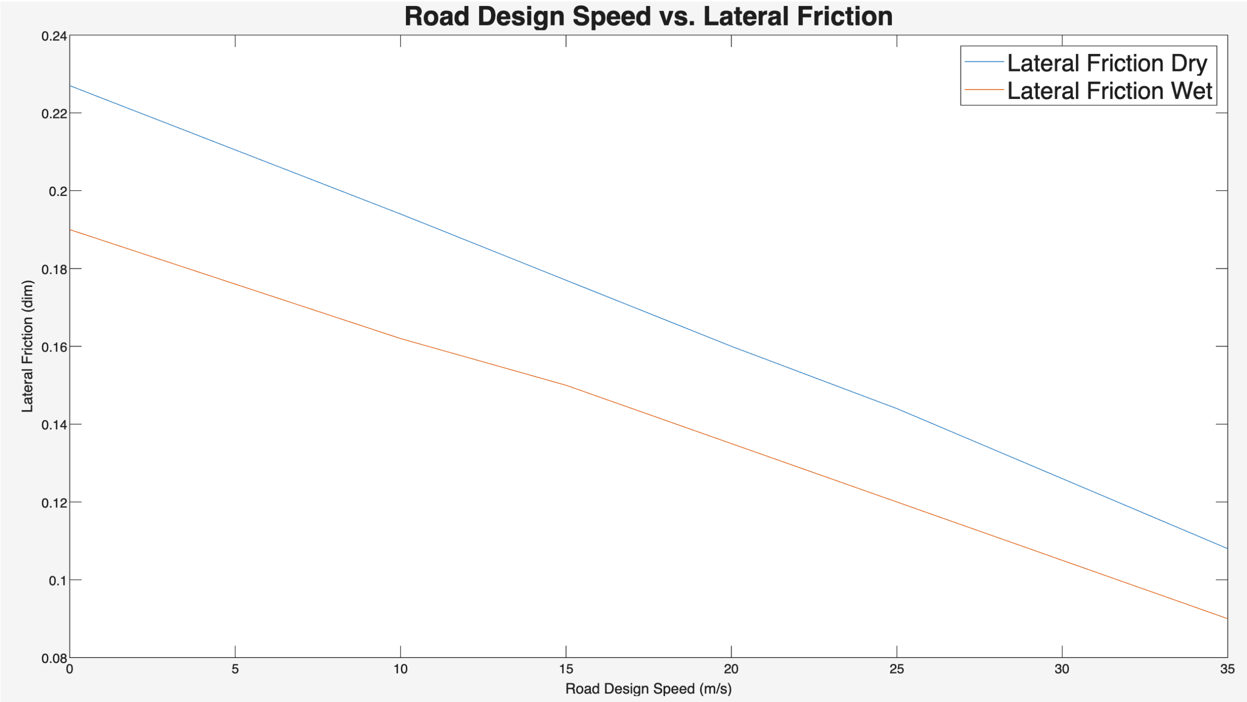
Assignment 7 Solution

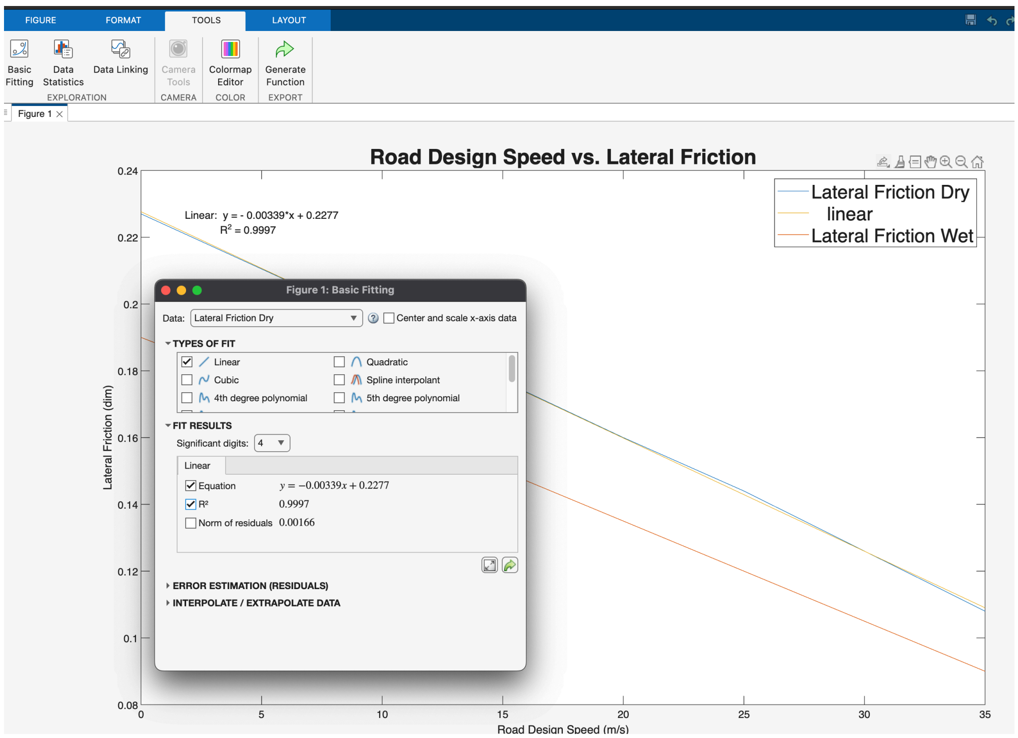
Problem 1 (7 points)

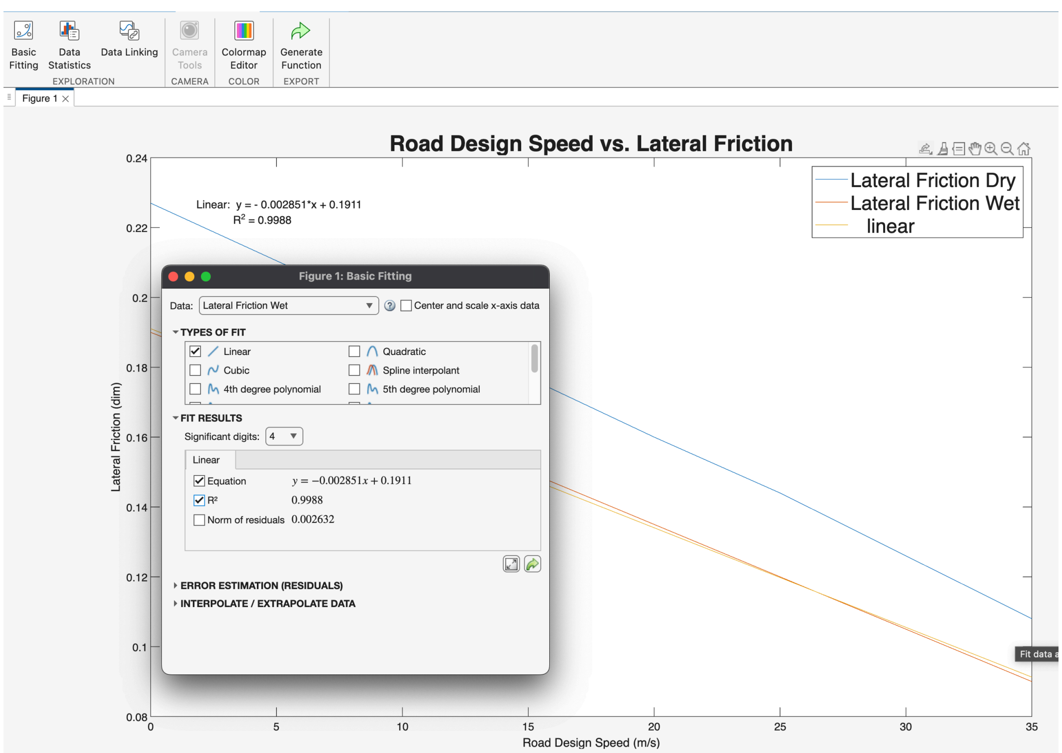
a. Use MATLAB to plot the values of (in y-axis) versus road design speed (x-axis). Plot both dry and wet pavement conditions in the same plot.

```
%% Part a
% Given inputs
RoadDesignSpeed = [0, 10, 15, 20, 25, 35]; % unit m/s
LateralFriction_Wet = [0.190, 0.162, 0.150, 0.135, 0.120, 0.090];
LateralFriction_Dry = [0.227, 0.194, 0.177, 0.160, 0.144, 0.108];

% Plot for dry condition
figure
plot(RoadDesignSpeed, LateralFriction_Dry, 'DisplayName','Lateral Friction Dry');

hold on % to draw both condition in same plot

% Plot for wet condition
plot(RoadDesignSpeed, LateralFriction_Wet, 'DisplayName','Lateral Friction Wet');

xlabel('Road Design Speed (m/s)')
ylabel('Lateral Friction (dim)')
title('Road Design Speed vs. Lateral Friction', FontSize=20)

% Legend
lgd = legend;
lgd.FontSize = 18;
```



b. Estimate the best linear regression model to predict for a given value of road design speed. Obtain equations for both dry and wet pavement conditions. Write down the equation to predict given design speed for both pavement conditions. Use the Basic Fitting functions in MATLAB to accomplish the task.



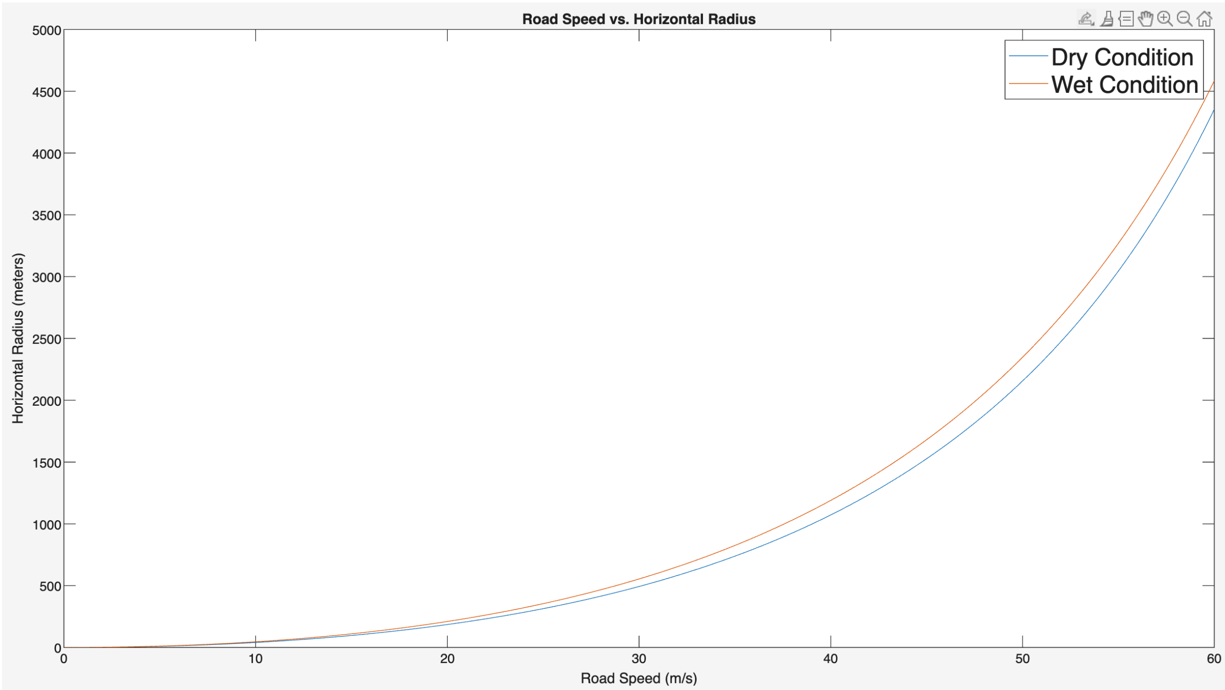


c. Create a MATLAB function to estimate the radius of the curve (R) given the design speed (v), superelevation rate (e), and the pavement condition. Calculate using the equations derived in part (a) of the problem. Your function includes as input, a string variable ('wet' or 'dry') to indicate if the horizontal radius of the road is to be calculated using dry or wet pavement conditions.

```
% Inputs
% Design speed in m/s
% Superelevation rate in %/100
% Pavement condition in string

% Output
% Horizontal radius in meter


function Radius = RadiusCalculation(RoadDesignSpeed, superelevation_rate, pavement_cond)

    g = 9.81; % gravitational constant (9.81 m/s^2)

    if strcmp(pavement_cond, 'Dry')
        f = -0.00339 .* RoadDesignSpeed + 0.2277; % equation got from part a
        Radius = (RoadDesignSpeed.^2) / (g .* (superelevation_rate + f)); % radius calculation

    elseif strcmp(pavement_cond, 'Wet')
        f = -0.002851 .* RoadDesignSpeed + 0.1911; % equation got from part a
        Radius = (RoadDesignSpeed.^2) / (g .* (superelevation_rate + f)); % radius calculation
    end % end if

end % end function (optional)
```

d. Create a separate MATLAB script (not a function) that uses the function created in part (c) to estimate the horizontal radius of the curve for speeds ranging from 0 to 35 meters/second at intervals of 0.1 m/s. Assume a constant value of superelevation rate at 0.06. Your new script created in this task, plots the horizontal radius as a function of road design speed (in the x-axis), for both dry and wet pavement conditions. Show both curves in the same plot to facilitate comparisons. Label the curves accordingly (use the legend).

```
% In case of both
elseif strcmp(pavement_cond, 'Both')

    % Calculates both dry and wet condition
    for index = 1:length(RoadSpeed)
        Calc_Radius_Dry(index, :) = RadiusCalculation(RoadSpeed(index), superelevation_rate, 'Dry');
        Calc_Radius_Wet(index, :) = RadiusCalculation(RoadSpeed(index), superelevation_rate, 'Wet');

    end % end for

    % Part d
    figure
    plot(RoadSpeed, Calc_Radius_Dry, 'DisplayName','Dry Condition');
    hold on
    plot(RoadSpeed, Calc_Radius_Wet, 'DisplayName','Wet Condition');
    title('Road Speed vs. Horizontal Radius')
    xlabel('Road Speed (m/s)')
    ylabel('Horizontal Radius (meters)')
    lgd = legend;
    lgd.FontSize = 18;
```



e. Use the plot created in part (d) to estimate the horizontal radius on the Smart Road in Blacksburg if the design speed of the road was 55 mph.

Radius of Smart Road When 55 mph in Dry and Wet conditions are 302 m and 341 m., respectively.

```
% Part e
RoadSpeedMph = RoadSpeed * 2.237; % Convert the m/s to mph

% Find the index where the closest point is to the 55mph because of the decimal places
[~, idx] = min(abs(RoadSpeedMph - 55));

% Radius for both dry and wet condition when 55mph
Radius_Dry_SmartRd = Calc_Radius_Dry(idx);
Radius_Wet_SmartRd = Calc_Radius_Wet(idx);

disp(['Radius of Smart Road When 55 mph in Dry and Wet conditions are ', num2str(Radius_Dry_Sma
```

```
Command Window
Radius of Smart Road When 55 mph in Dry and Wet conditions are 301.9396m, 340.8832m, respectively.
>> ✦ Press ⇧⌘P to generate code with Copilot
```

Problem 2 (7 points)

a. Create a MATLAB function to estimate the minimum distance between transit unit (Smin), the minimum headway (hmin), and the guideway capacity (Cguideway).

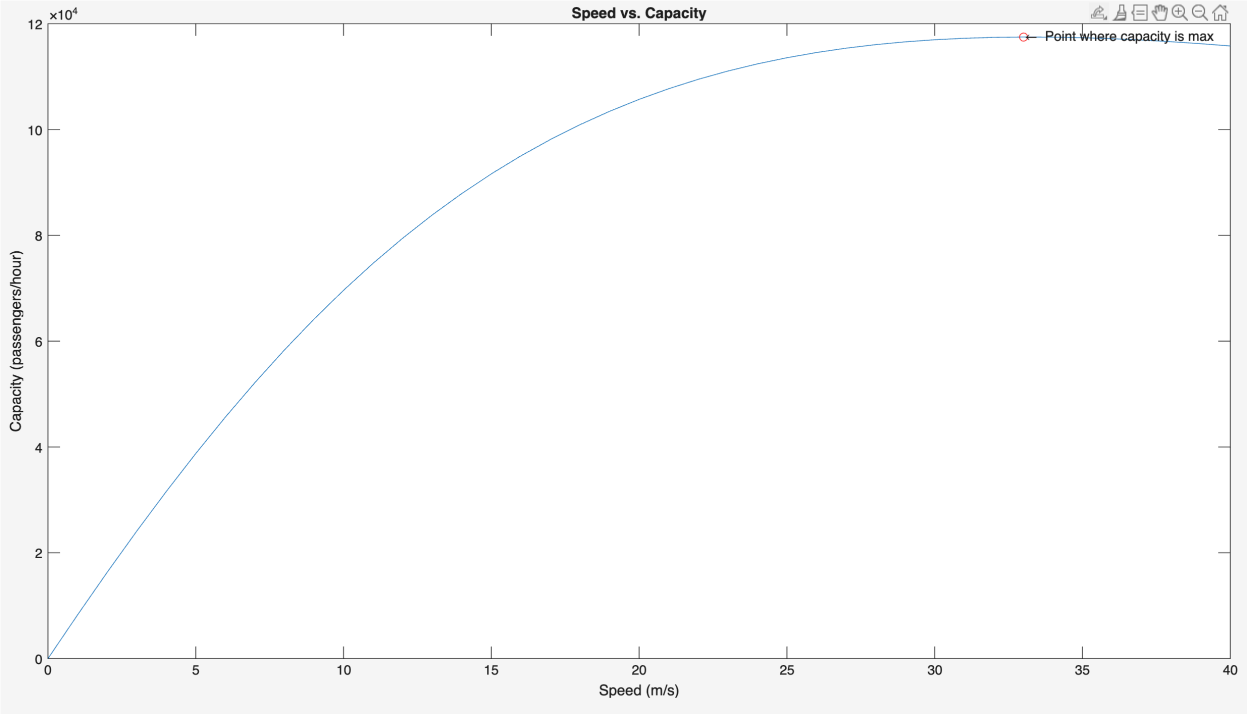
```
function [Smin, hmin, C_guideway] = CapacityCalculation(v, L, N, S0, Cv, b1, b2, tr)

    % Equation (2): Minimum safe distance between transit units
    Smin = (N * L) + S0 + (v .* tr) + (v.^2) / (2 .* b2) - (v.^2) / (2 .* b1);

    % Equation (3): Minimum headway
    hmin = Smin ./ v;

    % Equation (4): Guideway capacity
    C_guideway = (3600 * Cv * N) ./ hmin;

end % function end
```

b. Create another MATLAB script to call the function created in part(a) and evaluate the guideway capacity from zero to 40 meters/second. Use vector operations. No need for a FOR loop.

```
%% Part b

% Define the parameters
v = 0:40; % speed in m/s
L = 22; % length of each car (meters/cars)
N = 8; % number of cars in a transit unit (cars)
S0 = 100; % desired minimum distance between transit units after the both trains stop (meters)
Cv = 80; % capacity of each car (passengers per car)
b1  = 3; % the average deceleration rate of the lead transit unit (m/s^2)
b2 = 1.2; % the average deceleration rate of the following transit unit (m/s^2)
tr = 3; % the reaction time of the transit unit conductor (seconds)

% calculates the capacity
[Smin, hmin, Capacity] = CapacityCalculation(v, L, N, S0, Cv, b1, b2, tr);
```

c. Plot the guideway capacity (y axis) as a function of transit unit line speed with the parameters above.

```
%% Part c and d
figure
plot(v, Capacity)
xlabel('Speed (m/s)')
ylabel('Capacity (passengers/hour)')
```



d. Find the transit line speed that maximizes the value of Cguideway.

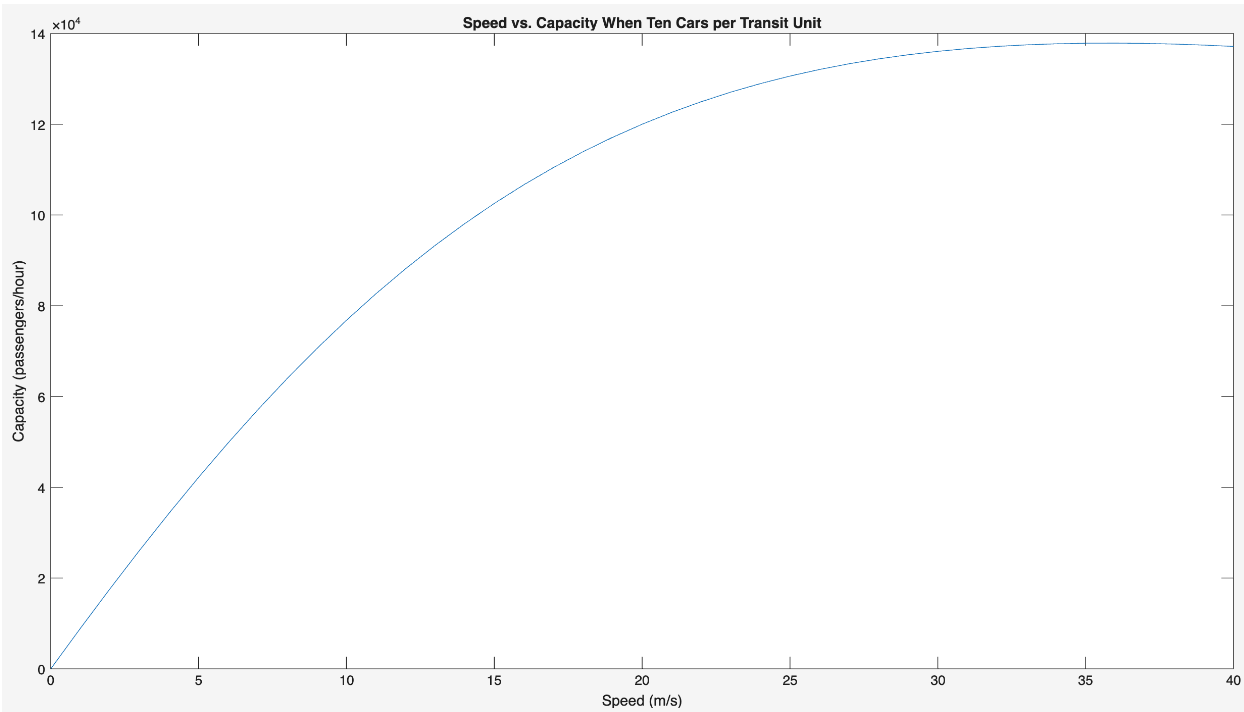
Speed for maximum subway capacity is 33m/s. The theoretical capacity of the subway is estimated to be 117,469 passengers/hour. Note that a typical highway with three lanes per side carries 6,000 vehicles per hours (~7,000 passengers per hour). The subway theoretical capacity is 16 times higher. Subways seldom reach the theoretical capacity because the load factor (number of passengers per vehicle) may only be 65% of the capacity per vehicle. Nevertheless, the subway may carry 45,000 to 75,000 passengers per hour depending upon the configuration.

```
% part d
[Max_Capacity, MaxIndex] = max(Capacity); % find the index where maximum capcaity
max_capacity_speed = v(MaxIndex);
disp(['Speed when the maximum capacity is ', num2str(max_capacity_speed), 'm/s and its capacity is ', n
hold on
plot(v(MaxIndex), Capacity(MaxIndex), 'or')
text(v(MaxIndex), Capacity(MaxIndex),'\leftarrow Point where capacity is max') % optional text
title('Speed vs. Capacity')
```

Command Window

```
Speed when the maximum capacity is 33m/s and its capacity is 117469.2932 passengers/hour
```

e. Plot the guideway capacity (y axis) as a function of transit unit line speed (v) when N is ten cars per transit unit.

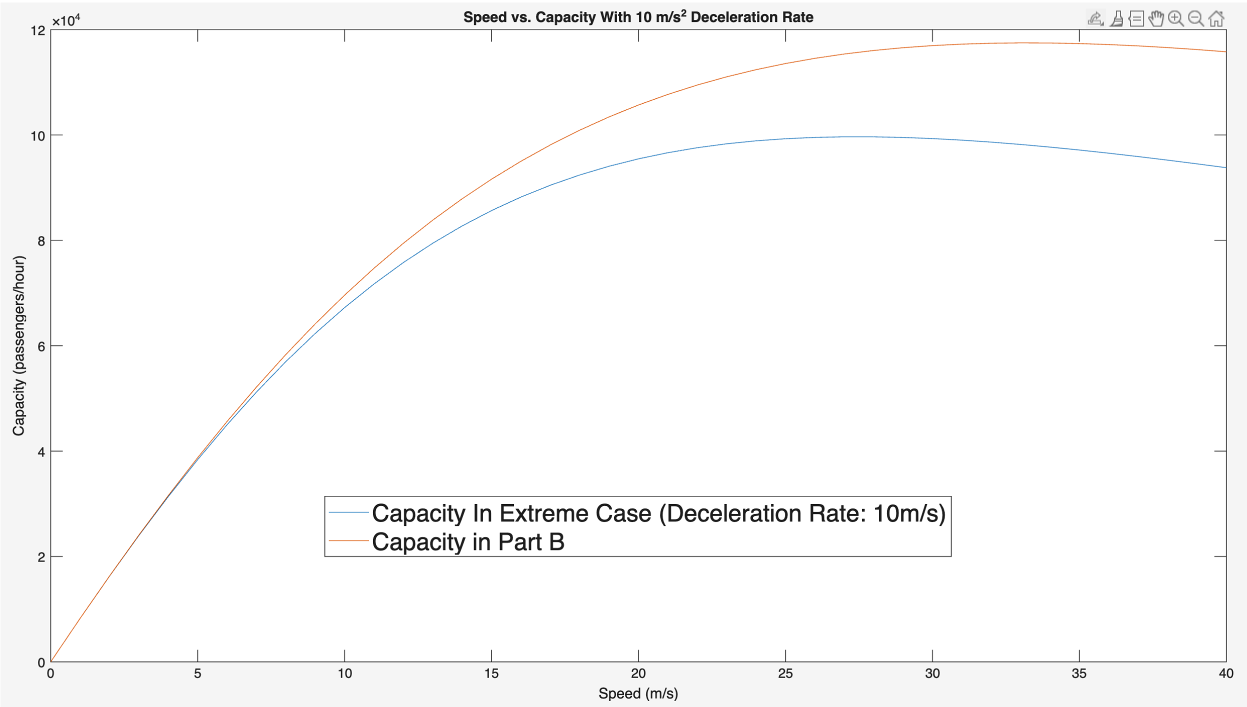



```
%% Part e
% Define the parameters
Ten_N = 10; % 10 cars per unit
[Smin_tenN, hmin_tenN, CapacityWhenTenN] = CapacityCalculation(v, L, Ten_N, S0, Cv, b1, b2, tr);
figure
plot(v, CapacityWhenTenN)
xlabel('Speed (m/s)')
ylabel('Capacity (passengers/hour)')
title('Speed vs. Capacity When Ten Cars per Transit Unit')
```

f. To be conservative in the operations, some transit agencies study the extreme case where the lead transit unit derails and comes to stop at a deceleration rate of 10 m/s2. Find the maximum guideway capacity and compare with part(d).

The maximum capacity when 10m/s deceleration is 99,644passengers/hour. There is a 17,825 passenger/hour difference compared with part (d).

```
%% Part f
% Define the parameters
b1_decel_10  = 10; % deceleration rate is now 10m/s^2
[Smin_decel_10, hmin_decel_10, CapacityWhenDecel_10]  = CapacityCalculation(v, L, N, S0, Cv, b1_decel_1
figure
plot(v, CapacityWhenDecel_10, 'DisplayName','Capacity In Extreme Case (Deceleration Rate: 10m/s)');
xlabel('Speed (m/s)')
ylabel('Capacity (passengers/hour)')
title('Speed vs. Capacity With 10 m/s^2 Deceleration Rate')

% Plot the data in part b (optional)
hold on
plot(v, Capacity, 'DisplayName','Capacity in Part B');

% Legend
lgd = legend('Location','best');
lgd.FontSize = 18;

% find the maximum guideway capacity when deceleration rate is 10m/s
Max_Capacity_When_Decel_10 = max(CapacityWhenDecel_10);
disp(['The maximum capacity when 10m/s deceleration is ', num2str(Max_Capacity_When_Decel_10), ' passen
```




```
The maximum capacity when 10m/s deceleration is 99644.4017 passengers/hour
```

Problem 3 (6 points)

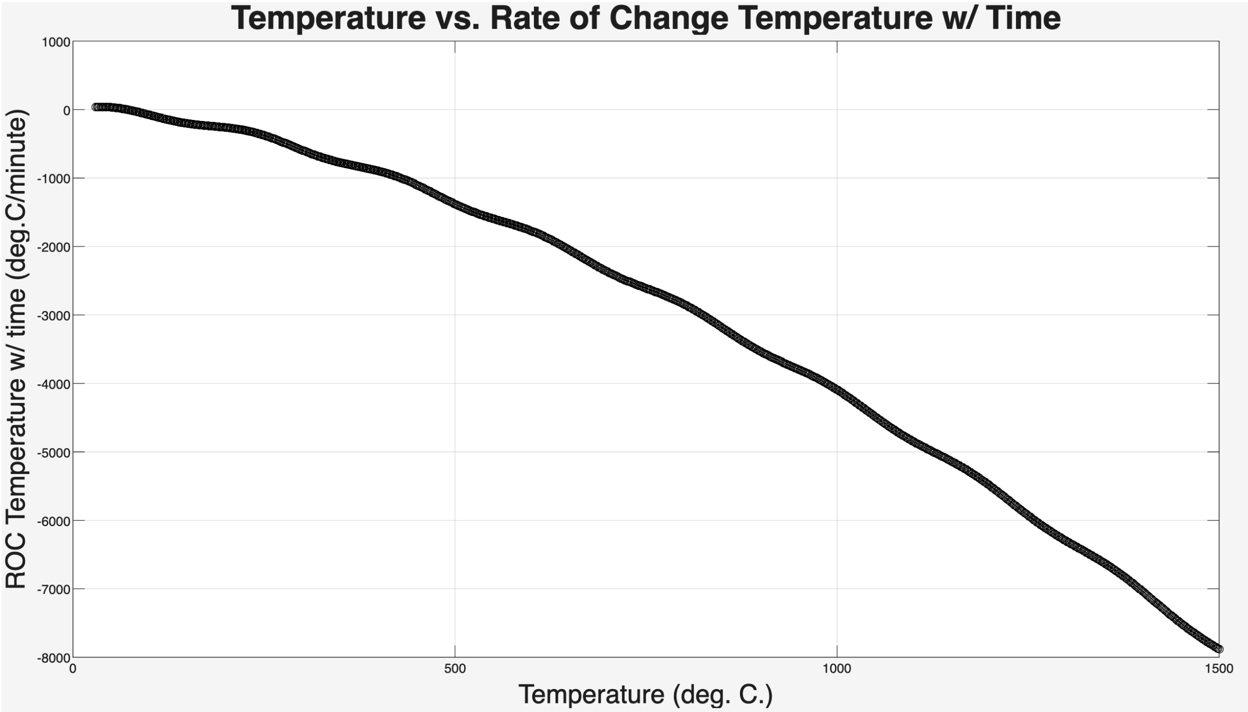
a) Create a function in Matlab to estimate the rate of change of temperature (dT/dt) as a function of time (t).

```
% CEE 3804 Spring 2026 Problem 3 Solution
% Programmer: Jeongwoo Park

%% Problem 3 - Part a
% Function to estimate the value of dT/dt
% Inputs
% K1: A heat release constant (1/minute)
% K2: A second constant (deg.C.)
% T: The temperature of the material (deg. C)
% Ta: The room temperature (deg. C)
% m: An adjustment factor (dim)

% Output: dT/dt: The rate of change of temperature with time (deg. C/minute)

function RateofChangeTemp = RateofChangeTempCalculation(T, Ta, K1, K2, m)
   RateofChangeTemp = -(K1 .* (T-Ta).^m) + (K2 * K1*sin(T/Ta));
```

b. Create a Matlab script that uses the Matlab function created in part (a) to estimate (dT/dt). Use the values of and for steel as 0.097 (1/minute) and 450 (deg. C.), respectively. The value of m is 1.55 and ambient temperature is 30 deg. C. Evaluate the function created in part (a) with 1000 linearly spaced steel temperature (T) values ranging from 1500 deg. C to ambient temperature.

```
%% Part b

% Define the parameters
T=linspace(30,1500,1000);  % 1000 linearly spaced steel temperature (deg. C)
K1 = 0.097; % value of heat release constant (deg.C)
K2 = 450; % second constant (deg.C)
Ta = 30; % room temperature (deg. c)
m = 1.55; % adjustment factor (dim)

% Call the function in part a
RateofChangeTemp = RateofChangeTempCalculation(T, Ta, K1, K2, m);
```

c. Create a plot of (dT/dt) versus Temperature.Label appropriately.




```
%% Part C
plot(T,RateofChangeTemp,'o-k')
xlabel('Temperature (deg. C.)','FontSize',20)
ylabel('ROC Temperature w/ time (deg.C/minute)','FontSize',20)
title('Temperature vs. Rate of Change Temperature w/ Time', 'FontSize', 25)
grid
```