07.04.2019

MIDDLE EAST TECHNICAL UNIVERSITY

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

EE 7566 Homework #1

Vehicle Dynamics and

Electrified Vehicle Powertrains

GÖKHAN ÇAKAL – **2332120**

1.)

These are torque and speed couplers. These are used in order to couple the outputs of two different mechanical drive sources such as electric motor and internal combustion engine. In speed couplers, torque outputs of two drives are proportional to each other with a ratio and speeds of two drives are added as stated below. As an example of speed coupler, we can give planetary gear unit.

Speed coupler

$$\omega_{out} = k_1\omega_{in1}+k_2\omega_{in2}$$

$$T_{out} = \frac{T_{in1}}{k_1} = \frac{T_{in2}}{k_2}$$

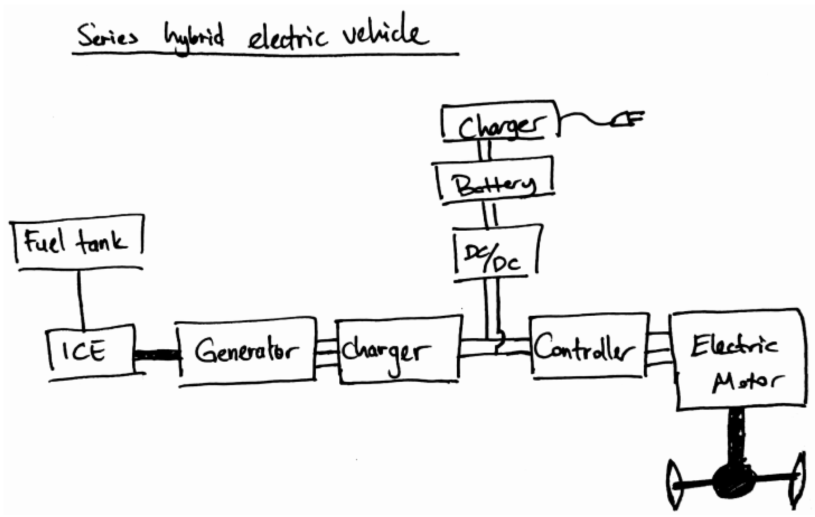
In torque coupler, torques of two drives are added and speeds are related with a ratio as shown below. Front and rear wheels of a car is an example of torque coupler. They have the same speeds but their torques are added.

Torque coupler

$$T_{out} = k_1T_{in1}+k_2T_{in2}$$

$$\omega_{out} = \frac{\omega_{in1}}{k_1} = \frac{\omega_{in2}}{k_2}$$

2.)



Advantages:

- ICE speed is adjusted at its maximum efficient point, independently
- There is no complicated gear box and transmission device
- Easy design

Disadvantages:

- Generator adds additional weight and cost. It decreases overall efficiency.
- ICE, generator and electric motor should be same sized (not valid for range extender)
- Energy from ICE is converted twice, decreasing efficiency.

Most common application of series hybrid configuration is range extenders as we will discuss.

3.)

Electrical continuous variable transmission device (eVT) is fully electromagnetic transmission device. Its structure is similar to conventional electric motors. It consists of two concentric rotors and a stator. The inner rotor is wound rotor and outer rotor contains permanent magnets or it is squirrel cage rotor. Stator is as the same as in conventional electric motor stator. Since there is no direct mechanical contact in and eVT, the efficiency is higher and the system is more reliable. It has inherent overload protection. Also, it can eliminate vibrations. Thus and eVT combines several functionalities into a single useful device, making the system compact.

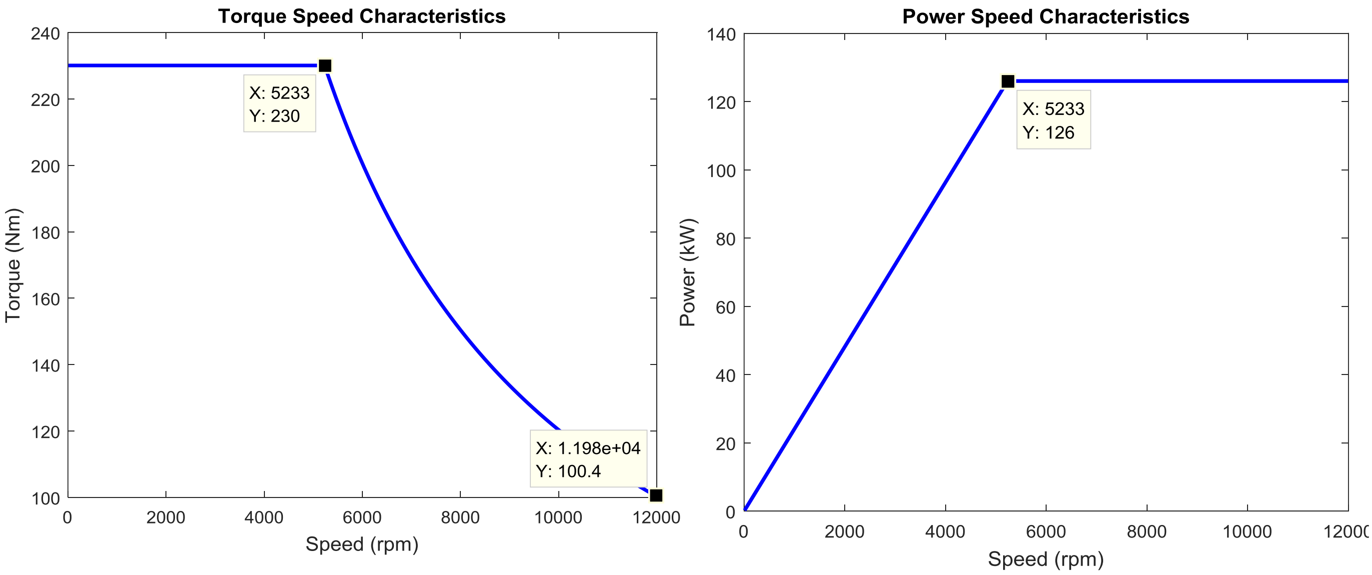
4.)

a)

```
Maximum machine speed is 12000 rpm.
fx >>
```

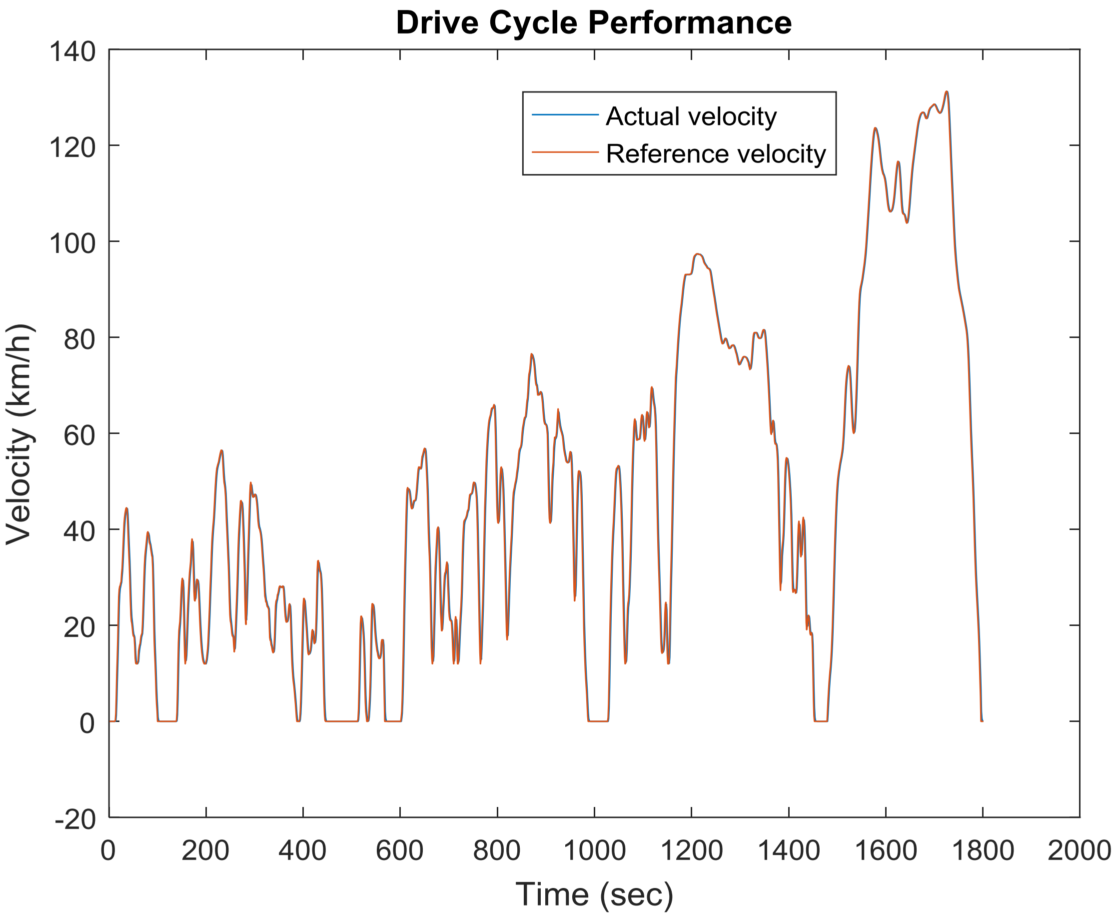
Note: Since the equations are already written in Matlab code at the end, I didn't write them here again.

b)



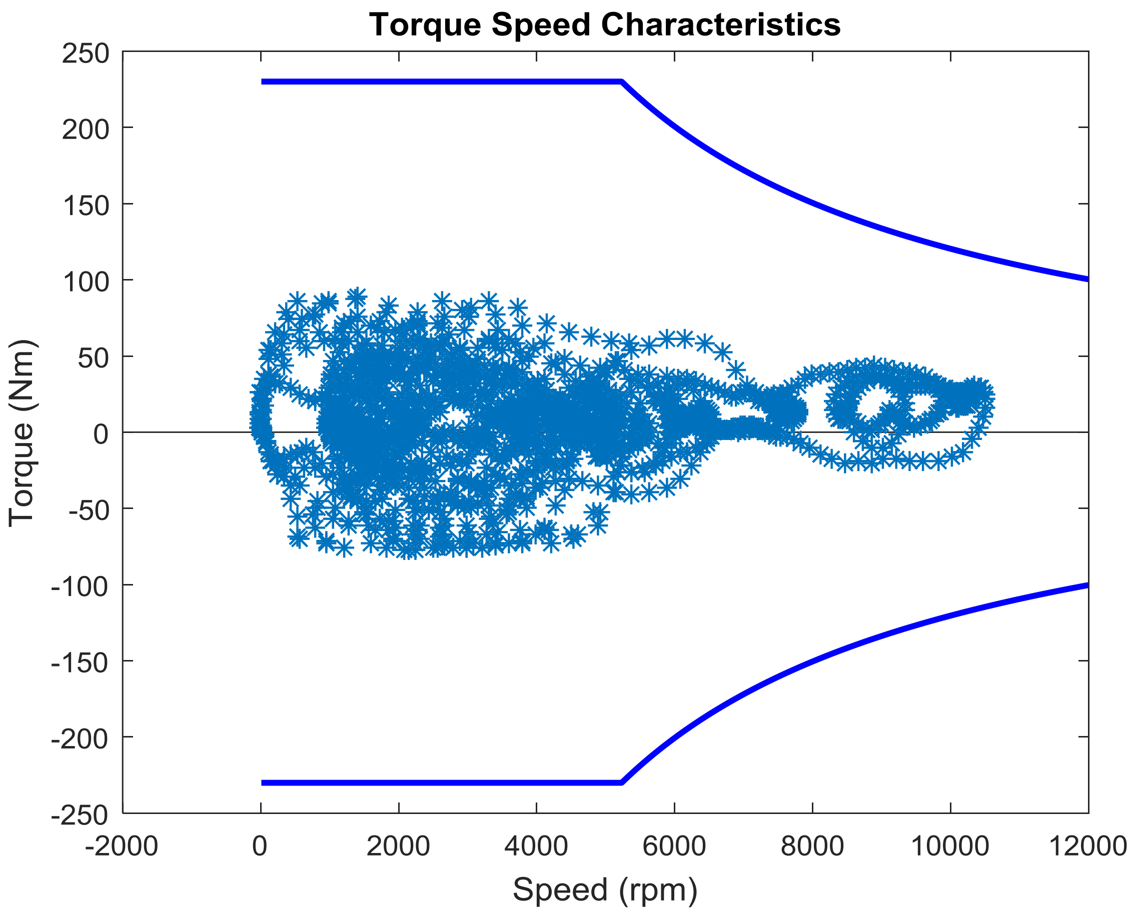
As can be seen above, base speed is 5233 rpm. Below base speed, we are in constant torque region. Above base speed, we are in constant power region. Maximum power and torque is 126 kW and 230 Nm, respectively. Maximum speed is 12000 rpm as found in part a.

c)



Here, we have followed the drive cycle with perfect match except one-two steps of lag. This lag is due to controller schematic and inevitable. Here, I used proportional constant (Kp) of 0.7. As we increase Kp, our controller gives stronger responses against speed changes. More dynamic system can be achieved. Power of the machine should be able to answer the need of faster response. With increasing Kp, our vehicle becomes more dynamic like we are in sport mode. Low Kp values results in smoother responses and power consumption may be decreased like we are in eco mode. Vehicle controllers use different Kp values in different operation modes like eco or sport mode. With increasing Kp, fuel consumption also increases.

d)



e)

```
Fuel consumption is 0.111 kWh/km.
Fuel economy is 0.056 TL/km.
fx >>
```

f)

```
Total regenerated energy is 27.6%.
fx >>
```

Regenerative energy is the energy retrieved during braking. Instead of losses of mechanical brakes, brake energy is used to charge back the battery. Actually, it is the one of the main advantages of electric vehicles over conventional internal combustion engines. As shown above, with regenerative braking, we restore 27.6% of the energy we used for traction in WLTP drive cycle. It is actually important amount of battery capacity. In city driving, since we have to use brakes frequently, regenerative braking provides important amount of saving. However, in highway driving, we rarely use brakes. Thus, we see advantage of regenerative braking mostly in city driving.

g)

		Capacity of battery In kWh	Electric machine power in kW	Electric machine torque in Nm	Mass of drivetrain in kg (incl. battery)	Cost of drivetrain in $	Fuel consumption in kWh	Fuel economy In TL/km	Total Regen. energy in %
BEV with 350 km range	0.05g & 750 W	38.93	126.02	230	460.1	13657	38.93	0.056	27.6
	0.05 g & 0 W	32.82	123.28	226	426.7	12315	32.82	0.047	27.2
BEV with 500 km range	0.05g & 750 W	56.33	133.24	243	554.6	17455	56.33	0.056	28.5
	0.05 g & 0 W	47.52	129.45	238	506.6	15527	47.52	0.048	28.1
BEV REX with 350+150 km range	0.05g & 750 W	39.10	127.86	235	489.2	14563	94.99	0.111	27.9
	0.05 g & 0 W	32.99	126.02	230	456.8	13259	80.15	0.094	27.6

5.)

When accessories are on,

```
ICE power output is 4.27 kW.
Capacity of fuel tank is 5.76 liters.
fx >>
```

When accessories are off,

```
ICE power output is 4.25 kW.
Capacity of fuel tank is 4.86 liters.
fx >>
```

This results make sense since as we open the accessories, fuel consumption will increase and to compensate this we need to increase fuel tank capacity.

a)

When accessories are on,

```
Mixed fuel consumption in WLTP is 4.42 kWh.
Fuel economy is 0.111 TL/km.
fx >>
```

When accessories are off,

```
Mixed fuel consumption in WLTP is 3.73 kWh.
Fuel economy is 0.094 TL/km.
fx >>
```

6.)

a)

Before deciding, let's compare these two. In REX, battery is smaller. Some of its range is supplied from internal combustion engine. Thus, charging times of the BEV will be more painful. Also, due to battery size, BEV is heavier and bulkier. However, since REX contains extra ICE and generator, initial drivetrain cost of BEV is less, which is an advantage. The bottleneck of REX is that its fuel economy is almost twice expensive compared to BEV.

If I was a rich man, I would definitely buy BEV and locate level 3 charger in my home to overcome fast charging times of BEV. Thus, my choice is BEV.

b)

As machine power increases, more energy is delivered in certain amount of time. This means faster acceleration and more dynamic performance. However, more power also means more fuel consumption and worse fuel economy.

c)

$$t_{charging} = \frac{E_{batt}}{eff_{charger} * eff_{batt} * P_{charger}}$$

```
Charging time with level 1 charger is 23.33 hours.
Charging time with level 2 charger is 2.19 hours.
>>
```

Matlab Code

```
clc; clear all; close all;

% This is Homework 1 of EE 7566 Electrical Vehicles course
% Prepared by Gökhan ÇAKAL - 2332120

%% Definitions

range = 350;              %km, required range of EV
accesories_state = 0;     % state of accessories, 1 or 0
rex_mode = 1;             % range extender mode on off, 1 or 0
range_rex = 150;          %km, range extension with range extender

Kp = 0.7;    % Kp constant for acceleration
g = 9.8;                  %m/s2, gravitional acceleration
vel_maxspeed_veh = 150/3.6;  %m/s,  max vehicle speed
vel_wind = 0;               %m/s,  wind speed
acc_maxspeed = 0.05*g;    %m/s2,  acceleration at maximum speed
gear_ratio = 9.0478;      % gear ratio of electric motor to wheels
r_wheel = 0.3;            %m, radius of wheels
d_air = 1.25;             %kg/m3, density of air
area_fr = 2.57;           %m2, frontal area
cd = 0.26;                % aerodynamic drag coefficient
m_body = 1000;            %kg, mass of the body only
m_power_pre = 0;          %kg, predicted mass of the power train
c_mass = 1.05;            % increase in the mass due to rotating masses
fr = 0.006;               % coefficient of rolling resistance
Pow_accesories = 750;     %W, power consumed by accesories
k_adh = 0.9;              % Adhesive coefficient of tires to ground surface
k_load_acc = 0.5;         % load distribution during acceleration
k_load_dec = 0.65;        % load distribution during decelearation
time_0_100 = 7.5;         %sec, 0-100 km/h time

eff_mc_inv = 0.92;       % efficiency of electric machine and inverter
eff_gear_diff = 0.97;     % efficiency of gearbox + differential
eff_batt = 0.95;          % efficiency of battery pack
eff_charger = 0.98;        % efficiency of charger
eff_ice = 0.35;           % efficiency of internal combustion engine

m_spec_mc_inv = 1.1e3;  %W/kg,  specific mass of electric machine and inverter
m_spec_ice = 0.55e3;    %W/kg, specific mass of internal combustion engine
vol_spec_mc_inv = 2.6e3;  %W/l, specific volume of electric machine + inverter
cost_spec_ice = 50;        %$/kW,  specific cost of internal combustion engine
cost_charger = 300;  %$, charger cost
cost_spec_mc_inv = 30;      % $/kW,  specific cost of electric machine and inverter
cost_fueltank = 150;  % $, cost of fuel tank
m_charger = 10;            %kg, charger mass
m_fuel_tank = 5;          %kg, fuel tank mass
m_emine = 70;  %kg, mass of the driver
price_electric = 0.5;      %TL/kwh, electric price
price_gas = 6.25;         %TL/l, gasoline price
dens_gas = 9.7;          % kWh/l, gasoline energy density

delta_mass = 10;  % initiate loop

while delta_mass > 5     % difference in predicted and actual mass is less than 5 kg,
stop the loop
```

```
%% Part 1

circ_wheel = 2 * pi * r_wheel;         %m, circumferential of wheels
n_maxspeed_wheel = 60 * vel_maxspeed_veh / circ_wheel ;  %rpm, max speed of the wheel
n_mc_maxspeed = n_maxspeed_wheel * gear_ratio; %rpm, max speed of the machine
m_total = m_body + m_power_pre + m_emine;      %kg, mass of the vehicle

Fw_maxspeed = 0.5 * d_air * area_fr * cd * (vel_maxspeed_veh+vel_wind)^2;  %N,
aerodynamic drag force at max speed
Fr = fr * m_total * g;      %N, rolling resistance
Fg = 0;                 %N, grading force

Ft_maxspeed = c_mass * m_total * acc_maxspeed + (Fw_maxspeed + Fr + Fg);   %N, traction
force at max speed
Tw_maxspeed = Ft_maxspeed * r_wheel;  %Nm, torque at wheels

T_mc_maxspeed = Tw_maxspeed / gear_ratio / eff_gear_diff;   %N, motor torque at max speed
W_mc_maxspeed = n_mc_maxspeed * pi/30;   %rad/s, motor speed at max speed

Pow_mc_part1 = T_mc_maxspeed * W_mc_maxspeed;   %W, plant traction power
Pow_batt_part1 = Pow_mc_part1 + Pow_accesories;  %W, power capacity of the battery



%% Part 2

Ft_max_acc = k_adh * m_total * g * k_load_acc;   %N, Maximum traction afford of tires
during acceleration
Ft_max_dec = k_adh * m_total * g * k_load_dec;   %N, Maximum traction afford of tires
during decelaration

Tmax_mc_max = Ft_max_acc * r_wheel / gear_ratio / eff_gear_diff;  %Nm, maximum allowed
traction torque by machine before slipping
Tmax_mc_init = Fr * r_wheel / gear_ratio / eff_gear_diff;   %Nm, initial maximum torque
assumption for machine



dT = 0.1;   %sec, time resolution
time_elements = time_0_100 / dT;  %elements, number of time elements determining
resolution
% dT = time_0_100 / time_elements;  %sec, time resolution

time_act = zeros(1,time_elements);  % defining time array

Pow_mc = Pow_mc_part1; %W, initial power assumption of the machine
vel_endtime = 0;  %m/s, start while loop

while vel_endtime < 100/3.6   %m/s, loop until velocity speed reaches 100 km/h

    for Tmax_mc = Tmax_mc_init:1:Tmax_mc_max  %Nm, machine maximum torque finder

        W_mc_base = Pow_mc / Tmax_mc;    %rad/s,    base speed of the machine
        vel_veh_base = W_mc_base / gear_ratio * r_wheel;  %m/s, base vehicle speed

        vel_veh = zeros(1,time_elements);  %m/s, initializing vehicle speed array
        Fw = zeros(1,time_elements);  %N, initializing aerodynamic drag force array
        Ft = zeros(1,time_elements);  %N, initializing traction force array
        T_mc = zeros(1,time_elements);  %Nm, initializing machine torque array
        W_mc = zeros(1,time_elements);  %Nm, initializing machine speed array

        for t = 1:1:time_elements
            time_act(t) = t * dT;   %sec, actual time array
            W_mc(t) = vel_veh(t) / r_wheel * gear_ratio; %rad/s, machine speed
```

```
% determine machine torque
         if vel_veh(t) < vel_veh_base;      % find torque of the machine at velocity of
the vehicle
            T_mc(t) = Tmax_mc;     %Nm, if vehicle is below base speed, maximum torque is
achieved
         else
            T_mc(t) = Pow_mc / W_mc(t);    %Nm, if base speed is exceeded, constant power
region is valid
         end
             % machine torque is found at that specific speed

            Ft(t) = T_mc (t) * gear_ratio * eff_gear_diff / r_wheel;  %N, traction force
            Fw(t) = 0.5 * d_air * area_fr * cd * (vel_veh(t)+vel_wind)^2;  %N,
aerodynamic drag force

            vel_veh(t+1) = vel_veh(t) + ( Ft(t) - Fw(t) - Fr  ) * dT / (c_mass *
m_total);  %m/s, vehicle speed at next time instant

        end

        vel_endtime = vel_veh(end);   %m/s, achieved velocity at the end of time

        if vel_endtime > 100/3.6
           break;
        end

    end

    Pow_mc = Pow_mc + 100;  %W, if enough speed is now achieved, increase the power of
the machine
%     plot(T_mc); hold on;
end

a=1;



%% part 3

load('C:\Users\DELL\Documents\Dersler\EE 7566 Electric Vehicles\HW 1\drive
cycles\cycles_wltp.mat');  % loading WLTP drive cycle
vel_ref = WLTP_class_3.Data / 3.6;    %m/s,  reference speed of drive cycle
vel_act = zeros(1,numel(vel_ref));  %m/s, actual vehicle velocity
vel_err = zeros(1,numel(vel_ref));  %m/s, error in velocity
Fw_3 = zeros(1,numel(vel_ref));  %N, drag force array in part 3
Ft_req = zeros(1,numel(vel_ref));  %N, required traction force
Ft_act = zeros(1,numel(vel_ref));  %N, actual applied traction force
W_mc_act = zeros(1,numel(vel_ref));  %rad/s, actual machine speed
T_limit_mc = zeros(1,numel(vel_ref));  %Nm, maximum torque that machine can supply for
that specific speed
Ft_limit_mc = zeros(1,numel(vel_ref));  %N, maximum force that machine can supply for
specific speed

% Define maximum traction force limits due to tire adhesion

Ft_max_tires =   k_adh * m_total * g * k_load_acc;   %N, Maximum traction force that
tires can apply
Ft_min_tires = - k_adh * m_total * g;   %N, Minimum traction force that tires can apply
(consider mechanical brakes)

dT3 = 1;  %sec, time resolution of drive cycle
```

```
for t = 1:1:numel(vel_ref)

    vel_err(t) = vel_ref(t) - vel_act(t);  %m/s,    error in velocity
    acc_req = Kp * vel_err(t);         %m/s^2, acceleration required by loop

    Fw_3(t) = 0.5 * d_air * area_fr * cd * (vel_act(t)+vel_wind)^2;  %N, aerodynamic drag
force
    Ft_req(t) = c_mass * m_total * acc_req + Fr + Fw_3(t);      %N, required traction
force by loop

    % define traction force limits that you machine can supply

    W_mc_act(t) = vel_act(t) / r_wheel * gear_ratio;  %rad/s, actual machine speed
    if vel_act(t) < vel_veh_base
        T_limit_mc(t) = Tmax_mc; %Nm, maximum torque that machine can supply for that
specific speed
    else T_limit_mc(t) = Pow_mc / W_mc_act(t); % in field weakening region
    end

    Ft_limit_mc(t) = T_limit_mc(t) * gear_ratio / r_wheel * eff_gear_diff;  %N, maximum
force that machine can supply for specific speed

    Ft_act(t) = max( Ft_min_tires ,min( min(Ft_req(t), Ft_max_tires), Ft_limit_mc(t)) );
%N, actually applied traction force considering limits

    vel_act(t+1) = vel_act(t) + ( Ft_act(t) - Fw_3(t) - Fr  ) * dT3 / (c_mass * m_total);
%m/s, vehicle speed at next time instant

end


dist_wltp = sum(vel_act) / 1e3;    %km, total distance travelled with wltp drive cycle

% battery energy calculation

Ft_min_mc =  - k_adh * m_total * g * k_load_dec;   %N, minimum force that is applied by
rear wheels (thus machine)

Ft_mc_act = max(Ft_min_mc,Ft_act);   %N, traction force applied by machine.
Ft_brakes = Ft_act - Ft_mc_act;   %N, traction force applied by mechanical brakes. It is
on stage during acceleration regenerative braking is not enough.

T_mc_act = Ft_mc_act*r_wheel / gear_ratio / eff_gear_diff;   %Nm, actual machine torque
Pow_mc_act = T_mc_act .* W_mc_act;           %W, machine output power applied to the
vehicle at every instant

E_batt_wltp = sum(Pow_mc_act * dT3 / eff_batt / eff_mc_inv +
accesories_state*Pow_accesories * dT3 / eff_batt) / 36e5;         %kWh, battery capacity
for one WLTP drive cycle

E_batt_final = E_batt_wltp * range / dist_wltp;           %kWh, battery capacity required
for given range of WLTP drive cycle

Pow_regenerated_wltp = -min(0,Pow_mc_act);     %W, instantaneous regenerated energy in
wltp
Pow_tractive_wltp = max(0,Pow_mc_act);         %W, instantaneous tractive power in wltp

E_regenerated_wltp = sum(Pow_regenerated_wltp * dT3 * eff_mc_inv * eff_batt) / 36e5;
%kWh, regenerated energy in wltp
E_tractive_wltp = sum(Pow_tractive_wltp * dT3 / eff_mc_inv / eff_batt) / 36e5;   %kWh,
energy used for tractive efford in wltp
```

```
% range extender mode

Pow_mc_avg = mean(Pow_mc_act);  %W, average power of electric machine
Pow_ice = Pow_mc_avg / eff_charger / eff_mc_inv;  %W, power output of internal combustion engine
Pow_gen = Pow_mc_avg / eff_charger;    %W, power output of generator
E_batt_rex = E_batt_wltp * range_rex / dist_wltp;       %kWh, battery energy required for range extension
E_fueltank_rex = E_batt_rex / (eff_ice*eff_mc_inv*eff_batt*eff_charger);  %kWh, energy stored in fuel tank for rex
vol_fueltank = E_fueltank_rex / dens_gas;   % l , volume of fuel tank for rex


% mass calculations

Pow_batt = Pow_mc / eff_mc_inv  + Pow_accesories;   %W, battery power

m_batt = E_batt_final*1000 / (200 - 3*Pow_batt/E_batt_final/1000) + 120;    %kg,  battery mass
m_mc_inv = Pow_mc / eff_mc_inv/ m_spec_mc_inv ;   %kg, mass of the machine and inverter
m_ice = Pow_ice / m_spec_ice;  %kg, mass of ice
m_gen = Pow_gen / m_spec_mc_inv;  %kg, mass of generator

m_power_act = m_batt + m_mc_inv + m_charger + rex_mode*( m_fuel_tank + m_ice + m_gen + m_charger);  %kg, mass of drivetrain
m_total_final = m_body + m_power_act + m_emine;  %kg, total mass of the vehicle including

delta_mass = abs( m_total - m_total_final );  %kg, error in actual and predicted mass

m_power_pre = m_power_pre + 10;   %kg, increase predicted power mass

end


%% machine characteristics

for W_mc = 1:1:W_mc_maxspeed

    if W_mc < W_mc_base;      % check base speed
            T_mc(W_mc) = Tmax_mc;    %Nm, if machine is below base speed, maximum torque is achieved
            Pow_mc_char(W_mc) = T_mc(W_mc) * W_mc;   %W, machine power characteristics
         else
            T_mc(W_mc) = Pow_mc / W_mc;   %Nm, if base speed is exceeded, constant power region is valid
            Pow_mc_char(W_mc) = Pow_mc;   %W, machine power characteristics

    end

end
W_mc = 1:1:W_mc_maxspeed;

figure;
stem(W_mc_act*30/pi,T_mc_act,'*','linestyle','none');
hold on;
plot(W_mc*30/pi,T_mc,'b',W_mc*30/pi,-T_mc,'b','linewidth',2);
% plot(W_mc*30/pi,T_mc,'b','linewidth',2);

title('Torque Speed Characteristics')
xlabel('Speed (rpm)')
ylabel('Torque (Nm)')
```

```
figure;
plot(W_mc*30/pi,Pow_mc_char/1e3,'b','linewidth',2);
title('Power Speed Characteristics')
xlabel('Speed (rpm)')
ylabel('Power (kW)')


%% speed traction

figure;
time = 1:1:numel(vel_ref);                  %sec, time array
plot(time,vel_act(1:end-1)*3.6,time,vel_ref*3.6);  % plot reference and actual speed

title('Drive Cycle Performance')
xlabel('Time (sec)')
ylabel('Velocity (km/h)')
legend('Actual velocity','Reference velocity')

%% results

cost_mc_inv = Pow_mc/1e3 * cost_spec_mc_inv;        % $, cost of mc and inv
cost_batt = E_batt_final * (200 + 13 * Pow_batt/E_batt_final/1000);  % $, cost of batt
cost_ice = cost_spec_ice * Pow_ice/1e3; % $, cost of ice
cost_gen = Pow_ice/1e3 * cost_spec_mc_inv;    % $, cost of generator

cost_power = cost_mc_inv + cost_batt + cost_charger + rex_mode*( cost_fueltank + cost_ice
+ cost_gen + cost_charger );    % $, cost of powertrain

elect_cost = price_electric * E_batt_final;    %TL, electric cost for full range
fuel_ec = (elect_cost + rex_mode * vol_fueltank * price_gas ) / ( range +
rex_mode*range_rex) ;   %TL/km, fuel economy

fuel_cons = E_batt_final + rex_mode*E_fueltank_rex;  %kWh, fuel consumption for a given
range
percent_regen = E_regenerated_wltp / E_tractive_wltp * 100;    % percent of energy
regenerated


fprintf('Capacity of battery is %.2f kWh.\n', E_batt_final);
fprintf('Electric machine power is %.2f kWh.\n', Pow_mc/1e3);
fprintf('Electric machine torque is %.0f Nm.\n', Tmax_mc);
fprintf('Mass of drivetrain is %.1f kg.\n', m_power_act);
fprintf('Cost of drivetrain is %.0f $.\n', cost_power);
fprintf('Fuel consumption is %.2f kWh.\n', fuel_cons);
fprintf('Fuel economy is %.3f TL/km.\n', fuel_ec);
fprintf('Total regenerated energy is %.1f%%.\n', percent_regen);

fprintf('Maximum machine speed is %.0f rpm. \n', n_mc_maxspeed);
fprintf('Fuel consumption is %.3f kWh/km. \n', fuel_cons / ( range +
rex_mode*range_rex));
fprintf('Fuel economy is %.3f TL/km.\n', fuel_ec);
fprintf('ICE power output is %.2f kW.\n', rex_mode*Pow_ice/1e3);
fprintf('Capacity of fuel tank is %.2f liters.\n', vol_fueltank*rex_mode);
fprintf('Mixed fuel consumption in WLTP is %.2f kWh.\n',
fuel_cons/(range+range_rex)*dist_wltp);
fprintf('Fuel economy is %.3f TL/km.\n', fuel_ec);
fprintf('Charging time with level 1 charger is %.2f hours.\n',
E_batt_final/eff_batt/eff_charger/1.8);
fprintf('Charging time with level 2 charger is %.2f hours.\n',
E_batt_final/eff_batt/eff_charger/19.2);
```