Table of Contents


```
% MAE 219 Assignment 2
% Original Author: Allison Okamura, September 20, 2015
% Last Modified By: David Lim, October 10, 2024

% Clear the workspace.
clear;
```

Constants

```
% device and human parameters
m = 0.03;   % effective mass at the handle, kg
b = 1;   % viscous damping, Ns/m
kh = 500;   % human hand stiffness, N/m
bh = 10;   % human hand damping, Ns/m

% virtual wall parameters
kwall = 500;   % wall stiffness, N/m
% kwall = i) 500, iv) 50, v) 10000
xwall = 0.025;   % wall position, m
% xwall = i) 0.025, ii) 0.05, iii) -0.05

% times for dynamic simulation
tstart = 0;   % s
tend = 10;   % s
T = 0.1*10^-3;   % time increment, s
t = (tstart:T:tend)';   % time vector

% human input
omega = 0.4*2*pi;   % frequency of user's desired motion, rad/s
A = 0.04;   % amplitude of user's desired motion, m
xd = A*sin(omega*t) - 0.01;   % user's desired hand position, in m
vd = A*omega*cos(omega*t);   % user's desired hand velocity, in m/s
```

State Tracking

```
xh = zeros(length(t),1);    % handle position
vh = zeros(length(t),1);    % handle velocity
ah = zeros(length(t),1);    % handle acceleration
fa = zeros(length(t),1);    % force applied by the actuator
ffelt = zeros(length(t),1); % force felt by the human
```

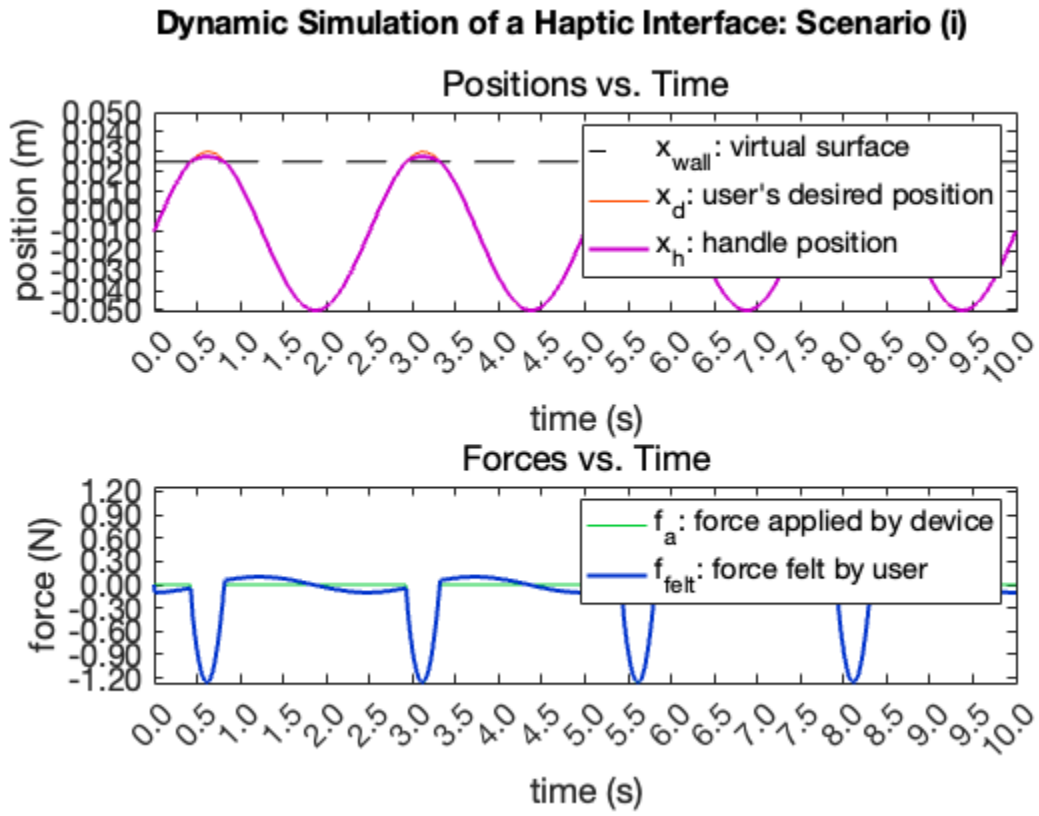
Dynamic Simulation

```
for i = 1:length(t)
    % integrate the main state derivatives
    if (i == 1)
        % first time step has no difference between desired and actual
        % handle position
        vh(i) = vd(i);
        xh(i) = xd(i);
    else
        % simple Euler integration (you could use something more accurate!)
        vh(i) = vh(i-1) + ah(i-1) * T;
        xh(i) = xh(i-1) + vh(i-1) * T;
    end

    % force applied by the virtual environment
    if (xh(i) > xwall)
        fa(i) = kwall * (xwall - xh(i));
    else
        fa(i) = 0;
    end

    % force between the hand and the handle
    fh = kh * (xd(i) - xh(i)) + bh * (vd(i) - vh(i));

    % force felt by the user
    ffelt(i) = -fh;

    % damping force
    ff = -b * vh(i);

    % Compute the sum of forces on the handle: applied force, human force,
    % and friction force.
    ftotal = fa(i) + fh + ff;

    % Compute the handle's new acceleration for the next iteration.
    ah(i) = ftotal / m;

end
```

Plotting

```
figure(1); clf;

% tick mark calculations
N = 5;
% find max
xmax = max( abs([xd;xh]) );
% create N tick marks above and below zero
xtic = round( xmax/N, 1,'significant');
xtics = [-fliplr(xtic:xtic:xmax) 0 xtic:xtic:xmax];
% find max
```

```
fmax = max(abs([fa;ffelt]));
% create N tick marks above and below zero
ftic = round( fmax/N, 1,'significant');
ftics = [-fliplr(ftic:ftic:fmax) 0 ftic:ftic:fmax];

% positions in top subplot
subplot(2,1,1)
h = plot(t, xwall*ones(1,length(t)), t, xd, t, xh);
set(h(1),'Color',[0 0 0],'LineWidth',1.0,'LineStyle','--')
set(h(2),'Color',[1 .3 0],'LineWidth',1)
set(h(3),'Color',[.8 0 .8],'LineWidth',2)
xlabel('time (s)')
ylabel('position (m)')
legend('x_{wall}: virtual surface','x_d: user''s desired position','x_h:
handle position')
title('Positions vs. Time','FontWeight','Normal')
axis([tstart tend -xmax xmax])
set(gca,'FontSize',16)
xticks(0:0.5:tend)
xtickformat('%.1f')
yticks(xtics)
ytickformat('%.3f')

% forces in bottom subplot
subplot(2,1,2)
h = plot(t, fa, t, ffelt);
set(h(1),'Color',[0 .8 .2],'LineWidth',1)
set(h(2),'Color',[0 .2 .8],'LineWidth',2)
xlabel('time (s)')
ylabel('force (N)')
legend('f_a: force applied by device','f_{felt}: force felt by user')
title('Forces vs. Time','FontWeight','Normal')
axis([tstart tend -fmax fmax])
set(gca,'FontSize',16)
xticks(0:0.5:tend)
xtickformat('%.1f')
yticks(ftics)
ytickformat('%.2f')

sgtitle('Dynamic Simulation of a Haptic Interface: Scenario
(i)','FontSize',16,'FontWeight','Bold')
```



Published with MATLAB® R2024a