ME 565 - Vehicle Dynamics Simulation 4

Abhishek Nair

Model Description

The model is comprised of the following variables, declared in the variable declaration of the code in the appendix, as follows:

$W = 3{,}000\ lbs$
$W_s = 2{,}700\ lbs$
$x_1 = 3.5\ ft$
$x_2 = -4.5\ ft$
$h = -1.0\ ft$
$t = 6.0\ ft$
$I_z = 40{,}000\ lbs\text{-}ft^2$
$I_x = 15{,}000\ lbs\text{-}ft^2$
$c = 0.5\ ft$
$\left.\frac{\partial L}{\partial \Phi}\right|_f = 8{,}000\ lbs\text{-}ft$
$\left.\frac{\partial L}{\partial \Phi}\right|_r = 5{,}000\ lbs\text{-}ft$
$\left.\frac{\partial L}{\partial \dot{\Phi}}\right|_f = 1{,}000\ lbs\text{-}ft/sec$
$\left.\frac{\partial L}{\partial \dot{\Phi}}\right|_f = 500\ lbs\text{-}ft/sec$
$p = d = 12\ in$
$\eta = 15{:}1$
$\epsilon = 1$ (steering gearbox efficiency, NOT roll steer coefficient – let's call it 1)
$K_s = 10\ in\text{-}lbs/deg$
$t_m = 3\ in.$

Cornering Coefficients:
For the linear part of the simulation use: $C_i = 140\ lbs/deg$

The following formula for nonlinear tire stiffness can be used where W is the weight on a given tire in pounds (lbs): $C_i = (0.2 * W_i - 0.0000942 * W_i^2)\ lbs/deg$

Steering Compliance:
Use the following equations from lecture to determine the handwheel input required to produce a given steer angle at the tire:

$$\delta_{hw} = \frac{2M_t p}{d\eta\epsilon K_s} + \delta\frac{\eta d}{p} \qquad M_t = t_m F_{lat}$$

1: 2-DoF Vehicle (Linear & Nonlinear)

1.1:

Using the code labelled in the appendix for part 1.1, the following eigenvalues were obtained for the state space model equation 10.22 from the textbook:

$$\begin{bmatrix} \dot{\beta} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \dfrac{-C_1 - C_2}{mu} & \left(\dfrac{-x_1 C_1 - x_2 C_2}{mu^2} - 1 \right) \\ \dfrac{-x_1 C_1 - x_2 C_2}{I_z} & \dfrac{-x_1{}^2 C_1 - x_2{}^2 C_2}{I_z u} \end{bmatrix} \begin{bmatrix} \beta \\ r \end{bmatrix} + \begin{bmatrix} \dfrac{C_1}{mu} \\ \dfrac{x_1 C_1}{I_z} \end{bmatrix} \begin{bmatrix} \delta_1 \end{bmatrix} \qquad (10.22)$$

For speed 30: eigenvalues = -8.68 +/- 3.32i

For speed 60: eigenvalues = -4.34 +/- 3.53i

Negative eigenvalues in both speeds indicate that the vehicle is still controllable by the driver.

1.2:

The code labelled for part 1.2 was adapted from equation 10.51 in the textbook, which is as follows:

$$\frac{r_{ss}}{\delta_{1,ss}} = \frac{u}{l_2 + u^2 K_2} \qquad (10.51)$$

Yaw rate was calculated for the following speeds and tabulated:

Speed (mph)	r / δ
0	0
10	1.79823281677221
20	3.40111507634933
30	4.67816338915332
40	5.58810538485489
50	6.16045113355465
60	6.46032687957616
70	6.55948945647426
80	6.52070705387490
90	6.39268399162211
100	6.21059436607627
110	5.99867500491298
120	5.77306468893142

Table 1: Yaw rate calculated for different speeds.

The yaw rate vs speed was also plotted in the following figure:

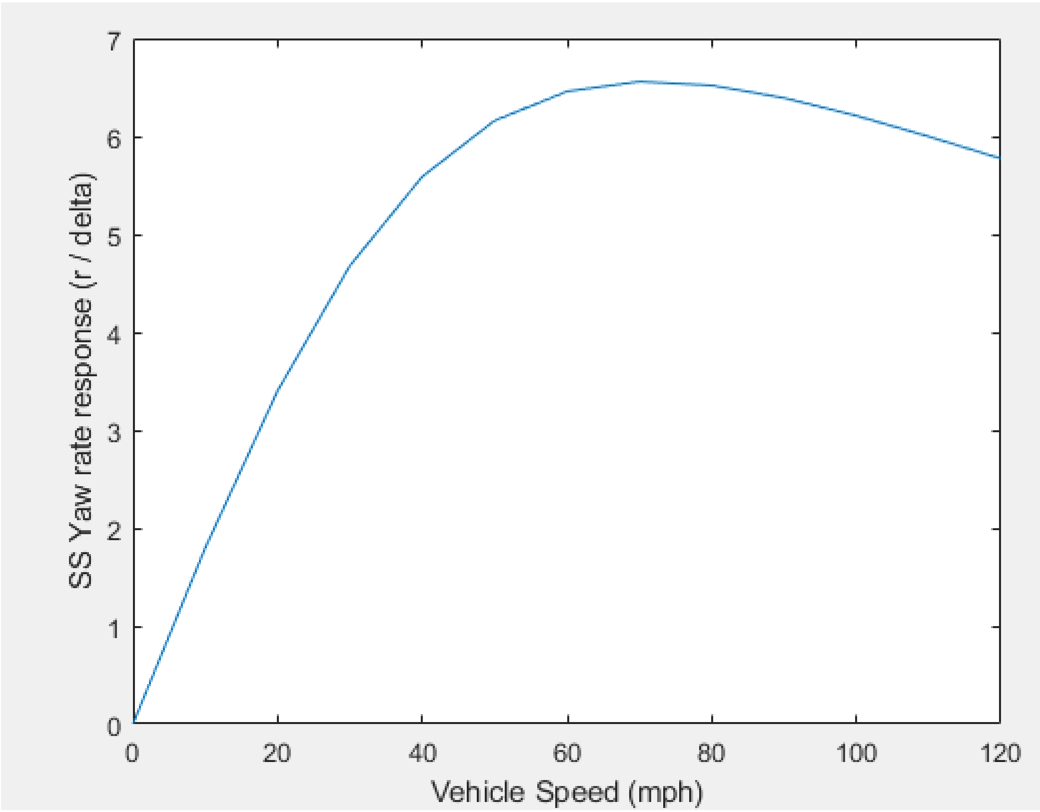


Figure 1: Yaw rate response plotted for different speeds

1.3

Using equation R = u / r, which is rearranged to r = u / R and R is constant at 400 ft, the required yaw rate for each speed is calculated.

Speed (mph)	Required Yaw rate	r / δ	β / δ
10	0.0367	1.798	0.485
20	0.0733	3.401	0.268
30	0.11	4.678	-0.045
40	0.1467	5.588	-0.404
50	0.183	6.160	-0.7699
60	0.22	6.460	-1.114
70	0.293	6.559	-1.424
80	0.2	6.521	-1.694
90	0.33	6.393	-1.926
100	0.367	6.211	-2.124
110	0.403	5.999	-2.292
120	0.44	5.773	-2.434

Table 2: Yaw rate response and drift angle response calculated for different speeds.

The table above collects data at 5 seconds, where the system is assumed to have achieved steady state (since I was not able to see any changes in values). The calculated yaw rate response values are identical to those of part 1.2, and thus the model may be assumed to be accurate. The Beta response is used to confirm the understeery nature of the vehicle, as it is noted to have a transitionary speed between 20 and 30 mph through a sign change.

1.4

The time response of steering input including human limitations are as followed, plotted using the code labelled for section 1.4.

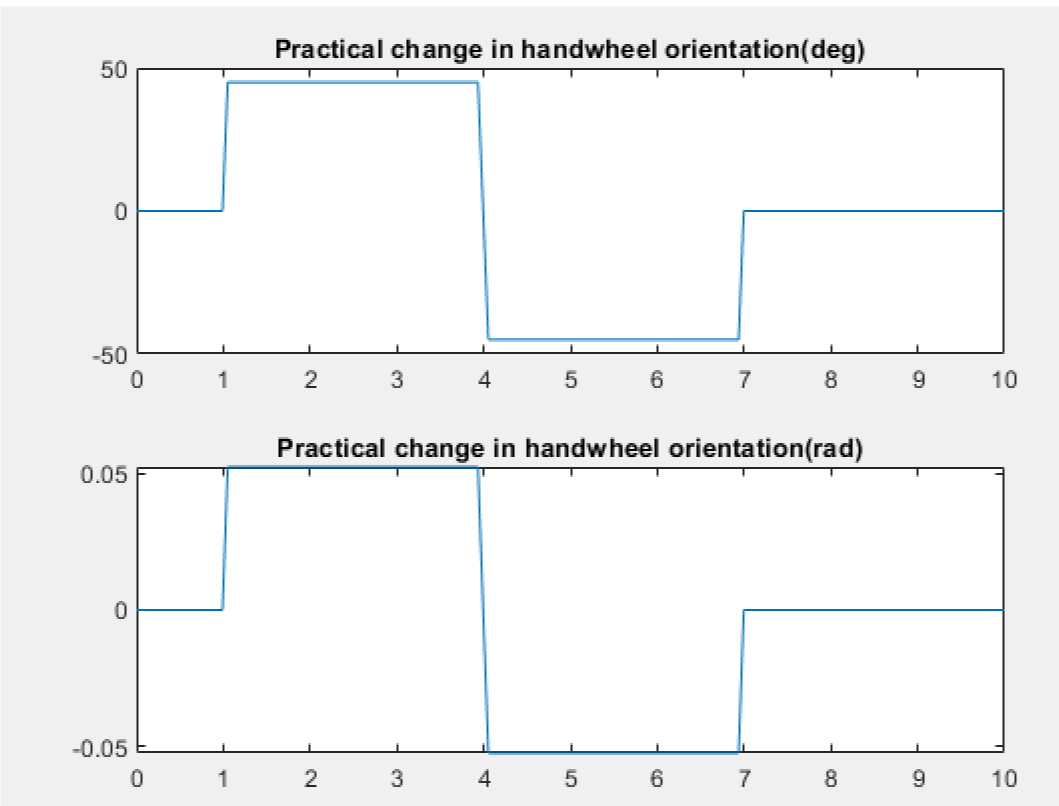


Figure 2: Handwheel input in degrees and radians

1.5

The handwheel input calculated in the previous section is first converted into steering input using the following equation 7.25 from the textbook, with the handwheel input taken as $delta_{in}$ and the output as $delta_{out}$:

$$\eta = \frac{\delta_{in}}{\delta_{out}}, \tag{7.25}$$

Using the code labelled for section 1.5, the drift angle β and yaw rate r are calculated at 30 and 60 mph as a response to the converted steering input from section 1.4. General observations are that increasing the speed drastically increased the drift angle, indicating a much wider turn at 60 mph compared to 30 mph. Increasing speed also increases the yaw rate response, as well as the settling time, as it is noted to have a greater spike as the steering input changes before setting into a higher steady state at 60 mph compared to 30 mph.

Increasing speed increases the understeery response of the vehicle, noted by wider turns and a less responsive yaw rate settling time. The increase in yaw rate from 0-30 mph is also noted to be more than 30-60 mph.

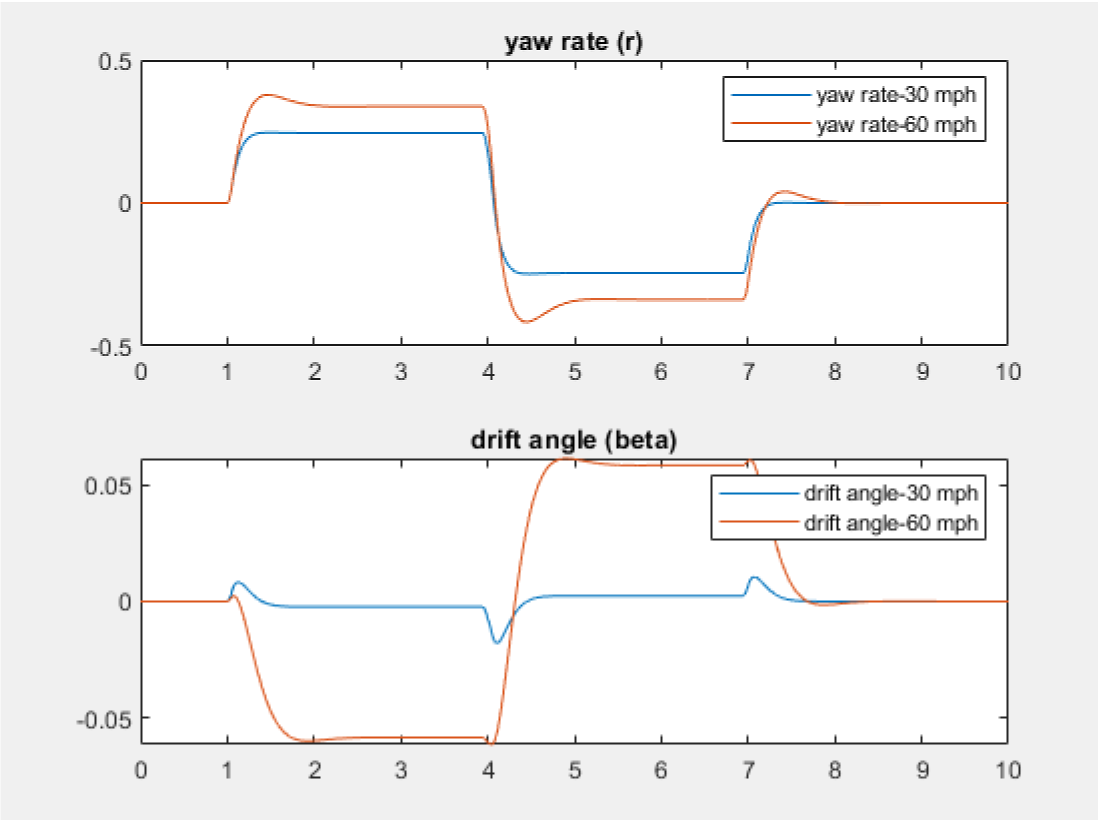


Figure 3: Yaw rate response and drift angle plotted for given input

1.6

I first started by recalculating the tire coefficients and related variables as in the code labelled for section 1.6, then calculating the steady state drift angle and yaw rate responses. The results for a 60/40 front bias are as follows:

Speed (mph)	Required Yaw rate	r / δ	β / δ
10	0.0367	1.98	0.4799
20	0.0733	3.46	0.163
30	0.11	4.27	-0.217
40	0.1467	4.57	-0.569
50	0.183	4.56	-0.859
60	0.22	4.39	-1.085
70	0.293	4.14	-1.259
80	0.2	3.89	-1.392
90	0.33	3.63	-1.495
100	0.367	3.39	-1.575
110	0.403	3.17	-1.639
120	0.44	2.98	-1.69

Table 3: Yaw rate response and drift angle response calculated for different speeds.

Comparing the 60/40 front bias to linear tires, the understeery nature of the vehicle seems to be amplified, as the vehicle with a 60/40 front bias seems to have a much lower steady state yaw rate response compared to the vehicle with linear tires. In comparing the drift angle response to those from the vehicle with linear tires, the transition is indicated to be at about the same speed as in part 1.3, between 20 and 30 mph where the sign change in the table occurs. The values however seem to be lower across the board, further indicating an understeery vehicle nature.

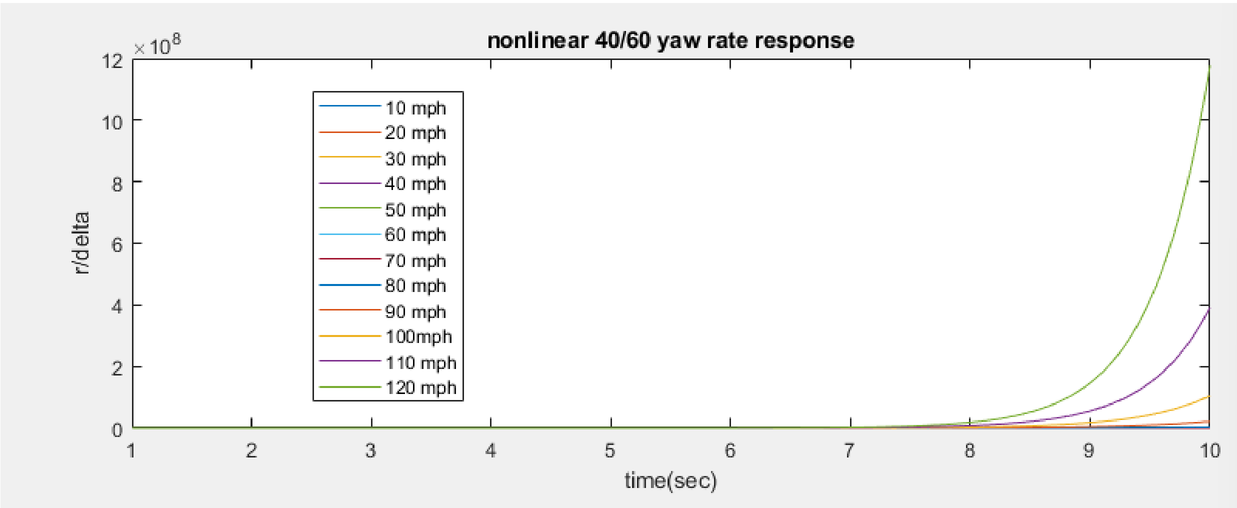
One discrepancy was noted in the calculations, however. Using the following equation 10.52 from the textbook:

$$u_{\beta=0} = \sqrt{\frac{-l_2 C_2 x_2}{m x_1}}. \tag{10.52}$$

I calculated the transition speed to be 17.8 mph, which is lower than the range in which this speed is expected in the table above. I attribute this error to possible limits in experimental accuracy, where my model could have rounded off certain variables, or I may have approximated a variable incorrectly.

To repeat this process for the 40/60 rear biased vehicle, I first recalculated the tire coefficient values which I then plugged into the same code to produce the following section.

I was unable to calculate the steady-state drift angle and yaw rate responses for a 40/60 rear bias and produce a table for the same, due to reasons that will be apparent in observing the following figure, which is the drift angle and yaw rate response of the system at various speeds simulated from 0 to 10 seconds:



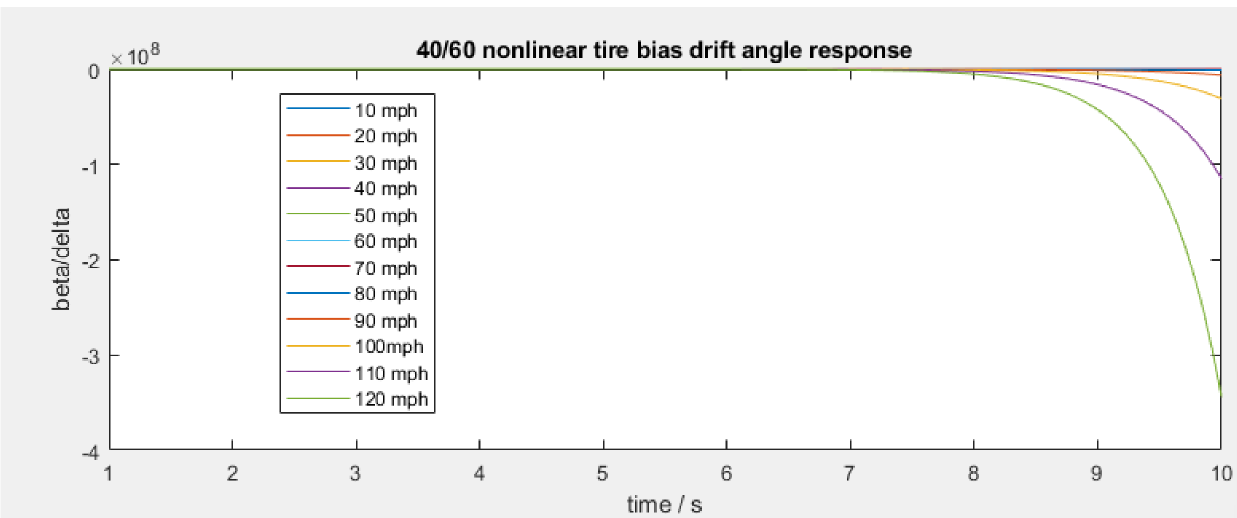


Figure 4: Yaw rate response and drift angle plotted vs time

As can be seen in the figures, the yaw rate tends to infinity at all speeds above 60 mph, and the drift angle tends to negative infinity for the same. This is indicative of an oversteering vehicle, where in this case, speeds exceeding 60 mph will require the vehicle to turn right to go left (which explains the negative drift angle!) In comparing all the other results below 60 mph, they appear to be straight lines in the above graphs, indicating neutral steer.

Since the handwheel input is identical as that of parts 1.4 through 1.5, I decided to reuse the same code in achieving that input, and in turn, calculate the drift angle and yaw rate response over 10 seconds for the front and rear biased vehicles. Please note that the responses for the drift angle and yaw rate at 60 mph for the rear biased 40/60 vehicle could not be calculated due to the responses tending to infinity, as explained in figure 5.

The yaw rate and drift angle for the 60/40 front bias were first calculated for the given input, which are plotted in the following figure:

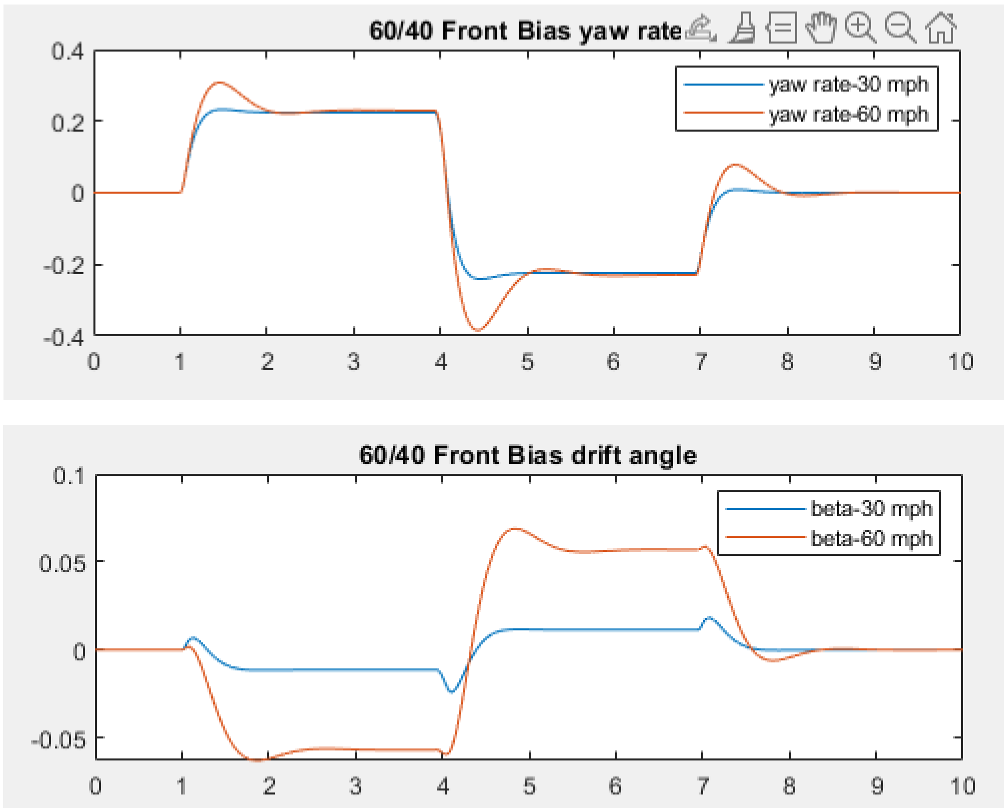


Figure 5: Yaw rate response and drift angle plotted vs time.

The yaw rate and drift angle for the 40/60 rear biased vehicle were also calculated using an identical process and the same code (but with adjusted variables), to provide the following figure. Please note that only 30 mph was calculated, as at 60 mph, the yaw rate tends to infinity and the drift angle tends to negative infinity.

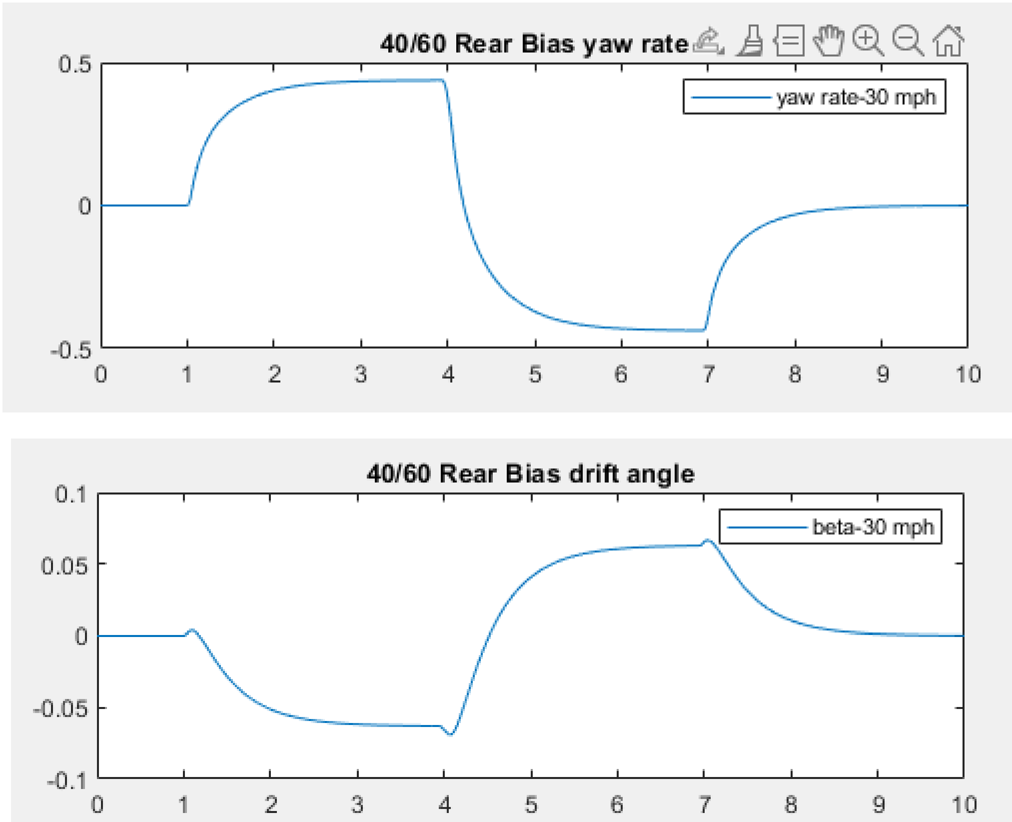


Figure 6: Yaw rate response and drift angle plotted vs time.

These results are then superimposed upon one another and separated into individual test runs for 30 and 60 mph for yaw rate and drift angle, which is as follows:

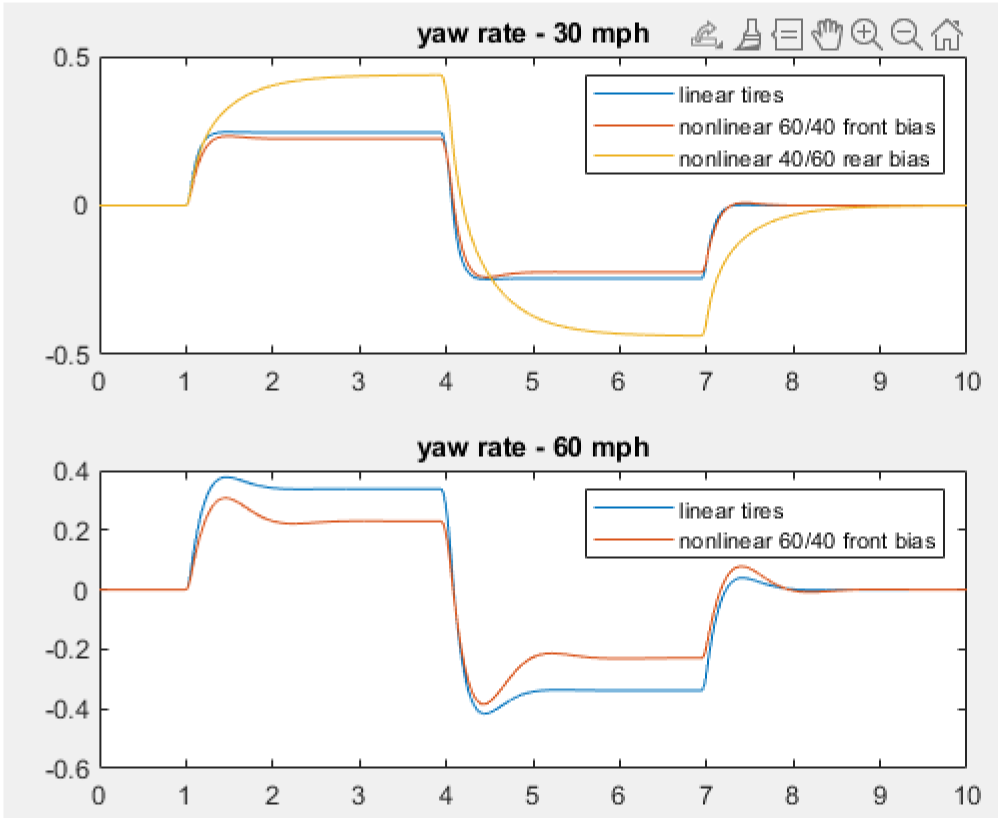


Figure 7: Yaw rate response plotted vs time.

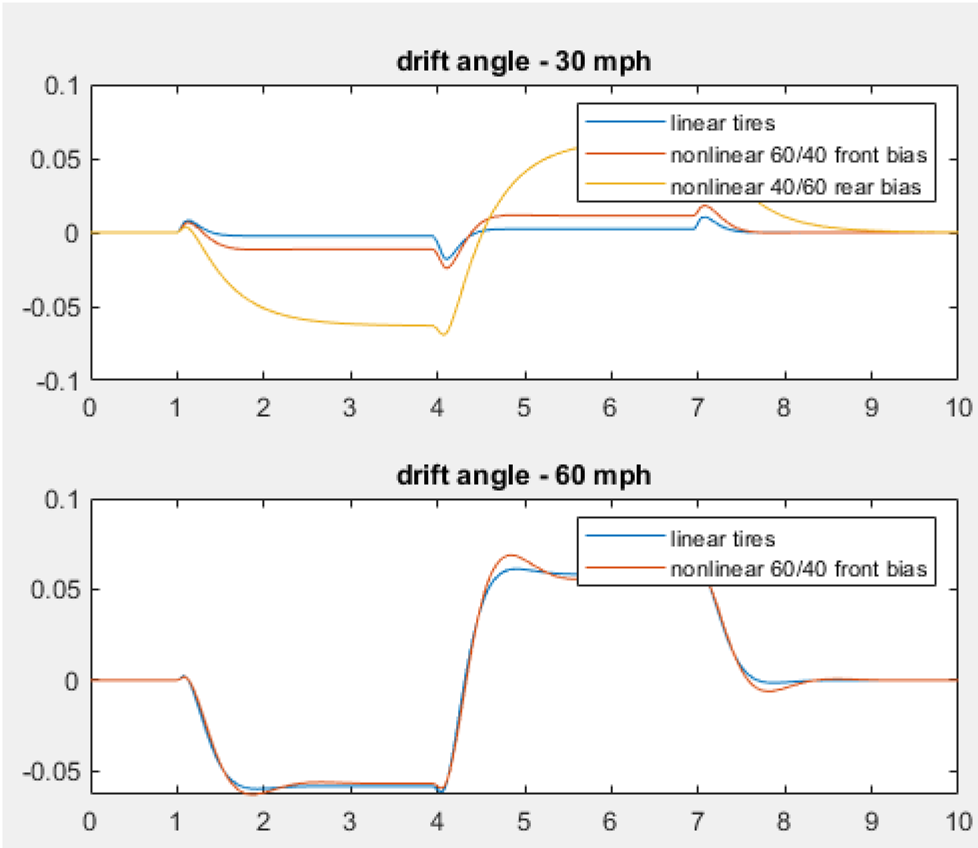


Figure 8: drift angle plotted vs time.

General observations for these two tire configurations compared to the linear model, seen in figure 7 are as follows:

- For the 60/40 front bias, it is noted that the increased understeer as discovered in the previous section causes a much lower yaw rate as compared to the linear tires (where the vehicle is understeering but not as much). As speed increases, so does the yaw response time, as can be seen as a greater and wider peak for the orange line in figure 7.2 when steering input changes. Increasing speed also causes the negative y axis lateral force to increase (which is acceleration, in turn affecting β). This causes a negative β to arise, which increases proportionally with speed which can be seen by comparing the blue and orange lines. Again, increasing speed also increases the drift angle settling time. These conclusions are all in line with those obtained for linear tires.
- For the vehicle with 40/60 rear bias causing oversteer, the variables at 60 mph cannot be calculated, since yaw rate tends to infinity, and the drift angle tends to negative infinity. Since the vehicle is oversteering, it is noted that the yaw rate response is much higher than both other vehicle models tested, even when comparing this model at 30 mph to the other two at 60 mph. This is in line with the understanding of oversteering vehicles having a much higher yaw rate for the same speed and handwheel input (indicating the model is correct!). The drift angle response is again like the two other models, where a negative β is caused by a lateral acceleration in the negative y direction.

Part 2: 3-DoF Vehicle (Linear)

2.1

For this section, the code used in section 1.1 was rewritten with a 3DOF state space model yaw rate response, equation 12.58 from the textbook which is as follows:

$$\frac{r}{\delta_1} = \frac{u}{(x_1 - x_2) + u^2 \frac{\left(-C_b m + \left(C_b\left(C_{\varnothing,1} + C_{\varnothing,2}\right) - C_a\left(x_1 C_{\varnothing,1} + x_2 C_{\varnothing,2}\right)\right)\frac{m_s h}{K_\varnothing}\right)}{C_1 C_2 (x_1 - x_2)}}. \qquad (12.58)$$

The K_{phi} values were also calculated for the vehicle with roll and no roll using the following equation 12.59 from the textbook as follows:

$$K_{2,\varnothing} = \frac{-C_b m + \left(C_b\left(C_{\varnothing,1} + C_{\varnothing,2}\right) - C_a\left(x_1 C_{\varnothing,1} + x_2 C_{\varnothing,2}\right)\right)\frac{m_s h}{K_\varnothing}}{C_1 C_2 (x_1 - x_2)}. \qquad (12.59)$$

This was then used to obtain the yaw rate response iterated through speeds 0-120 mph. The following figure and table were obtained:

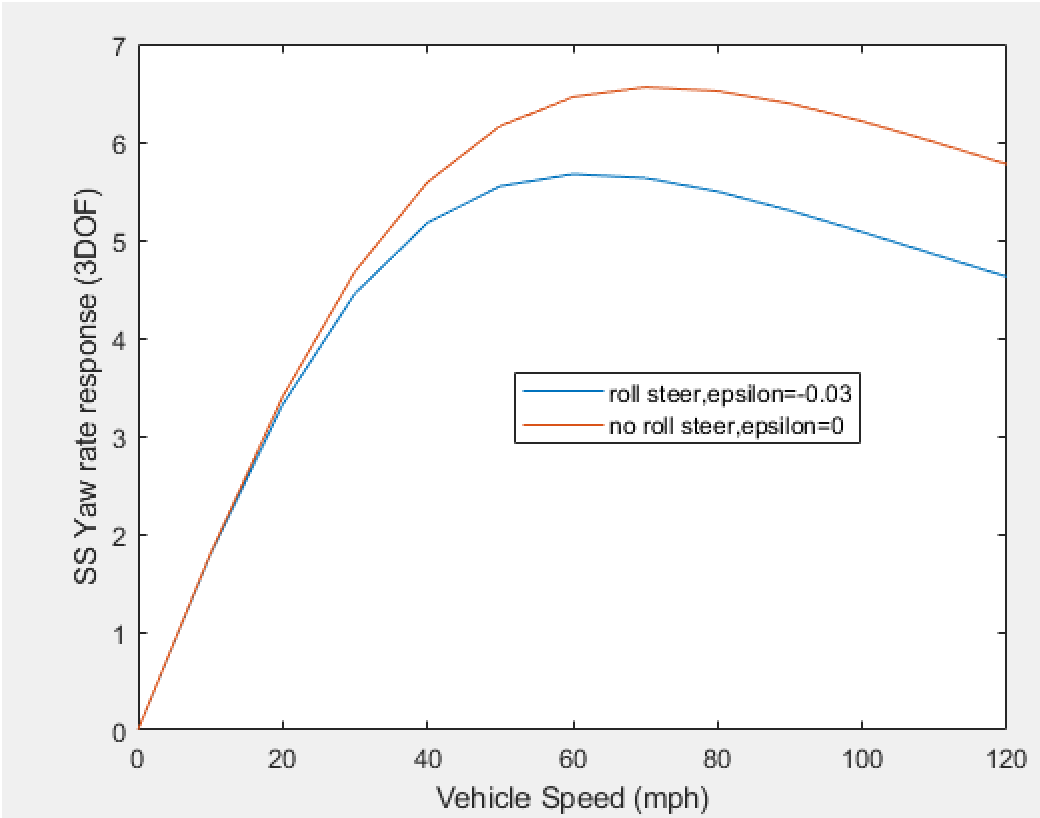


Figure 9: SS Yaw rate response plotted vs vehicle speed.

Speed (mph)	r / δ (ε = -0.03)	r / δ (ε = 0)
0	0	0
10	1.78672410378147	1.79823281677221
20	3.32021646528460	3.40111507634933
30	4.454242876891534	4.67816338915332
40	5.17385565704649	5.58810538485489
50	5.548290931970978	6.16045113355465
60	5.67270258751100	6.46032687957616
70	5.633017051322833	6.55948945647426
80	5.49409932903726	6.52070705387490
90	5.30035377388949	6.39268399162211
100	5.08039727822377	6.21059436607627
110	4.85188935193005	5.99867500491298
120	4.62530410494879	5.77306468893142

Table 4: Yaw rate response calculated for different speeds for two steering coefficients.

A general observation from comparing the model with roll steer and no roll steer is that introducing the negative roll steer introduced further understeer into the system. This is evidenced by the blue line in figure X where the system has roll steer having a lower yaw rate response at all speeds iterated, compared to the orange line where the system has no roll steer, which has a comparatively higher yaw rate at all speeds. I confirmed this by calculating the K_{us} and $u_{transition}$, where the vehicle with roll had values of 0.0009702 (greater than 0, understeer) and 28.7 mph respectively.

2.2

Using the drift angle equation 12.54, as well as the simplified yaw rate equation 12.58, code implementing the two equations was written and iterated from speeds 10 to 120 mph. The code achieving the same has been attached in the appendix for section 2.2

$$\beta = \left(\frac{-C_b}{C_a u} - \frac{mu}{C_a} + (C_{\varnothing,1} + C_{\varnothing,2}) \frac{m_s h u}{C_a K_\varnothing} \right) r + \frac{C_1}{C_a} \delta_1 \qquad (12.54)$$

$$\frac{r}{\delta_1} = \frac{u}{(x_1 - x_2) + u^2 \frac{\left(-C_b m + \left(C_b \left(C_{\varnothing,1} + C_{\varnothing,2} \right) - C_a \left(x_1 C_{\varnothing,1} + x_2 C_{\varnothing,2} \right) \right) \frac{m_s h}{K_\varnothing} \right)}{C_1 C_2 (x_1 - x_2)}}. \qquad (12.58)$$

The response from the same has been summarized in the following table:

Speed (mph)	Required Yaw rate	r / δ	β / δ
10	0.0367	1.787	0.488
20	0.0733	3.32	0.286
30	0.11	4.453	0.006
40	0.1467	5.174	-0.3
50	0.183	5.548	-0.594
60	0.22	5.672	-0.856
70	0.293	5.63	-1.08
80	0.2	5.493	-1.27
90	0.33	5.302	-1.431
100	0.367	5.09	-1.556
110	0.403	4.869	-1.66
120	0.44	4.665	-1.753

Table 5: Yaw rate response and drift angle response calculated for different speeds.

General observations in comparing this data to the previous section 2.1, the yaw rate response at steady state is noted to be near identical between 2.1 and 2.2, confirming that the reworked model is in fact valid. Comparing the data here to section 1.3 with linear tires and no roll effects, a similar observation as in section 2.1 with figure 9 is made, where these additional roll effects to the car cause it to be even more understeery.

Again, like my observation in section 1.6, the sign change in drift angle occurs at between 30 and 40 mph which is markedly different to my calculated transition speed of 28.7 mph. This is admittedly quite close to the transition window I obtained in my code, but however was a large setback that confused me in my analysis of this section. In going through my code and checking for bugs or issues with the formula, I had to attribute this difference to the limitations of experimental accuracy.

2.3 & 2.4

In beginning to tackle this problem, I first divided the steering inputs into the following 9 experimental runs. I use this model naming convention in all the figures in this section, and the following table may be used as a helpful reference in referencing the front and rear roll steer coefficients for each experimental run:

Experiment number	Front roll steer coefficient ε_f	Rear roll steer coefficient ε_r
1	0.04	0.04
2	0.04	0
3	0.04	-0.04
4	0	0.04
5	0	0
6	0	-0.04
7	-0.04	0.04
8	-0.04	0
9	-0.04	-0.04

Table 6: Experimental runs, and their corresponding coefficients.

In defining these 9 experimental runs, I then wrote code that iterated through all 9 runs, following a process like that in sections 1.5 and 1.6. This provided me with the following graphs that compare the yaw rate response and drift angle response of all 9 models for the same given input developed in section 1.4. These figures are as follows:

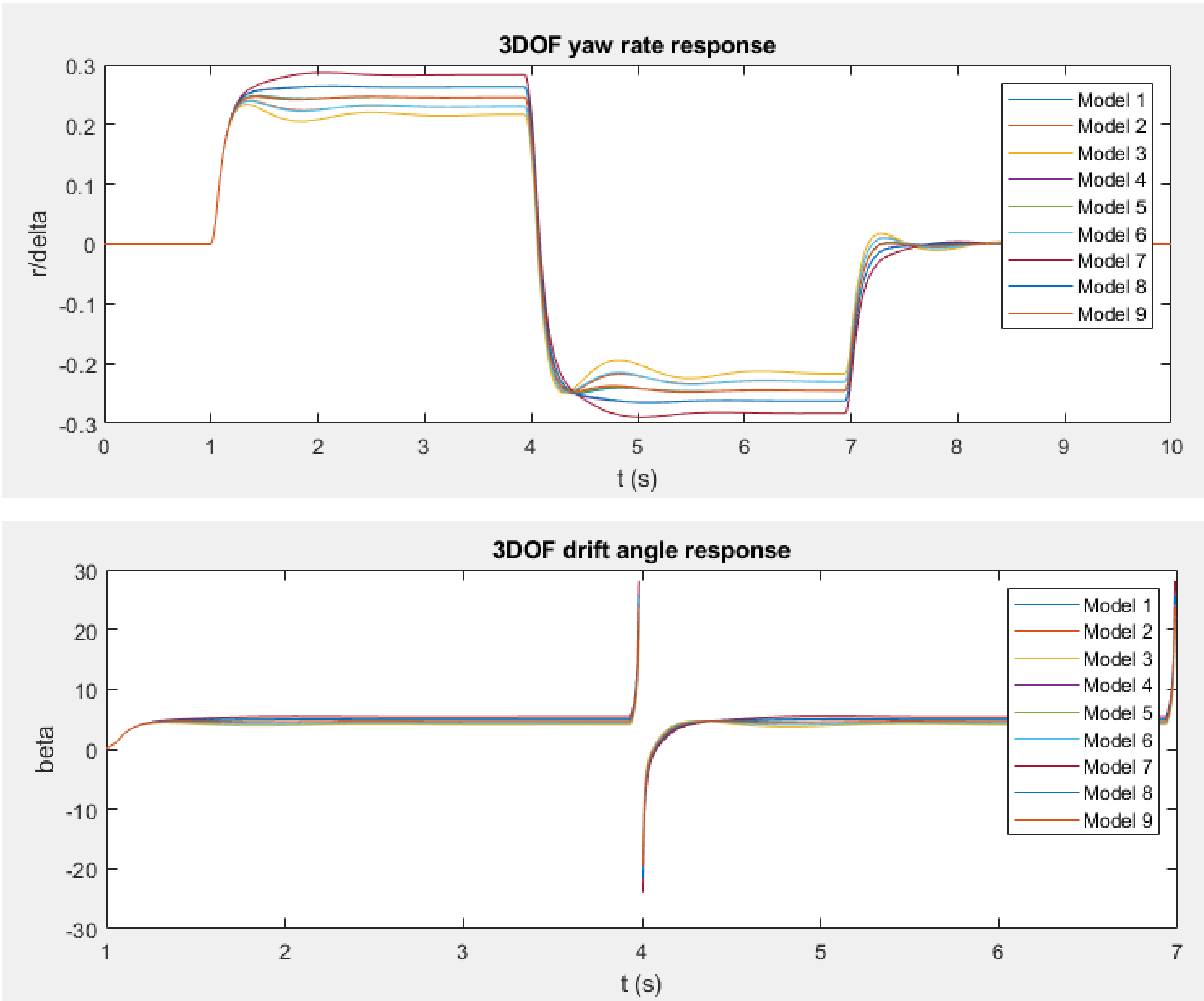


Figure 10: 3 DOF Yaw rate response and drift angle plotted vs time.

In observing the yaw rate response, model 3 with $\varepsilon_f = 0.04$ and $\varepsilon_r = -0.04$ showcases the least yaw rate response, while model 7 with $\varepsilon_f = -0.04$ and $\varepsilon_r = 0.04$ showcases the most yaw rate response. This indicates that the two extreme opposite front and rear steering biases produce the most understeery and oversteery vehicles, indicating the overall effect steering biases have on vehicle responsiveness.

I then proceed to analyze every model for its yaw rate response graph at various speeds, adapting the code from section 1.3 in calculating steady state yaw rate responses and graphing the same. The plots are as follows:

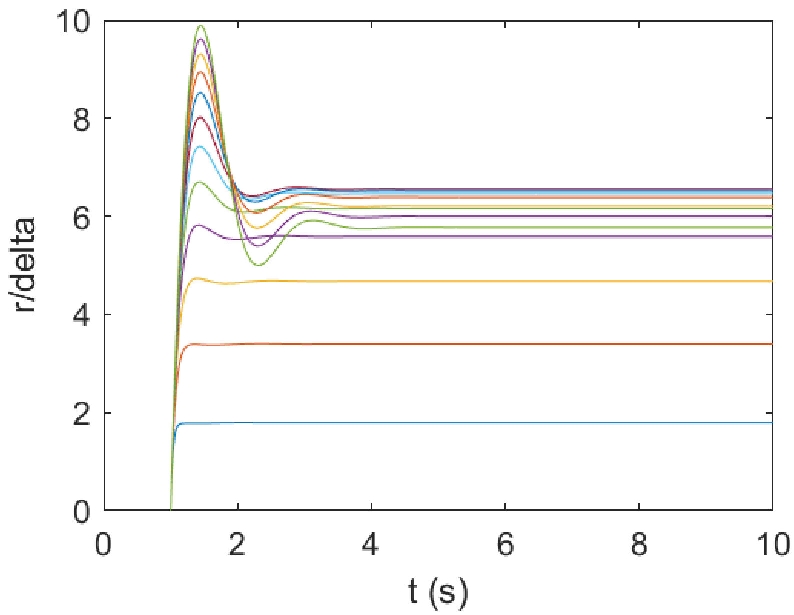


Figure 11: Model 1 Yaw response

Model 1 with ε_f = 0.04 and ε_r = 0.04 has a quick under 4 second response at all speeds and has very little oscillation even at high speed. The large peaks indicate oversteer, but since there is very little oscillation, I expect the vehicle to be quite stable even if slightly oversteery.

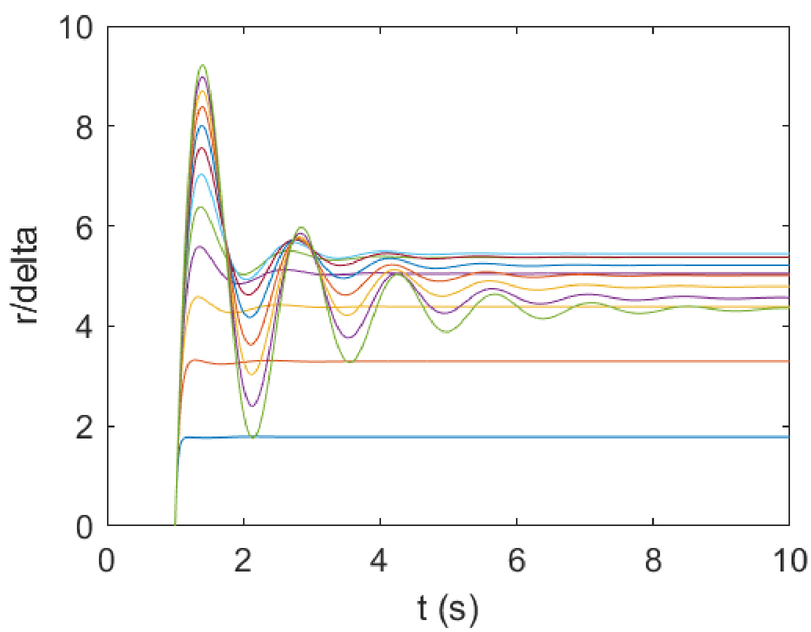


Figure 12: Model 2 Yaw response

Model 2 with ε_f = 0.04 and ε_r = 0 has a similar oversteery peak as in model 1 but oscillates much longer. This indicates that this model is less stable than model 1 despite having similar oversteer characteristics and may be difficult to drive.

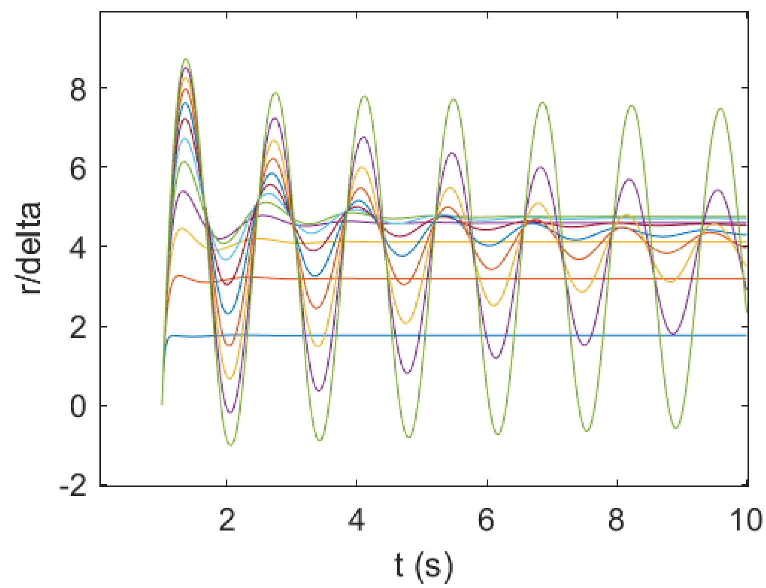


Figure 13: Model 3 Yaw response

Model 3 with ε_f = 0.04 and ε_r = -0.04 showcases no steady state response with oscillations continuing beyond 10 seconds. This model thus appears to be an oversteering vehicle that is fishtailing, with high oversteer and no stability making it the worst handling model. Increasing speed also seems to worsen these negative characteristics.

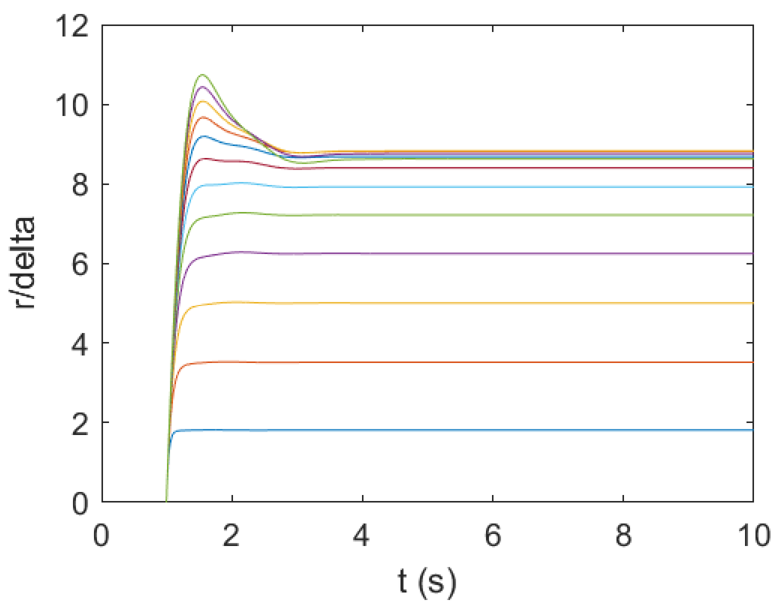


Figure 14: Model 4 Yaw response

Model 4 with $\varepsilon_f = 0$ and $\varepsilon_r = 0.04$ showcases a well to over damped system which remains uniform across the tested speeds. This indicates an understeering response to me, as there are no oscillations, even at high speeds (essentially a more damped version of model 1).

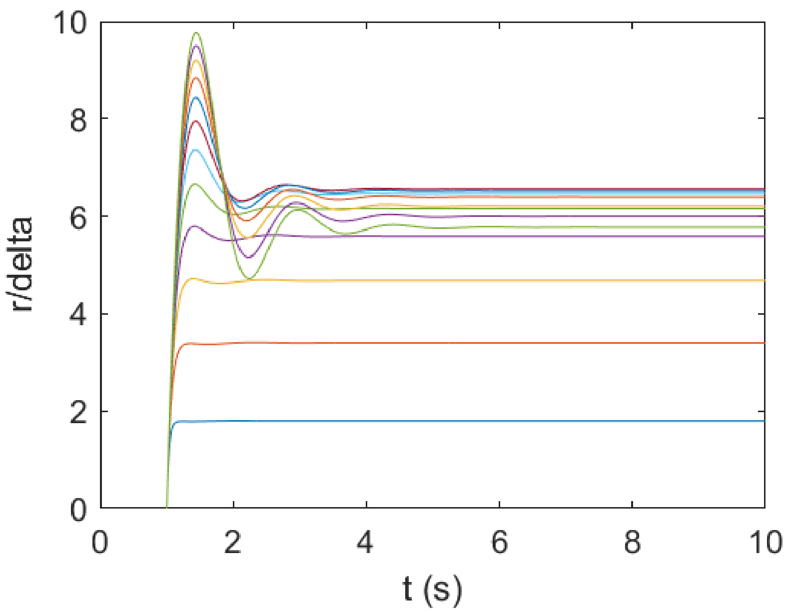


Figure 15: Model 5 Yaw response

Model 5 with $\varepsilon_f = 0$ and $\varepsilon_r = 0$ is the model with no roll steer, identical to part 1.3. This confirms the validity of the model as identical results are found as a previous result.

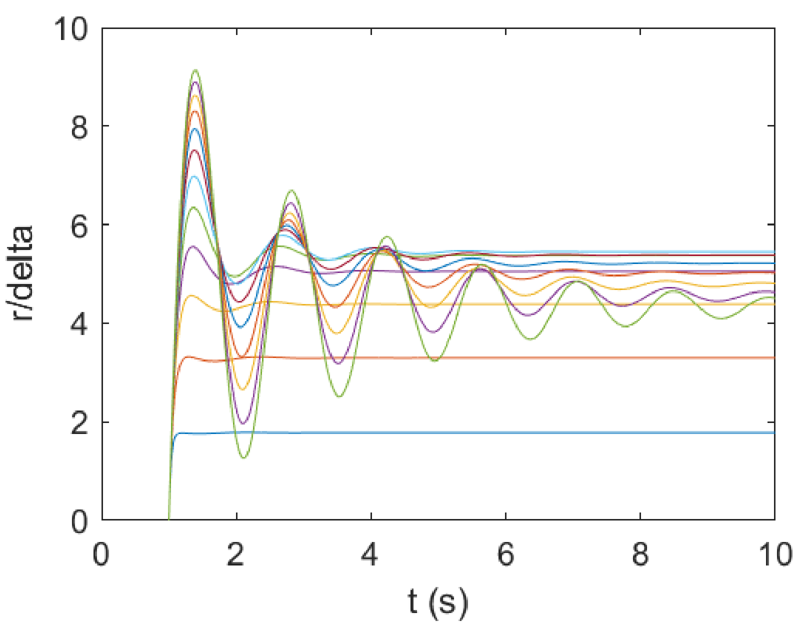


Figure 16: Model 6 Yaw response

Model 6 with $\varepsilon_f = 0$ and $\varepsilon_r = -0.04$ is a slightly better version of model 3, showcasing quite a few of the previous model's weaknesses including a propensity to oversteer and extreme oscillation causing severe instability. Since low speeds do reach steady state and higher speeds slowly approach the same, it is noted to be more stable than model 3.

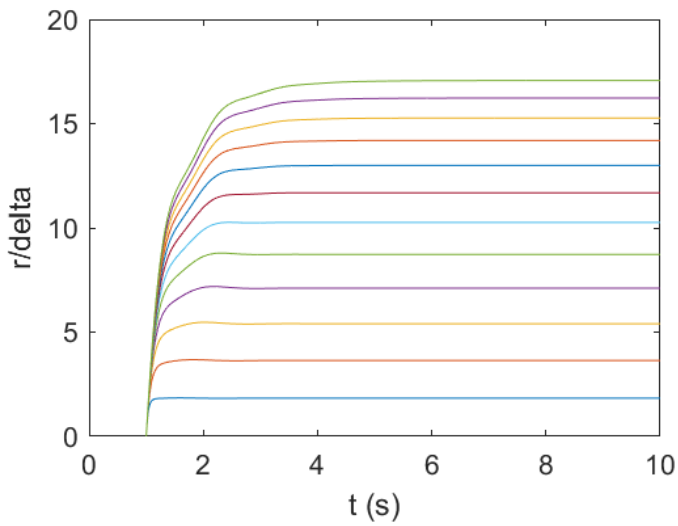


Figure 17: Model 7 Yaw response

Model 7 with $\varepsilon_f = -0.04$ and $\varepsilon_r = 0.04$ has no oversteering overshoot or oscillations, indicating an overdamped system. This handling model will greatly understeer but will be very stable.

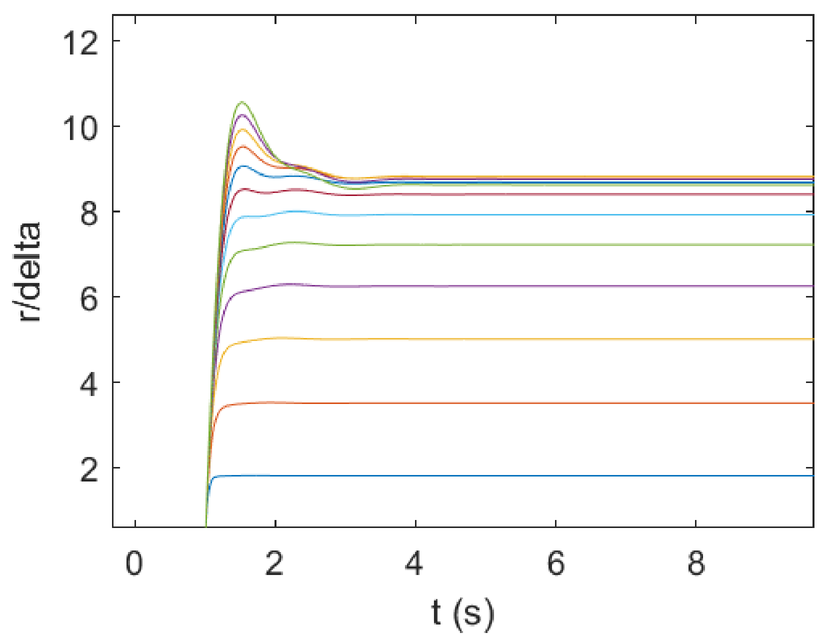


Figure 18: Model 8 Yaw response

Model 8 with ε_f = -0.04 and ε_r = 0 showcases a similarly well damped system as in model 4. This system is again stable but has a propensity to understeer at higher speeds.

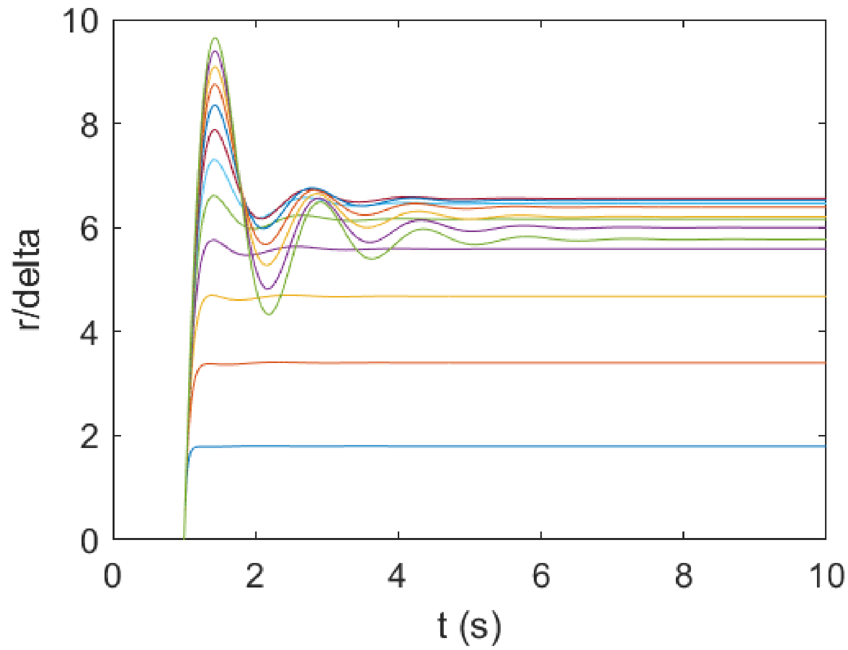


Figure 19: Model 9 Yaw response

Model 9 with ε_f = -0.04 and ε_r = 0.04 has a system that is both understeering especially at higher speeds and is likely to fishtail due to uncontrolled oscillations and oversteer at higher speeds. The system exhibits neutral steer at mid to low speeds, however.

Overall, the general observations noted of positive and negative roll steer coefficients are as follows:

- To increase stability, one can have a negative or zero front roll steer coefficient, or positive coefficients for both front and rear. All these damps the system more and stabilize it, but in turn may introduce understeer.
- To increase responsiveness and oversteer, a negative rear and positive front coefficient may be introduced. This lets the vehicle feel ‘pointier’ and have a more sensitive front end, but this effect cannot be pushed too far such as with both coefficients being negative, as that introduces severe instability and oscillation.

The ideal vehicle will be balanced to be slightly oversteering but having good stability at all speeds.

Part 3: 3-DoF Vehicle (Non-Linear)

3.1

In solving for the tire stiffness step change, I used the following equation:

$$\frac{t}{2}\left(W_{left} - W_{right}\right) = dv * |h| * \frac{m_s}{2}$$

Along with the sum of Wleft and Wright being equal to half the sprung mass. I then used the code for part 3.1 to calculate the following graph varying the speeds in order to calculate the tire stiffness:

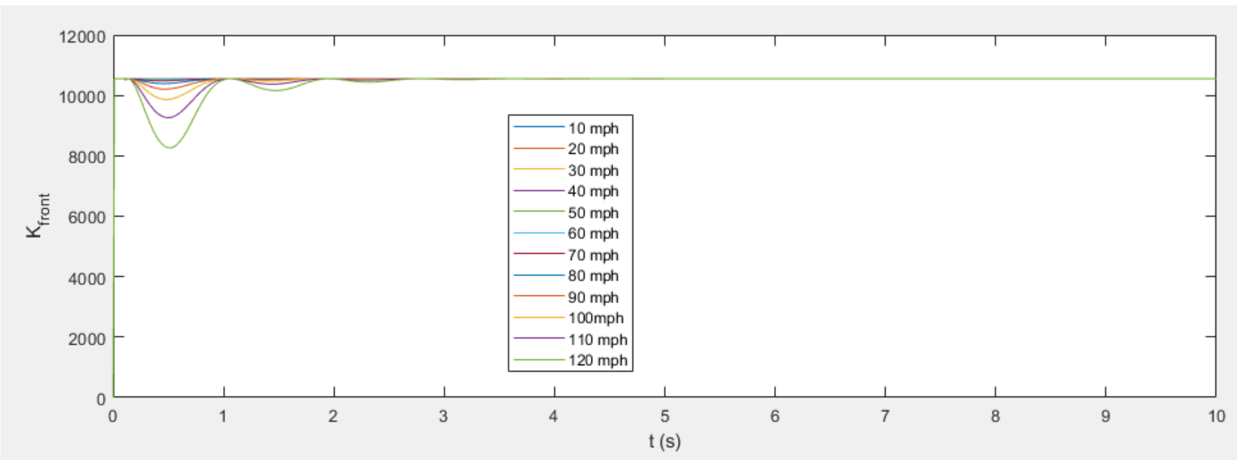


Figure 20: Nonlinear tire constant for the front tires

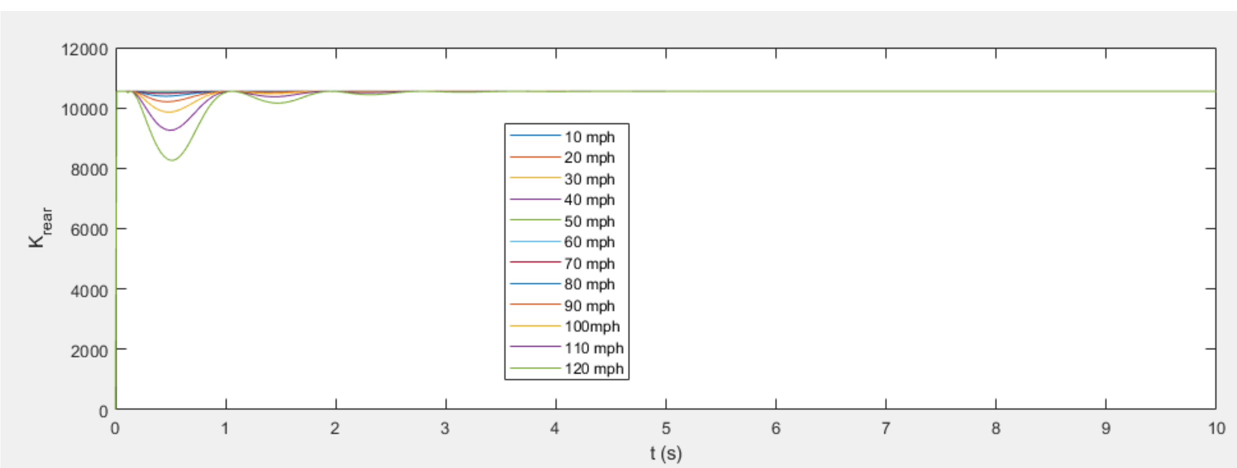


Figure 21: Nonlinear tire constant for the rear tires

Through these graphs, the weight distribution caused by body roll may be witnessed as the oscillations observed. These tend to increase in amplitude and wavelength as the speed increases.

It is however noted to be lower than the linear model, however. I then used the variables calculated in Figures 20 and 21 to obtain the following yaw rate response:

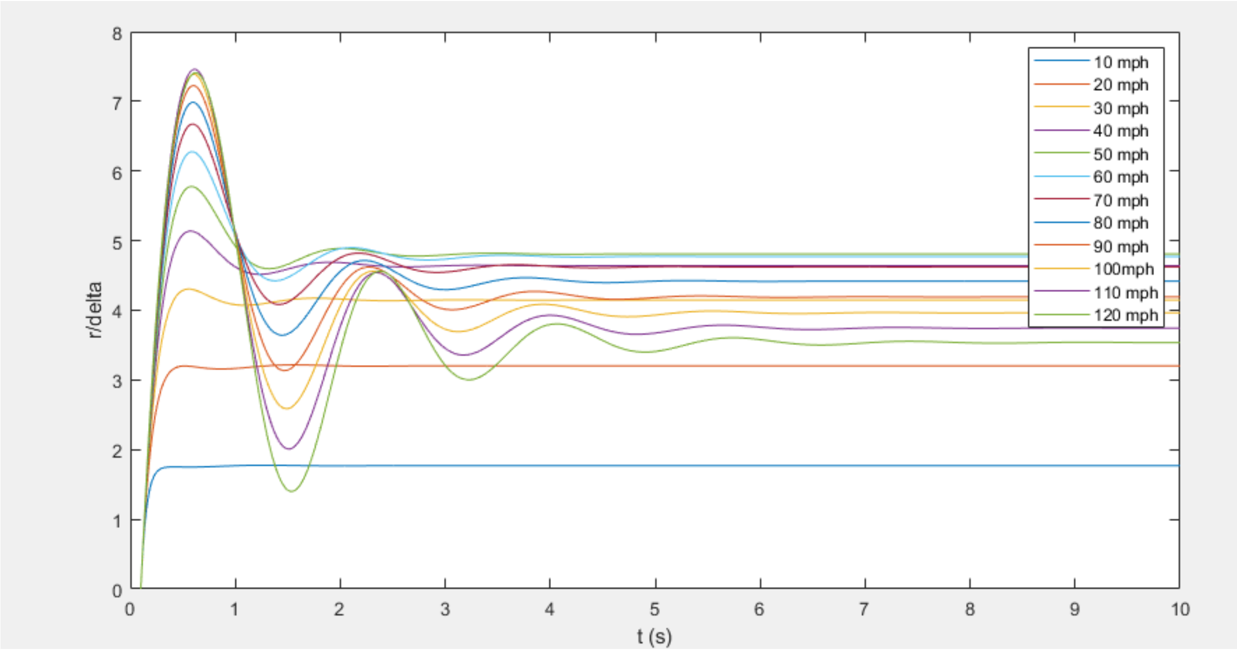


Figure 21: Yaw rate response at various speeds

The yaw response curve also exhibits nonlinear oscillations, caused by the body roll affecting the yaw response and in turn causing these uneven peaks and troughs compared to the model in section 2.

3.2

To achieve this objective, I used the code as given in section 3.2 of the appendix to graph 4 scenarios where Kphi and Dphi were both increased and decreased respectively. The figures of the same are as follows:

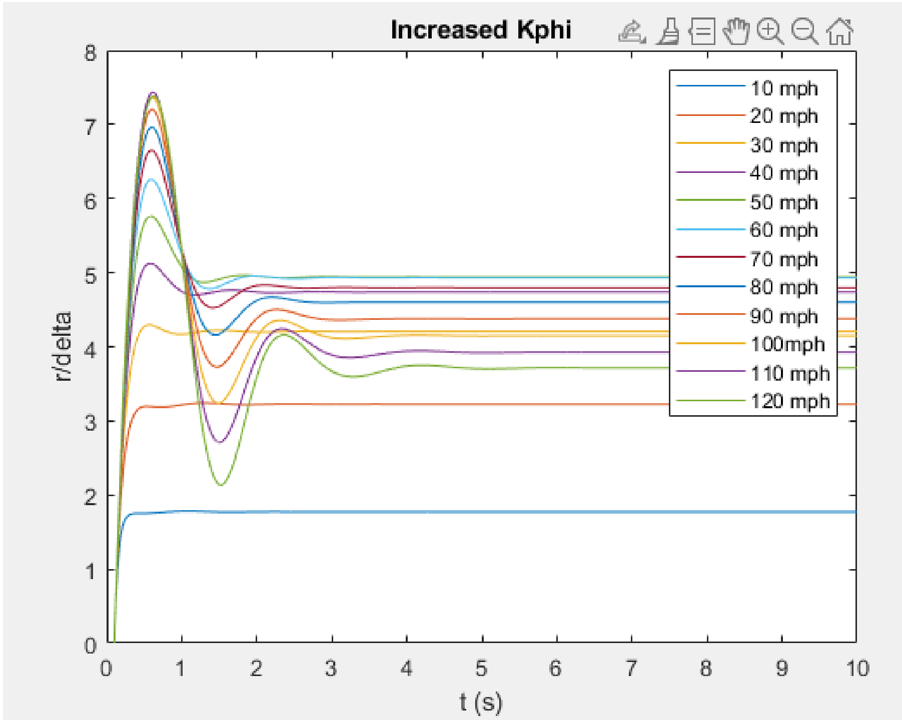


Figure 22: Yaw rate response for varying speed, increasing Kphi

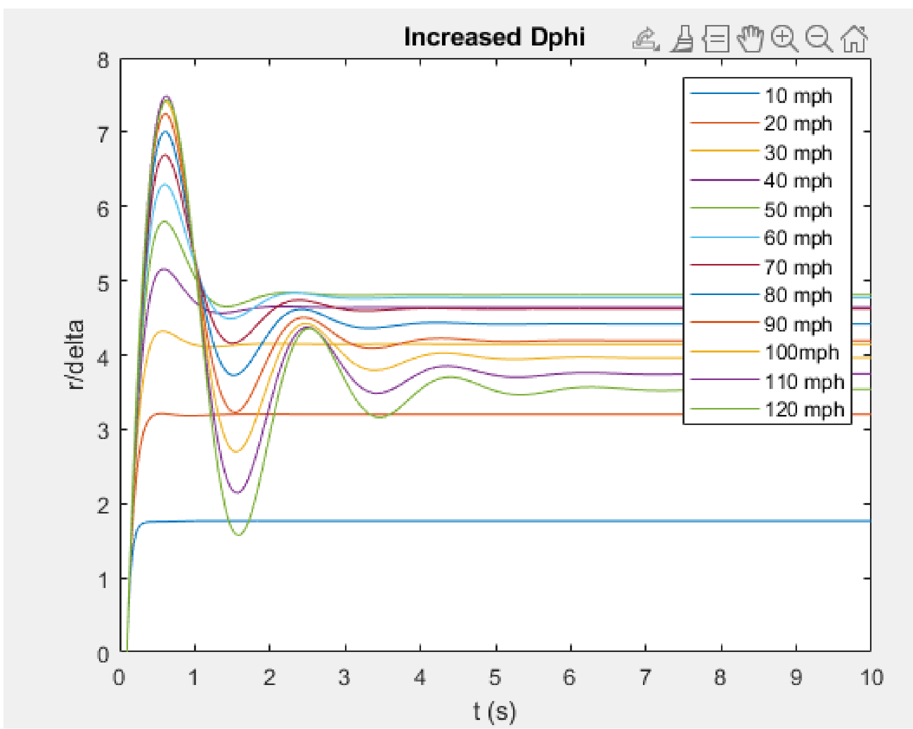


Figure 23: Yaw rate response for varying speed, increasing Dphi

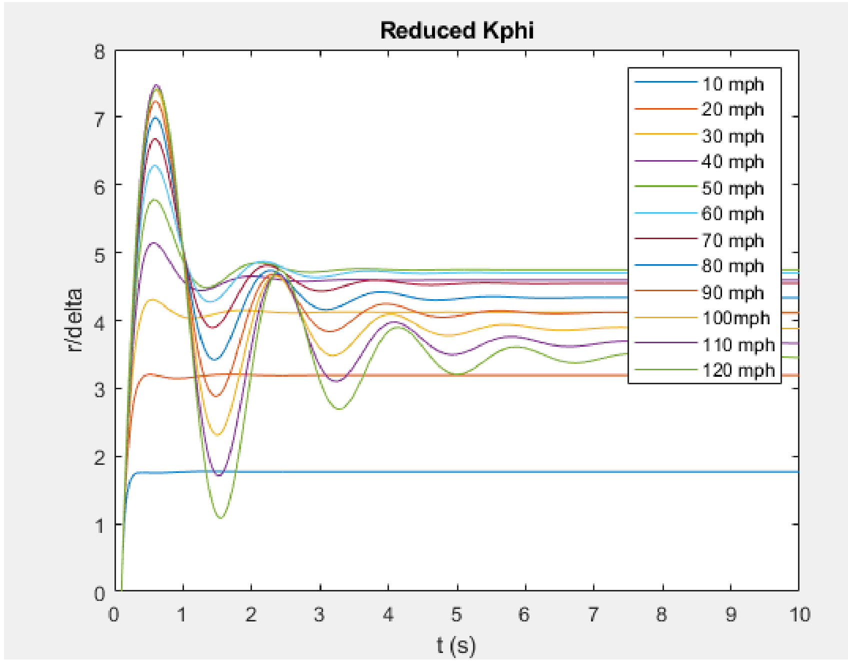


Figure 24: Yaw rate response for varying speed, decreasing Kphi

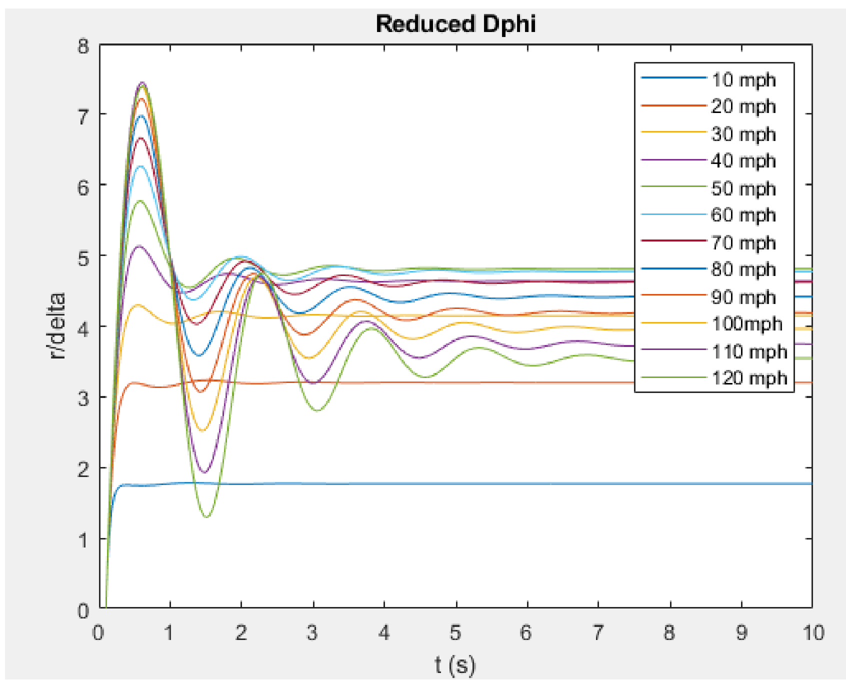


Figure 25: Yaw rate response for varying speed, decreasing Dphi

Increasing Kphi and Dphi increases the vehicle's stability (increased damping caused by Dphi increasing) as well as responsiveness (caused by increased roll stiffness caused by increasing

Kphi). Decreasing both seems to have the opposite effect where the stability and responsiveness are reduced.

Kphi is noted to impact responsiveness (directly proportional), and Dphi is noted to impact stability (directly proportional)

Extra Credit:

I unfortunately was not able to mathematically solve both these extra credit questions, but I did think of how I would go about solving them.

For part 1, I would possibly re-solve the code for section 1.5 with the following line of code amended:

```
deltainput=deltahw*pi/180/n;
```

In this line, n is taken to be 15 to convert handwheel input to steer angles. I would amend it to 1 and directly calculate section 1.5 with the input in radians to possibly get the answer to this section.

For part 2, since I am not sure what precise neutral steer to oversteer balance is preferred as a vehicle may have different dynamics on corner entry, mid-corner and exit, I would use a reference I occasionally use in designing sim-racing tuning setups in Forza, linked below:

https://suspensionsecrets.co.uk/how-to-remove-oversteer-and-understeer/

Ignoring fixed parameters, I would focus on the advice on suspension parameters based on my desired vehicle behavior.

Conclusion

In conclusion, this study employed comprehensive vehicle dynamics simulations, utilizing MATLAB to explore the handling responses of various vehicle models under different steering inputs and configurations. I employed linear and nonlinear tire assumptions for models with 2 and 3 degrees of freedom to analyze the vehicle's steering input, roll steer coefficients as well as tire setups. I obtained the following major conclusions from completing this project:

1. Increasing speed increases yaw rate and drift angle
2. Roll steer greatly affects how a vehicle's stability and steering nature is defined.
3. Changing tire characteristics significantly affect the vehicle's stability and responsiveness.

One way I think this simulation could be improved would be in developing a time based model in Simulink instead of MATLAB, as it would allow easy expandability and the ability to change the type of car while still easily being able to refactor the model as required.

As a whole, this study greatly improved my understanding of the course material from the latter half of this class, and has allowed me to apply the same even in personal hobbies, including simracing.

Appendix:

Code:

```
% Variable declaration

clear;
clc;

W=3000;%total, lbs
m=W/32.2;%conversion to required units

Ws=2700;%sprung mass, lbs
ms=Ws/32.2;%conversion to required units

x1=3.5; %feet
x2=-4.5; %feet
h=-1; %feet
t=6; %feet

Iz=40000;%lbs-ft^2
Ix=15000;%lbs-ft^2

c=0.5;%ft

dldphif=8000;%lbs-ft
dldphir=5000;%lbs-ft
dldphidf=1000;%lbs-ft/sec
dldphidr=500;%lbs-ft/sec

p=12; %inches
d=12; %inches

n=15; %efficiency
e=1; %steering gear box efficiency

Ks=10; %in-lbs/deg
tm=3; %inches

%linear tire coefficient - comment out for part 2
Ci=140*180/pi*2; %converted to lbs/rad

C1=Ci;
C2=Ci;


%Function declaration

function [db,b,dr,r]=yaw(delta,u,C1,C2,m,x1,x2,I,time,dt) %calculates yaw rate per
instance

% State array initialization
b = zeros(1,length(time));
db = zeros(1,length(time));
```

```
r = zeros(1,length(time));
dr = zeros(1,length(time));

% Matrixes
A =[ (-C1-C2)/(m*u) (-x1*C1-x2*C2)/(m*u^2)-1;
 (-C1*x1-C1*x2)/I (-(x1)^2*C1-(x2)^2*C2)/(I*u)];
B =[C1/(m*u);
 x1*C1/I];
for i = 1:1:(length(time)-1)
 xv = A*[b(i);r(i)] + B*delta(i);
 %output with xv
 db(i)= xv(1);
 dr(i)= xv(2);
 %update state variables
 b(i+1) = db(i)*dt + b(i);
 r(i+1) = dr(i)*dt + r(i);
end
end



function
[v,dv,r,dr,phi,dphi,ddphi,Wdiff,Wsum,Cf,Cr]=yawroll(delta1,u,Dphi,Kphi,x1,x2,m,ms)
W=3000;
Ws=2700;
h=-1;
c=0.5;
Iz=40000;%lbs-ft^2
Ix=15000;%lbs-ft^2
t= 10;
dt = 0.01;
time = linspace(0,t,t/dt);
ef=0;
er=-0.03;
% State array initialization
v = zeros(1,length(time));
dv= zeros(1,length(time));
r = zeros(1,length(time));
dr = zeros(1,length(time));
phi = zeros(1,length(time));
dphi = zeros(1,length(time));
ddphi = zeros(1,length(time));
Cf = zeros(1,length(time));
Cr = zeros(1,length(time));
Cphif = zeros(1,length(time));
Cphir = zeros(1,length(time));
Ca = zeros(1,length(time));
Cb = zeros(1,length(time));
Cc = zeros(1,length(time));
Wdiff = zeros(1,length(time));
Wsum = zeros(1,length(time));
Wright = zeros(1,length(time));
Wleft = zeros(1,length(time));
Cf_left = zeros(1,length(time));
Cf_right = zeros(1,length(time));
```

```
for i = 1:1:(length(time)-1)
 % Matrixes
 A1=[m, 0, -ms*h;
 0, Iz/32.2, -ms*h*c;
 -ms*h, -ms*h*c, Ix/32.2];
 A=inv(A1);
 B=[-Ca(i)/u, -Cb(i)/u-m*u, 0, Cphif(i)+Cphir(i), Cf(i), Cr(i);
 -Cb(i)/u, -Cc(i)/u, 0, x1*Cphif(i)+x2*Cphir(i), x1*Cf(i), x2*Cr(i);
 0, ms*h*u, -Dphi, -Kphi, 0, 0];

 xval = A*B*[v(i);r(i);dphi(i);phi(i);delta1(i);0];

 %output with xval
 dv(i)= xval(1);
 dr(i)= xval(2);
 ddphi(i)= xval(3);
 %update state variables
 v(i+1) = dv(i)*dt + v(i);
 r(i+1) = dr(i)*dt + r(i);
 dphi(i+1) = ddphi(i)*dt + dphi(i);
 phi(i+1) = dphi(i)*dt + phi(i);
 %update Cf,Cr,Cphif,Cphir,Ca,Cb,Cc,K2phi
 %front
 Wdiff(i+1)= ms/2*dv(i)*2/t;%Wdiff for one wheel,g=32.2,/2 for one wheel
 Wsum(i+1)=(1/2)*Ws;
 Wleft(i+1)=(Wdiff(i+1)+Wsum(i+1))/2;
 Wright(i+1)=Wsum(i+1)-Wleft(i+1);
 %left
 Cf_left(i+1)=(0.2*(Wleft(i+1))-0.0000942*(Wleft(i+1))^2)*180/pi;
 %right
 Cf_right(i+1)=(0.2*(Wright(i+1))-0.0000942*(Wright(i+1))^2)*180/pi;
 %front total
 Cf(i+1)=Cf_right(i+1)+Cf_left(i+1);
 %rear
 Cr(i+1)=Cf(i+1);
 Cphif(i+1)=Cf(i+1)*ef;
 Cphir(i+1)=Cr(i+1)*er;
 Ca(i+1)=Cf(i+1)+Cr(i+1);
 Cb(i+1)=x1*Cf(i+1)+x2*Cr(i+1);
 Cc(i+1)=x1^2*Cf(i+1)+x2^2*Cr(i+1);
end
end




%----------------------------------------------------------------------

%1.1
u1=30*5280/3600; % 30 mph in ft/s
u2=60*5280/3600; % 60 mph in ft/s

I=Iz/32.2; %unit conversion for lbf

%Eigenvalue code adapted from lecture notes
```

```
A1 =[ (-Ci-Ci)/(m*u1) (-x1*Ci-x2*Ci)/(m*u1^2)-1;
 (-Ci*x1-Ci*x2)/I (-(x1)^2*Ci-(x2)^2*Ci)/(I*u1)];
eigval1 = eig(A1);
A2 =[ (-Ci-Ci)/(m*u2) (-x1*Ci-x2*Ci)/(m*u2^2)-1;
 (-Ci*x1-Ci*x2)/I (-(x1)^2*Ci-(x2)^2*Ci)/(I*u2)];
eigval2 = eig(A2);


%---------------------------------------------------------------------


%1.2
spd=linspace(0,120,13); % generates the speed values to be iterated

u1=spd*5280/3600;%unit conversion from mph to ft/sec

l2=8 ; %wheelbase / ft
K2=(-m*(x1*Ci+x2*Ci))/(Ci*Ci*l2);
utrans=sqrt((-l2)*C2*x2/(m*x1))*3600/5280;
for i=1:13
 ssyaw(i)=u1(i)/((l2)+u1(i)^2*K2);
end
figure(1)
plot(spd,ssyaw);
xlabel('Vehicle Speed (mph)');
ylabel('SS Yaw rate response (r / delta)');




%---------------------------------------------------------------------


%1.3
R=400;%turning radius in feet
l2=8 ; %wheelbase / ft
K2=(-m*(x1*Ci+x2*Ci))/(Ci*Ci*l2);
I=Iz/32.2; %unit conversion for lbf

speed=linspace(10,120,12); % generates the speed values to be iterated
u1=speed*5280/3600;%unit conversion from mph to ft/sec

for i=1:12
 delta(i)=(l2+K2*(u1(i))^2)/R;
end

t= 10; % I assume steady state is achieved within 5 seconds, max time 10s
dt = 0.01; %time step interval
time = linspace(0,t,t/dt); %generates the time grid to iterate

for i = 1:1:length(speed)
 uq3 = u1(i);
 deltaqq3 = delta(i);
 d_step(1:100) = 0;
 d_step(101:1000) = deltaqq3;
```

```
 [dbeta_loop,beta_loop,dr_loop,r_loop] = yaw(d_step,uq3,C1,C2,m,x1,x2,I,time,dt);
 brate(i,:) = beta_loop./d_step;
 rrate(i,:) = r_loop./d_step;
end


%---------------------------------------------------------------------


%1.4
t= 10; % I assume steady state is achieved within 5 seconds, max time 10s
dt = 0.01; %time step interval
time = linspace(0,t,t/dt); %generates the time grid to iterate
deltahw=zeros(1,t/dt); %starts at 0
for i=1:6
 deltahw(100+i)=7.5*i; %changes from 0-45 in 0.0625 s
end
deltahw(107:394)=45; %constant
for i=1:12
 deltahw(394+i)=45-7.5*i; %changes from 45 - -45 in 0.125 s
end
deltahw(407:694)=-45; %constant
for i=1:6
 deltahw(694+i)=-45+7.5*i; %changes from -45 - 0 in 0.0625 s
end

drads=deltahw*pi/180/n; %converts from degrees to radians

figure(3);
subplot(2,1,1);
plot(time,deltahw);
title('Practical change in handwheel orientation(deg)')
subplot(2,1,2);
plot(time,drads)
title('Practical change in handwheel orientation(rad)')


%---------------------------------------------------------------------


%1.5

deltainput=deltahw*pi/180/n;

[dbetaq5_1,betaq5_1,drq5_1,rq5_1]=yaw(deltainput,44,C1,C2,m,x1,x2,I,time,dt);
[dbetaq5_2,betaq5_2,drq5_2,rq5_2]=yaw(deltainput,88,C1,C2,m,x1,x2,I,time,dt);

figure(4);
subplot(2,1,1);
plot(time,rq5_1,time,rq5_2);
legend('yaw rate-30 mph','yaw rate-60 mph');
title('yaw rate (r)')
subplot(2,1,2);
plot(time,betaq5_1,time,betaq5_2);
legend('drift angle-30 mph','drift angle-60 mph');
```

```
title('drift angle (beta)')

%-----------------------------------------------------------------------



%1.6
%calculating the new biases

l2=8 ; %wheelbase / ft
speed=linspace(10,120,12); % generates the speed values to be iterated
u4=speed*5280/3600;%unit conversion from mph to ft/sec
R=400;%turning radius in feet

q6x1 =l2*60/100;
q6x2 =-l2*40/100;
q6c1 =2*(0.2*(4/10*W/2)-0.0000942*(4/10*W/2)^2)*180/pi;
q6c2 =2*(0.2*(6/10*W/2)-0.0000942*(6/10*W/2)^2)*180/pi;

q6K=(-m*(q6x1*q6c1+q6x2*q6c2))/(q6c1*q6c2*l2);


%Steering response (problem 3)

for i=1:12
 deltac2(i)=(l2+q6K*(u4(i))^2)/R;
end

for i = 1:1:length(speed)
 uq3c2 = u4(i);
 deltaqqqq3 = deltac2(i);
 delta_stepc2(1:100) = 0;
 delta_stepc2(101:1000) = deltaqqqq3;
 [dbeta_loopc2,beta_loopc2,dr_loopc2,r_loopc2] =
yaw(delta_stepc2,uq3c2,q6c1,q6c2,m,q6x1,q6x2,I,time,dt);
 betaratec2(i,:) = beta_loopc2./delta_stepc2;
 yawratec2(i,:) = r_loopc2./delta_stepc2;
end

figure(7)
subplot(2,1,1)
plot(time,betaratec2(1,:),time,betaratec2(2,:),time,betaratec2(3,:),time,betaratec2(4
,:),time,
betaratec2(5,:),time,betaratec2(6,:),time,betaratec2(7,:),time,betaratec2(8,:),time,b
etaratec2 (9,:),time,betaratec2(10,:),time,betaratec2(11,:),time,betaratec2( 12,:))
xlabel('time / s')
ylabel('beta/delta')
legend('10 mph','20 mph','30 mph','40 mph','50 mph','60 mph','70 mph','80 mph','90
mph','100mph','110 mph','120 mph')
title('40/60 nonlinear tire bias drift angle response')
subplot(2,1,2)
plot(time,yawratec2(1,:),time,yawratec2(2,:),time,yawratec2(3,:),time,yawratec2(4,:),
time,yawratec2(5,:),time,yawratec2(6,:),time,yawratec2(7,:),time,yawratec2(8,:),time,
yawratec2(9,:),time,yawratec2(10,:),time,yawratec2(11,:),time,yawratec2(12,:))
```

```
xlabel('time(sec)')
ylabel('r/delta')
legend('10 mph','20 mph','30 mph','40 mph','50 mph','60 mph','70 mph','80 mph','90
mph','100mph','110 mph','120 mph')
title('nonlinear 40/60 yaw rate response')


%yaw rate r and drift angle B vs time for steering input 1.4

deltainput=deltahw*pi/180/n;
[dbetaq5_3c2,betaq5_3c2,drq5_3c2,rq5_3c2]=yaw(deltainput,u1,q6c1,q6c2,m,q6x1,q6x2,I,t
ime,dt);
[dbetaq5_4c2,betaq5_4c2,drq5_4c2,rq5_4c2]=yaw(deltainput,u2,q6c1,q6c2,m,q6x1,q6x2,I,t
ime,dt);

figure(8);
subplot(2,1,1);
plot(time,betaq5_3c2,time,betaq5_4c2);
legend('beta-30 mph','beta-60 mph');
title('40/60 Rear Bias drift angle');
subplot(2,1,2);
plot(time,rq5_3c2,time,rq5_4c2);
legend('yaw rate-30 mph','yaw rate-60 mph');
title('40/60 Rear Bias yaw rate');

figure(9)
subplot(2,1,1);
plot(time,betaq5_3c2);
legend('beta-30 mph');
title('40/60 Rear Bias drift angle');
subplot(2,1,2);
plot(time,rq5_3c2);
legend('yaw rate-30 mph');
title('40/60 Rear Bias yaw rate');

figure(10)
subplot(4,1,1);
plot(time,betaq5_1,time,betaq5_1c1,time,betaq5_3c2);
legend('linear tires','nonlinear 60/40 front bias','nonlinear 40/60 rear bias');
title('drift angle - 30 mph');
subplot(4,1,2);
plot(time,betaq5_2,time,betaq5_2c1);
legend('linear tires','nonlinear 60/40 front bias');
title('drift angle - 60 mph');
subplot(4,1,3);
plot(time,rq5_1,time,rq5_1c1,time,rq5_3c2);
legend('linear tires','nonlinear 60/40 front bias','nonlinear 40/60 rear bias');
title('yaw rate - 30 mph');
subplot(4,1,4);
plot(time,rq5_2,time,rq5_2c1);
legend('linear tires','nonlinear 60/40 front bias');
title('yaw rate - 60 mph');
```

```
%----------------------------------------------------------------------


%2.1

%Variable declaration identical to part 1

u1=30*5280/3600; % 30 mph in ft/s - 44
u2=60*5280/3600; % 60 mph in ft/s - 88

Ca=Cf+Cr;
Cb=x1*Cf+x2*Cr;
Cc=x1^2*Cf+x2^2*Cr;

Kphi=(dldphif+dldphir)+ms*32.2*h;
Dphi=(dldphidf+dldphidr);

ef=0;
er=-0.03;

Cphif=Cf*ef;
Cphir=Cr*er;
Cphir2=0;


K2phi=(ms*h/Kphi*(-Ca*(x1*Cphif+x2*Cphir)+Cb*(Cphif+Cphir))-Cb*m)/(l2*Cf*Cr);
K2phie0=(ms*h/Kphi*(-Ca*(x1*Cphif+x2*Cphir2)+Cb*(Cphif+Cphir2))-Cb*m)/(l2*Cf*Cr);

t= 10; %system is simulated for 10 seconds
dt = 0.01; %10 ms step interval
time = linspace(0,t,t/dt); %time iteration

%3DOF SS yaw rate response, same from part 1

speed=linspace(0,120,13);
u1=speed*5280/3600;%converting to ft/sec
for i=1:13
 r(i)=u1(i)/((l2)+u1(i)^2*K2phi);
end
for i=1:13
 r2(i)=u1(i)/((l2)+u1(i)^2*K2phie0);
end
figure(1)
plot(speed,r,speed,r2);
xlabel('Vehicle Speed (mph)');
ylabel('SS Yaw rate response (3DOF)');
legend('roll steer,epsilon=-0.03','no roll steer,epsilon=0')




%----------------------------------------------------------------------

%2.2
%Adapted from 1.3 code
```

```
R=400;%turning radius in feet

speed=linspace(10,120,12); %speeds to iterate through
u1=speed*5280/3600;%this converts the mph speed to ft/s

for i=1:12
 d(i)=(l2+K2phi*(u1(i))^2)/R;
end

for i = 1:1:length(speed)
 uq8 = u1(i);
 dq8 = d(i);
 dstep(1:100) = 0;
 dstep(101:1000) = dq8;
 [v_loop,dv_loop,r_loop,dr_loop,phi_loop,dphi_loop,ddphi_loop] =
yawroll(dstep,uq8,Cf,Cr,Cphif,Cphir,Ca,Cb,Cc,Dphi,Kphi,x1,x2,m,ms);
 betarate(i,:) = v_loop./dstep./uq8;
 yawrate(i,:) = r_loop./dstep;
end


%-----------------------------------------------------------------------
%2.3-4

d9=dhw*pi/180/n;
Cphifp=0.04*Cf;
Cphif0=0;
Cphifn=-0.04*Cf;
Cphirp=0.04*Cr;
cphir0=0;
Cphir_n=-0.04*Cr;
cphif9=[0.04*Cf; 0; -0.04*Cf];
cphir9=[0.04*Cr; 0; -0.04*Cr];
for i=1:3
 for j=1:3
 Cphif=cphif9(i);
 Cphir=cphir9(j);

[v_loop,dv_loop,r_loop,dr_loop,phi_loop,dphi_loop,ddphi_loop]=yawroll(d9,u1,Cf,Cr,Cph
if,Cphir,Ca,Cb,Cc,Dphi,Kphi,x1,x2,m,ms);
 row = 3*(i-1)+j;
 vs_loop(row,:)=v_loop./d9; %drift angle
 rs_loopdel(row,:)=r_loop./d9; %yaw response
 rs_loop(row,:)=r_loop; %yaw
 end
end

figure(3)
subplot(2,1,1)
plot(time, rs_loopdel(1,:),time, rs_loopdel(2,:),time,
rs_loopdel(3,:),time,rs_loopdel(4,:),time, rs_loopdel(5,:),time,
rs_loopdel(6,:),time, rs_loopdel(7,:),time,rs_loopdel(8,:),time, rs_loopdel(9,:));
xlabel('t (s)');
ylabel('beta');
```

```
title('3DOF drift angle response');
legend('Model 1','Model 2','Model 3','Model 4','Model 5','Model 6','Model 7','Model
8','Model 9')
subplot(2,1,2)
plot(time, rs_loop(1,:),time, rs_loop(2,:),time, rs_loop(3,:),time,
rs_loop(4,:),time,rs_loop(5,:),time, rs_loop(6,:),time, rs_loop(7,:),time,
rs_loop(8,:),time, rs_loop(9,:));
xlabel('t (s)');
ylabel('r/delta');
title('3DOF yaw rate response');
legend('Model 1','Model 2','Model 3','Model 4','Model 5','Model 6','Model 7','Model
8','Model 9')
for i=1:3
 for j=1:3
 Cphif=cphif9(i);
 Cphir=cphir9(j);
 for k = 1:12
 uq8 = u8(k);
 deltaq8 = delta(k);
 delta_step(1:100) = 0;
 delta_step(101:1000) = deltaq8;

[v_loop,dv_loop,r_loop,dr_loop,phi_loop,dphi_loop,ddphi_loop]=yawroll(delta_step,uq8,
Cf,Cr,Cphif,Cphir,Ca,Cb,Cc,Dphi,Kphi,x1,x2,m,ms);
 %vs_loop(row,:)=v_loop; %v for one pair of cphi
 rs_loopq9(k,:)=r_loop./delta_step; %r for one pair of cphi
 end
 figure(4);
 subplot(3,3,3*(i-1)+j);
 plot(time,rs_loopq9);
 xlabel('t (s)')
 ylabel('r/delta')
 %legend('10 mph','20 mph','30 mph','40 mph','50 mph','60 mph','70 mph','80
mph','90mph','100 mph','110 mph','120 mph')
 end
end


%-------------------------------------------------------------------


%3.1


for i=1:12
 delta(i)=(l2+K2phi*(u1(i))^2)/R;
end
for i = 1:1:length(speed)
 u11 = u1(i);
 d11 = delta(i);
 ds(1:10) = 0;
 ds(11:1000) = d11;
 [v_loop,dv_loop,r_loop,dr_loop,phi_loop,dphi_loop,ddphi_loop,Wdiff,Wsum,Cf,Cr]
=yawroll(ds,u11,Dphi,Kphi,x1,x2,m,ms);
 brate(i,:) = v_loop./ds./u11;
```

```
 yrate(i,:) = r_loop./ds;
 Cf3(i,:) = Cf;
 Cr3(i,:) = Cr;
end

figure(1)
plot(time,yawrate(1,:),time,yawrate(2,:),time,yawrate(3,:),time,yawrate(4,:),time,yaw
rate(5,:),time,yawrate(6,:),time,yawrate(7,:),time,yawrate(8,:),time,yawrate(9,:),tim
e,yawrate(10,:),time,yawrate(11,:),time,yawrate(12,:))
xlabel('t (s)')
ylabel('r/delta')
legend('10 mph','20 mph','30 mph','40 mph','50 mph','60 mph','70 mph','80 mph','90
mph','100mph','110 mph','120 mph')
figure(2)
subplot(2,1,1)
plot(time,Cfpart3(1,:),time,Cfpart3(2,:),time,Cfpart3(3,:),time,Cfpart3(4,:),time,Cfp
art3(5,:),time,Cfpart3(6,:),time,Cfpart3(7,:),time,Cfpart3(8,:),time,Cfpart3(9,:),tim
e,Cfpart3(10,:),time,Cfpart3(11,:),time,Cfpart3(12,:))
xlabel('t (s)');
ylabel('K_f_r_o_n_t')
legend('10 mph','20 mph','30 mph','40 mph','50 mph','60 mph','70 mph','80 mph','90
mph','100mph','110 mph','120 mph')
subplot(2,1,2)
plot(time,Crpart3(1,:),time,Crpart3(2,:),time,Crpart3(3,:),time,Crpart3(4,:),time,Crp
art3(5,:),time,Crpart3(6,:),time,Crpart3(7,:),time,Crpart3(8,:),time,Crpart3(9,:),tim
e,Crpart3(10,:),time,Crpart3(11,:),time,Crpart3(12,:))
xlabel('t (s)');
ylabel('K_r_e_a_r')
legend('10 mph','20 mph','30 mph','40 mph','50 mph','60 mph','70 mph','80 mph','90
mph','100mph','110 mph','120 mph')




%---------------------------------------------------------------------


%3.2


for i = 1:1:length(speed)
 u12 = u1(i);
 d12 = delta(i);
 ds(1:10) = 0;
 ds(11:1000) = d12;
 [v_loop,dv_loop,r_loop,dr_loop,phi_loop,dphi_loop,ddphi_loop,Wdiff,Wsum,Cf,Cr] =
yawroll(ds,u12,2500,15000,x1,x2,m,ms);
 brate12(i,:) = v_loop./ds./u12;
 yrate12(i,:) = r_loop./ds;
end
for i = 1:1:length(speed)
 u12 = u1(i);
 d12 = delta(i);
 ds(1:10) = 0;
 ds(11:1000) = d12;
```

```
 [v_loop,dv_loop,r_loop,dr_loop,phi_loop,dphi_loop,ddphi_loop,Wdiff,Wsum,Cf,Cr]
=yawroll(ds,u12,1000,9000,x1,x2,m,ms);
 brate12(i,:) = v_loop./ds./u12;
 yrate12(i,:) = r_loop./ds;
end
for i = 1:1:length(speed)
 u12 = u1(i);
 d12 = delta(i);
 ds(1:10) = 0;
 ds(11:1000) = d12;
 [v_loop,dv_loop,r_loop,dr_loop,phi_loop,dphi_loop,ddphi_loop,Wdiff,Wsum,Cf,Cr]
=yawroll(ds,u12,1500,15000,x1,x2,m,ms);
 brate12(i,:) = v_loop./ds./u12;
 yrate12(i,:) = r_loop./ds;
end
figure(1)
plot(time,yrate12(1,:),time,yrate12(2,:),time,yrate12(3,:),time,yrate12(4,:),time,yra
te12(5,:),time,yrate12(6,:),time,yrate12(7,:),time,yrate12(8,:),time,yrate12(9,:),tim
e,yrate12(10,:),time,yrate12(11,:),time,yrate12(12,:))
xlabel('t (s)')
ylabel('r/delta')
title('Increased Kphi')
legend('10 mph','20 mph','30 mph','40 mph','50 mph','60 mph','70 mph','80 mph','90
mph','100mph','110 mph','120 mph')
for i = 1:1:length(speed)
 u12 = u1(i);
 d12 = delta(i);
 ds(1:10) = 0;
 ds(11:1000) = d12;
 [v_loop,dv_loop,r_loop,dr_loop,phi_loop,dphi_loop,ddphi_loop,Wdiff,Wsum,Cf,Cr]
=yawroll(ds,u12,2500,10300,x1,x2,m,ms);
 brate12(i,:) = v_loop./ds./u12;
 yrate12(i,:) = r_loop./ds;
end
figure(2)
plot(time,yrate12(1,:),time,yrate12(2,:),time,yrate12(3,:),time,yrate12(4,:),time,yra
te12(5,:),time,yrate12(6,:),time,yrate12(7,:),time,yrate12(8,:),time,yrate12(9,:),tim
e,yrate12(10,:),time,yrate12(11,:),time,yrate12(12,:))
xlabel('t (s)')
ylabel('r/delta')
title('Increased Dphi')
legend('10 mph','20 mph','30 mph','40 mph','50 mph','60 mph','70 mph','80 mph','90
mph','100mph','110 mph','120 mph')
for i = 1:1:length(speed)
 u12 = u1(i);
 d12 = delta(i);
 ds(1:10) = 0;
 ds(11:1000) = d12;
 [v_loop,dv_loop,r_loop,dr_loop,phi_loop,dphi_loop,ddphi_loop,Wdiff,Wsum,Cf,Cr]
=yawroll(ds,u12,1500,9000,x1,x2,m,ms);
 brate12(i,:) = v_loop./ds./u12;
 yrate12(i,:) = r_loop./ds;
end
figure(3)
```

```
plot(time,yrate12(1,:),time,yrate12(2,:),time,yrate12(3,:),time,yrate12(4,:),time,yra
te12(5,:),time,yrate12(6,:),time,yrate12(7,:),time,yrate12(8,:),time,yrate12(9,:),tim
e,yrate12(10,:),time,yrate12(11,:),time,yrate12(12,:))
xlabel('t (s)')
ylabel('r/delta')
title('Reduced Kphi')
legend('10 mph','20 mph','30 mph','40 mph','50 mph','60 mph','70 mph','80 mph','90
mph','100mph','110 mph','120 mph')
for i = 1:1:length(speed)
 u12 = u1(i);
 d12 = delta(i);
 ds(1:10) = 0;
 ds(11:1000) = d12;
 [v_loop,dv_loop,r_loop,dr_loop,phi_loop,dphi_loop,ddphi_loop,Wdiff,Wsum,Cf,Cr]
=yawroll(ds,u12,1000,10300,x1,x2,m,ms);
 brate12(i,:) = v_loop./ds./u12;
 yrate12(i,:) = r_loop./ds;
end
figure(4)
plot(time,yrate12(1,:),time,yrate12(2,:),time,yrate12(3,:),time,yrate12(4,:),time,yra
te12(5,:),time,yrate12(6,:),time,yrate12(7,:),time,yrate12(8,:),time,yrate12(9,:),tim
e,yrate12(10,:),time,yrate12(11,:),time,yrate12(12,:))
xlabel('t (s)')
ylabel('r/delta')
title('Reduced Dphi')
legend('10 mph','20 mph','30 mph','40 mph','50 mph','60 mph','70 mph','80 mph','90
mph','100mph','110 mph','120 mph')
```