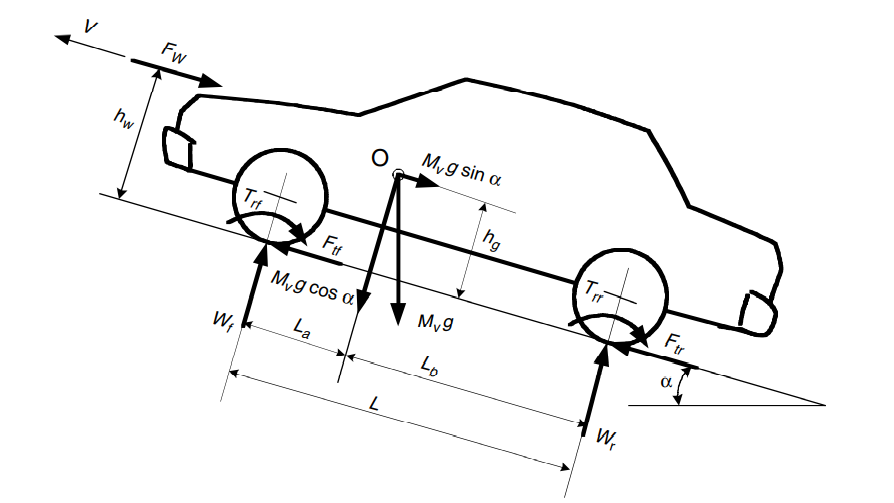
Theoretical Background

For Vehicle Modelling, we have followed the book M.Ehmadi, Y.Gao and A.Emadi [1].



The total traction force required at the drive wheel is comprised of

****

Where,

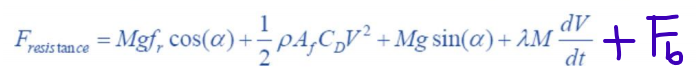
Fr is Rolling Resistance

Fw is Aerodynamic Drag

Fg is Grade Resistance

Fa is Acceleration Resistance

Fb is Braking Force



To propel the vehicle, the Motor should provide traction force equal to the Fresistance . To find the torque required by the wheel, the traction force is multiplied by dynamic radius of the vehicle. The relation between torque required at the wheel and torque provided by the motor is.



Where,

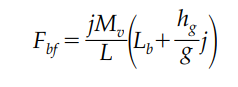
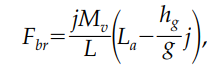
Ig – Gear ratio of the transmission

Io – Gear ration of the final Drive

etat – Efficiency of the driveline from the power plant to the driven wheels

Tp – Torque output from the motor.

The braking force can be broken down in force required on the front wheel and rear wheel.

Where,

Mv- Mass of the vehicle

L- Wheel base of the vehicle

j- Deceleration of the vehicle

hg – Height of the center of gravity of the vehicle to the ground

La and Lb - Horizontal distances between the vehicle gravity center to the center of the front and rear wheels

[1] Modern Electric, Hybrid Electric, and Fuel Cell Vehicles: Fundamentals, Theory, and Design, by [Mehrdad Ehsani](https://www.amazon.in/s/ref=dp_byline_sr_book_1?ie=UTF8&field-author=Mehrdad+Ehsani&search-alias=stripbooks) (Author), [Yimin Gao](https://www.amazon.in/s/ref=dp_byline_sr_book_2?ie=UTF8&field-author=Yimin+Gao&search-alias=stripbooks) (Author), [Ali Emadi](https://www.amazon.in/s/ref=dp_byline_sr_book_3?ie=UTF8&field-author=Ali+Emadi&search-alias=stripbooks) (Author)

%% Output type coloumn Vector, Select Column and import. The speed column and name the varaible as SpeedMph

%%torque cap and speed cap left , cant keep speed cap as it is related to

%%distance of the breake pedal travelled. Only max braking force cap.

%%Convert SpeedMph to m/s

SpeedMPH = 1.6\*SpeedMPH; %%miles perhour to kmph

SpeedMph=SpeedMPH;

%%SpeedMph=0.409\*SpeedMph;

SpeedMph=0.2778\*SpeedMph;

%Vehicle constants

wheel\_radius=0.318; %%Tesla model S https://www.errolstyres.co.za/content/tyre-overall-rolling-diameter

grade\_deg=0; %%maybe grade change

roll\_coef=0.01; %%Car tyre on smooth tarmac road

area=2.4; %%from tesla S

aero\_coeff=0.24; %%from tesla S

inertia\_coeff=0.04; %% assumption

vehicle\_mass=2200; %%from tesla S mass(car+driver)

drive\_eff=0.88; %% from tesla S

wheel\_base=2.959;

la=1.4795;

lb=1.4795;

cg\_height=0.4572; %%http://www.roperld.com/science/TeslaModelS.htm

%Initialization for variables

torque=zeros(length(SpeedMph),1);

roll\_ress=zeros(length(SpeedMph),1);

aero\_ress=zeros(length(SpeedMph),1);

acc\_ress=zeros(length(SpeedMph),1);

grade\_ress=zeros(length(SpeedMph),1);

braking\_force=zeros(length(SpeedMph),2);

for i = 2 : length(SpeedMph)

if SpeedMph(i)-SpeedMph(i-1) >= 0

braking=1;

else

braking=-1;

end

roll\_ress(i)= calc\_rollingres(vehicle\_mass,roll\_coef,grade\_deg);

aero\_ress(i)= calc\_aerodrag(SpeedMph(i),area,aero\_coeff);

acc\_ress(i)=calc\_accress(SpeedMph(i),SpeedMph(i-1),vehicle\_mass,inertia\_coeff);

grade\_ress(i)= calc\_grade(vehicle\_mass,grade\_deg);

if braking ==1

torque(i)=calc\_torque(roll\_ress(i),aero\_ress(i),acc\_ress(i),grade\_ress(i),wheel\_radius,drive\_eff);

else

braking\_force(i,1)=calc\_fbraking(vehicle\_mass,SpeedMph(i),SpeedMph(i-1),wheel\_base,lb,cg\_height);

braking\_force(i,2)=calc\_rbraking(vehicle\_mass,SpeedMph(i),SpeedMph(i-1),wheel\_base,la,cg\_height);

torque(i)=wheel\_radius\*braking\_force(i,1);

end

end

subplot(2,1,1)

plot(SpeedMPH,braking\_force(:,1))

subplot(2,1,2)

plot(SpeedMPH,braking\_force(:,2))

function roll\_ress= calc\_rollingres(mass,roll\_coef, grade\_deg)

roll\_ress= mass\*9.81\*roll\_coef\*cosd(grade\_deg);

end

function aero\_ress= calc\_aerodrag(next\_speed,area,aero\_coeff)

vehiclespeed= next\_speed^2;

aero\_ress = 0.5\*1.27\*area\*aero\_coeff\*vehiclespeed;

end

function acc\_ress = calc\_accress(next\_speed, prev\_speed, mass, inertia\_coeff)

acc = (next\_speed-prev\_speed)/1;

acc\_ress = (mass + mass\*inertia\_coeff)\*acc;

end

function grade\_ress = calc\_grade(mass,grade\_deg)

grade\_ress = mass\*9.81\*sind(grade\_deg);

end

function torque = calc\_torque(roll\_ress,aero\_ress,acc\_ress,grade\_ress,wheel\_radius,drive\_eff)

torque = (wheel\_radius\*(roll\_ress+aero\_ress+acc\_ress+grade\_ress))/9.73;

torque = torque/drive\_eff;

end

function rbrakingf =calc\_rbraking(mass,next\_speed,prev\_speed,wheel\_base,la,cg\_height)

decel=next\_speed-prev\_speed/1;

rbrakingf=(decel\*mass/wheel\_base)\*(la-(cg\_height\*decel/9.81))/9.73;

end

function fbrakingf =calc\_fbraking(mass,next\_speed,prev\_speed,wheel\_base,lb,cg\_height)

decel=next\_speed-prev\_speed/1;

fbrakingf=(decel\*mass/wheel\_base)\*(lb+(cg\_height\*decel/9.81))/9.73;

end