ABSTRACT

PASSIVE GYROSCOPIC STABILIZATION OF A SOLID BODY

SIMULATED USING A COMPUTATIONAL SPATIAL MODEL

By

Jonathan Hull

May 2015

This thesis examines the spatial motion of a simple solid body with a gyroscopic stabilizer following a defined path. The body's motion is determined using a simplified spatial computational spatial model. The track was designed to be equivalent to a single swerving maneuver on a roadway and the body's motion was analyzed at a multitude of nodes before and after the addition of a gyroscope. The question was to determine whether the addition of a fixed gyroscope could be used to passively reduce the magnitude of the subsequent spatial motion of a body during a maneuver. The simulation found that the body's motion overwhelmingly dominated that of the gyroscope due to the very large moments of inertia of the body. While gyroscopic effects were seen, the computational model predicts that they will minimally affect the body.

PASSIVE GYROSCOPIC STABILIZATION OF A SOLID BODY SIMULATED USING A COMPUTATIONAL SPATIAL MODEL

A THESIS

Presented to the Department of Mechanical and Aerospace Engineering

California State University, Long Beach

In Partial Fulfillment

of the Requirements for the Degree

Master of Science in Mechanical Engineering

Committee members:

David Stout, Ph.D. (Chair)
Jalal Torabzadeh, Ph.D.
Ramin Esfandiari, Ph. D.

College Designee:

Antonella Sciortino, Ph.D.

By Jonathan L. Hull

BS, 2006, California Polytechnic State University, San Luis Obispo

May 2015

ACKNOWLEDGEMENTS

Firstly, I would like to thank Dr. Stout for all of his help, advice, and guidance as this project would have been near impossible to complete without him. I have sincerely appreciated working with him, and he shall always have my highest esteem and respect.

Secondly, I would like to thank Dr. Hamel and the CSULB Department of Mechanical and Aerospace Engineering. Dr. Hamel was immensely helpful to start this effort, seemingly so long ago, and the Department made it all possible.

Lastly, but not least, I would like to thank my family. They have always supported my efforts and encouraged me to continue by looking for that light at the end of the tunnel even when I doubted it was there. I would have long ago gone adrift without their support and care.

TABLE OF CONTENTS

Page

CHAPTER Page

LIST OF TABLES

TABLE Page

LIST OF FIGURES

FIGURE Page

FIGURE Page

LIST OF ABBREVIATIONS

CG	Center of Gravity
GCS	Global Coordinate System
LCS	Local Coordinate System
TCS	Trajectory Coordinate System

LIST OF NOMENCLATURE

A	Transformation Matrix
bN	Total Quantity of Boundary Nodes
C_t	Rotational Damper Constant
$F_{A,R}$	Applied Forces Vector, Orthogonal
$F_{A,\theta}$	Applied Forces Vector, Rotational
F_C	Centripetal Force
$F_{Q,R}$	Quadratic Forces Vector, Orthogonal
$F_{Q,\theta}$	Quadratic Forces Vector, Rotational
g	Earth Gravitational Constant
G	Unit Vector Transformation Matrix
$\bar{G}$	Unit Vector Transformation Matrix in LCS
h	Body Support Height
H	Height From Point O to CG
i	Node
i_{bN}	Boundary Node
i_{rN}	Radial Node
i_{tN}	Tangent Node
$\bar{I}_{\theta\theta}$	Body Inertia Tensor in LCS

$I_{\theta\theta_{Gyro}}$	Inertia Tensor of Gyroscope
J_{ϕ}	Polar Moment of Inertia of Body in Direction of Phi
k	Track Curvature
K_t	Rotational Spring Constant
L1	Body Length
L2	Body Width
L3	Body Height
m	Mass of Body
M	Slope of Tangent Track
M_O	All Moments About Point O in FIGURE 7
M_{θ}	Gyroscopic Moment in Theta
M_{ϕ}	Gyroscopic Moment in Phi
M_{ψ}	Gyroscopic Moment in Psi
N	Nodal Quantity
P	Length of Hypotenuse of vt-P-R
R	Radius of Curved Track
$\ddot{R}$	System Resultant Accelerations, Orthogonal
rN	Total Quantity of Radial Nodes
s	Arc Length within Zone
S	Arc Length
sB	Boundary Distance
S_B	Total Length of Boundary

S_{Lane}	Width of Lange Change
S_{x2}	Total X Length of Zone 2
S_{x3}	Total X Length of Zone 3
S_{y2}	Total Y Length of Zone 2
S_3	Total Length of Zone 3
t	Time
tN	Total Tangent Nodes
T	Kinetic Energy of Gyroscope
$\bar{u}$	CG Position Vector in LCS
v	Speed of Body
x1	LCS X Position Coordinate
$x2$	Total X-length of Zone 2
X	GCS X Position Coordinate
y1	LCS Y Position Coordinate
$y2$	Total Y-length of Zone
Y	GCS Y Position Coordinate
z1	LCS Z Position Coordinate
Z	GCS Z Position Coordinate
α	Co-Interior Angle of Triangle in FIGURE 6
β	Co-Interior Angle of Triangle in FIGURE 6
γ	Co-Interior Angle of Triangle in FIGURE 6
γ_{Gyro}	Angular Orientation of Gyroscope

$\dot{\gamma}_{Gyro}$	Angular Velocity of Gyroscope
ζ	Angle of Rotation of Tangent Track in FIGURE 6
ε	Initial Roll Perturbation
θ	Pitch
$\dot{\theta}$	Pitch Rate
$\ddot{\Theta}$	System Resultant Accelerations, Rotational
Θ_{Body}	Angular Orientation Vector of Body
θ_G	Gyroscopic Pitch
$\dot{\theta}_G$	Gyroscopic Angular Velocity
π	Pi
ϕ	Roll
$\dot{\phi}$	Roll Rate
$\ddot{\phi}$	Roll Acceleration
ψ	Yaw
$\dot{\psi}$	Yaw Rate
$\ddot{\psi}$	Yaw Acceleration
Ω_{Body}	Angular Velocity Vector of Body
$\overline{\Omega}_{Body}$	Angular Velocity Vector of Body in LCS
$\overline{\Omega}_{Gyro}$	Angular Velocity Vector of Gyroscope in LCS

CHAPTER 1

INTRODUCTION

It is an all too common experience for the modern automotive driver to encounter a cargo vehicle on a public roadway that has overturned or experienced some sort of accident. According to US Department of Transportation data from 2012 [1], rollover caused 5 percent of all fatal crashes and 3 percent of all nonfatal crashes involving large trucks. In terms of large truck crashes by severity [1], this corresponds to 152 fatalities and 4,000 injuries for 2012. Further, truck rollover caused an additional 4,000 accidents resulting in property damage only [1].

As such a seemingly common event leads to so much damage and personal tragedy for those involved, one considers what might be done to help society? One could reasonably argue that it would be a great benefit to the driving, and consuming, public if a means were engineered to help reduce this type of accident. A mechanical stabilizing device could be an ideal means to effect accident reduction as rollover is caused by a transient reduction of vehicular stability. Further, if this stabilizer was passive in its effect, then vehicles could be equipped such that the driver would not need to provide additional input or alter his driving technique. In this way, the stabilizer could provide an additional type of redundancy to the vehicle's existing safety systems.

However, it is a difficult problem to design a passive mechanical stabilizer to alter a large vehicle's dynamics. Fortunately, history lends some guidance and one

readily finds past attempts to use gyroscopes as both passive and active mechanical stabilizers in road vehicles, ships, and rail cars. These past attempts have had varying degrees of success and acceptance but have not specifically been applied to tractor-trailers. This thesis proposes a means by which to evaluate a passive gyroscopic stabilizer using a simplified computational body model. The proposed model simulates the system spatially using computational dynamics and allows an evaluation to be made of the body's performance before and after the addition of a simple gyroscope. Further, the proposed model is intended to be conceptually and mathematically approachable in its derivation for the benefit of those who might use it and expand upon it in the future.

1.1. Inspiration for Gyroscopic Passive Dampening

Experienced bicyclists and motorcyclists learn in the course of their training that their vehicle's trajectory is opposite in direction to the motion of the steering column. For example, should a motorcyclist wish to move left he must steer right. The reverse is true for moving right, and a likewise situation exists for bicyclists moving at high speeds. This phenomenon is explained via gyroscopic precession and nutation caused by the vehicle's wheels rotating with a high angular velocity. Motorcyclists have an additional gyroscopic effect induced by the engine crankshaft while bicyclists have a lesser corollary via the drivetrain cranks, and chain rings. In both cases, the vehicle wheels are composed of qualitatively large masses distributed at a distance from a central spindle and rotating very quickly. In the study of dynamics such a device is considered a gyroscope and has unique properties while rotating about its spindle. A gyroscope will attempt to maintain its initial inertial reference frame after rotation. This is to say that the gyroscope resists motion that is contrary to maintaining the initial position of its axis of

rotation after it has begun rotating. The effect of this phenomenon is an induced moment that is mutually orthogonal to the input motion. The classic classroom example of this is the student holding a bicycle wheel forthright by its axle while sitting in a rotatable chair. After a few turns of the wheel by the teacher, the student is able to rotate himself and his chair simply by twisting the now spinning wheel. This effect is the same as that experienced by both motorcyclists and bicyclists with the motorcyclist's experience amplified due to the relative high speeds and masses involved.

Therefore, in expanding upon this common experience, it is wondered whether a gyroscope can be used to passively induce a moment via a body's motion that is contrary to that motion? If the body's motion would be such that it would cause the body to overturn, as in the case of a vehicle, then a contrary moment may help to lessen the potential to overturn. Therefore, the gyroscope could be used as a passive dynamic stabilizing device for a large tractor-trailer.

1.2. Problem Design

The scope of this thesis is to develop a simplified computational model that can assess the performance of a gyroscope fixed within a moving body, and whether this has a general positive effect on undesirable body motions passively. The body shall be a simple rectangular prism with dimensions and mass mimicking that of a commercially available American tractor-trailer. Undesirable body motions are those motions that would cause the body to become singular with respect to its path, as in the body overturning. A reasonable assessment of the body's status and whether its disposition has either improved, worsened, or is a mix of the two can be determined by analysis of the body's moments. Chapter 3 presents the computational process necessary for this

assessment, input assumptions, and derivations of the preprocessor. In Chapter 4, the body's spatial equations of motion are derived for the main processor, while Chapter 5 derives the gyroscope's spatial equations of motion used in the gyroscopic processor. Chapter 6 demonstrates how the body and gyroscopic models are combined to determine the resultant with the presentation of those results. Finally, Chapter 7 presents concluding thoughts and future directions for continued research. The computational model is implemented in MATLAB using m-code script files whose source code is presented in the appendices.

CHAPTER 2

LITERATURE REVIEW

Throughout the history of ground vehicle design there has always been the issue of the vehicle's propensity to overturn during a maneuver when conditions permit. This is also a potential hazard for a vehicle's trailer, and in the case of commercial cargo trailers, significant injury, death, and loss of property can occur if the trailer overturns. Accordingly, significant effort has been placed into creating tractor-trailer stability models and to this end work has generally focused on creating a system of equations to adequately model the system with subsequent verification of that model. Some elementary approaches use a quasi-static summation of the body moments [2] with and without suspension compliance [3]. Other models focus on modeling the system [4] by means of mass-acceleration and free-body diagramming techniques to generate a set of linear differential equations. Other work has developed a set of differential equations of motion based on the system's kinematics with the addition of multiple coordinate systems and a control model [5].

In parallel, and unrelated to the development of tractor-trailer stability models, is the issue of rail car stability and rollover. Much like tractor-trailers, rail cars are also prone to the hazard of overturning in certain conditions. Modeling approaches in this realm have taken similar lines of thought as those of trailers, but with multibody spatial analysis [6] and the use of multiple coordinate systems [7]. Further work has focused on

the induced moments of the cars' wheels and analyzed by a numerical processor [8].

While the currently available models help to predict vehicle roll, for both the tractor-trailer and rail car, they do not offer solutions for amelioration. In the case of tractor-trailers, currently available commercial products for roll control tend to focus on electronic control by interfacing into the vehicle's braking system [9]; rail cars do not have similar turnkey products available. However, since the early 20th century various parties have worked on providing a solution to vehicle roll stability using gyroscopes. Such technology has been implemented in the Brennan monorail, Schilovski gyro car, and the Ford Gyron to name but a few [10]. Gyroscopic stabilization technology has also been implemented in oceangoing vessels [10,11] to control the ship's roll due to waves. Much like the development of ground vehicle and rail car roll models, gyroscopic stabilization models have focused on development of a set of differential equations of motion [11] specific to the device.

Regarding the state of the art of tractor-trailers, no known model exists that combines the elements of all three independent efforts. A computational trailer spatial model has not been combined with the track trajectory assumption of the rail car, nor combined with a passive gyroscopic stabilizer. Additionally, while existing efforts have independently created portions of the modelling elements presented in this thesis, their derivations tend towards less expedient implementation and usage. The model presented herein addresses the issues of simplifying a trailer analytical model while assessing its spatial performance with and without a passive gyroscopic stabilizer. The proposed model builds upon well-known concepts and combines them in a novel fashion. Further,

the derivations presented address the issue of expediency by permitting their implementation in modular, computational processors.

CHAPTER 3

COMPUTATIONAL PROCESS AND PREPROCESSOR DERIVATIONS

3.1. Computational Process Introduction

To begin development of the analytical model presented in this thesis, the general computational process and ideology were adapted from Shabana's work on railroad dynamics [6]. The primary inspiration taken from Shabana's work is in the discretization of a defined path geometry to provide a body with a motion trajectory and motion data. The computational process and associated MATLAB functions, as implemented, are shown in Figure 1.

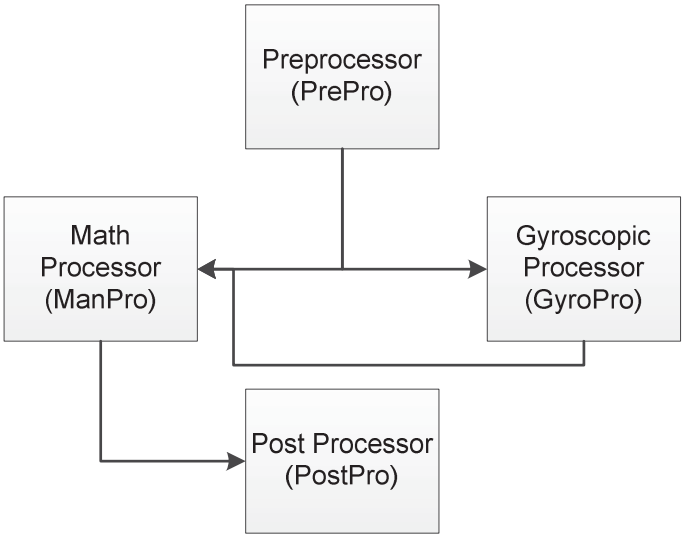

FIGURE 1. Computational process showing that the preprocessor prepares a data array that is read by both the main processor and the gyroscopic processor simultaneously. The gyroscopic processor in turn feeds its data array to the main processor for completion of the final data array, which is formatted for viewing, by the post processor.

The body's resultant motion data is calculated via a custom written math processor run in MATLAB whose input data is provided by a preprocessor. The preprocessor's calculations are determined a priori by the assumptions of the general model. The math processor's output is in turn passed into the post processor for generating plots and formatting data for review. In parallel with the math processor is the gyroscopic processor which uses the same input data as the math processor to augment the main data for the resultant solution. In this way, the final output data of the body with the gyroscopic stabilizer can be compared to data without the gyroscopic stabilizer as the outputs are segregated.

The first block in the process flow shown in Figure 1 is the preprocessor. The preprocessor's data output is directly determined from the assumptions applied to the general model and is further determined by direct application of geometric relationships. The preprocessor calculates 11 outputs for use as inputs to the math and gyroscopic processors and are detailed in Table 1.

This model is unique from previous work by its combination of several topics that have been treated separately and by its focus on simplification. Previous tractor-trailer modeling efforts [3, 4] have focused on generating 2D equations of motion for the system via mass-acceleration, free-body diagramming, or quasi-static assumptions. Such work also neglected the induced spatial body moments and a vehicle path trajectory. Previous work on rail cars applied a discretized track model to a rail car by using global, local, and trajectory coordinate systems [6]. While such an approach simplifies concerns for later application of constraints, it necessitates calculations of multiple coordinate transfers that tend towards computational burden.

Finally, existing work on gyroscopic stabilization has focused on complex differential equation analysis of the bodies without application via a discretized computational approach [11]. The approach presented in this thesis attempts to address all issues simultaneously by use of a model that is highly approachable to the user in terms of its analytical complexity but yields relevant performance data for a gyroscopic stabilizer within a trailer. The issue of body motion, resultant body effects, and a means by which to pacify that motion are considered all while using a model that is still qualitatively approachable from a user's perspective.

TABLE 1. Preprocessor Variables and Units

Processor Variable	Description	Units	Math Notation
N	N	Not Applicable	N
S	Arc Length	in	S
CH	Radius of curvature	in^{-1}	1/R
PHI	Roll	rad	ϕ
THETA	Pitch	Rad	θ
PSI	Yaw	Rad	ψ
X	X coordinate	In	X
Y	Y coordinate	In	Y
Z	Z coordinate	In	Z
DPHI	Roll rate	rad/sec	$\dot{\phi}$
DPSI	Yaw Rate	rad/sec	$\dot{\psi}$

3.2. General Assumptions, Body Geometry, and Coordinate System

To simplify the model into a suitable form for investigation of a gyroscopic stabilizer, it is assumed that the body is a simple rectangular prism, at some distance from

a plane, and connected to a defined path. The full list of assumptions is detailed in Table 2 [12].

TABLE 2. Computational Model Assumptions

Assumption No.	Description	Definition
1	Body Shape	Simple Rectangular Prism
2	Body Dimensions and Mass	1988 Pines 48ft tractor trailer
3	Path	Idealized Swerving
4	Body Path Coupling	Massless
5	No Pitch	Body is assumed to roll and yaw only, pitch is considered zero
6	Speed	55 mph (968 in/sec)
7	Maneuver Time	5 seconds
8	Lane Distance	12 feet [13]

As the purpose of this model is to determine the effect of a gyroscopic stabilizer in a body for trailer applications, the simple rectangular prism is modeled using basic dimensions and mass from a 1988 Pines tractor-trailer [12]. This is done in order to create a highly simplified tractor-trailer geometry. The body is offset from the XY ground plane by a distance h that is to account for the suspension system offset. The distance h is assumed massless and is used as a means by which to avoid calculating multiple coordinate transfers. Note that if distance h were not massless then its rotational inertia would have to be considered and a trivial origin offset within the body's inertia tensor would not suffice.

In a method similar to previously developed models [6], the general model uses multiple coordinate systems; a local coordinate system (LCS) is attached to the moving

body and a global coordinate system (GCS) is attached to a global origin. However, this model differs from previous designs [6] by the lack of a trajectory coordinate system (TCS) and associated transformations. Within the current model, the motion of the body in the GCS is determined via Euler angle transformations of the body's motion in the LCS [14]. In this way, the body's motion can be determined within its own coordinate system from the track geometry and global assumptions while the motion is then mapped back to the GCS through a single Euler angle transformation.

To provide simplification to the model, the LCS origin is attached to the XY ground plane and the body is offset by a distance as explained. This is done in order to remove another set of transformations and simplify the calculations. The alternative would be to attach the TCS to the track, transform the trajectory defined by the TCS to the LCS, and then finally transform the body's motion in the LCS to the GCS. Such an approach can increase the computational complexity without necessarily adding greater insight into the overall gross behavior.

Finally, the current model differs from previous designs by the use of a rotary spring and damper as shown in Figure 2. The spring and damper add a higher degree of fidelity to the model as these devices are encountered in actual applications. Further, the use of a spring and damper enables the body's motion to both be initiated and to avoid a numerical singularity during the simulation. Without a spring, once the body engages in motion under gravity the body will continue to fall until it encounters the XY ground plane; this would be represented by a numerical singularity as the body has now effectively crashed in physical terms. Figure 2 and Figure 3 show the general body model and its relation to the GCS.

3.3. Track Geometry and Discretization

Figure 4 shows the design of the predefined track and associated nodal discretization. The equations of motion evaluate the body at a single instance, and in order to generate a motion history, it is necessary to evaluate the body at multiple instances. This is accomplished through discretization of the track into an arbitrary number of nodes and evaluating the equations of motion at each node. The number of nodes can be defined just prior to a simulation run with a higher number of nodes leading to smoother motion topography at the expense of computational time.

The track is designed to simulate an idealized single object avoidance maneuver performed by a body while moving. The body is initially travelling along a straight path, makes a sudden swerve to the left at some radius R, and continues along a new straight path that is tangent to the exit of the radial curve. The path is broken into four segments, zone 1, zone 2.1, zone 2.2, and zone 3. These zones define the initial straight boundary, curve entrance, curve exit, and the exit tangent [6]. The curve radius can be arbitrarily determined and ultimately depends on the actual body's configuration. For the simulation in this thesis, the radius is arbitrarily selected to provide a measurable response.

As the body travels along the track, it is assumed that the body remains tangent to the track's geometry. Further, since the speed and maneuver time are defined in the assumptions, the distance traveled per node is determined from geometric relationships. Note that zone 2 is composed of segments 2.1 and 2.2; this is done for computational implementation of the model.

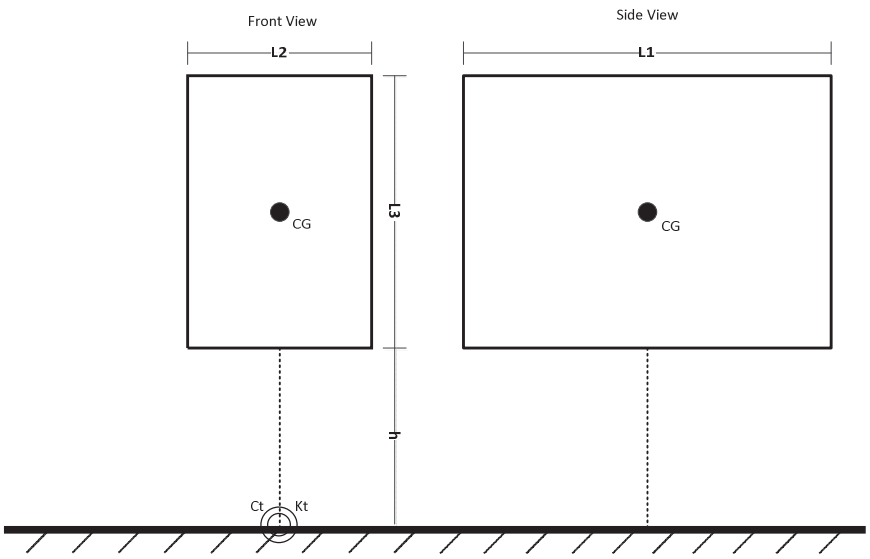

FIGURE 2. Diagram representation of body above the ground plane and connected via a massless offset with a rotary spring and rotary damper.

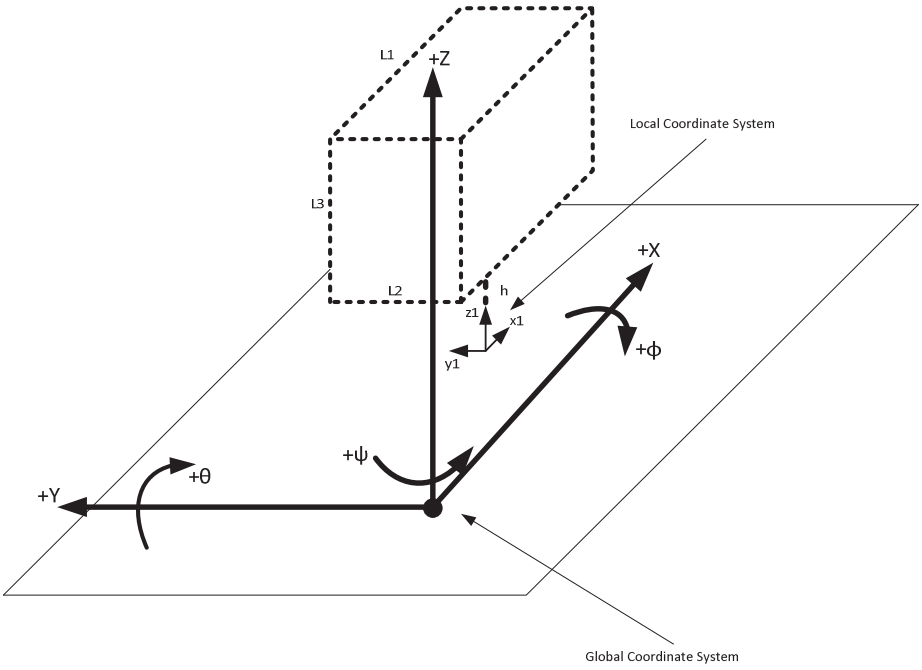

FIGURE 3. Diagram of body in space showing the relation between the global coordinate system and the local coordinate system. The body is offset from the global coordinate system by a massless connection while the local coordinate system is attached to the XY plane.

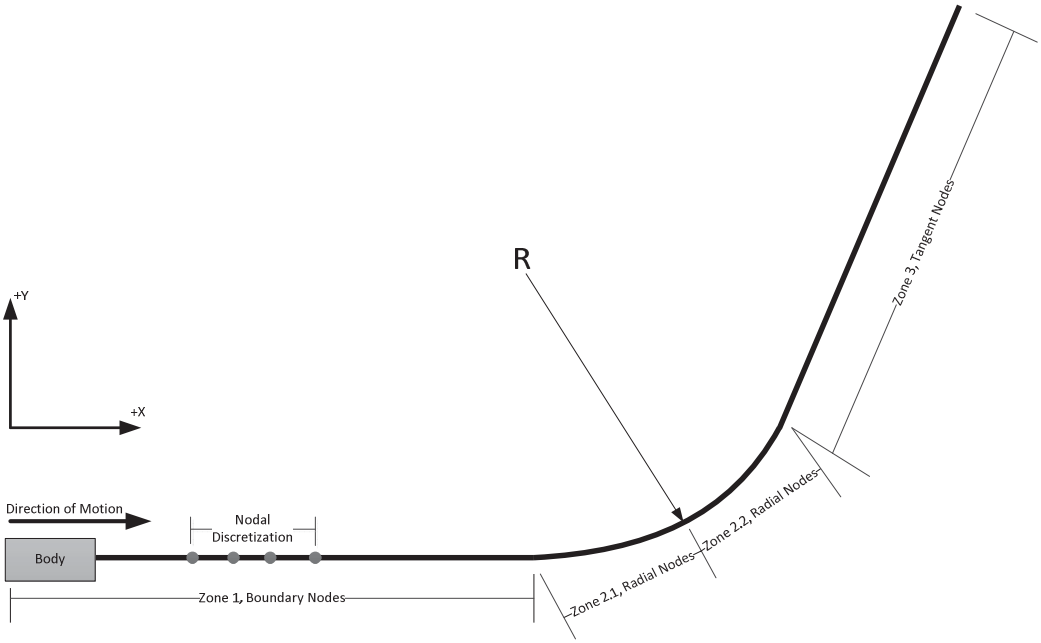

FIGURE 4. Diagram of the body moving along the associated track zones and nodal discretization. The track is broken into three zones for computational purposes and geometric analysis.

3.4. Track Geometry

As described within this thesis, the preprocessor defines eleven variables for use in the main processor. Of these eleven, the X, Y, and Z coordinates need to be determined first as other preprocessor variables rely upon the body's XYZ location. Since the body is assumed to be traveling along a perfectly flat XY plane, and the GCS is attached to that plane, the Z coordinate is always zero. In order to calculate the nodal XY positions, it is necessary to apply descriptive geometry to the track as shown in Figure 5.

The schematic representation of the track in Figure 5 can be considered as consisting of three elements connected together. A single straight segment is connected to another straight segment, at some arbitrary slope, via a tangent radial curve with some arbitrary radius.

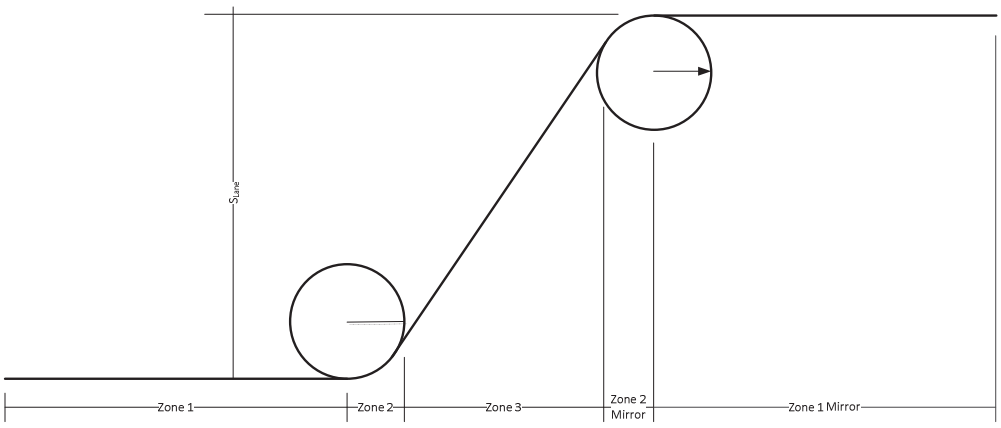

FIGURE 5. Diagram showing the general geometry of the track and associated zones.

These three curves compose zones 1, 2, and 3. In order to fully define the motion within the model, it is necessary to mirror zones 1 and 2. Mirroring zones 1 and 2 allows the body to establish its geometric boundary conditions. Without such an assumption it would not be possible to establish entry and exit points within the model based on the geometry. To define the body on the track an assumed lane distance, body speed, and maneuver time are made. The lane distance S_{LANE} is equal to 12 feet which is the center-to-center distance of two freeway lanes in California [13]. The speed is assumed as 55 miles per hour, and the maneuver time assumed as 5 seconds.

3.5. Track Nodal Coordinates, Zone 1

As the path is in line with respect to X in zone 1 and zero with respect to Z, the only coordinate to calculate is X as Y is always equal to zero. Therefore, the initial X ordinate at node 1 is zero and is initialized by:

$$x_1(i_{bN} = 0) = 0 \tag{1}$$

Subsequent X ordinates are calculated by dividing the assumed zone 1 length over the

desired number of nodes and summing recursively:

$$x_1(i_{bN} + 1) = x_1(i_{bN}) + \frac{sB}{bN} \tag{2}$$

By dividing the assumed length of zone 1 over the desired nodal quantity, zone 1 is discretized with the nodal quantity determining the discretization width.

3.6. Track Nodal Coordinates, Zone 2

Zone 2 is composed of two segments, the radial entrance and the radial exit [6], and these are labeled as zones 2.1 and 2.2 respectively. It is necessary to split zone 2 into two halves in order to simplify dependent calculations within the preprocessor. Unlike zone 1, zone 2 is circular and is composed of both X and Y coordinates. For computational implementation, zone 2 is first discretized and then split into sub-zones 2.1 and 2.2.

The respective X and Y distances of all of zone 2 can be calculated via trigonometric relations using the assumed lane distance, speed, and maneuver time. Figure 6 shows the trigonometric constructions necessary to calculate the X and Y lengths of zone 2. Firstly, to determine the X and Y lengths of zone 2, it is necessary to determine angle ζ which is the angle swept by the used radial portion of the track. Using ζ, it is then possible to find the respective zone lengths through elementary trigonometric relations.

Per Figure 6, we utilize right triangle vt-P-R and right triangle R-R-$\frac{P}{2}$ rotated by angle ζ and containing interior angle β. The hypotenuse of this right triangle R-R-$\frac{P}{2}$ is exactly half that of right triangle vt-P-R.

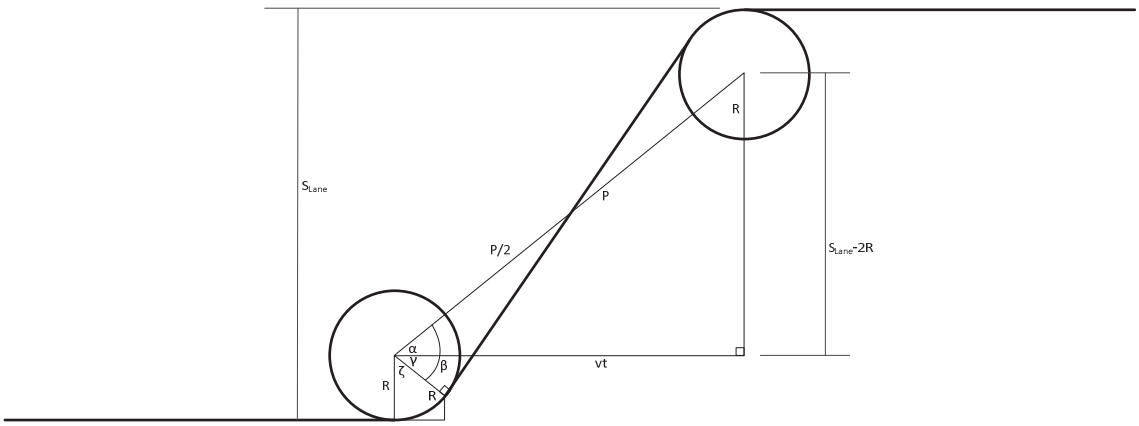

FIGURE 6. Trigonometric construction to calculate XY components of radial curved track section of Zone 2.

Angle β in Figure 6 is found from triangle R-R-$\frac{P}{2}$:

$$\beta = \cos^{-1}\left(\frac{2R}{\sqrt{(vt)^2 + (S_{Lane} - 2R)^2}}\right) \tag{3}$$

Angle α is found from right triangle vt-P-R:

$$\alpha = \tan^{-1}\frac{S_{Lane} - 2R}{vt} \tag{4}$$

However, per Figure 6, ζ and γ form a right angle and therefore after rearranging and substituting terms:

$$\zeta = \frac{\pi}{2} - (\beta - \alpha) \tag{5}$$

With angle ζ and the known radius of curvature, the x-length of zone 2 is found by the sine of ζ:

$$x2 = R \sin\zeta \tag{6}$$

The y-length of zone 2 is determined from the difference between the radius and the

y-component of the swept length:

$$y2 = R(1 - \cos\zeta) \tag{7}$$

X-component discretization within zone 2 is initialized at the first zone 2 node, which is also the last zone 1 node. Subsequent nodes are determined by dividing the entire x-length of zone 2 by the desired nodal quantity and summing recursively:

$$x_2(i_{rN} + 1) = x_2(i_{rN}) + \frac{S_{x2}}{rN} \tag{8}$$

As the y-coordinates follow the path of a circle, subsequent coordinates can be determined via the x-coordinates and the equation of a circle:

$$y_2(i_{rN}) = -\sqrt{R^2 - x_2(i_{rN}) - x_1(i_{bN} = \ bN)^2} + R \tag{9}$$

As the x-coordinates have already been discretized within zone 2, the y-coordinates are accordingly discretized via their dependency.

3.7. Track Nodal Coordinates, Zone 3

Similar to zone 2, zone 3 is composed of X and Y coordinates with all Z coordinates again equal to zero. However, unlike zone 2, zone 3 does not have a radius of curvature and is a straight line with an arbitrary slope that is tangent to zone 2 and its mirror. Therefore, the beginning of zone 3 lies at the terminus of zone 2, and the converse being true for zone 2's mirror. As zone 3 is a straight line, the standard expression of a line with a slope can be used. The length of zone 3 is found by taking the difference of the maneuver distance and two times the x-distance of zone 2, S_{x2}:

$$S_{x3} = vt - 2S_{x2} \tag{10}$$

Note that the first node of zone 3 is the last node of zone 2. X-coordinate discretization is completed by dividing the length over the number of desired zone 3 nodes:

$$x_3(i_{tN}+1) = x_3(i_{tN}) + \frac{S_{x3}}{tN} \tag{11}$$

The y-coordinates of zone 3 are found from the basic equation of a line at some slope. To determine the y-coordinates, it is necessary to use the x-coordinates of zone 3 as input to determine the line's slope. The line's slope is determined by the lane distance and points of radial tangency per Figure 6:

$$M = \frac{S_{Lane} - 2S_{y2}}{vt - 2S_{x2}} \tag{12}$$

With the slope in hand, the y-coordinates can be calculated by using the distance of zone 2 as the y-intercept and summing recursively:

$$y_3(i_{tN}) = M[x_3(i_{tN}) - x_2(i_{rN} = rN)] + S_{y2} \tag{13}$$

With both the x and y coordinates of the zone 3 nodes known, zone 3's geometry is now fully defined and discretized.

3.8. Arc Length

To determine the time increment and track curvature at each node it is necessary to calculate the arc length [6]. The arc length is the length of traversed track as measured from the origin to any specific node. The arc length can be determined exclusively from the track geometry and previous nodal coordinate expressions. Firstly, the arc length at each node within zone 1 must be determined. This is readily accomplished by dividing the total zone length over desired nodes and summing recursively. The first node of zone 1 is initialized as zero and subsequent nodes are found by summing the previous nodal arc length with the arc length increment:

$$s_1(i_{bN}+1) = s_1(i_{bN}) + \frac{S_B}{bN} \tag{14}$$

The arc length for zone 2 is determined by first splitting zone 2 into two halves. This process is not necessary in itself for calculating the arc length, but simplifies later computational implementation. Zone 2 is therefore composed of an entrance half and an exit half, and is implemented by halving the desired zone nodes and counting within.

The first node of sub-zone 2.1 is initialized as the last node of zone 1. Subsequent nodes within sub-zone 2.1 are found by summing the previous node with the length increment:

$$s_{2.1}\left(i_{\frac{rN}{2}}+1\right)=s_{2.1}\left(i_{\frac{rN}{2}}\right)+\frac{R\zeta}{rN} \tag{15}$$

The nodal arc lengths of sub-zone 2.2 are determined in much the same way as sub-zone 2.1. The first node of sub-zone 2.2 is initialized as the last node of sub-zone 2.1. The arc length at all nodes following the initial node are determined by summing the previous nodal arc length with the arc length increment:

$$s_{2.2}\left(i_{\frac{rN}{2}}+1\right)=s_{2.2}\left(i_{\frac{rN}{2}}\right)+\frac{R\zeta}{rN} \tag{16}$$

The nodal arc lengths within zone 3 are determined by initializing the first node as the last node of sub-zone 2.2. Remaining nodal arc lengths are determined likewise to zones 1 and 2 through summation of the previous nodal arc length and the arc length increment. The arc length increment of zone 3 is the total arc length divided by the zone 3 quantity of nodes. The total arc length of zone 3 is determined from Figure 6:

$$S_3=\sqrt{(vt-2S_{x2})^2+\left(S_{Lane}-2S_{y2}\right)^2} \tag{17}$$

The nodal arc lengths are therefore:

$$s_3(i_{tN}+1) = s_3(i_{tN}) + \frac{S_3}{tN} \tag{18}$$

With the arc lengths of zones 1 through 3 now known, the arc length can be used in later calculations of the nodal time increment and parameterization of variables with respect to arc length, as needed.

3.9. Time at Node

The time at each node is calculated by dividing the arc length by the body's constant speed. As the body is assumed to have a constant speed, and the body is traversing a distance given by the arc length, the time readily determined. Note that the times determined within sub-zone 2.2 must be rounded down in order to avoid numerical errors within the preprocessor. To begin, the time increments within zone 1 are initialized by dividing the arc length at the first node by the body's speed. The time at subsequent nodes is found by dividing their arc lengths by the body's speed:

$$t_1(i_{bN}) = \frac{s_1(i_{bN})}{v} \tag{19}$$

The first node of sub-zone 2.1 is initialized as the last node of zone 1. The time at the first node of sub-zone 2.1 is found by dividing the associated arc length by the body's constant speed. The time increments at subsequent nodes are likewise found by dividing their associated arc lengths by the body's constant speed:

$$t_{2.1}\left(i_{\frac{rN}{2}}\right) = \frac{s_{2.1}\left(i_{\frac{rN}{2}}\right)}{v} \tag{20}$$

Sub-zone 2.2 is initialized with its first node being the last node of sub-zone 2.1. The time at all other nodes within sub-zone 2.2 is found by dividing the associated nodal arc length by the body's constant speed:

$$t_{2.2}\left(i_{r\frac{N}{2}}\right) = \frac{s_{2.2}\left(i_{r\frac{N}{2}}\right)}{v} \tag{21}$$

The time at the first node of zone 3 is found by initializing as the last node of sub-zone 2.2. Subsequent nodes are found the arc length to that node by the body's constant speed:

$$t_3(i_{tN}) = \frac{s_3(i_{tN})}{v} \tag{22}$$

With the time fully defined within zone 3, the time increment for each node within all zones is now known.

3.10. Curvature

The track consists of two straight sections connected tangentially via a semi-circular curved section. For both straight sections the curvature is zero at each node, and for the curved section, the curvature is constant at each node [6]. Therefore, the curvatures within zones 1 and 3 are always equal to zero, and the curvatures within zone 2 at all nodes are equal to:

$$k(i_{rN}) = \frac{1}{R} \tag{23}$$

3.11. Yaw Angle

The yaw angle ψ can be calculated from the path curvature using a standardized form [6]. As the body must turn for it to yaw, its yaw angle will be zero for the initial straight section of zone 1:

$$\psi_1(i_{bN}) = 0 \tag{24}$$

As the body enters the curved radial section of zone 2, it will begin to turn and hence yaw. The first node of sub-zone 2.1, $\psi_{2.1}|_1$, is initialized as the last node of zone 1. To

use the expression for the yaw angle [6], the initial and final nodes of the arc length and curvature must be defined for sub-zone 2.1. The initial and final arc lengths within sub-zone 2.1 are defined as $s_{2.1}|_1$ and $s_{2.1}|_{\frac{rN}{2}}$, respectively. The curvature at the final node within sub-zone 2.1 is defined as $k_{2.1}|_{\frac{rN}{2}}$. Using these definitions, the nodal yaw angle within sub-zone 2.1 is:

$$\psi_{2.1}\left(i_{\frac{rN}{2}}\right) = \psi_{2.1}|_1 + \frac{1}{\left(s_{2.1}|_{\frac{rN}{2}} - s_{2.1}|_1\right)}\left(\left(\frac{k_{2.1}|_{\frac{rN}{2}}}{2}\right)\left(s_{2.1}\left(i_{\frac{rN}{2}}\right) - s_{2.1}|_1\right)^2\right) \tag{25}$$

Note that the total nodes rN within zone 2 are divided by two to account for the two sub-zones. The first node of sub-zone 2.2, $\psi_{2.2}|_1$, is initialized from the last node of sub-zone 2.1. Like sub-zone 2.1, let the initial and final arc lengths within sub-zone 2.2 be defined as $s_{2.2}|_1$ and $s_{2.2}|_{\frac{rN}{2}}$, respectively. Let the curvature at the first node within sub-zone 2.2 be defined as $k_{2.2}|_1$. The expression for the yaw angle [6] within sub-zone 2.2 becomes:

$$\begin{aligned}\psi_{2.2}\left(i_{\frac{rN}{2}}\right) = \psi_{2.1}|_{\frac{rN}{2}} + \frac{1}{\left(s_{2.2}|_{\frac{rN}{2}} - s_{2.2}|_1\right)}\left(-\frac{k_{2.2}|_1}{2}\left(s_{2.2}\left(i_{\frac{rN}{2}}\right) - s_{2.2}|_{\frac{rN}{2}}\right)^2\right) \\ + \left(\left(\frac{k_{2.2}|_1}{2}\right)\left(s_{2.2}|_{\frac{rN}{2}} - s_{2.2}|_1\right)\right)\end{aligned} \tag{26}$$

As the body exits zone 2, it ceases to turn and therefore ceases to yaw leaving its yaw angle fixed for all remaining nodes thereafter. Within zone 3 the yaw angle remains constant as the final yaw angle upon exit from zone 2 where $\psi_{2.2}|_{rN}$ is the final yaw angle in sub-zone 2.2:

$$\psi_3(i_{tN}) = \psi_{2.2}|_{rN} \tag{27}$$

With the yaw angles defined within all three zones, the only angle yet to be defined is the body's roll.

3.12. Body Roll

To determine the body's roll, the body is modeled as a sprung mass rotating about a fixed point with a rotary spring and damper. This is essentially a version of the classic rotating spring-damper-mass problem. Several assumptions must be made for the body to negotiate the track's geometry. The body is assumed to be a simple solid rectangular prism supported above the track by a massless support. The point at which the support intersects the track is the local body origin and is attached to the XY plane. The centripetal force and weight act at the center of gravity and both are located at the centroid. All angles are considered small, and therefore permit linearization, enabling the body to be modeled using a second order, linear, non-homogeneous, differential equation. Figure 7 shows the free-body diagram of the solid body at roll angle ϕ. To find an expression for the roll angle with respect to time, the moments about O are first summed:

$$\sum M_O = J_\phi \ddot{\phi} = mgH \sin\phi - F_C H \cos\phi - C_t \dot{\phi} - K_t \phi \tag{28}$$

All angles shall be considered small therefore permitting linearization. Therefore, for small angles of ϕ, $\sin\phi \approx 0$ and $\cos\phi \approx 1$. The moments about O are linearized:

$$\sum M_O = J_\phi \ddot{\phi} = mgH\phi - F_C H - C_t \dot{\phi} - K_t \phi \tag{29}$$

As the equation of motion is now a second order, linear, non-homogeneous differential equation its solution can be determined by finding the sum of the homogeneous solution and the particular solution [15].

Note that the centripetal force is expressed using the standard relation, where m is mass, v is speed, and r is radius of curvature:

$$F_C = \frac{mv^2}{R} \tag{30}$$

The solution to ϕ as a function of time is assumed to take the general form [15]:

$$\phi(t) = \phi(t)_H + \phi(t)_P \tag{31}$$

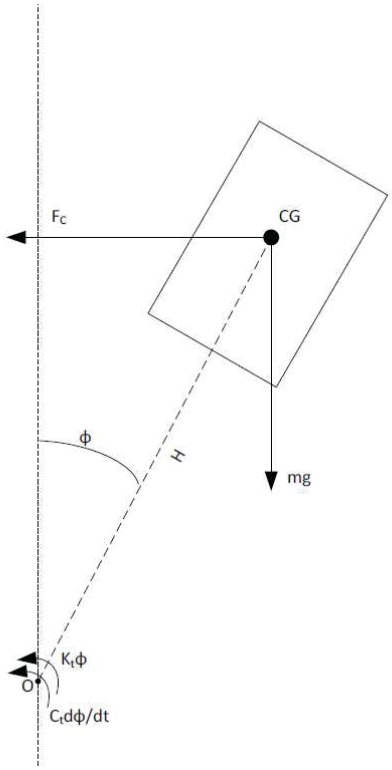

FIGURE 7. Free-body diagram of solid body at roll angle ϕ. The body is supported by a massless support with a rotary spring and damper and acted upon by the centripetal force and its own weight.

First the homogeneous solution will be determined. Terms are substituted, collected, and placed into standard form using equation 48 [15]:

$$\ddot{\phi} + \frac{C_t}{J_\phi}\dot{\phi} + \frac{(K_t - mgH)}{J_\phi}\phi = 0 \tag{32}$$

Therefore, the roots of the characteristic equation to the standardized linear homogeneous differential equation are:

$$\lambda_{1,2} = \frac{1}{2}\left(-\frac{C_t}{J_\phi} \pm \sqrt{\left(\frac{C_t}{J_\phi}\right)^2 - 4\frac{(K_t - mgH)}{J_\phi}}\right) \tag{33}$$

The roots of the characteristic equation can assume one of three scenarios: two distinct real roots, two double roots, or a complex conjugate pair. The third scenario, complex conjugate pair, represents oscillatory motion and most accurately reflects the motion seen in real-world applications such as a ground vehicle. Therefore, it is assumed that the body's response will be oscillatory and this motion will be enforced in the model's solution. In order to enforce oscillatory motion, the discriminant must be negative. In the case of a complex conjugate pair, the solution to the homogeneous differential equation, using standard notation, assumes the form [15]:

$$\phi(t)_H = e^{-\sigma t}(c_1 \cos \eta t + c_2 \sin \eta t) \tag{34}$$

Angles σ and η are defined in terms of the characteristic equation [15] with:

$$\sigma = \frac{1}{2}a_1 \tag{35}$$

and:

$$\eta = \frac{1}{2}\sqrt{4a_2 - {a_1}^2} \tag{36}$$

where:

$$a_1 = \frac{C_t}{J_\phi} \tag{37}$$

and a_2 is:

$$a_2 = 4\frac{K_t - mgH}{J_\phi} \tag{38}$$

With σ, η, a_1, and a_2 defined, the homogeneous portion of the general solution is known. The particular portion of the general solution is found using the centripetal force. The product of F_C and H is constant and therefore a polynomial of degree zero. The particular solution can be found using the method of undetermined coefficients by choosing a degree-zero polynomial, and substituting into equation 32 [15]. This yields:

$$\phi(t)_P = -\frac{F_C H J_\phi}{K_t - mgH} \tag{39}$$

After substituting equations 34 and 39 into equation 31, the general solution is:

$$\phi(t) = e^{-\sigma t}(c_1 \cos \eta t + c_2 \sin \eta t) - \frac{F_C H J_\phi}{K_t - mgH} \tag{40}$$

Initial conditions must be applied to solve for c_1 and c_2. For the two initial conditions we assume zero initial angular velocity and a very small initial angle, ε:

$$\phi(0) = \varepsilon, \dot{\phi}(0) = 0 \tag{41}$$

It is reasonable to assume angular velocity is zero at $t = 0$ as motion has yet to occur. However, if the roll angle itself is also zero then the solution will be undefined and so therefore the roll angle must have some initial value which is assumed. This has the effect of initiating motion of the model by a qualitatively small angle. Were the roll angle equal to zero the model becomes unable to determine which direction the motion should initiate and therefore is undefined. In a real world application, this is experienced as a locking point, wherein the system or device cannot move one way or the other and so becomes locked into position.

With the initial conditions defined, c_1 and c_2 can be found by first substituting ε into equation 61:

$$\phi(0) = \varepsilon = \ c_1 \ - \frac{F_C H J_\phi}{K_t - mgH} \tag{42}$$

Constant c_2 is found by differentiating equation 61 with respect to time and substituting equation 63 for the initial angular velocity:

$$\dot{\phi}(0) = \ 0 = \sigma c_1 - \eta c_2 \tag{43}$$

Therefore, equation 64 can be rearranged to define c_1, and c_2 can be expressed in terms of c_1:

$$c_1 = \varepsilon + \frac{F_C H J_\phi}{K_t - mgH}, \quad c_2 = \frac{\sigma}{\eta} c_1 \tag{44}$$

It should be noted that J_ϕ is the rotational inertia of the body in the roll plane along the x-axis. As the body's LCS is attached to the track for model simplification, the rotational inertia must therefore be offset accordingly. This is most simply determined via the parallel axis theorem:

$$J_\phi = \frac{L2 \cdot L3^3}{12} + L2 \cdot L3 \left(h + \frac{L3}{2}\right)^2 \tag{45}$$

4.10. Roll Velocity

The roll velocity as a function of time is readily determined from the main roll angle equation. To find the roll velocity, the first derivative with respect to time is taken with all terms retaining their previously stated definitions:

$$\dot{\phi}(t) = \ e^{-\sigma t}\big((c_2 \eta \cos \eta t - c_1 \eta \sin \eta t) - \sigma(c_1 \cos \eta t - c_2 \sin \eta t)\big) \tag{46}$$

As the roll velocity is calculated as a function of time, it is calculated using the existing time increments.

3.13. Roll Acceleration

Similar to the roll velocity, the roll acceleration is readily determined by taking the second derivative with respect to time of the roll equation with all terms retaining their previously stated definitions:

$$\ddot{\phi}(t) = e^{-\sigma t}\big(\sigma^2(c_1\cos\eta t + c_2\sin\eta t) - \eta^2(c_1\cos\eta t + c_2\sin\eta t) - 2\sigma\eta(c_2\cos\eta t + c_2\sin\eta t)\big) \tag{47}$$

Again, like the roll angle and roll velocity, the roll acceleration is found by using the existing time increments.

3.14. Yaw Velocity

The same limitations of determining the yaw angle remain with the yaw velocity. The yaw velocity will only exist so long as the body is yawing, therefore within zone 1 the yaw velocity is zero:

$$\dot{\psi}_1(i_{bN}) = 0 \tag{48}$$

Within zones 2 and 3 the case is different as the yaw velocity is not equal to zero. For zones 2 and 3, a general form of the yaw velocity $\dot{\psi}_{2,3}(t)$ can be obtained by taking the first derivative of the general yaw angle equation [6] with respect to time:

$$\dot{\psi}_{2.1,2.2,3}(t) = \frac{k_1(s(i)-s_0)(\dot{s}(i)-\dot{s}_0) - k_0(s(i)-s_1)(\dot{s}(i)-\dot{s}_1)}{s_0-s_1} - \frac{k_0}{2}(\dot{s}_0-\dot{s}_1) - \frac{\left(\frac{k_0}{2}(s(i)-s_1)^2 - \frac{k_1}{2}(s(i)-s_0)^2\right)(\dot{s}_0-\dot{s}_1)}{(s_0-s_1)^2} \tag{49}$$

Within equation 49, k_0 and $k_1$are the initial and final curvatures, respectively, within the associated sub-zones and zones. The initial and final arc lengths, and their associated speeds, are represented as s_0, s_1, $\dot{s}_0$, and $\dot{s}_1$. The arc length and arc length speed at any

given node are represented as $s(i)$ and $\dot{s}(i)$. The arc length speed $\dot{s}(i)$ at a given node within the zones 2 or 3 can be simply determined by the quotient of the arc length $s(i)$ at that node divided by the associated time increment $t(i)$:

$$\dot{s}_{2.1,2.2,3}(i) = \frac{s(i)}{t(i)} \tag{50}$$

The initial yaw velocity in sub-zone 2.1 is initialized as the final yaw velocity from zone 1, which is zero. The initial yaw velocity of sub-zone 2.2 is initialized as the final yaw velocity from sub-zone 2.1. Finally, the initial yaw velocity of zone 3 is initialized as the final yaw velocity from sub-zone 2.2. With the yaw velocities for zone 3 now defined, the yaw velocity at all nodes for all 3 zones is obtained.

3.15. Conclusions

The mathematical model used to simulate a simplified body with a gyroscopic stabilizer implements the use of a preprocessor that feeds input data into a series of subsequent processors. The preprocessor determines all of the input motion based upon assumed geometry, speed, and spatial positioning. The calculations performed by the preprocessor are determined by the assumptions of the general model, and the geometric design of the body and track. The preprocessor's output is compiled into a data array that is then used as the input for the body and gyroscopic processors to calculate their respective data outputs.

CHAPTER 4

MAIN BODY EQUATIONS OF MOTION

Once the output data are calculated and prepared by the preprocessor, they are used as input for the body equations of motion. Through the equations of motion the moments induced upon the body by its motion along the track can be determined. Note that the body's equations of motion do not model the moments induced by the gyroscope.

4.1. Equations of Motion and Moments

The dynamic equations of motion for a single solid body defined in three dimensional space may be represented as [14]:

$$\begin{bmatrix} m_{RR} & m_{R\theta} \\ m_{\theta R} & m_{\theta\theta} \end{bmatrix} \begin{bmatrix} \ddot{R} \\ \ddot{\Theta} \end{bmatrix} = \begin{bmatrix} F_{A,R} \\ F_{A,\theta} \end{bmatrix} + \begin{bmatrix} F_{Q,R} \\ F_{Q,\theta} \end{bmatrix} \tag{51}$$

The spatial equations of motion are merely a more general application of the second law of motion. The F_A terms represent the applied forces on the body, and as there are no applied forces yet in the model, these terms are zero. The F_Q terms represent the quadratic forces arising from the body's motion, and these are the terms of interest as they are the forces induced by the body in each of its degrees of freedom.

4.2. Transformation Matrices

To avoid the transformations involved with using a trajectory coordinate system [8], the current model uses an LCS that is attached directly to the track and the body

offset through its moments of inertia. Figure 8 shows the body in relation to the track, the GCS, and its LCS.

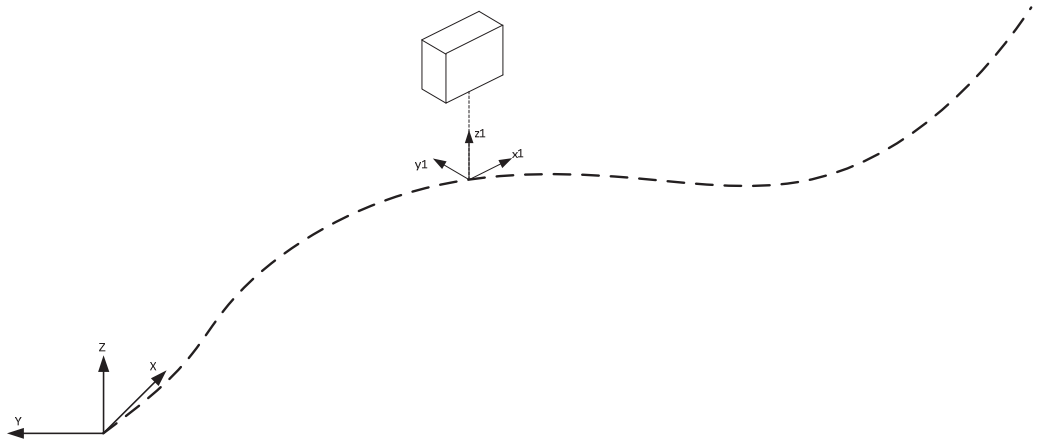

FIGURE 8. Diagram showing the body traversing the defined path with its local coordinate system in relation to the global coordinate system.

As the LCS moves along the track its mutually perpendicular axes remain tangent to, and perpendicular to, the track at all times. In this way the LCS follows the track's path exactly, and as the body is aligned within the LCS, the track's geometry is communicated to the body. If this method were not employed, the problem would become of how to communicate the track geometry to the body, the body motion arising from that track geometry, and then that body motion back to the GCS? While using an offset through the body's moment of inertia requires modifying the inertia tensor from what it would otherwise be, it is nonetheless highly effective and avoids otherwise lengthy calculations.

The current arrangement therefore requires only two coordinate systems and a single transformation as opposed to three coordinate systems and two transformations [8]. In order to overcome the distance of the centroid positioned over the LCS origin, the inertia tensor is adjusted using the parallel axis theorem. This allows the LCS origin to remain attached to the track but for the body to be positioned above, and still move relative to, the track.

The current model uses the transformation sequence Z-X-Y with positive axes defined by the right hand rule as depicted in Figure 3. The transformation matrix of the LCS to the GCS therefore becomes:

$$A = \begin{bmatrix} \cos\psi & -\cos\phi\sin\psi & \sin\phi\sin\psi \\ \sin\psi & \cos\phi\sin\psi & -\cos\phi\sin\psi \\ 0 & \sin\phi & \cos\phi \end{bmatrix} \tag{52}$$

The first derivative of the transformation matrix with respect to time is found by differentiating A with respect to time, t:

$$\dot{A} =$$

$$\begin{bmatrix} -\sin\psi\,\dot{\psi} & \sin\phi_1\sin\psi\,\dot{\phi} - \cos\phi\cos\psi\,\dot{\psi} & \cos\phi\sin\psi\,\dot{\phi} + \cos\psi\sin\phi\,\dot{\psi} \\ \cos\psi\,\dot{\psi} & -\cos\psi\sin\phi\,\dot{\phi} - \cos\phi\sin\psi\,\dot{\psi} & \sin\phi\sin\psi\,\dot{\psi} - \cos\phi\cos\psi\,\dot{\phi} \\ 0 & \cos\phi\,\dot{\phi} & -\sin\phi\,\dot{\phi} \end{bmatrix} \tag{53}$$

4.3. Unit Vector Transformation

The unit vector transformation is the transformation of the unit vectors from the LCS to the GCS. The unit vector transformation follows the same sequence of rotations as the transformation matrix. The unit vector along the z1 axis is taken with respect to Z and is therefore unity. The unit vector along the x1 axis is found by transforming x1 by

Z. Finally, the unit vector along the y1 axis is found by transforming y1 through z and x respectively. When all three unit vectors are summed they become:

$$G = [G_i \quad G_j \quad G_k] = \begin{bmatrix} 0 & \cos\psi & -\cos\phi\sin\psi \\ 0 & \sin\psi & \cos\phi\cos\psi \\ 1 & 0 & \sin\phi \end{bmatrix} \tag{54}$$

The unit vector transformation matrix can be defined with respect to the body coordinate system by pre-multiplying with the transpose of the transformation matrix [14]:

$$\bar{G} = A^T G \tag{55}$$

The derivative of the unit vector transformation with respect to the LCS is found by differentiating $\bar{G}$ with respect to time [14]:

$$\dot{\bar{G}} = \begin{bmatrix} 0 & 0 & 0 \\ \cos\phi\dot{\phi} & 0 & 2\cos\phi\sin\phi\,\dot{\phi} - 2\cos\phi\sin\phi\,\dot{\phi}\sin^2\psi - 2\cos\phi\sin\phi\,\dot{\phi}\cos^2\psi \\ -\sin\phi\,\dot{\phi} & 0 & \dot{\phi}\cos^2\phi - \dot{\phi}\cos^2\phi\sin^2\psi - \dot{\phi}\cos^2\phi\cos^2\psi + \dot{\phi}\cos^2\psi\sin^2\phi + \dot{\phi}\sin^2\phi\sin^2\psi - \dot{\phi}\sin^2\phi \end{bmatrix} \tag{56}$$

4.4. Position Vector

The location of the body's centroid with respect to the LCS' origin is depicted in Figure 9. As the centroid does not move relative to the LCS origin in the X direction, the local x1 component is always equal to 0. All angles are assumed to be small and therefore the displacement of the centroid is linear with respect to the LCS origin.

Using existing notation [14] the position vector within the LCS is found through elementary trigonometric relations, and with small angles is linearized:

$$\bar{u} = \left[0 \quad \phi\left(h + \frac{L3}{2}\right) \quad h + \frac{L3}{2}\right]^T \tag{57}$$

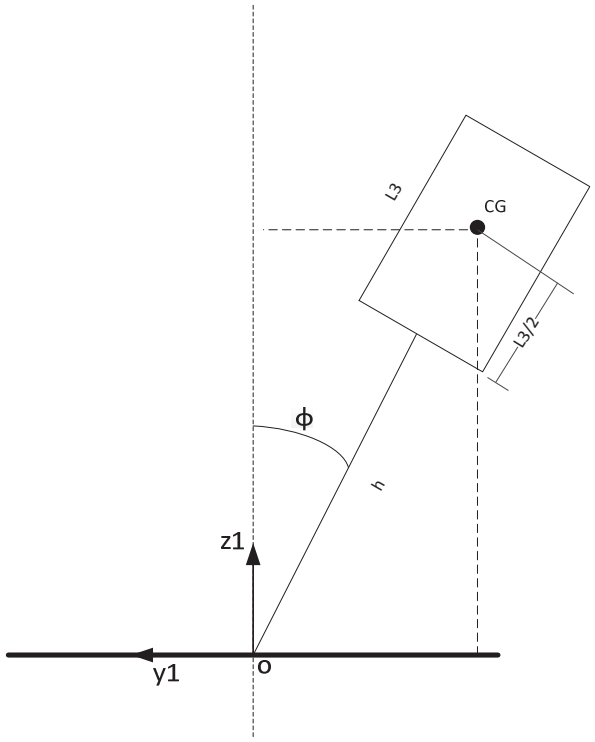

FIGURE 9. Diagram of geometry used to determine the body's position vector.

4.5. Angular Velocities

The Euler angles can be expressed as a single vector giving the orientation of the body at a single point [14]. For the model under consideration, pitch is neglected, and therefore is always equal to zero. If Θ is the total orientation of the body, with ϕ being roll and ψ being yaw, the orientation thus becomes:

$$\Theta_{Body} = \begin{bmatrix} \phi \\ 0 \\ \psi \end{bmatrix} \tag{58}$$

The angular velocity can be expressed by taking the first derivative of the orientation vector with respect to time:

$$\dot{\Theta}_{Body} = \Omega_{Body} = [\dot{\phi} \quad 0 \quad \dot{\psi}]^T \tag{59}$$

The angular velocity can be redefined in terms of the LCS by pre-multiplying by the LCS defined unit vector matrix [14]:

$$\bar{\Omega}_{Body} = \bar{G}\dot{\Theta} \tag{60}$$

4.6. Inertia Tensor

The inertia tensor of the body within the LCS can be expressed as a symmetric matrix composed of its constituent elements, the moments of inertia [14]:

$$\bar{I}_{\theta\theta} = \begin{bmatrix} i_{xx} & i_{xy} & i_{xz} \\ i_{xy} & i_{yy} & i_{yz} \\ i_{xz} & i_{yz} & i_{zz} \end{bmatrix} \tag{61}$$

As the LCS origin is offset from the body's centroid it is necessary to ensure that the inertia tensor is offset accordingly. This is accomplished by determining the inertial components respectively at the centroid and then using the parallel axis theorem to account for the origin offset. The variable descriptions are the same as applied in prior relations and diagrams. The subscript CG denotes that the component is taken at the centroid with all references being made within the LCS. The bar over the respective components denotes the distance from the LCS origin to the body centroid using standard notation. The moment of inertia along the x1 axis:

$$i_{xx} = \frac{m}{12}(L2^2 + L3^2) + m(\bar{y}^2 + \bar{z}^2) \tag{62}$$

The moment of inertia along the y1 axis:

$$i_{yy} = \frac{m}{12}(L1^2 + L3^2) + m(\bar{x}^2 + \bar{z}^2) \tag{63}$$

The moment of inertia along the z1 axis taken at the centroid:

$$i_{zz} = \frac{m}{12}(L1^2 + L2^2) + m(\bar{x}^2 + \bar{y}^2) \tag{64}$$

The centroidal moments of inertia along the cross axes are zero and are offset from the origin by multiplying the mass by the centroidal location from the respective axis. The xy1 moment of inertia is:

$$i_{xy} = -m\bar{x}\bar{y} \tag{65}$$

The xz1 moment of inertia is:

$$i_{xz} = -m\bar{x}\bar{z} \tag{66}$$

The yz1 moment of inertia is:

$$i_{yz} = -m\bar{y}\bar{z} \tag{67}$$

4.7. Quadratic Forces

The quadratic forces for a solid body's motion consist of the linear forces and the rotational forces [14]:

$$F_Q = \begin{bmatrix} F_{Q,R} \\ F_{Q,\theta} \end{bmatrix} \tag{68}$$

As this thesis is concerned primarily with the rotational quadratic forces, only these will be considered. The rotational quadratic forces can be expressed as the product of the transpose of the unit vector matrix in terms of the LCS, and the cross-product of the inertia tensor, the angular velocity, and the unit vector transformation [14]:

$$F_{Q,\theta} = -\bar{G}^T \left[\bar{\Omega}_{Body} \times \left(\bar{I}_{\theta\theta} \bar{\Omega}_{Body} \right) + \bar{I}_{\theta\theta} \dot{\bar{G}} \dot{\theta} \right] \tag{69}$$

4.8. Conclusions

As the body model consists of only the body, the quadratic forces are only those in play and therefore only those which are considered. Further, this thesis is concerned chiefly with the moments induced by the body and accordingly only the rotational quadratic forces are considered as these are the very moments under consideration. With

the body moments determined, it is now necessary to determine the moments induced by the addition of a gyroscope. With the gyroscopic moments known, the original body moments can be augmented and the difference between the two determined.

CHAPTER 5

GYROSCOPIC EQUATIONS

The gyroscope considered in this model is a simple gyroscope consisting of a single rotor rotating on a massless shaft. As shown in FIGURE 10 and FIGURE 11 the gyroscope's centroid is coincident with the body's centroid and there is no relative movement between the two. Therefore the yaw and roll angles experienced by the body are the same as those experienced by the gyroscope by virtue of their fixed relative positions.

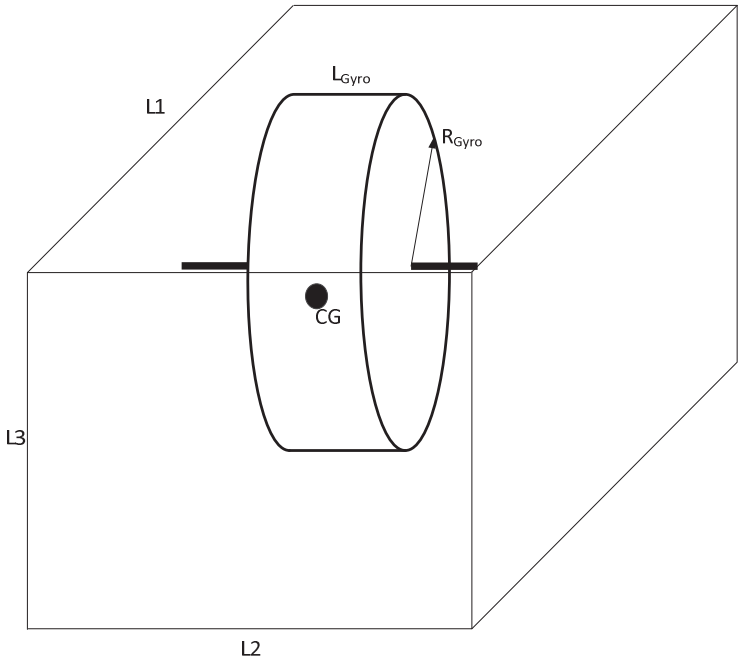

FIGURE 10. Diagram of body showing relative spatial position of gyroscope along with basic gyroscopic geometry. Notice that the gyroscope consists of a single cylindrical rotor along a massless shaft.

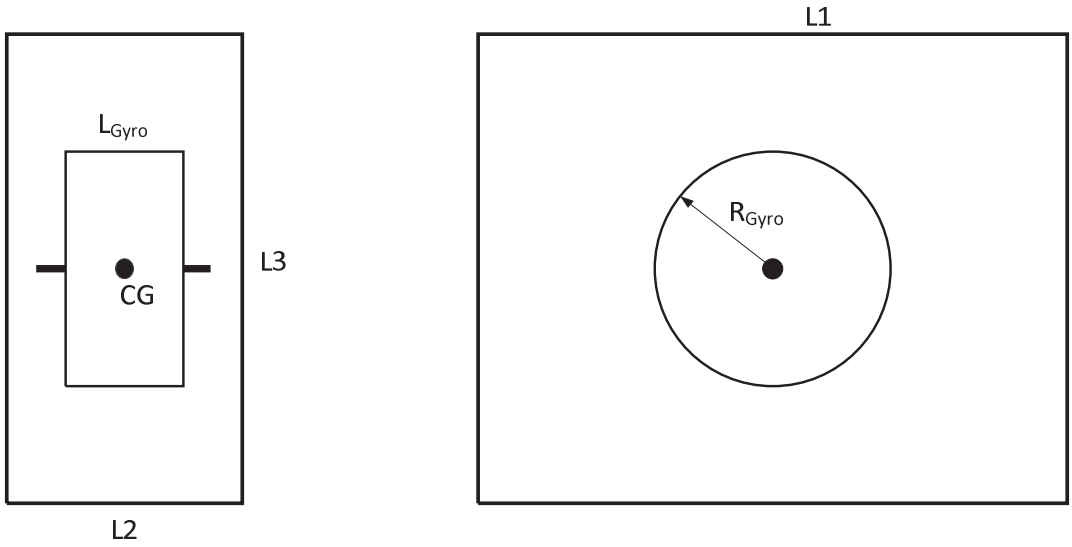

FIGURE 11. Diagram of gyroscope with its associated geometric parameters as viewed orthogonally.

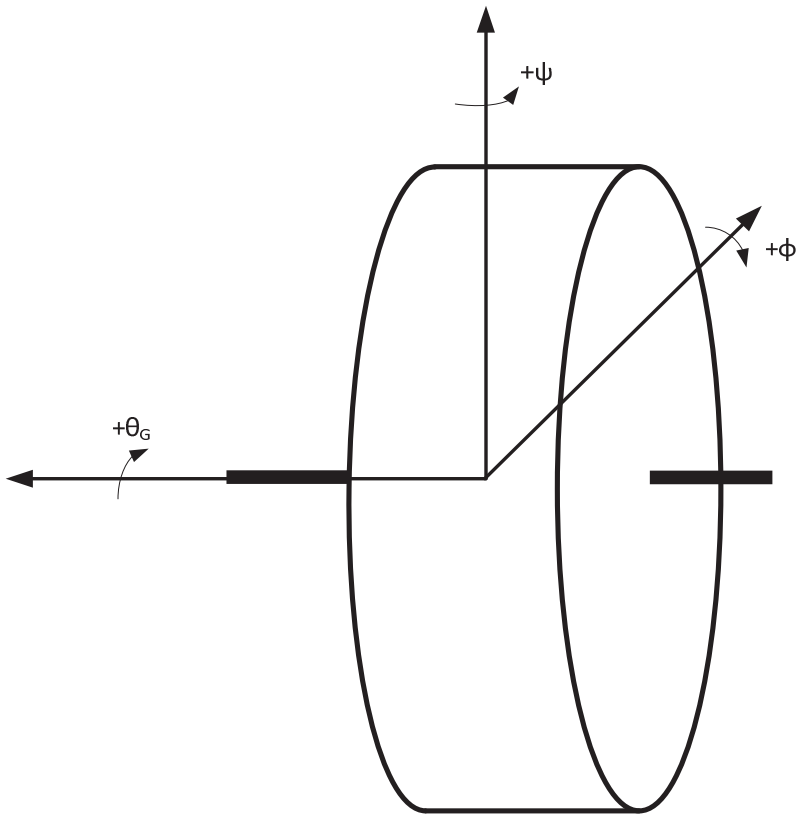

FIGURE 12. Rotational axes of gyroscope.

The positive directions of rotation along the gyroscope's rotational axes are the same as those of the body and shown in FIGURE 12. While the yaw and roll angles of the body and gyroscope are identical, the pitch angle is not. Unlike the body, the gyroscope is rotating and therefore has a positive spin and non-zero θ. To avoid confusion with the generalized θ, the gyroscope's pitch is labeled with a G and is θ_G. As the gyroscope follows the right hand rule with constant angular velocity in spin, nutation and precession will occur depending upon which rotational axis is active. For example, with a constant spin, a motion along ψ will cause a corresponding motion along ϕ, and vice versa. The relative directions of the rotations are dependent upon the direction of the spin, and for this model, it is critical to note that the spin must be negative. If this is not the case then positive inputs will generate positive outputs and the system will have a positive feedback. A negative spin will create a negative feedback system and therefore ameliorate the input motions.

With a negative spin, as the gyroscope begins to yaw counterclockwise through negotiation of a radial curve, then it will also begin to roll counterclockwise into the center of curvature and in opposition to its centripetal motion. This is the opposite motion predicted with a positive spin gyroscope whereas the body yaws counterclockwise a clockwise roll ensues. Therefore, the moments induced by a gyroscope with negative spin are contrary to the moments induced on the body by its motion. Analytically, the moments induced by the gyroscope in the three mutually perpendicular directions can be expressed in terms of the kinetic energy of the gyroscope [14].

The roll moment M_ϕ can be expressed as [14]:

$$M_\phi = \frac{\partial}{dt}\left(\frac{\partial T}{\partial \dot{\phi}}\right) - \frac{\partial T}{\partial \phi} \tag{70}$$

The pitch moment M_{θ_G} can be expressed as [14]:

$$M_{\theta_G} = \frac{\partial}{dt}\left(\frac{\partial T}{\partial \dot{\theta}_G}\right) - \frac{\partial T}{\partial \theta_G} \tag{71}$$

Finally, the yaw moment M_ψ can be expressed as [14]:

$$M_\psi = \frac{\partial}{dt}\left(\frac{\partial T}{\partial \dot{\psi}}\right) - \frac{\partial T}{\partial \psi} \tag{72}$$

With these expressions is it now possible to determine the moments induced by the gyroscope and thereby determine its effect on the body.

5.1. Angular Velocity and Transformations

The angular velocity of the gyroscope in terms of the LCS can be expressed as the product of the LCS unit vector transformation matrix and the first time derivative of the orientation vector:

$$\bar{\Omega}_{Gyro} = \bar{G}\dot{\gamma}_{Gyro} \tag{73}$$

The orientation vector γ_{Gyro} and the associated angular velocity $\dot{\gamma}_{Gyro}$ use the same generalized angles as the body with the exception of pitch due to the gyroscope's spin:

$$\gamma_{Gyro} = [\psi \quad \theta_G \quad \phi]^T, \qquad \dot{\gamma}_{Gyro} = [\dot{\psi} \quad \dot{\theta}_G \quad \dot{\phi}]^T \tag{74}$$

The unit vector transformation matrix in terms of the LCS remains the same as that of the body:

$$\bar{G}_{Gyro} = A^T G \tag{75}$$

Which expands into:

$$\bar{G}_{Gyro} = \begin{bmatrix} 0 & \cos^2\psi + \sin^2\psi & 0 \\ \sin\phi & 0 & \cos^2\phi\cos^2\psi + \cos^2\phi\sin^2\psi + \sin^2\phi \\ \cos\phi & 0 & \cos\phi\sin\phi - \cos\phi\sin\phi\sin^2\psi - \cos\phi\sin\phi\cos^2\psi \end{bmatrix} \tag{76}$$

5.2. Kinetic Energy

To obtain the moments induced by the gyroscope, one can use the gyroscope's kinetic energy. The kinetic energy of the gyroscope is expressed as [14]:

$$T = \frac{1}{2}\bar{\Omega}_{Gyro}{}^{T} I_{\theta\theta\,Gyro}\bar{\Omega}_{Gyro} \tag{77}$$

As the gyroscope is symmetric about its y-axis, the mass moment along the x-axis is equal to that along the z-axis, with all cross axes being equal to zero:

$$I_{\theta\theta\,Gyro} = \begin{bmatrix} i_{xx} & 0 & 0 \\ 0 & i_{yy} & 0 \\ 0 & 0 & i_{xx} \end{bmatrix} \tag{78}$$

After substitution and expansion of terms, the kinetic energy of the gyroscope is obtained:

$$\begin{aligned} T = i_{xx}\Bigg(& \left(\dot{\phi}\cos\theta_G - \dot{\psi}\cos\phi\sin\theta_G\right)\left(\frac{\dot{\phi}\cos\theta_G - \dot{\psi}\cos\phi\sin\theta_G}{2}\right) \\ & + \left(\dot{\theta}_G + \dot{\psi}\sin\phi\right)\left(\frac{\dot{\theta}_G + \dot{\psi}\sin\phi}{2}\right)\Bigg) \\ & - i_{zz}\left(\dot{\phi}\sin\theta_G + \dot{\psi}\cos\phi\cos\theta_G\right)\left(\frac{\dot{\phi}\sin\theta_G + \dot{\psi}\cos\phi\cos\theta_G}{2}\right) \end{aligned} \tag{79}$$

With the kinetic energy now known, the gyroscopic moments can be evaluated.

5.3. Moment Along Phi

The moment along ϕ can be determined from the kinetic energy of the gyroscope [14]. Unlike the case for the body, the gyroscope has a nonzero θ due to its spin and therefore is included in the transformation and derived relations. The moment M_ϕ be expressed as the difference shown in equation 109. First the partial derivative of the kinetic energy with respect to ϕ is evaluated:

$$\frac{\partial T}{\partial \phi} = i_{xx}\left(\dot{\psi}^2 \cos\phi \sin\phi \cos^2\theta_G + \dot{\phi}\dot{\psi} \sin\phi \sin\theta_G \cos\theta_G + \dot{\theta}_G\dot{\psi} \cos\phi\right) - i_{zz}\left(\dot{\psi}^2 \cos\phi \sin\phi \cos^2\theta_G + \dot{\phi}\dot{\psi} \sin\phi \sin\theta_G \cos\theta_G\right) \tag{80}$$

Next, the partial derivative with respect to the roll velocity is evaluated:

$$\frac{\partial T}{\partial \dot{\phi}} = i_{xx}\left(\dot{\phi} \cos^2\theta_G - \dot{\psi} \cos\phi \cos\theta_G \sin\theta_G\right) + i_{zz}\left(\dot{\phi} \sin^2\theta_G + \dot{\psi} \cos\phi \cos\theta_G \sin\theta_G\right) \tag{81}$$

And finally when differentiated with respect to time:

$$\frac{\partial}{dt}\left(\frac{\partial T}{\partial \dot{\phi}}\right) = -i_{xx}{\beta_{1,\phi}}' + i_{zz}{\beta_{2,\phi}}' \tag{82}$$

Where the first term ${\beta_{1,\phi}}'$ is:

$${\beta_{1,\phi}}' = \dot{\psi}\dot{\theta}_G \cos^2\theta_G - \ddot{\phi} \cos^2\theta_G - \dot{\psi}\dot{\theta}_G \sin^2\theta_G + \ddot{\psi} \cos\theta_G \sin\theta_G + 2\dot{\phi}\dot{\theta}_G \cos\theta_G \sin\theta_G \tag{83}$$

Differentiating with respect to time causes higher order terms in ψ to appear as the yaw acceleration. However, as the yaw velocity is constant, its acceleration is equal to zero and can therefore be neglected. Therefore, the second term ${\beta_{2,\phi}}'$ is:

$${\beta_{2,\phi}}' = \dot{\psi}\dot{\theta} \cos^2\theta_G - \ddot{\phi} \cos^2\theta_G - \dot{\psi}\dot{\theta}_G \sin^2\theta_G + 2\dot{\phi}\dot{\theta}_G \cos\theta_G \sin\theta_G \tag{84}$$

5.4. Moment Along Psi

The moment M_ψ can be determined similarly to that of ϕ [14]. As the kinetic energy does not contain any yaw terms, the partial derivative of the gyroscopic kinetic energy with respect to the yaw is zero:

$$\frac{\partial T}{\partial \psi} = 0 \tag{85}$$

However, the kinetic energy does contain yaw velocity terms and therefore the partial derivative with respect to the yaw velocity is not zero:

$$\begin{aligned}\frac{\partial T}{\partial \dot{\psi}} = {} & i_{xx}\left(\dot{\psi} - \dot{\phi}\cos\phi\cos\theta_G\sin\theta_G - \dot{\psi}\cos^2\phi\cos^2\theta_G + \dot{\theta}_G\sin\phi\right) \\ & + i_{zz}\left(\dot{\psi}\cos^2\phi\cos^2\theta_G + \dot{\phi}\cos\phi\cos\theta_G\sin\theta_G\right)\end{aligned} \tag{86}$$

When the partial derivative is taken with respect to time it becomes:

$$\frac{\partial}{dt}\left(\frac{\partial T}{\partial \dot{\psi}}\right) = i_{xx}{\beta_{1,\psi}}' - i_{zz}{\beta_{2,\psi}}' \tag{87}$$

Again, neglecting higher order spin and yaw terms, the first term ${\beta_{1,\psi}}'$ is:

$$\begin{aligned}{\beta_{1,\psi}}' = {} & \dot{\theta}_G\dot{\phi}\cos\phi - \dot{\phi}\dot{\theta}_G\cos\phi\cos^2\theta_G + \dot{\phi}\dot{\theta}_G\cos\phi\sin^2\theta_G \\ & - \ddot{\phi}\cos\phi\cos\theta_G\sin\theta_G + 2\dot{\psi}\dot{\phi}\cos\phi\sin\phi\cos^2\theta_G \\ & + 2\dot{\psi}\dot{\theta}_G\cos\theta_G\sin\theta_G\cos^2\phi + \dot{\phi}^2\cos\theta_G\sin\phi\sin\theta_G\end{aligned} \tag{88}$$

And the second term ${\beta_{2,\psi}}'$ is:

$$\begin{aligned}{\beta_{2,\psi}}' = {} & \dot{\phi}\dot{\theta}_G\cos\phi\sin^2\theta_G - \dot{\phi}\dot{\theta}_G\cos\phi\cos^2\theta_G - \ddot{\phi}\cos\phi\cos\theta_G\sin\theta_G \\ & + 2\dot{\psi}\dot{\phi}\cos\phi\sin\phi\cos^2\theta_G + 2\dot{\psi}\dot{\theta}_G\cos\theta_G\sin\theta_G\cos^2\phi \\ & + \dot{\phi}^2\cos\theta_G\sin\phi\sin\theta_G\end{aligned} \tag{89}$$

5.6. Moment Along Theta

The moment M_{θ_G} is obtained in the same manner as both M_ϕ and M_ψ [14]. The partial derivative of the kinetic energy with respect to θ_G is:

$$\frac{\partial T}{\partial \theta_G} = -i_{xx}\beta_{1,\theta_G} + i_{zz}\beta_{2,\theta_G} \tag{90}$$

Where the first term β_{1,θ_G} is:

$$\begin{aligned}\beta_{1,\theta_G} = {} & \dot{\phi}^2 \cos\theta_G \sin\theta_G + 2\dot{\phi}\dot{\psi}\cos\phi\cos^2\theta_G - \dot{\phi}\dot{\psi}\cos\phi \\ & - \dot{\psi}^2\cos\theta_G\sin\theta_G\cos^2\phi\end{aligned} \tag{91}$$

And the second term β_{2,θ_G} is:

$$\begin{aligned}\beta_{2,\theta_G} = {} & \cos\theta_G \sin\theta_G \dot{\phi}^2 + 2\dot{\phi}\dot{\psi}\cos\phi\cos^2\theta_G - \dot{\phi}\dot{\psi}\cos\phi \\ & - \dot{\psi}^2\cos\theta_G\sin\theta_G\cos^2\phi\end{aligned} \tag{92}$$

The partial derivative of the gyroscope's kinetic energy with respect to the pitch velocity is:

$$\frac{\partial T}{\partial \dot{\theta}_G} = i_{xx}\left(\dot{\theta}_G + \dot{\psi}\sin\phi\right) \tag{93}$$

When differentiated with respect to time and neglecting higher order terms, equation 93 becomes:

$$\frac{\partial}{dt}\left(\frac{\partial T}{\partial \dot{\theta}_G}\right) = i_{xx}\left(\dot{\psi}\dot{\phi}\cos\phi\right) \tag{94}$$

5.7. Gyroscopic Theta

As shown in the gyroscopic derivations and associated equations, the sine and cosine of the rotor angle θ_G frequently appear. As the rotor is spinning its angle cannot be assumed small, and therefore its sine and cosine terms cannot be linearized.

Accordingly this necessitates that the rotor's angle θ_G be calculated. Fortunately, as the gyroscope possesses a constant spin, its total angular rotation at a given point is readily determined.

The angle swept by the rotor can be determined by multiplying the rotor's rotational speed by the time increment at that point, where $\dot{\theta}_G$ is constant:

$$\theta_G(i) = \dot{\theta}_G \cdot t(i) \tag{95}$$

Within zone 1, the rotor angle at the first node is initialized with an angle of 0 radians. Subsequent angles at respective nodes are calculated by multiplying the rotor speed and the nodal time increment. Zone 2 is managed in a similar fashion to previously developed nodal discretization calculations. The first node of sub-zone 2.1 is initialized as the last node of zone 1. Subsequent nodes within sub-zone 2.1 are determined recursively by multiplying the rotor speed by the time increment at the respective node. Sub-zone 2.2 is initialized by the last node of sub-zone 2.1, and subsequent nodes within zone 2.2 can be determined by the product of the rotor speed and the time increment at the respective node. Finally, the rotor's angular displacement in zone 3 is determined much the same as with zones 1 and 2. The first node of zone 3 is initialized as the last node of sub-zone 2.2. Subsequent nodes are determined from the product of the rotor speed and the respective time increment. With all three zones defined, the entire angular displacement of the rotor can therefore be determined for the entirety of the body's travel.

5.8. Conclusions

The moments induced by the gyroscopic stabilizer can be obtained via its kinetic energy. The kinetic energy of the gyroscope is obtained using the classic expression with the moment of inertia symmetric about the Y spin axis. As the gyroscope experiences the

same yaw and roll as that of the body, these angles are identical. However, the gyroscope has a constant spin and therefore has a constant pitch velocity unlike the body. The gyroscope must have a negative spin in order to force a negative feedback by the body's left swerving maneuver. With the gyroscopic moments now obtained, the original body moments from the main processor can be augmented and a performance assessment made.

CHAPTER 6

SIMULATION RESULTS

Both the body's moments and the gyroscope's moments were calculated independently from the same input data. As the gyroscope applies moments to the body, its performance can be determined by simply summing the two results per the original body equations of motion.

6.1. Combined Results

With the moments on the body known, and the moments induced by the gyroscope known, it is possible to find the combined motion of the body and gyroscope. If the body moments are those due to the quadratic forces:

$$(M_\theta)_{Body} = F_{Q,\theta} \tag{96}$$

And the gyroscope's forces are those which are applied to the body:

$$(M_\theta)_{Gyro} = F_{A,\theta} = \begin{bmatrix} M_\phi \\ M_{\theta_G} \\ M_\psi \end{bmatrix} \tag{97}$$

Then the resultant motion can be expressed as the sum of the two:

$$(M_\theta)_{Combined} = F_{Q,\theta} + F_{A,\theta} \tag{98}$$

6.2. Simulation Inputs

As stated in this thesis, the body is simulated as a rectangular prism creating a highly simplified model of a tractor trailer. To further this the body is modeled using data from a 1988 Pines 48ft trailer [12] as shown Table 3. The trailer is assumed to be

traveling at 55 mph, and assumed to take 5 seconds to make a single lane to lane swerving maneuver on a Californian freeway. The maneuver time is arbitrarily selected to provide a reasonable estimate of actual vehicle maneuver times. The gyroscope is modeled as a single rotating disk whose axis is parallel to the ground with a massless spindle. All units are standardized to inches, seconds, slugs, and radians for distance, time, mass, and angle respectively.

The gyroscope's material is selected as depleted uranium in order to provide for a realistic material which has high density, is easily obtained, and can be used structurally. The gyroscope's final geometry is selected based on what could reasonably be expected to be placed into an actual application while maintaining a high aspect ratio in order to increase its moments of inertia along the roll and yaw axes.

The nodal quantity was selected to provide an acceptable resolution of the track geometry while maintaining a reasonable run time on a quad-core PC. Increased nodes increase the track geometry resolution at the expense of additional computational time.

6.3. Simulation Results

Figure 13 displays the roll moment, M_{ϕ}, with respect to time for both the body and the combined resultant of the body with the gyroscope as the body travels the radial curve. As can be seen in the figure, the body's moments and the resultant moments are nearly indistinguishable from one another. This is due to the relative magnitudes of the moments between the body and the gyroscope. On average the body's moments have an order or magnitude of 10^{41} in-lb while the gyroscope's moments have an average order of magnitude of 10^{17} in-lb. When the two moments are summed to obtain the resultant, the body moments dominate, and therefore the final motion is nearly identical to what it was

initially. Multiple runs of the simulation with varying gyroscopic parameters generate similar results in that the body dominates the gyroscope. This is most plausibly explained by the relative magnitudes of the two body's properties.

While the gyroscope has adequate mass compared to the body, the body's geometry has very large moments of inertia and therefore the body's motion will dominate that of the gyroscope. Figure 14 displays the roll moments for the body and the gyroscope on separate axes for comparison of topography. While the magnitudes imply that the body's motion will dominate, the topographies show that the gyroscope's general motion is indeed responding to that of the body's. In many instances one can see that peaks and valleys in the body's moments are responded to with opposite moments of the gyroscope. This is particularly apparent as the body exists the radial curve where a large negative moment is countered with an opposing positive moment of the gyroscope. This suggests that the gyroscope is producing a motion dampening effect, but that its magnitude does not have a significant enough effect when confronted by the body.

Figure 15 displays the pitch moment M_θ with respect to time for both the body and the resultant of the body and gyroscope. Again, the outcome is very similar to that seen with the body and gyroscope for M_ϕ. The body's motion dominates that of the gyroscope and so the body does not effectively see the gyroscopic effects. Figure 16 shows the body and the resultant yaw moments M_ψ. As is the case for both M_ϕ and M_θ, one can see that the body moments dominate the gyroscopic moments and so the resultant is predominantly the motion of the body. However, unlike M_ϕ and M_θ, the yaw moments do not exhibit as large a difference in magnitude and this is shown in Figure 17. While

TABLE 3. Simulation Input

Model	Input	Value
Main Body Properties	m	984.876 slug
	h	72.1 in
	$L1$	565 in
	$L2$	102.5 in
	$L3$	114.5 in
	C_t	6×10^5 in-lb-sec
	K_t	1×10^8 in-lb
	ε	7.731×10^{-3} rad
	v	968 in/s
	g	386.088 in/s^2
Gyroscope Properties	L_{Gyro}	6 in
	R_{Gyro}	48 in
	ρ	2.14468277435E-2 slug/in^3 (Density, depleted uranium)
	$\dot{\theta}_G$	-10000 RPM
Environmental Properties	t	5 sec
	R	1200 in
	S_{Lane}	144 in
	Zone 1 Nodal Quantity, bN	28
	Zone 2 Nodal Quantity, rN	26
	Zone 3 Nodal Quantity, tN	36
	Zone 1 Distance	1000 in

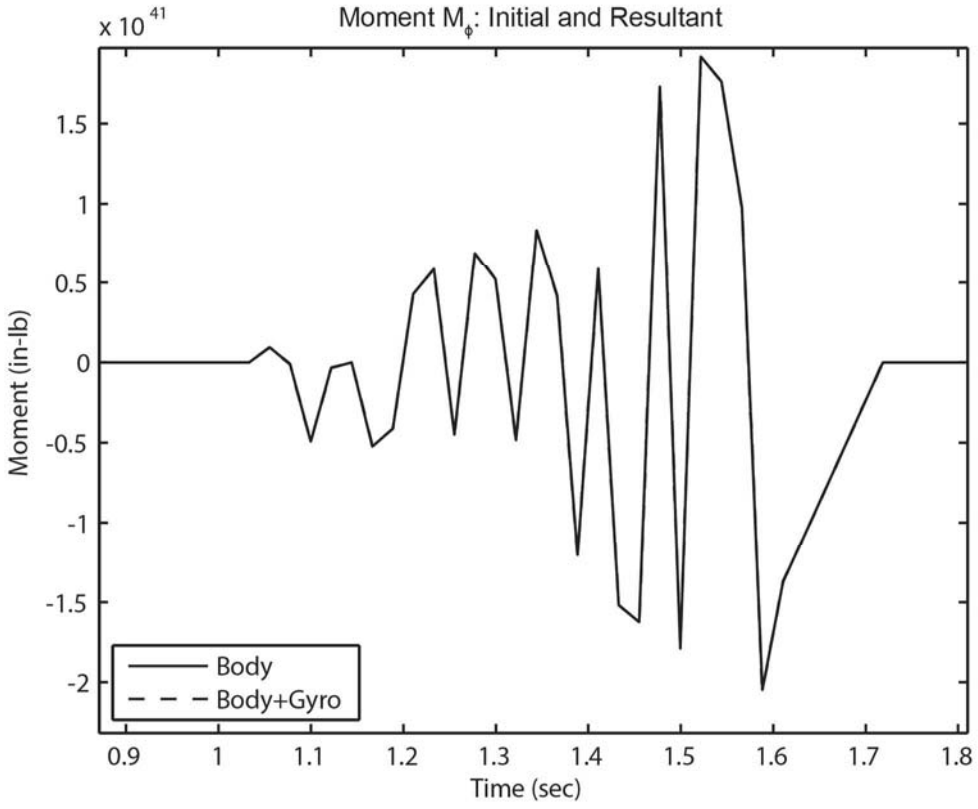

FIGURE 13. Roll moment M_ϕ with respect to time of the body without the gyroscopic stabilizer, and of the body after addition of the gyroscopic stabilizer. The body's response dominates that of the gyroscope causing the resultant to be nearly identical to that of the body's response.

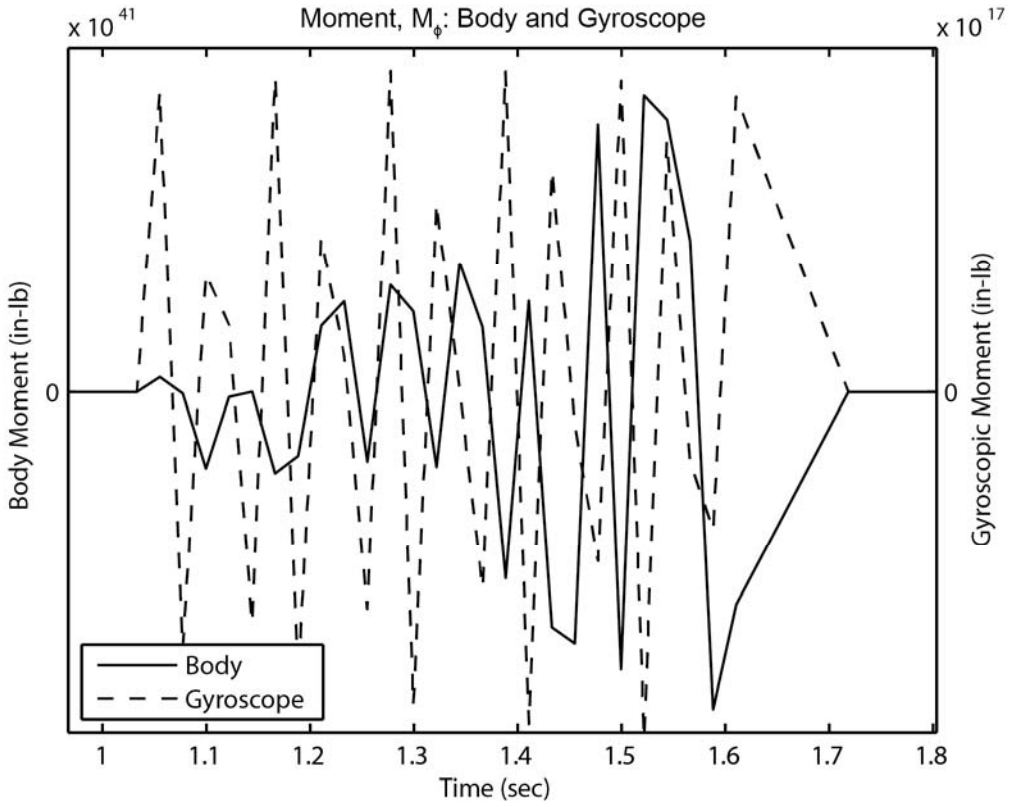

FIGURE 14. Roll moment M_ϕ of the body and of the gyroscope on separate axes. The magnitude of the body's moments are many orders of magnitude larger than that of the gyroscope and so dominate in the resultant. However the gyroscope's motion is responding to that of the body with many peaks of the body's motion plot coinciding with valleys of the gyroscope's plot.

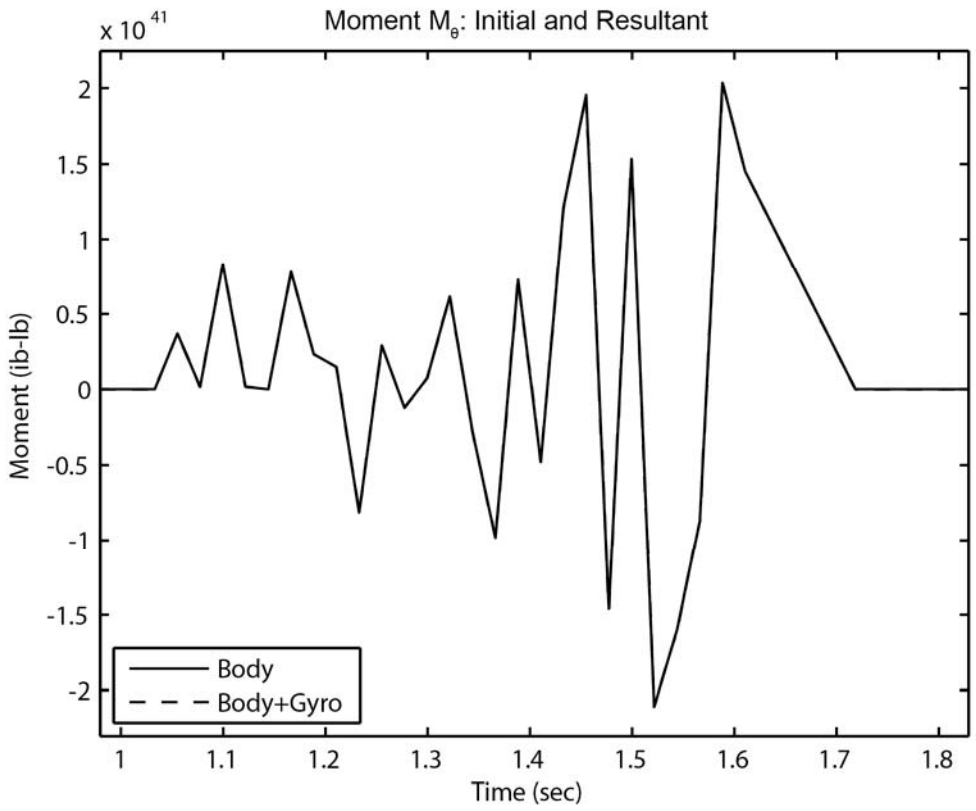

FIGURE 15. Pitch moment M_θ with respect to time of the body before and after the addition of the gyroscope. Like the other axes, the body's motion dominates that of the gyroscope and so the resultant is predominantly that of the body.

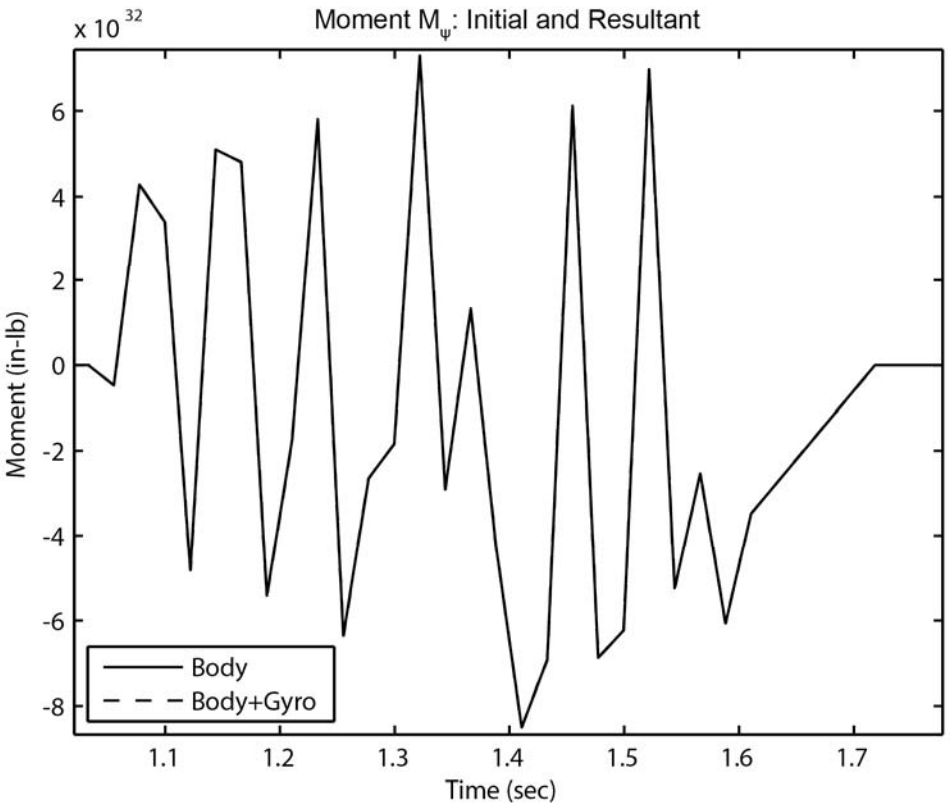

FIGURE 16. Total yaw moment of the body and the combined body-gyroscope resultant. As with the other axes the body's motion dominates that of the gyroscope and so the resultant is predominantly that of the body making the resultant nearly indistinguisable from the body alone.

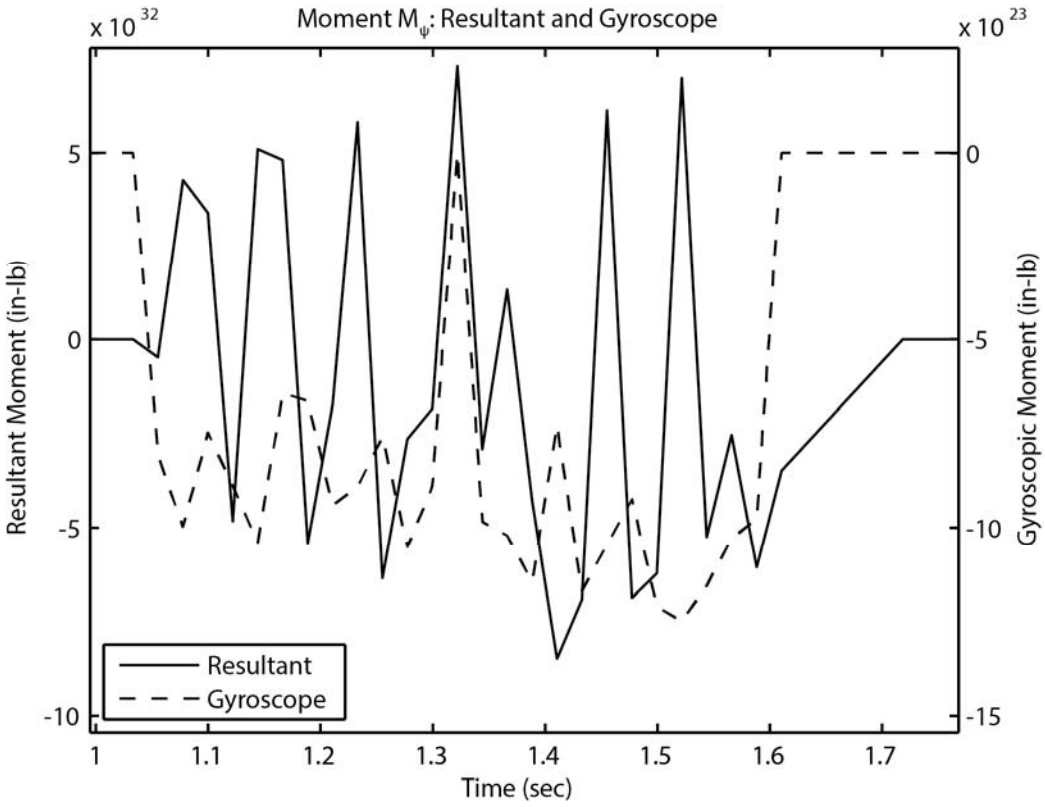

FIGURE 17. Yaw moment of the body resultant and the gyroscope on separate axes. The body's moments continue to dominate those of the gyroscope but with less of a difference. The gyroscope starts with an initial moment due to the small angle displacement from the roll initial conditions.

the ϕ and θ directions see differences in magnitude of greater than 20 times, the difference in magnitude between the body and resultant moments is only 10 times.Note in Figure 17 that the gyroscope begin with a small initial moment due to the initial angular displacement of the body in ϕ. Similar to Figure 15, one sees multiple instances where a peak or valley of the body's M_{ψ} has an opposite peak or valley from the gyroscope. Again, this suggests that the gyroscope is indeed responding oppositely to the body's input moments, but that the shear maganitude of difference between the two eliminates any effect from the gyroscope.

6.4. Simulation Conclusion

The simulation data shows that the body's moments will dominate those of the gyroscope. This effectively negates any effects that the gyroscope would potentially have on the body and therefore reduces any stabilizing effects. However, in the case of the yaw moments, there is a slight difference from the trend. The difference in magnitude between the body and gyroscope is nearly half that as seen for the other axes.

CHAPTER 7

CONCLUSION AND FUTURE DIRECTIONS

The problem presented in this thesis was to assess the performance of a simplified solid body equipped with a passive gyroscopic stabilizer using a simplified computational spatial model. The model was successfully able to calculate the body's dynamic performance via its induced moments both before and after the addition of a gyroscopic stabilizer. Additionally, the model's derivation has been presented in a simplified form and to assist in maintainability and usability. However, despite the model's successful calculations, it was unable to confirm that a fixed passive gyroscopic stabilizer has a measureable effect on the body's motion. The greatest cause of this is due to the relative magnitudes between the moments of inertia of the body and gyroscope, respectively. To this end, the body was of such sufficiently large geometry that any induced moments provided by the gyroscope were overwhelmingly dominated by the body and therefore had minimal effect on the resultant.

However, despite this, the model still retains utility in providing a qualitatively accessible computational model for simple bodies, with potential application in other studies. While the magnitude of a tractor-trailer's inertia tensor appears to make a fixed passive gyroscopic stabilizer an inappropriate choice, this may not be the case for a smaller body. It is entirely conceivable that a small body, on the order of a personal vehicle, may be well suited for passive gyroscopic stabilization depending on the desired

final performance criteria. Further, due to the modular nature of the computational model, it is possible to remove individual functions and substitute new ones thereby expanding its features or altering them. The gyroscopic processor, for example, requires input data from a source that meets its basic input requirements. With appropriate input data provided, the module could be used to compute gyroscopic moments for a multitude of differently shaped bodies. The other computational modules can be similarly altered for more advanced investigations.

Potential future direction of the current model is envisaged as an expansion of the gyroscopic model and refinement of the preprocessor. Regarding the gyroscopic model, it would be desirable to view what an actively controlled gyroscopic stabilizer's performance might be. An actively controlled stabilizer would allow the gyroscope to adapt its roll, yaw, and spin in response to those of the body. A dual acting stabilizer is an additional possibility with the two gyroscopes acting in unison upon the body. Concerning the preprocessor, it would be desirable, and likely to yield greater insight, if general path geometry could be input rather than the static path defined in Figure 5. Therefore, while the results of the current system configuration do not support the use of a fixed passive gyroscopic stabilizer for a tractor-trailer, an alternative configuration may yield very different results, and this would be the recommended next step for research.

APPENDICES

APPENDIX A

MAIN PROCESSOR MATLAB SOURCE CODE

```
%*************************************************************************
%ManPro: Main Processor, V1
%*************************************************************************
%This is the main script file to calculate the body moments and
%gyroscopic moments.  PrePro is first run in order to prepare all the data
%for calculation.  Once prepared, ManPro calculates the moments for each
%node iteratively.  PostPro can be run, or not, as it simply formats plots
%for easy review.
%
%Body moments are calculated based on the main equations of motion (EOM).
%The equations of motion calculate moments at a single instance, therefore
%it is necessary to perform calculations at each node to get a history of
%the body's motion.  Output of each count is dumped to QV_THETA array which
%is then used for analysis.  QV_THETA contains the history of the body
%moving along track.  Each count creates a matrix from the EOM, these could
%be used for analysis is so desired with minor adjustments to the main body
%loop.
%
%PARENT FUNCTION: None.
%
%*************************************************************************
%VARIABLE DEFINITIONS
%*************************************************************************
%Main Variables within ManPro (Main Processor)
%psi1_i              Psi angle of main body w/o gyroscope at a single node
%phi1_i              Phi angle of main body w/o gyroscope at a single node
%theta1_i            Theta angle of main body w/o gyroscope at a single node
%dphi1_i             dPhi/dt of main body w/o gyroscope at a single node
%dpsi1_i             dPsi/dt of main body w/o gyroscope at a single node
%genDTheta           Orientation vector theta, theta (euler angle) = 0,
%                    no pitch.
%A1                  Main body euler transformation
%dA1                 First time derivative of euler transformation
%I1_bar_thetatheta   Inertia tensor in LCS
%G1bar               Unit vector euler transformation
%dG1bar              Frist time derivative of unit vector euler
%                    transformation
%M                   Mass matrix
%omegaBarTilda       Angular velocity matrix, in LCS and skew symmetric
%omegaBar            Angular velocity matrix, in LCS, not skew symmetric
%Qv_theta            Quadratic angualr forces
%QV_THETA            Transpose of Qv_theta, for formatting into other
%                    functions
%GYRO                Gyroscopic processor and associated outputs
%RESULTANT           Body quadratic mmoments + gyroscopic moments, total
%                    moments on body w/ gyroscope
%DA                  Output array of PrePro (pre-processor) function
%bNodes              Nodes in Zone 1, boundary
%tNodes              Nodes in Zone 3, tangent
%rNodes              Nodes in Zones 2.1, 2.2, radial
%H_CG                Location of main body CG
%L1                  Body length
%L2                  Body width
%L3                  Body height, above rails, excluding suspension
%h                   Suspension height; rails
%*************************************************************************
%GLOBAL VARIABLES & DEFINITIONS
%*************************************************************************
%
%Global variables are used to pass data back and forth between functions.
%
%NOTE: Variables which contain the C appended to them are the corrected
%arrays. In order to avoid overlapping and double-counting nodes, the
%loop outputs of each zone are corrected and then summed to have the
%total nodes input by the user.  If not, the nodes will overlap at
%each terminus, as that terminus is the start of a new zone, causing
%the overlap.
%
%   bN rN tN                Zone nodal counts
```

```
%   x                       x track coordinates
%   y                       y track coordinates
%   z                       z track coordinates
%   s                       Arc length
%   ch                      Curvature
%   m                       Mass
%   n                       node
%   psi                     Psi, yaw angle
%   t                       Time
%   phi                     Phi, roll angle
%   dphi                    dPhi/dt, roll velocity
%   lambda1,C, J, L, m_h    Intermediary roll calculations
%   dpsi                    dPsi/dt, yaw velocity
%   ddphi                   d/dt(dPhi/dt), roll acceleration
%   Mth                     Gyroscopic moments
%   thgy                    Gyroscopic theta with respect to time
%
%*************************************************************************
%
%
global bN rN tN
global x1 x2 x3 x1C x2C x3C x2S
global y1 y2 y3 y1C y2C y3C y2S
global z1 z2 z3 z1C z2C z3C
global s1 s1C s21 s21C s22 s22C s3 s2C s3C
global ch1 ch1C ch2C ch3 ch3C ch21 ch22 ch21C ch22C m21 m22
global n1 n1C n2 n2C n3 n3C
global psi1 psi1C psi21 psi22 psi21C psi22C psi3 psi3C psi2C
global t1 t1C t21 t21C t22 t22C t23 t23C t3
global phi1 phi1C phi21 phi21C phi22 phi22C phi2 phi2C phi3 phi3C
global dphi1 dphi1C dphi21 dphi21C dphi22 dphi22C dphi2 dphi2C dphi3 dphi3C
global lambda1 lambda2 C1 C2 J L1 L2 L3 H_CG m h
global dpsi1 dpsi21 dpsi22 dpsi2 dpsi1C dpsi21C dpsi22C dpsi3 dpsi3C
global ddphi1 ddphi1C ddphi21 ddphi21C ddphi22 ddphi22C ddphi2 ddphi2C ddphi3 ddphi3C
global Mth1 Mth1C Mth21 Mth22 Mth2C Mth3 Mth3C
global thgy1 thgy1C thgy2C thgy21 thgy22 thgy3 thgy3C
%
clear all %Clear out all data
%
clc %Clear out workspace
%
disp('****Starting Simulation****');
disp('--');
disp('--');
pause(1);
disp('Running PrePro');
disp('--');
pause(1);
PrePro %Run pre-processor and get data matrix DA to feed rest of processor
disp('PrePro Complete');
disp('--');
pause(1);
%
disp('Running ManPro');
disp('--');
pause(1);
format long
%
%*************************************************************************
%Body Processing
%*************************************************************************
%
for i = 1:1:numel(N) %begin counting per each node.
     psi1_i = DA(i,6);
     phi1_i = DA(i,4);
     theta1_i = DA(i,5);
     dphi1_i = DA(i,10);
     dpsi1_i = DA(i,11);
     genDTheta1_i = [dphi1_i; 0; dpsi1_i];
```

```
      A1(:,:,i) = a1transform(psi1_i, phi1_i, theta1_i);
      dA1(:,:,i) = a1dottransform(psi1_i,phi1_i,dpsi1_i,dphi1_i);

      I1_bar_thetatheta(:,:,i) = I1bar(m, phi1_i, H_CG, L1, L2, L3);

      G1bar(:,:,i) = transpose(A1(:,:,i))*...
                     g1transform(psi1_i, phi1_i, theta1_i);

      dG1bar(:,:,i) = g1bardot(phi1_i, psi1_i, dphi1_i);

      M(:,:,i) = mass1(m, G1bar(:,:,i), I1_bar_thetatheta(:,:,i), ...
                 A1(:,:,i), L1, L2, L3, h, phi1_i);

      omegaBarTilda(:,:,i) = omegabar_tilda(A1(:,:,i),dA1(:,:,i));

      omegaBar(:,:,i) = omegabar(G1bar(:,:,i),genDTheta1_i);

      Qv_theta(:,:,i) = -transpose(G1bar(:,:,i))*...
                        (cross(omegaBar(:,:,i),(I1_bar_thetatheta(:,:,i)*...
                        omegaBar(:,:,i)))+I1_bar_thetatheta(:,:,i)*...
                        dG1bar(:,:,i)*genDTheta1_i); %Calculate QV_theta matrix
      QV_THETA = transpose(Qv_theta(:,:)); %Change to vertical array
end
%*************************************************************************
%Gyroscopic Processing
%*************************************************************************
%Run gyroscopic moment calcualtions contained within gyroscopic function
%
disp('Running GyroPro');
disp('--');
pause(1);
GYRO = GyroPro(bNodes, tNodes, rNodes, ixx_gyro, iyy_gyro, dtheta_gyro);
disp('GyroPro Complete');
disp('--');
pause(1);
%
%*************************************************************************
%
%
%Calculate combined moments of body and gyroscope.  Resultant motion is
%sum of body's moments and gyro's moments.
%
RESULTANT = QV_THETA + GYRO;
%
%*************************************************************************
%Post Processing
%*************************************************************************
%Run post processing script which formats plots for review; optional, can
%comment out if desired and analyze nodal data in application of choice.
%
disp('ManPro Complete');
disp('--');
pause(1);
disp('Running PostPro');
disp('--');
pause(1);
PostPro
disp('PostPro Complete');
pause(1);
disp('--');
disp('--');
disp('****Simulation Complete****');
%*************************************************************************
```

APPENDIX B

PRE-PROCESSOR MATLAB SOURCE CODE

```
%*************************************************************************
%PrePro: Pre Procressor, V1
%*************************************************************************
%This is the main script to process all of the inputs for the main motion
%equations.  PrePro calculates the track nodes, nodal positions, time, and
%euler rotation angles.  All track geometry is designed into the coordinate
%functions directly.  Each of the below functions calculates the specific
%variable at the current node.  For example, the third element in the X
%coordinate array is the X coordinate that belongs to node 3, and so on.
%
%NOTE: It is important that the order in which the functions are called
%remains as-is, or are carefully rearracnged.  Certain functions require
%data that is calcualted from other functions.  This data is in turn called
%out as global variables to make it available globally.
%
%PARENT FUNCTION: This script is called in ManPro
%
%*************************************************************************
%VARIABLE DEFINITIONS
%*************************************************************************
%NOTE: All global variables are defined in the ManPro documentation.
%N                      Nodal count array of node numbers
%X                      X track coordinate array
%Y                      Y track coordinate array
%Z                      Z track coordinate array
%THETA                  Body pitch angle array
%S                      Arc length array
%T                      Time array
%CH                     Curvature array
%PSI                    Yaw angle array
%PHI                    Roll angle array
%DPHI                   Roll velocity array
%DDPHI                  Roll acceleration array
%DPSI                   Yaw velocity array
%DA                     Preprocessed data matrix
%alpha                  Intermediary variable for radial calculations
%beta                   Intermediary variable for radial calculations
%zeta                   Intermediary variable for radial calculations
%*************************************************************************
%
format short
%Define global variables for use across functions
global bN rN tN
global x1 x2 x3 x1C x2C x3C x2S
global y1 y2 y3 y1C y2C y3C y2S
global z1 z2 z3 z1C z2C z3C
global s1 s1C s21 s21C s22 s22C s3 s2C s3C
global ch1 ch1C ch2C ch3 ch3C ch21 ch22 ch21C ch22C m21 m22
global n1 n1C n2 n2C n3 n3C
global psi1 psi1C psi21 psi22 psi21C psi22C psi3 psi3C psi2C
global t1 t1C t21 t21C t22 t22C t23 t23C t3
global phi1 phi1C phi21 phi21C phi22 phi22C phi2 phi2C phi3 phi3C
global dphi1 dphi1C dphi21 dphi21C dphi22 dphi22C dphi2 dphi2C dphi3 dphi3C
global lambda1 lambda2 C1 C2 J L1 L2 L3 H_CG m h
global dpsi1 dpsi21 dpsi22 dpsi2 dpsi1C dpsi21C dpsi22C dpsi3 dpsi3C
global ddphi1 ddphi1C ddphi21 ddphi21C ddphi22 ddphi22C ddphi2 ddphi2C ddphi3 ddphi3C
global Mth1 Mth1C Mth21 Mth22 Mth2C Mth3 Mth3C
%
disp('Running VarInput');
disp('--');
pause(1);
VarInput %Input user defined variables through VarInput script.
disp('VarInput Complete');
disp('--');
pause(1);
%
%------------------------
%Calculate angles for radial distances
%to pass to other functions
```

```
alpha = atan(sLane-2*rTurn)/(vMan*tMan);
beta = acos((2*rTurn)/sqrt((vMan*tMan)^2+(sLane-2*rTurn)^2));
zeta = Pi/2 - (beta-alpha);
%
%Call functions
%************************************************************************
N = nodalCount(bNodes, tNodes, rNodes);
X = xCount(bNodes, tNodes, rNodes, sBuff, rTurn, vMan, tMan, zeta, sLane);
Y = yCount(bNodes, tNodes, rNodes, sBuff, rTurn, vMan, tMan, zeta, sLane);
Z = zCount(bNodes, tNodes, rNodes, sBuff, rTurn, vMan, tMan, zeta);
THETA = thCount(bNodes, tNodes, rNodes, sBuff, rTurn, vMan, tMan, ...
    sLane, zeta);
S = sCount(bNodes, tNodes, rNodes, sBuff, rTurn, vMan, tMan, ...
    sLane, zeta);
T = tCount (bNodes, tNodes, rNodes, sBuff, rTurn, vMan);
CH = chCount(bNodes, tNodes, rNodes, sBuff, rTurn, vMan, tMan, ...
    sLane, zeta);
PSI = psiCount(bNodes, tNodes, rNodes, sBuff, rTurn, vMan, tMan, ...
    sLane, zeta);
PHI = phiCount(bNodes, tNodes, rNodes, Kt, Ct, m, g, L1, L2, L3, h, rTurn,...
              vMan, eps);
DPHI = dphiCount(bNodes, tNodes, rNodes, Kt, Ct, m, g, L1, L2, L3, h, rTurn, vMan, eps);
DDPHI = ddphiCount(bNodes, tNodes, rNodes, Kt, Ct, m, g, L1, L2, L3, h, rTurn, vMan,
eps);
DPSI = dpsiCount(bNodes, tNodes, rNodes);
%
%Compile data into single matrix
DA = [N S CH PHI THETA PSI X Y Z DPHI DDPHI DPSI];
```

APPENDIX C

VARIABLE INPUT SCRIPT MATLAB SOURCE CODE

```
%************************************************************************
%VarInput: Variable Input, V1
%************************************************************************
%This script consolidates all of the user defined variables into a single
%location for ease of modification.
%
%NOTE: All units are inches, seconds, slugs, pounds, and radians
%
%PARENT FUNCTION: This script is called in ManPro
%
%************************************************************************
%VARIABLE DEFINITIONS
%************************************************************************
%vMan                       Body speed, in/sec
%tMan                       Time of maneuver, sec
%sBuff                      Entry (zone 1) distance, in
%rTurn                      Radial curve radius, in
%sLane                      Center-center lane distance, in
%bNodes                     Desired zone 1 nodal count, #
%tNodes                     Desired zone 3 nodal count, #
%rNodes                     Desired zone 2 nodal count, #
%Pi                         Define pi for radial calculations
%g                          Define gravitational constant, in/s^2
%m                          Body mass, slug
%h                          Height of body above ground, in
%L1                         Body length, in
%L2                         Body width, in
%L3                         Body height, in
%Ct                         Rotational damper constant, in-lb-s
%Kt                         Rotational spring constant, in-lb
%H_CG                       CG location, in
%L_gyro                     Width of rotor, in
%R_gyro                     Gyro radius, in
%rho                        Density of rotor material, slug/in^3
%V_rotor                    Rotor volume, in^3
%m_rotor                    Rotor mass, slug
%rotor_speed                Speed of rotor, rev/min
%dtheta_gyro                Gyroscope angular velocity, rad/sec
%ixx_gyro                   Gyro mass moment along x axis, in^4
%iyy_gyro                   Gyro mass moment along y axis, in^4
%************************************************************************
%Body Inputs
%************************************************************************
pause(1);
vMan = 968;          %in/s, equivalent to 55mph
tMan = 5;           %sec, time to swerve
sBuff = 1000;        %in, Buffer distance
rTurn = 1200;        %in, turn radius
sLane = 144;         %in, lane distance (12ft)
bNodes = 28;         %#, nodes in buffer
tNodes = 36;         %#, tangent nodes
rNodes = 26;         %#, radial turn nodes
Pi = 3.14159265358979323846264338327795; %define pi
g = 386.088;         %in.s^2, gravity
eps = 7.731E-3;      %rad, epsilon, equiv. to 1 in distance
%************************************************************************
%TRAILER DATA:
%http://tractor-trailer.model.ntrci.org/test/test.cgi?model=1&navv=2
%************************************************************************
%Year: 1988
%Make: Pines
%Model: 48' Van
m = 984.876;         %mass, slugs
h = 72.1;            %in, height, lower rail
L1 = 565;            %in, total length, D+E+F+K
L2 = 102.5;          %in, Width, A
L3 = 114.5;          %in, Cab height, W-L
Ct = 6E5;            %in-lb-sec, rotary damper coefficient
Kt = 1E8;            %in-lb, spring coefficient
```

```
H_CG = h+L3/2;
%*************************************************************************
%GYRO DATA
%*************************************************************************
L_gyro = 6; %rotor thickness (in)
R_gyro = 48; %gyro radius (in)
rho = 2.14468277435E-2; %density depleted uranium, slug/in^3
rotor_speed = -10000; %speed in RPM
%*************************************************************************
%Intermediate calculations
V_rotor = Pi*R_gyro^2*L_gyro; %calculate volume to find mass
m_gyro = V_rotor*rho; %gyro mass(slugs)
dtheta_gyro = (rotor_speed*2*Pi)/60; %convert RPM to rad/s
ixx_gyro = 0.083333*m_gyro*(3*R_gyro^2+L_gyro^2); %x mass moment
iyy_gyro =  0.5*m_gyro*R_gyro^2; %y mass moment
%*************************************************************************
%PERFORM CHECK-OUT
%*************************************************************************
%Check that input data is OK with constraints, if not, exit.
Jphi = (L2*L3^3)/12+L2*L3*(h+L3/2)^2; %Calculate Jphi for checks
H = h+L3/2;%Calculate H for checks
A1 = Ct/Jphi;
A2 = (Kt-m*g*H)/Jphi;
B1 = Ct^2;
B2 = 4*m*Kt;
pause(1);
disp('Running VarInput data checks');
disp('--')
pause(1);
disp('Checking discriminant');
disp('--');
pause(1);
if (A1^2 - 4*A2)<0
    disp('Discriminant less than zero, check OK');
    disp('--')
    disp('Checking for underdamped system');
    disp('--');
    pause(1);
else
    disp('WARNING: Discriminant larger than zero');
    disp('--');
    pause(1);
end;

if B1 < B2
    disp('Underdamped system, check OK')
    disp('--');
    pause(1);
    disp('VarInput checks complete.');
    disp('--');
    pause(1);
else
    disp('WARNING: System is not underdamped.');
    disp('--');
    pause(1);
    end;
%*************************************************************************
%Clear Variables no longer needed
%*************************************************************************
clear Jphi H A1 B1 A2 B2 %clear variables since no longer needed
```

BIBLIOGRAPHY

BIBLIOGRAPHY

Primary Sources

[1] Federal Motor Carrier Safety Administration, 2014, "Large Truck and Bus Crash Facts 2012," U.S. Department of Transportation.

[2] Gillespie, T.D., 1992, *Fundamentals of Vehicle Dynamics*, Society of Automotive Engineers, Inc., Warrendale, PA, Chap. 9

[3] Winkler, C.B., and Ervin, R. D., 1999, "Rollover of Heavy Commercial Vehicles," The University of Michigan, Ann Arbor, MI.

[4] Alexander, L., Donath, M., Hennessey, M., Morellas, V., and Shankwitz, C., 1996, "A Lateral Dynamic Model of a Tractor-Trailer Experimental Validation," University of Minnesota, Minneapolis, MN.

[5] Tianjun, Z., Xiaojun, S., and Changfu, Z., 2009, "Development of a Tractor-Semitrailer Stability Control Model," 2009 2nd International Conference on Power Electronics and Intelligent Transportation Systems, IEEE, Washington DC, Vol. 1

[6] Shabana, A.A., Zaazaa, K. E., and Sugiyama, H., 2008, *Railroad Vehicle Dynamics, A Computational Approach*,CRC Press, Boca Raton, FL.

[7] Rathod, C., and Shabana, A. A., 2006, "Rail Geometry and Euler Angles," Journal of Computational and Nonlinear Dynamics, 1(July 2006), pp. 264 - 268.

[8] Shabana, A. A., Hamper, M. B., and O'Shea, J. J., 2013, "Rolling Condition and Gyroscopic Moments in Curve Negotiations," Journal of Computational and Nonlinear Dynamics, 8(January 2013).

[9] MeritorWabco, 2014, "Roll Stability Support (RSS)," www.meritorwabco.com/ Product,2,32,2,Roll-Stability-Support-(RSS).aspx

[10] Townsend, N. C., and Shenoi, R. A., 2011, "Gyrostabilizer Vehicular Technology," Applied Mechanics Reviews, 64(January 2011).

[11] Scarborough, J.B., 1958, *The Gyroscope Theory and Applications*, Interscience Publishers Inc., New York, NY, Chap. VIII.

[12] Simunovic, S., and N. Zisi, 2014, "FEM Models for Semitrailer Trucks," http://tractor-trailer.model.ntrci.org

[13] California Highway Design Manual, 2014, California Department of Transportation, pp. 300-1.

[14] Shabana, A. A., 2010, *Computational Dynamics, 3rd. ed,* John Wiley & Sons, Inc., New York, NY.

[15] Esfandiari, R. S., 2008, *Applied Mathematics for Engineers, 4th ed*, Atlantis Publishing, Los Angeles, CA, Chap. 2.

Seconday Sources

[16] Meirovitch, L., 2001, *Fundamentals of Vibrations,* Waveland Press, Inc., Long Grove, IL, Chap 2.

[17] Cossalter, V., 2006, *Motorcycle Dynamics 2nd ed.*, Lulu Press Inc., Raleigh, NC, Chap. 4.

[18] Popp, K., and Schiehlen, W., 2010, *Ground Vehicle Dynamics*, Springer-Verlag, Berlin, Chap. 2.

[19] Barickman, F., Elsasser, D., Albrecht, H., and Xu, G., 2011, "Tractor Semi-Trailer Stability Objective Performance Test Research-Roll Stability," National Highway Traffic Safety Administration, Washington, DC.