_x,pt_Cur_y,pt_Cur_x,Cur_Heading,Look_ahe
ad_distance)
pt_Cur_LA_y = Look_ahead_distance*cos(Cur_Heading) + pt_Cur_y;
pt_Cur_LA_x = Look_ahead_distance*sin(Cur_Heading) + pt_Cur_x;
```

```
area = pt_A_y*pt_B_x - pt_A_x*pt_B_y + pt_A_x*pt_Cur_LA_y - pt_A_y*pt_Cur_LA_x +
pt_B_y*pt_Cur_LA_x - pt_Cur_LA_y*pt_B_x;
if (area > 0)
   right = 0;
else
   right = 1;
end
if (right == 0)
   distance1 = distance;
else
  distance1 = -distance;
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%
function [distance,pt_Cur_LA_y,pt_Cur_LA_x]  = GetPerpenDistance(slope,
intercept,pt_A_y,pt_Cur_y,pt_Cur_x,Cur_Heading,vertical,Look_ahead_distance)

% Project ahead to ahead distance
pt_Cur_LA_y = Look_ahead_distance*cos(Cur_Heading) + pt_Cur_y;
pt_Cur_LA_x = Look_ahead_distance*sin(Cur_Heading) + pt_Cur_x;

temp = pt_Cur_LA_x - (slope*pt_Cur_LA_y) - intercept;
if (temp < 0)
   temp = -(temp);
end
if (vertical ~=1)
   distance = temp/sqrt((slope*slope)+1);
else
  %if it is a vertical line,
  %perpendicular distance is the difference between two x coordinates
   if (pt_Cur_LA_y >=pt_A_y)
      distance = pt_Cur_LA_y - pt_A_y;
   else
      distance = pt_A_y - pt_Cur_LA_y;
   end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%
function HeadingError = GetHeadingError(PathHeading,CurrentVehicleHeading)
% This block supports an embeddable subset of the MATLAB language.
% See the help menu for details.

PrelimHeadingError = PathHeading-CurrentVehicleHeading;

if abs(PrelimHeadingError) > pi %180
   HeadingError = 2*pi - abs(PrelimHeadingError);
   if ((PathHeading > 0) && (CurrentVehicleHeading < 0))
       HeadingError = -HeadingError;
   end
else
   HeadingError = PrelimHeadingError;
end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%
function CorrectedHeading = OutputCorrecter(RawHeading)

temp1=RawHeading/(2*pi);

temp2 = 2*pi*(temp1 - fix(temp1));
if temp2 > pi
   temp2 = (2*pi-temp2);
end
if (-temp2 > pi)
   temp2 = (temp2+2*pi);
end
CorrectedHeading = temp2;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%
function fsa=SSU(HeadingError,OffTrackError,intOffTrackError,delOT)
KpH = 0.4;
KpOT = 0.1;
KdOT = 0.001;
KiOT = 0.0001;

fsa = (180/pi)*(KpH*HeadingError+KpOT*OffTrackError+KdOT*delOT+KiOT*intOffTrackError);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%
function fsa=rSSU(HeadingError,OffTrackError,fsa,KpH,KpOT,Kfff)

% Use a minus sign in front of Kfff since front and rear steer angles are
% define opposite.  01/04/06
fsa = KpH*HeadingError+KpOT*OffTrackError-Kfff*fsa;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%
function next_segment = checkfornextsegment(path_ind,PATH,LA_Point);

% Projects current LA_Point onto the current PATH segment vector; if the
% projected length is greater than the segment length then it is time to
% shift to the next segment. See notes p. (35).
Segvectr = PATH(path_ind,:)-PATH(path_ind-1,:);
vecttoLA = LA_Point-PATH(path_ind-1,:);
SegLength = norm(Segvectr);
projectLength = vecttoLA*Segvectr'/SegLength;

if projectLength >=SegLength
   next_segment = 1;
else
   next_segment = 0;
end
```

E.2 Parameter Setting and Plotting: *VehicleSimDriverSlopes.m*

```
% Name: VehicleSimDriverSlopes
% Purpose: To simulate a vehicle as 4WS controller parameters are changed
% and under different vehicle speeds and under side slope conditions.

clear all

%PATHTYPE = 1 90 degree turns
%PATHTYPE = 2 Lateral shifts
%PATHTYPE = 3 Straight
PATHTYPE = 3;

SLOPEPATH = 1; % 5 deg step slope
% SLOPEPATH = 2; % sinusoidal slope (-5 deg to +5 deg, 1 period over 200 meters)

% LASLOPE = 1;  %look-ahead into SLOPEPATH turned off
LASLOPE = 2;  %looks ahead into SLOPEPATH by distance "dla" for upcoming slope

if PATHTYPE == 1
fprintf(fidout,'PATHTYPE\tSpeed\tTheta\tLASLOPE\tKHEAD\tKpOT\tKiOT\tKFF\tAREA\tMax_Err\tErr
or_Percentage\r\n');
str=sprintf('PATHTYPE\tSpeed\tTheta\tLASLOPE\tKHEAD\tKpOT\tKiOT\tKFF\tAREA\tMax_Err\tError
_Percentage\r\n');
disp(str);
speed = [2 4 6 8];

if PATHTYPE == 1 % 90 degree turn case
elseif PATHTYPE == 2 % Lateral Shifts case
else % Straight Case
   NumCases = 1;

   Case(1).KpOTrack = 0;
   Case(1).KpHEading = 0;
   Case(1).KiOTrack = 0;
   Case(1).K3 = 1;
   initial_simtime = 175;
   top_speed_ind = 4;
end
for case_ind = 1:NumCases
   for speed_ind = 1:top_speed_ind
         simtime = initial_simtime*2/speed(speed_ind);
         [Time,Y,Delf,Delr]=vehiclesim8(speed(speed_ind),SLOPEPATH,simtime,PATHTYPE,...

LASLOPE,Case(case_ind).KpHEading,Case(case_ind).KpOTrack,Case(case_ind).KiOTrack,Case(ca
se_ind).K3);
         [Total_Area,MaxOT,error_segments] = CalcOffTrackArea(Y,Time,simtime,PATHTYPE);

         total_pts = sum(abs(Y(:,6)) >= 0);
         err_pts = sum(abs(Y(:,6)) >= 0.025);
         per_err = (err_pts./total_pts)*100;
         max_err = max(abs(Y(:,6)));
         fox = Y(:,6);

         err = Y(:,6);
         big_err = abs(err) >= 0.025;
         maj_err = err.*big_err;
   end
end
fclose(fidout);
```

E.3 Calculating the Off-Tracking Area: *CalcOffTrackArea.m*

```
function [Total_Area,MaxOT,error_segments] = CalcOffTrackArea(Y,Time,simtime,PATHTYPE)
%  Name: CalcOffTrackArea
%  Purpose: Given output from simulation, to calculate the off-tracking area for that simulation

% Defining the State Variables
% y(1): lateral velocity in the vehicle coordinate system
% y(2): yaw rate
% y(3): yaw or heading angle
% y(4): rear steering angle
% y(5): x-coordinate of rear axle center in the world coordinate system
% y(6): y-coordinate of rear axle center in the world coordinate system
% y(7): x-coordinate of CG
% y(8): y-coordinate of CG
% y(9): front steering angle
% y(10): x-coordinate of front axle center in the world coordinate system
% y(11): y-coordinate of front axle center in the world coordinate system

SampTime = 0.2; % sec
Steps = simtime/SampTime;
MaxOT = 0;
Total_Area = 0;
Total_distance = 0;
new_rear = Y(1,5:6);
at_beginning = 1;
error_segments = zeros(size(Time));

if PATHTYPE == 1
   EndDistance = 248.8;
   % Since for the 90 degree turn course, the vehicle wraps around from
   % the starting point, we need to find where that point is in the Path.
   [error,wrap_ind] = find_nearest_path_point(Y(200:size(Y,1),10:11),[0 0]);
   first_time = 1;
else
   EndDistance = 104;
end

h = waitbar(0,'Please wait...');
for i = 1:Steps
   current_time = SampTime*(i-1); % We will just perform error calculations
```

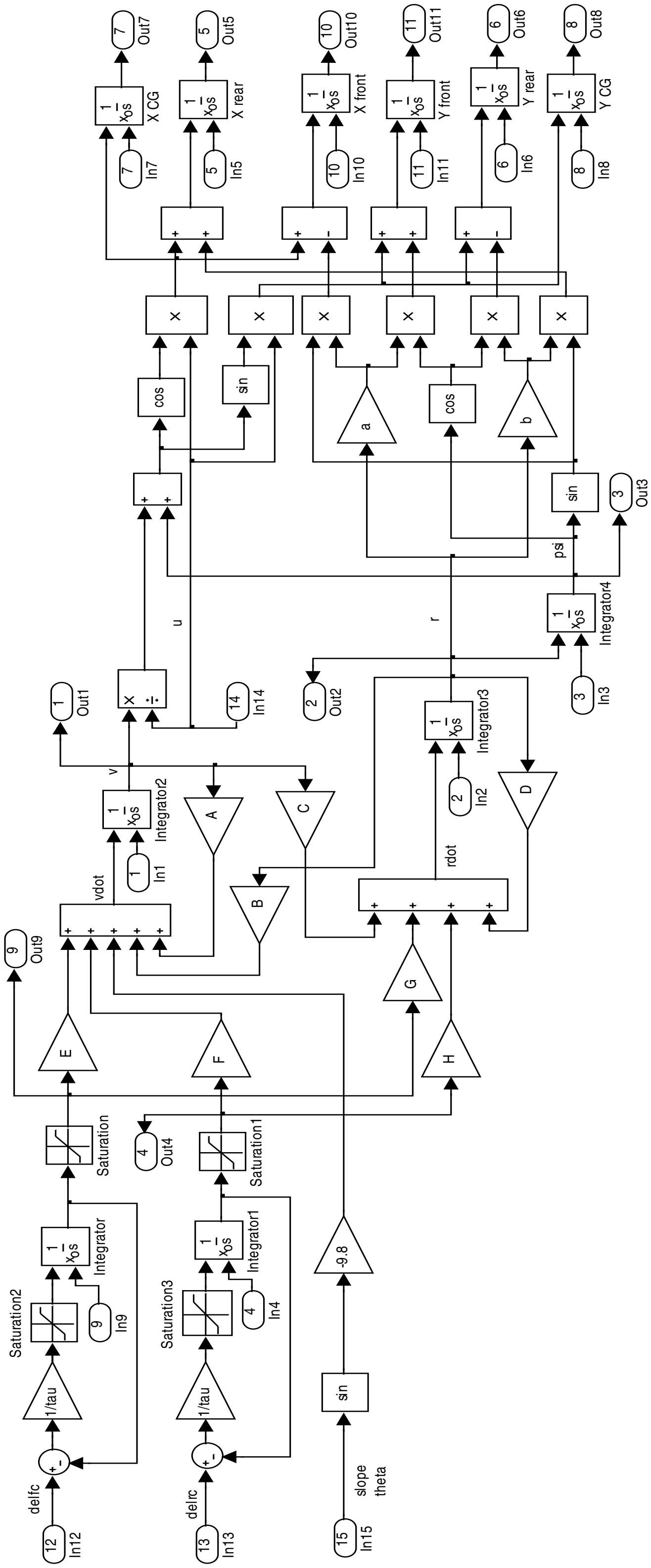
```
% at the SampTime Intervals
cur_ind=find(Time==current_time);
old_rear = new_rear;
new_rear = Y(cur_ind(1),5:6);
distance = norm(old_rear-new_rear);
Total_distance = Total_distance + distance;
if PATHTYPE == 1
   if first_time
      [pre_error,ind]= find_nearest_path_point(Y(1:wrap_ind,10:11),new_rear);
      if ind > 1 && ind < size(Y,1)
         error = findperpdistance(Y(ind-1:ind+1,10:11),new_rear);
      else
         error = pre_error;
      end
      if (wrap_ind - ind < 300) && (Total_distance > 10)
         first_time = 0;
         [pre_error,ind]= find_nearest_path_point(Y(wrap_ind-400:size(Y,1),10:11),new_rear);
         ind = ind+(wrap_ind-401);
         if ind > 1 && ind < size(Y,1)
            error = findperpdistance(Y(ind-1:ind+1,10:11),new_rear);
         else
            error = pre_error;
         end
      end
   else
      [pre_error,ind]= find_nearest_path_point(Y(wrap_ind-400:size(Y,1),10:11),new_rear);
      ind = ind+(wrap_ind-401);
      if ind > 1 && ind < size(Y,1)
         error = findperpdistance(Y(ind-1:ind+1,10:11),new_rear);
      else
         error = pre_error;
      end
   end
else
   [pre_error,ind]= find_nearest_path_point(Y(:,10:11),new_rear);
   if ind > 1 && ind < size(Y,1)
      error = findperpdistance(Y(ind-1:ind+1,10:11),new_rear);
   else
      error = pre_error;
   end
end
if ~at_beginning
   area = 0;
   if (Total_distance < EndDistance) % Stop accumulating error at the end of the path
      if error > 0.304 % Then off-tracking segment
         area = 0.304*distance;
         error_segments(cur_ind) = 1;
      elseif error > 0.304/2
         area = error*distance;
         error_segments(cur_ind) = 1;
      end
      Total_Area = Total_Area+area*2; % Multiply by 2 for two tires
      MaxOT = max([MaxOT error]);
   end
```

```
    else
        if Total_distance > 4.3
            at_beginning = 0;
            area = 0;
            if error > 0.304 % Then off-tracking segment
                area = 0.304*distance;
            elseif error > 0.304/2
                area = error*distance;
            end
            Total_Area = Total_Area+area;
            MaxOT = max([MaxOT error]);
        end
    end
    waitbar(i/Steps,h);
end
close(h);
err_ind = find(error_segments);

function dist_error=findperpdistance(Y,rearpoint);

[error,ind] = find_nearest_path_point([Y(1,:);Y(3,:)],rearpoint);
if ind == 2
    ind = 3;
end
tempx(1) = Y(2,1);
tempy(1) = Y(2,2);
tempx(2) = Y(ind,1);
tempy(2) = Y(ind,2);
newxr = rearpoint(1);
newyr = rearpoint(2);

% from http://mathworld.wolfram.com/Point-LineDistance2-Dimensional.html
dist_error = abs(((tempx(2)-tempx(1))*(tempy(1)-newyr)-((tempx(1)-newxr))...
   *(tempy(2)-tempy(1)))/sqrt((tempx(2)-tempx(1))^2+(tempy(2)-tempy(1))^2));
```

E.4 Locating the Points Along the Path: *find_nearest_path_point.m*

```
function [error,ind] = find_nearest_path_point(front_path,rear_point);
%  Name: find_nearest_path_point
%  Purpose: To find the next point in a Path to the point given

N = size(front_path,1);
dif = front_path - ones(N,1)*rear_point;
distance = sqrt(dif(:,1).^2+dif(:,2).^2);
[error,ind] = min(distance);
```

APPENDIX F. MATLAB SIMULINK VEHICLE MODEL



REFERENCES

Abidine, A. Z., B. C. Heidman, D. J. Hills, and S. K. Upadhyaya. 2004. Autoguidance system operated at high speed causes almost no tomato damage. *California Agriculture*, 58(1): 44-47.

ASAE Standards. 2004. S383.1: Roll-Over Protective Structures (ROPS) for Wheeled Agricultural Tractors. St. Joseph, MI: ASAE**.**

Baack, E. J. 2003. A zero slip vehicle dynamics model for virtual reality applications. MS Thesis. Ames, IA: Iowa State University, Department of Mechanical Engineering.

Bernard, J. E. 2005. ME 549: Vehicle Dynamics class notes. Ames, IA: Iowa State University, Fall Semester.

Constable, G., and B. Somerville. 2003. *A Century of Innovation: Twenty Engineering Achievements that Transformed Our Lives*. Washington, D.C.: National Academy of Engineering.

Cox, S. 2002. Information technology: the global key to precision agriculture and sustainability. *Computers and Electronics in Agriculture.* 36 (2): 93-111.

DE. 2005. GREENSTAR Guidance Parallel Tracking and AutoTrac Assisted Steering Systems Operator's Manual. Moline, IL: Deere and Company.

DE. 2006. John Deere 8030 Series Tractors. Moline, IL: Deere and Company.

Evans, B. S. 2006. Automatic steering control system model and simulation to analyze performance of off-road vehicles subject to sloped terrain and vehicle parameter changes. MS Thesis, Ames, IA: Iowa State University, Department of Mechanical Engineering.

Finck, C. 2004. Wow, It Works! *Farm Journal*. January.

Gillespie, T. D. 1992. *Fundamentals of Vehicle Dynamics*. Warrendale, PA: Society of Automotive Engineers, Inc.

Lambert, D., and J. Lowenberg-DeBoer. 2000. Precision agriculture profitability review. Site Specific Management Center. Purdue University. West Lafayette, IN: Purdue University Available at: www.agriculture.purdue.edu/ssmc. Accessed: 25 June 2006.

Metz, L. D. 1993. Dynamics of Four-Wheel-Steer Off-Highway Vehicles. SAE Paper No. 930765. Warrendale, PA: SAE.

Miller, M. A., and B. L. Steward. 2002. Control and Evaluation Methods for Multi-Mode Steering. In: Proc. Automation Technology for Off-Road Equipment, 357-366. St. Joseph, MI: ASAE.

Minnesota Beef Council. 2006. Beef Trivia. Available at: www.mnbeef.org. Accessed 12 July 2006.

Nise, N. S. 2004. *Control Systems Engineering*. New York, NY. John Wiley and Sons.

Pettygrove, G. S., R. E. Plant, and W. R. Reinert. 2000. Precision agriculture can increase profits and limit environmental impacts. *California Agriculture*. 54(4) 66-71.

Shearer, J. L., B. T. Kulakowski and J. F. Gardner. 1997. *Dynamic Modeling and Control of Engineering Systems.* Upper Saddle River, NJ. Prentice-Hall.

Smith, D. 2003. Guiding Principles. AgWeb. Available at: www.agweb.com. Accessed: 2 July 2006.

Steward, B. L. 2006. Four-wheel Steer Modeling and Simulation Notes. Ames, IA: Iowa State University. January.

Steward, B.L. and M. L. Westphalen. 2006. Development and evaluation of an active rear wheel steering system. *Final Project Report, John Deere DMW.* Ames, IA: Iowa State University.

Swanghart, H., and K. Rott. 1984. The influence of the tire tread on the rolling resistance and steering forces on undriven wheels. In Proc. 8th International Conference of the Society of Terrain Vehicle Systems, 855-888. Cambridge, UK: ISTVS.

Watson, M., and J. Lowen-DeBoer. 2002. Who will benefit from GPS Auto Guidance in the Corn Belt? Site Specific Management Center. West Lafayette, IN: Purdue University. Available at: www.agriculture.purdue.edu/ssmc. Accessed 10 July 2006.

Westphalen, M.L. 2004. Development of topographic maps and rear steering control for an agricultural vehicle through incorporation of posture and attitude measurements. MS Thesis. Ames, IA: Iowa State University, Department of Agricultural and Biosystems Engineering.