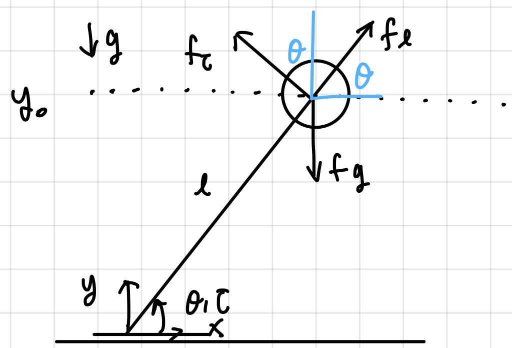
16-665 Legged Mobility Assignment

Jiyoon Park (andrew ID: jiyoonp)

#1

1)



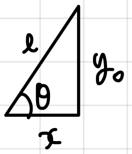
$f_{\ell x} = f_\ell \cos\theta$
$f_{\ell y} = f_\ell \sin\theta$

$f_{\tau x} = -f_\tau \sin\theta$
$f_{\tau y} = f_\tau \cos\theta$

$f_x = f_{\ell x} + f_{\tau x} = f_\ell \cos\theta - f_\tau \sin\theta$
$f_y = f_{\ell y} + f_{\tau y} - f_g = f_\ell \sin\theta + f_\tau \cos\theta - mg$

$\therefore f_x = f_\ell \cos\theta - f_\tau \sin\theta$
$f_y = f_\ell \sin\theta + f_\tau \cos\theta - mg$

2)



$\cos\theta = \frac{x}{\ell}$

$\sin\theta = \frac{y_0}{\ell}$

$\tau = f_\tau \ell \rightarrow f_\tau = \frac{\tau}{\ell}$

$f_x = f_\ell \cos\theta - f_\tau \sin\theta = f_\ell \frac{x}{\ell} - \frac{\tau}{\ell}\frac{y_0}{\ell} = f_\ell \frac{x}{\ell} - \frac{\tau y_0}{\ell^2}$

$f_y = f_\ell \sin\theta + f_\tau \cos\theta - mg = f_\ell \frac{y_0}{\ell} + \frac{\tau}{\ell}\frac{x}{\ell} - mg = f_\ell \frac{y_0}{\ell} + \frac{\tau x}{\ell^2} - mg$

$\therefore f_x = f_\ell \frac{x}{\ell} - \frac{\tau y_0}{\ell^2}$

$f_y = f_\ell \frac{y_0}{\ell} + \frac{\tau x}{\ell^2} - mg$

3) $f_y = f_\ell \frac{y_0}{\ell} + \frac{\tau x}{\ell^2} - mg = 0$

$f_\ell = \left(mg - \frac{\tau x}{\ell^2}\right)\frac{\ell}{y_0} = \frac{mg\ell}{y_0} - \frac{\tau x}{\ell y_0}$

$\therefore f_\ell = \frac{mg\ell}{y_0} - \frac{\tau x}{\ell y_0}$

4) $f_x = f_\ell \frac{x}{\ell} - \frac{\tau y_0}{\ell^2} = \frac{mg\ell^2 - \tau x}{y_0 \ell}\,\frac{x}{\ell} - \frac{\tau y_0}{\ell^2} = \frac{mgx\ell^2 - \tau x^2}{y_0 \ell^2} - \frac{\tau y_0}{\ell^2}$

$= \frac{mgx\ell^2 - \tau x^2 - \tau (y_0)^2}{y_0 \ell^2} = \frac{mgx\ell^2 - \tau\ell^2}{y_0 \ell^2} = \frac{mgx - \tau}{y_0}$

$\therefore f_x = \frac{mgx - \tau}{y_0}$

5) The relationship btw τ_A and Cop is

$$p = \frac{\tau_A}{mg} \qquad \tau = pmg$$

$$f_x = \frac{mgx - \tau}{y_0} = \frac{mgx - pmg}{y_0} = \frac{mg(x-p)}{y_0}$$

so, $m\ddot{x} = f_x = \dfrac{mg(x-p)}{y_0}$

6) $$\int_{-x_\tau}^{0} F_x dx = \int_{-x_\tau}^{0} \frac{mg}{y_0}(x-p)dx = \left[\frac{mg}{y_0}\left(\frac{1}{2}x^2 - px\right)\right]_{-x_\tau}^{0} = -\left(\frac{mg}{y_0}\left(\frac{1}{2}x_\tau^2 + px_\tau\right)\right)$$

$$= -\frac{mg}{2y_0}x_\tau^2 - \frac{mgpx_\tau}{y_0} = -\frac{mgx_\tau}{2y_0}(x_\tau + 2p)$$

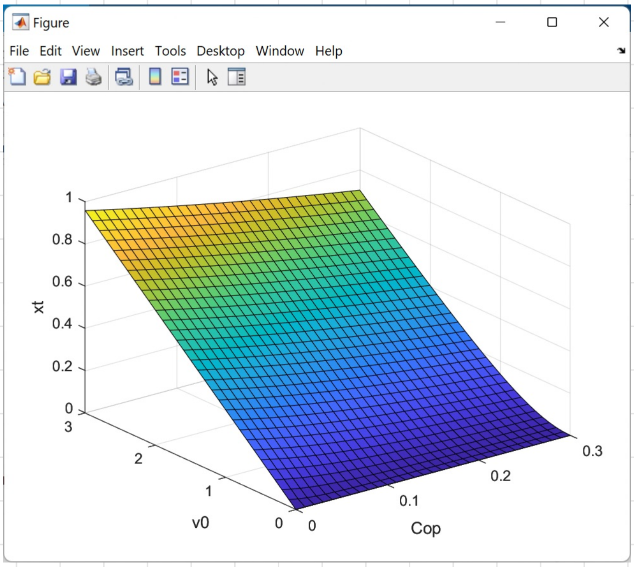
$$\cancel{-}\frac{\cancel{m}}{\cancel{2}}(V_0^2) = \cancel{-}\frac{\cancel{m}gx_\tau}{\cancel{2}y_0}(x_\tau + 2p)$$

$$y_0 V_0^2 = gx_\tau(x_\tau + 2p) = g(x_\tau)^2 + 2gpx_\tau$$

$$\Rightarrow g(x_\tau)^2 + 2gpx_\tau - y_0V_0^2 = 0$$

$$x_\tau = \frac{-2gp \pm \sqrt{4g^2p^2 + 4gy_0V_0^2}}{2g} = -p \pm \sqrt{p^2 + \frac{y_0V_0^2}{g}}$$

$$x_\tau = -p + \sqrt{p^2 + \frac{y_0V_0^2}{g}}$$



Looking at the plot, we see that as velocity increases the capture point x_τ increases as well. This is understandable in real world. Also, when the Cop increases x_τ decreases. The person should take another step when Cop leaves the polygon of support. For me, if Cop is greater than 0.25 then it would be rational for me to take a step. Before that, we can use an ankle strategy to maintain stability.

#2

1) part number : 500267

$$J_m = 5060 g cm^2 \times \frac{kg}{1000g} \times \left(\frac{1m}{100cm}\right)^2 = 0.000506 \, kgm^2$$

$$\tau_{max} = 964 \, kNm = 0.964 \, Nm$$

2) $$I_{on} = I_N \sqrt{\frac{\tau}{\frac{1}{2}\tau}} = I_N\sqrt{2}$$

$$\begin{aligned}\tau^*_{max} &= I_{on} \times \tau_{const} \\ &= I_N \sqrt{2} \times \tau_{const} \\ &= 4.06A \times \sqrt{2} \times 231 mNm/A \\ &= 231 \times 4.06\sqrt{2} \; mNm \\ &= 0.001 \times 937.86\sqrt{2} \; Nm \\ &= 1.326334 \, Nm\end{aligned}$$

$$\begin{aligned}\tau_{ext} &= r \, 1.37 mg \\ &= N \tau_{max}\end{aligned}$$

$$N = \frac{\tau_{ext}}{\tau_{max}} = \frac{1.37 \times 80kg \times 9.81 m/s^2 \times 0.05m}{1.326334 \, Nm}$$

$$= \frac{40.532 \, \cancel{kg} \, \cancel{m^2}}{\cancel{s^2} \, \cancel{Nm}} = 40.532$$

$\therefore \tau^*_{max} = 1.326334 Nm$

$N = 40.532$

3) $$F_s = K \Delta l \qquad \Delta l = \Delta I_y + \Delta I_m \qquad \Delta I_y = y_0 - y \qquad \Delta I_m = \frac{r}{N}\theta_m$$

$$= y_0 - y + \frac{r}{N}\theta_m$$

$$F_s = K\left(y_0 - y + \frac{r}{N}\theta_m\right)$$

$$J_m \ddot{\theta} = \tau_m - \tau_{ext} \quad \cdots (3)$$

$$\tau_{gear} = F_s \cdot r$$

$$\tau_{ext} = \frac{\tau_{gear}}{N} = \frac{F_s r}{N} = \frac{K\left(y_0 - y + \frac{r}{N}\theta_m\right) r}{N}$$

$$J_m \ddot{\theta} = \tau_m - \frac{F_s \cdot r}{N}$$

$$\ddot{\theta} = \frac{\tau_m}{J_m} - \frac{F_s \cdot r}{J_m \cdot N}$$

$$\therefore \ddot{\theta} = \frac{\tau_m}{J_m} - \frac{F_s \cdot r}{J_m \cdot N}$$

4) For the outer loop,

let $e_1 = y_{des} - y$

Then, $\dot{e}_1 = \dot{y}_{des} - \dot{y}$ since y_{des} is a constant, $\dot{e}_1 = -\dot{y}$

$\ddot{e}_1 = \ddot{y}_{des} - \ddot{y}$ $\qquad \ddot{e}_1 = -\ddot{y}$ .... (1)

$m\ddot{y} = F_s^{des} - mg$: system dynamics. .... (2)

$\ddot{y} = \frac{F_s^{des}}{m} - g$ .... (3)

using the error dynamics function, $\ddot{e} + k_d\dot{e} + k_p e = 0$ and (1)

$$\ddot{e} + k_d\dot{e} + k_p e = 0$$

$$-\ddot{y} + k_d\dot{e} + k_p e = 0$$

$$g - \frac{F_s^{des}}{m} + k_d\dot{e} + k_p e = 0 \quad \text{(using (3))}$$

$$\frac{F_s^{des}}{m} = g + k_d\dot{e} + k_p e$$

$$F_s^{des} = mg + k_d m\dot{e} + k_p e m$$

$$= m(g + k_d\dot{e} + k_p e) \text{ .... (4)}$$

$\therefore F_s^{des} = m(g + k_d\dot{e} + k_p e)$

For the inner loop,

$e = \theta_m^{des} - \theta_m$

$\dot{e} = \dot{\theta}_m^{des} - \dot{\theta}_m$ since θ_m^{des} is a constant, $\ddot{e} = -\ddot{\theta}_m$ .... (5)

$\ddot{e} = \ddot{\theta}_m^{des} - \ddot{\theta}_m$

we use the same error dynamics

$$\ddot{e} + k_d\dot{e} + k_p e = 0$$

$$-\ddot{\theta}_m + k_d\dot{e} + k_p e = 0 \text{ .... (6)}$$

$$J_m\ddot{\theta}_m = \tau_m - \tau_{ext} = \tau_m - \frac{F_s \cdot r}{N}$$

$$\ddot{\theta}_m = \frac{\tau_m}{J_m} - \frac{F_s \cdot r}{NJ_m} \text{ .... (7)}$$

(6) & (7)

$$\left(-\frac{F_s \cdot r}{NJ_m} + \frac{\tau_m}{J_m} = k_d\dot{e} + k_p e\right) \times NJ_m$$

$$-F_s \cdot r + N\tau_m = NJ_m k_d\dot{e} + NJ_m k_p e$$

$$N\tau_m = NJ_m k_d\dot{e} + NJ_m k_p e + F_s \cdot r$$

$$\tau_m = J_m k_d\dot{e} + J_m k_p e + \frac{F_s \cdot r}{N} \text{ .... (9)}$$

$$F_s = k(\Delta I_y + \Delta I_m) = k\left(y_0 - y + \frac{r\theta_m}{N}\right) = k\left(e_1 + \frac{r\theta_m}{N}\right) \text{ ....(10)}$$

(9) & c(0)

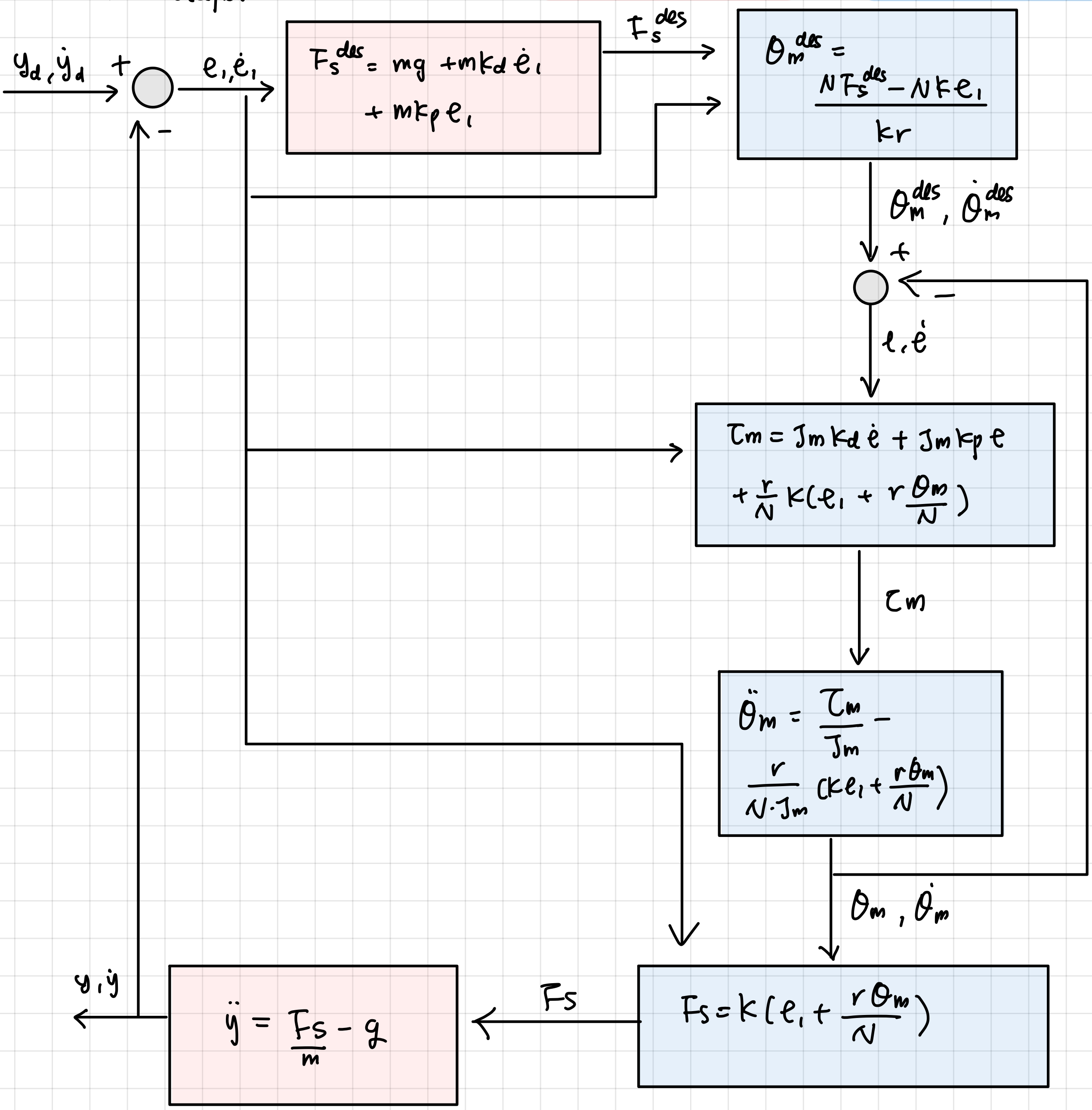
$$\tau_m = J_m k_d \dot{e} + J_m k_p e + \frac{F_s \cdot r}{N}$$

$$= J_m k_d \dot{e} + J_m k_p e + \frac{r}{N} k\left(e_1 + \frac{r\theta_m}{N}\right)$$

these are the basic calculations

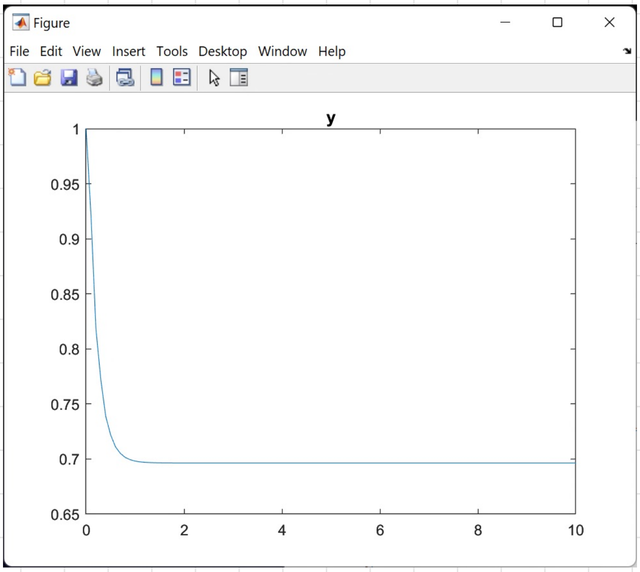
$$\therefore \tau_m = J_m k_d \dot{e} + J_m k_p e + \frac{r}{N} k\left(e_1 + \frac{r\theta_m}{N}\right)$$

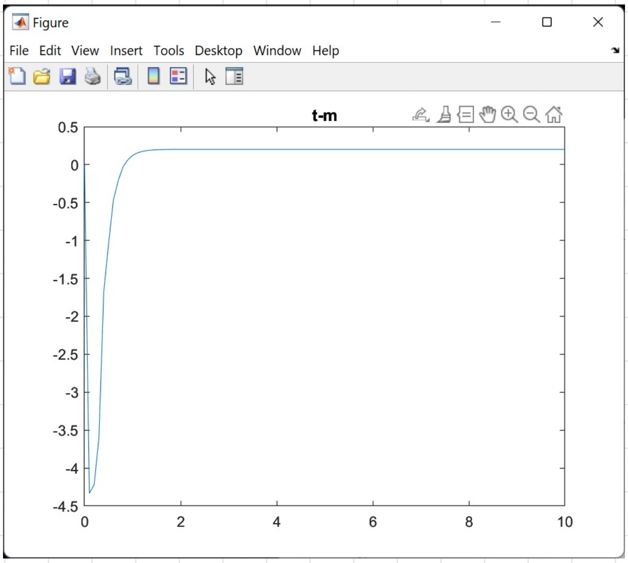
The flow diagram is (where the outer loop is in red and inner loop is in blue



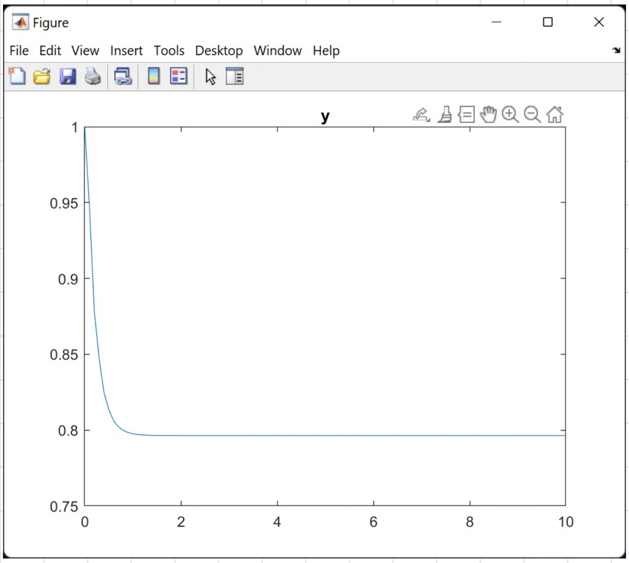
In the outer loop, we take in y, y_{des} as input then calculate F_s^{des}. We send this F_s^{des} to the inner loop then calculate θ_m^{des}. With the diffrence e of θ_m^{des} and θ_m, we calculate τ_m. τ_m is then recalculated to current θ_m and is finally calculated to F_s. The inner loop outputs F_s and the outer loop calculates that back to current y and loops the system again.

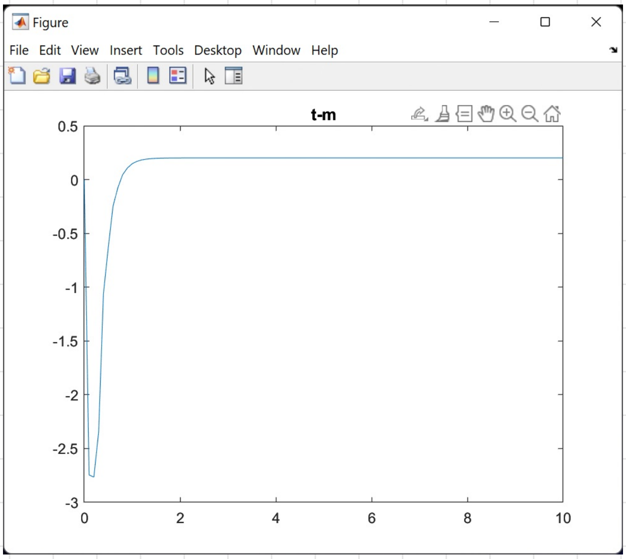
5) for y-des = 0.7



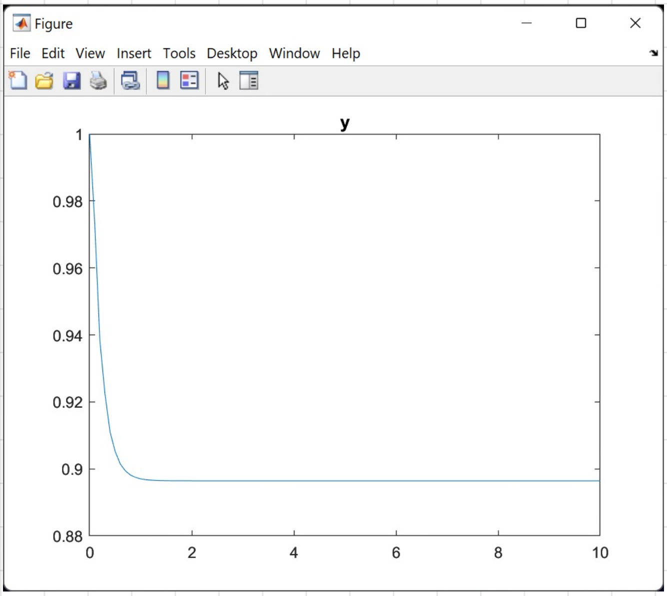


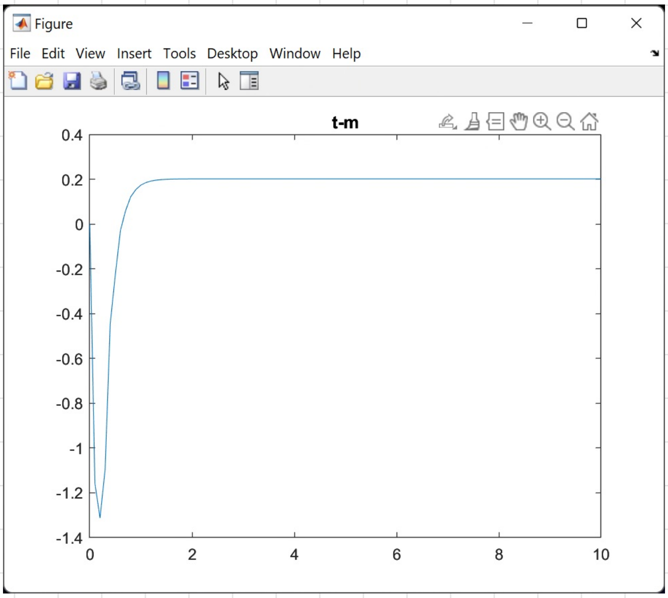
for y-des = 0.8





for y-des = 0.9





Question 3

1) The other two things are Swing Foot Point (FP) and Trunk (TR) (torso)

2)

FP: The PD tracking of nominal foot point plan is modified based on the joint angles (q), joint velocities (dq). It tracks the nominal swing foot plan using computed torque control. It uses pd control to compute the desired acceleration (nominal acceleration) as feedforward term plus PD-feedback tracking of the desired motion.

TR: They are updating the desired truck acceleration using PD tracking of desired truck pitch. It takes joint (q) and joint velocities (dq) and calculates the desired acceleration.

3)

$GRF_x \leq \mu GRF_y$ $\qquad\qquad$ $-\mu GRF_y \leq GRF_x$

$GRF_x - \mu GRF_y \leq 0$ $\qquad\qquad$ $-GRF_x - \mu GRF_y \leq 0$

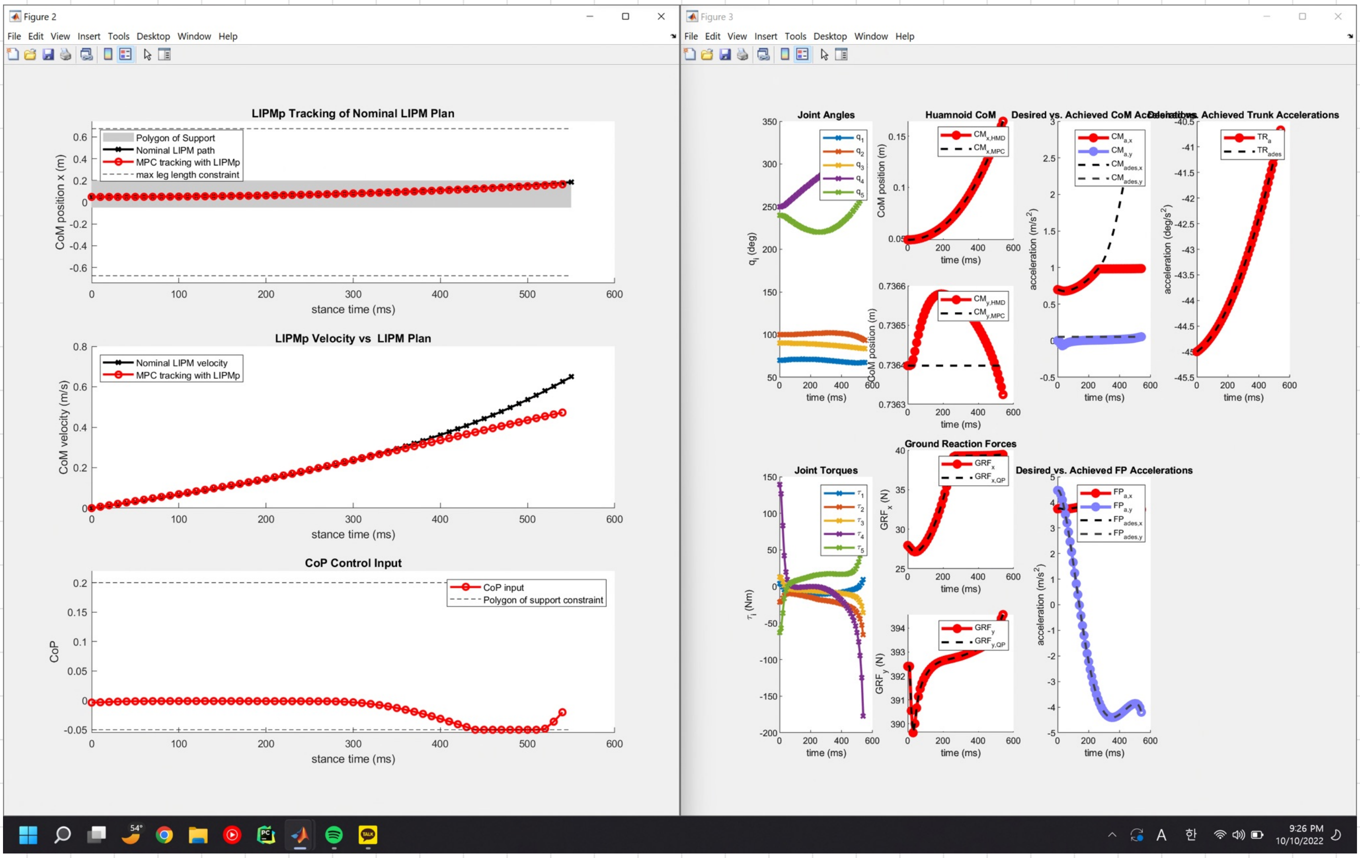
$\begin{bmatrix} 1 & -\mu \end{bmatrix} \begin{bmatrix} GRF_x \\ GRF_y \end{bmatrix} \leq 0$ $\qquad\qquad$ $\begin{bmatrix} -1 & -\mu \end{bmatrix} \begin{bmatrix} GRF_x \\ GRF_y \end{bmatrix} \leq 0$

$a_{11} = 1 \quad a_{12} = -\mu$ $\qquad\qquad$ $a_{21} = -1 \quad a_{22} = -\mu$

$$A_{ineq} = \begin{bmatrix} 0\,0\,0\,0\,0\,0\,0\,0\,0\,0 & 1 & -\mu \\ 0\,0\,0\,0\,0\,0\,0\,0\,0\,0 & -1 & -\mu \end{bmatrix} \qquad b_{ineq} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

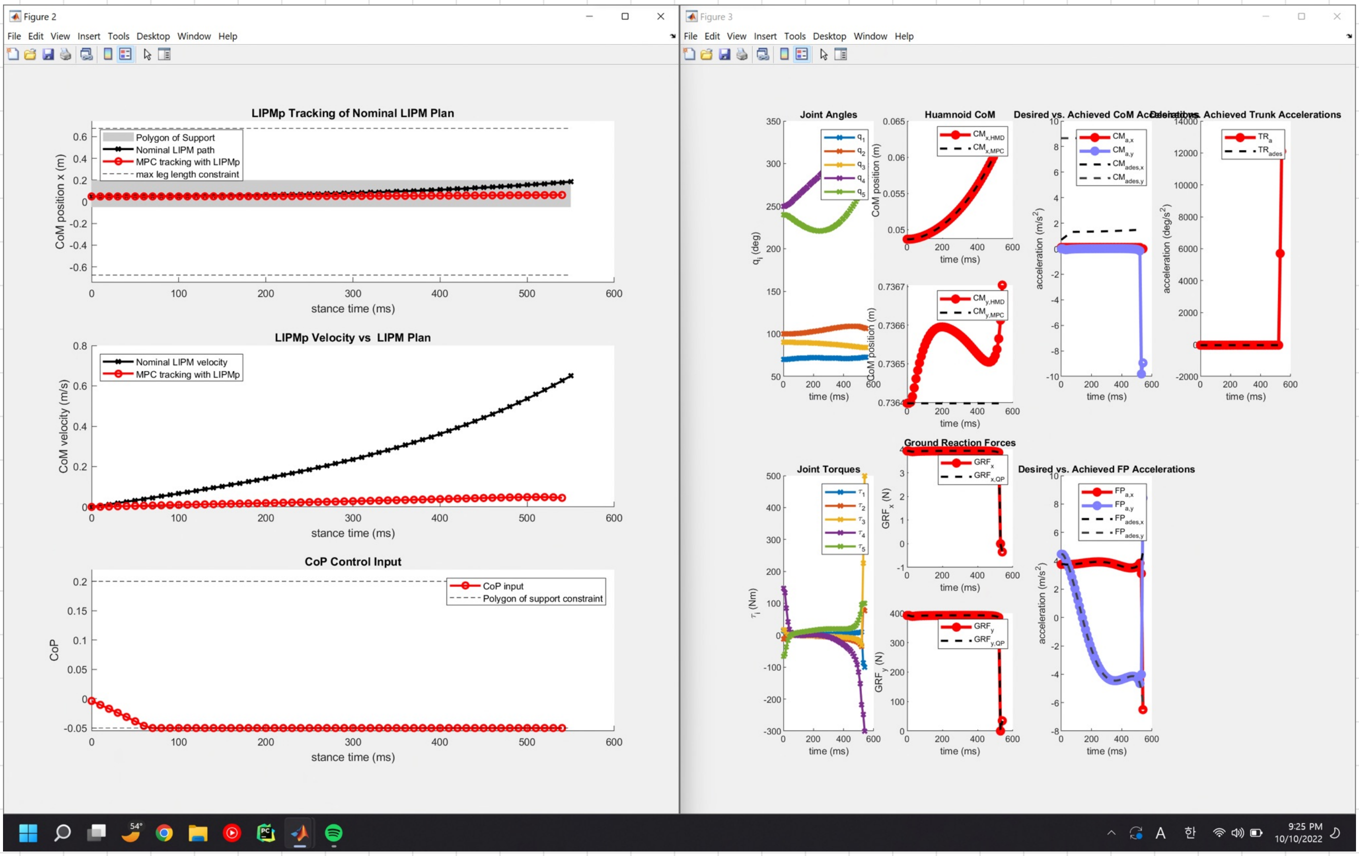
and I put it into my matlab code

4) $u = 0.1$



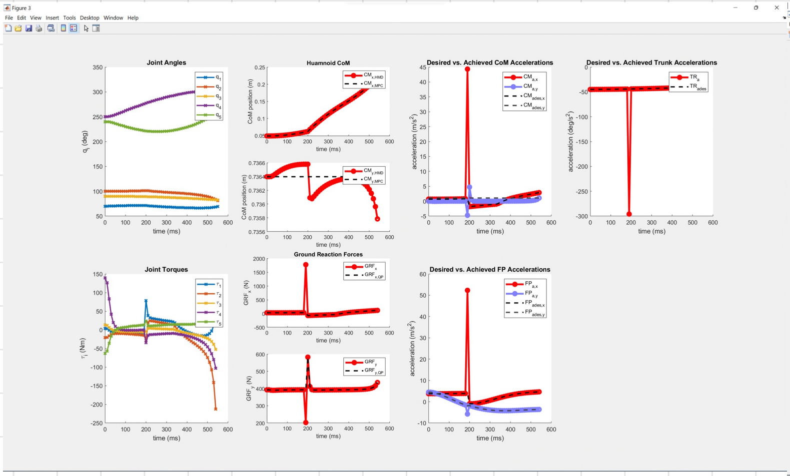
When $u=0.1$ we have the constraint $-0.1\,GRF_y \le GRF_x \le 0.1\,GRF_y$. Since $GRF_y = 400N$ our GRF_x constraint is $-40N \le GRF_x \le 40N$. We can see in the figure 3, Ground reaction forces plot that the GRF_x is indeed capped at 40N, Since it is capped at 40N, the acceleration and the velocity gets affected as well. This is shown in figure 2. It is clear that CoM velocity is not high enough to track the Nominal LIPM velocity. The Cop gets effected also as it is at reaches the bottom edge of polygon of support. The CoM position x seem to track the Nominal LIPM path, but towards the end, it is gaining more error. If it were to walk for a longer period of time the error would accumulate more.

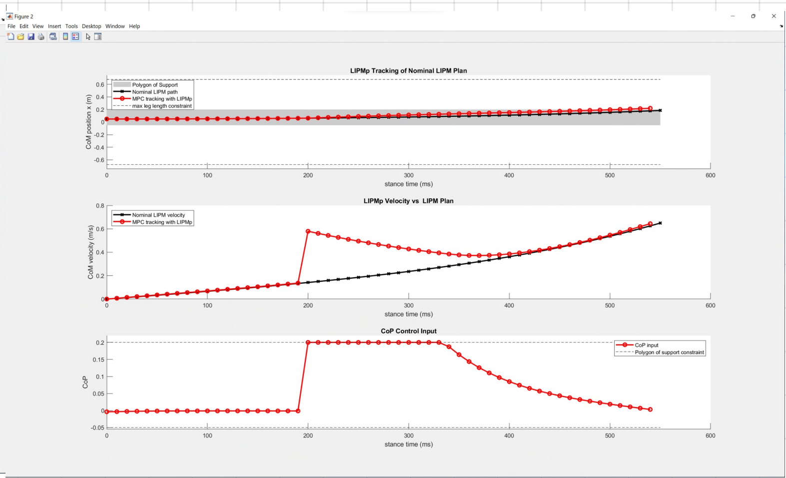
$u=0.01$



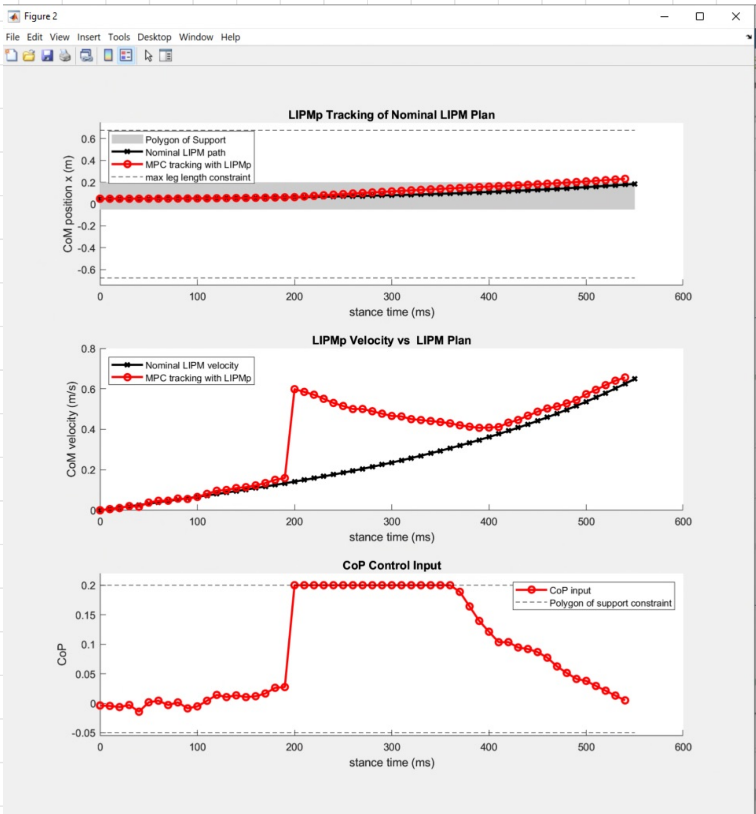
When $u=0.01$, then $-4N \le GRF_x \le 4N$. So, there is a bigger restriction to the Force. That means there will be a bigger difference in CoM velocity and CoM position. We first see in figure 3 that the GRF_x is indeed capped at 4N. Also, in figure 2, we see that the CoM velocity is very low compared to the Nominal LIPM velocity. And we also see that Cop input is at the very bottom close to the edge of (lower limit) polygon of support. This all results in CoM position X not being able to exactly track the Nominal LIPM path.

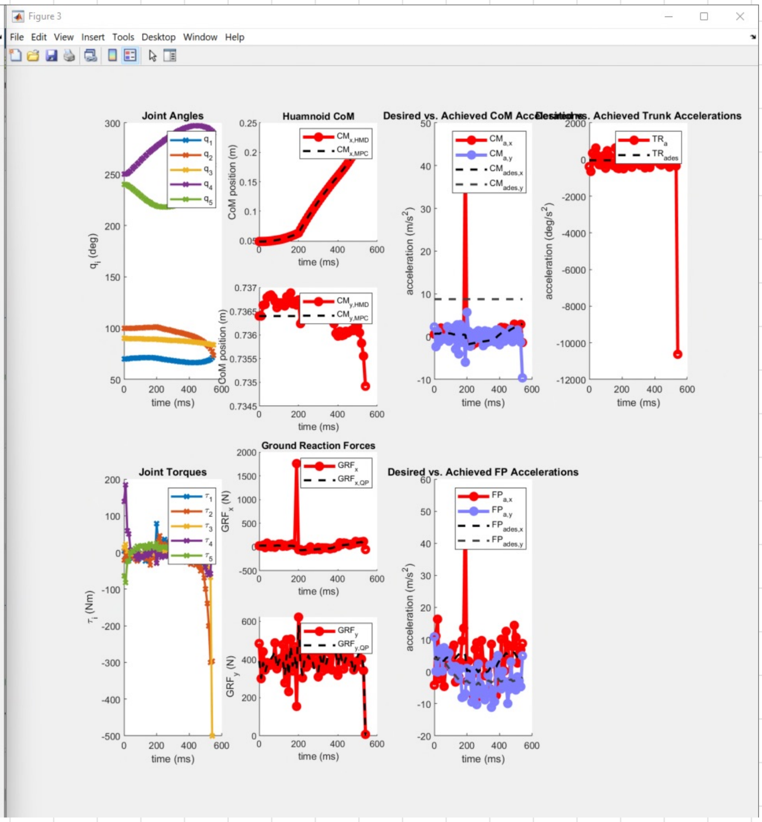
5)





When a disturbance impulse of +20Ns is applied, we can see that there is a spike at t=200 for the GRF, acceleration, velocity. This is natural since external force has been applied to it. However, the controller stablizes quickly and makes the acceleration/GRF/velocity go back to normal. Looking at figure 2, we can see that the CoM velocity stablizes after some time, but CoM position x leaves the polygon of support towards the end. Which means the robot will fall soon.



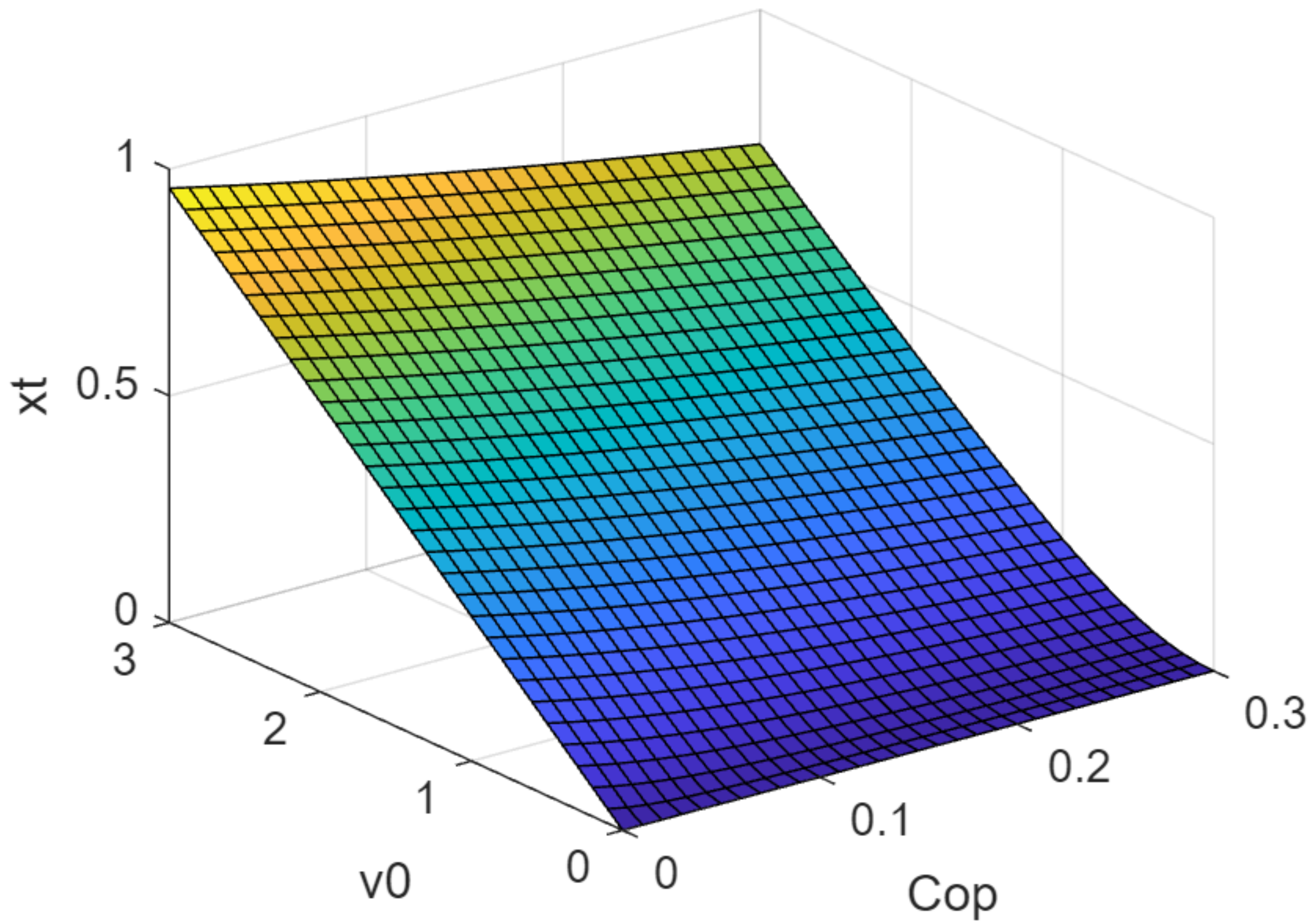


When I add noise of 5Nm then every value in figure 3 oscilates as there is actuator noise. It does not smoothly track the desired outputs and thus CoM velocity and CoM position gets affected a lot. So it becomes harder to track as wanted in the beginning.

Jiyoon Park HW2

Question 1

```
v0 = [0:0.1:3];
Cop = [0:0.01:0.3];
[v0_mesh,Cop_mesh] = meshgrid(v0,Cop);
y0 = 1;
g = 9.81;

xt = -Cop_mesh +1/g*(sqrt((g*Cop_mesh).^2+g*y0*v0_mesh.^2));

surf(Cop_mesh,v0_mesh,xt)
hold on
% xt2 = v0_mesh*(y0/g)*0.5;
% surf(Cop_mesh,v0_mesh,xt2)

xlabel('Cop');
ylabel('v0');
zlabel('xt')
```



Question 2

```
y_des = 0.9;
y_des_dot = 0;

Fs_des = 0;
Fs = 0;

y0=1;
y0_dot = 0;

y = y0;
y_dot = 0;

e1 = y_des - y;
e1_dot = 0;
```

```
t_m= 0;
theta_m=0;
theta_m_dot =0;

theta_m_des = 0;
theta_m_des_dot = 0;
```

```
kd1 = 15;
kp1 = 50;

kd2 = 130;
kp2 = 4200;

N = 40.532; %kg m / s^2 N
r = 0.05; % m
Jm = 0.000506; % kg m^2
m = 80; % kg
g = 9.81;% m/s^2
K = 20000; % kNm^-1

timestep = 0.1;
timestep2 = 0.01;
```

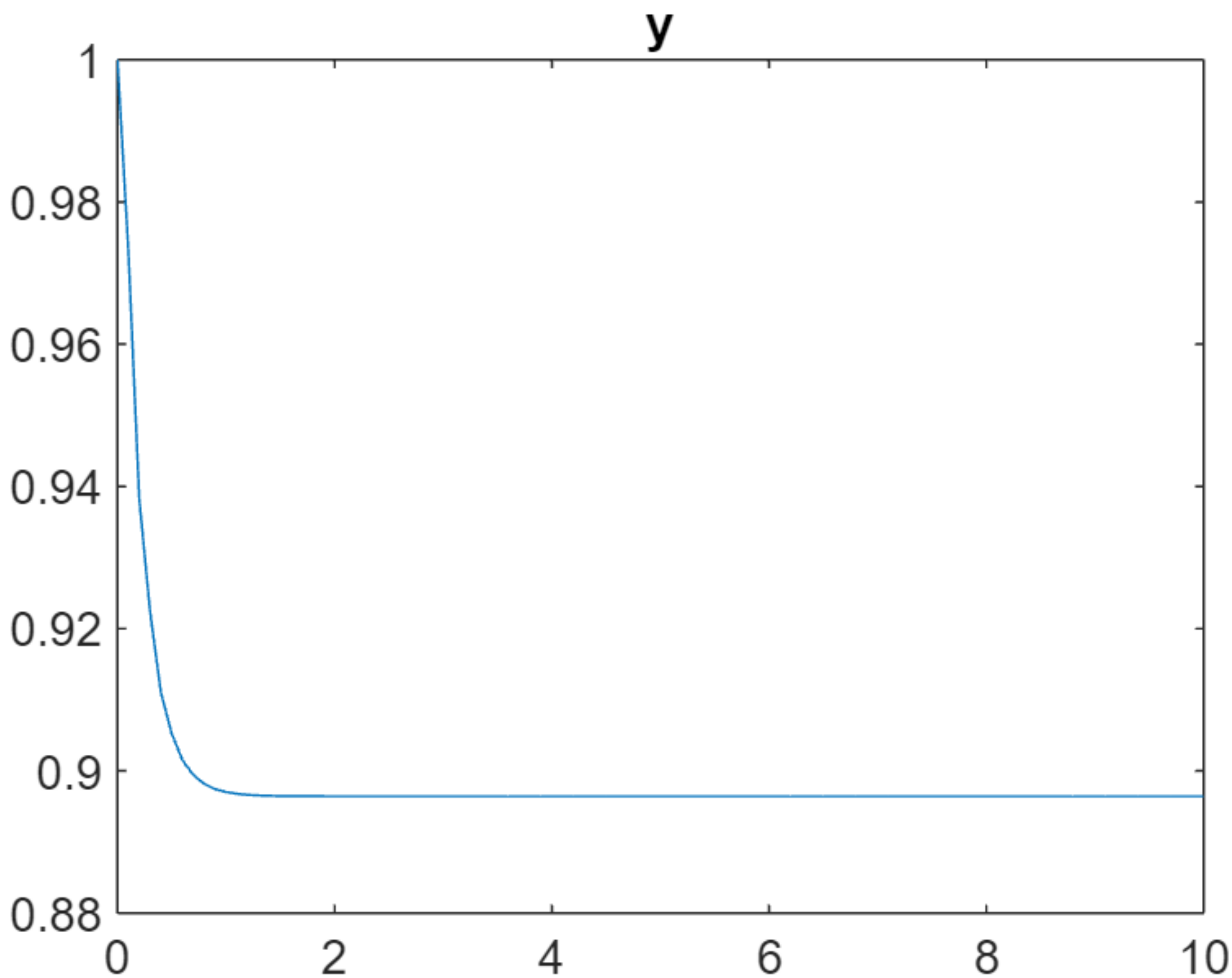
```
time = 0:timestep:10;
Fs_des_list = zeros(length(time), 1);
t_m_list = zeros(length(time), 1);

time2 = 0:timestep2:timestep;

Fs_list = zeros(length(time), 1);
theta_m_list = zeros(length(time2), 1);

theta_m_state_list = zeros(length(time2), 2);

y_list = zeros(length(time), 1);
y_list(1) = y;
y_state_list = zeros(length(time), 2);
y_state_list(1) = y;
```

```
for i=1:length(time) - 1

    index = i;

    e1 = y_des - y;
    e1_dot = -y_dot;
```

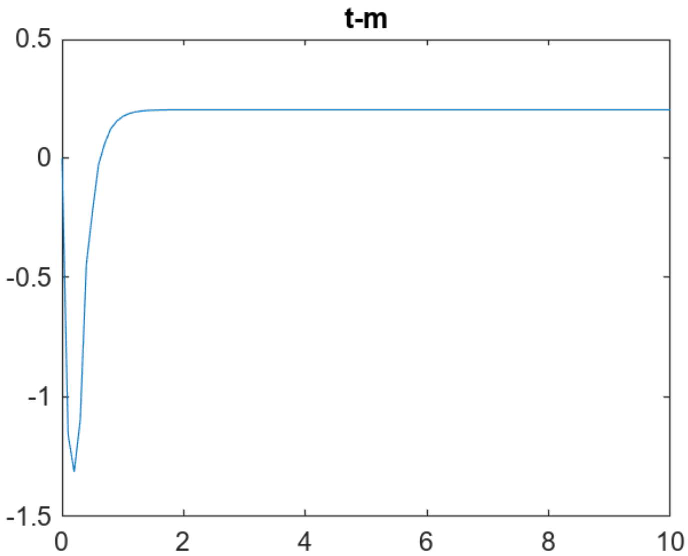
```
    Fs_des = m*g + m*kd1*e1_dot+m*kp1*e1;
    Fs_des_list(index) = Fs_des;

    for j=1:length(time2)- 1
        index_inner =j;

        theta_m_des = (N*Fs_des - N*K*e1)/(K*r);


        e = theta_m_des - theta_m;
        e_dot = theta_m_des_dot - theta_m_dot;

        t_m = Jm*kd2*e_dot+Jm*kp2*e+r*K/N *(e1+r*theta_m/N);
        [theta_m_time, theta_m_state] = ode45(@(t, x) func3(t, x, t_m, e1), [0, timestep2], the
        theta_m_state_list(index_inner+1, :) = theta_m_state(end, :);

        theta_m =theta_m_state(end, 1);
        theta_m_dot =theta_m_state(end, 2);

        Fs = K*(e1+(r*theta_m)/N);

    end

    t_m_list(index+1) = t_m;

    Fs_list(index+1) = Fs;
    [y_time, y_state] = ode45(@(t, x) func4(t, x, Fs), [0, timestep], y_state_list(index, :)');

    y = y_state(end, 1);
    y_dot = y_state(end, 2);
    y_state_list(index+1, :) = y_state(end, :);

    y_list(index+1)=y;

end
```

```
% figure
% plot(time, Fs_list)
% hold on
% plot(time, Fs_des_list)
% title('Fs')
% legend('Fs', 'Fs-des')
% figure
% plot(time, Fs_des_list - Fs_list)
% title('Fs-des - Fs')

figure
plot(time, y_list)
```

```
title('y')
```




```
figure

plot(time, t_m_list)
title('t-m')
```




```
function out = func3(t, x, t_m, e1)
    N = 40.532; %kg m / s^2 N
    r = 0.05; % m
    Jm = 0.000506; % kg m^2
    m = 80; % kg
    g = 9.81;% m/s^2
    K = 20000; % kNm^-1

    A = [0 1;
        -(r^2*K)/(N^2*Jm) 0]; % -
    B=[0;
        t_m/Jm - (r*K*e1)/(N*Jm)]; % -
    out = A*x+B;
end

function out = func4(t, x, Fs)
    N = 40.532; %kg m / s^2 N
    r = 0.05; % m
    Jm = 0.000506; % kg m^2
    m = 80; % kg
    g = 9.81;% m/s^2
    K = 20000; % kNm^-1

    A = [0 1;
        0 0];
    B=[0;
        Fs/m-g];

    out = A*x+B;
```

```
end
```