Table of Contents

```
close all
clear all
```

parameters setting (Choose one of the 3 cases)

```
set_params_case1
% set_params_case2
% set_params_case3
```

initial condition

```
x_m = 0;
xd_m = 0;
xdd_m = 0;
x_s = 0;
xd_s = 0;
xdd_s = 0;

tau_op = 0;
f_m = 0;
tau_m = 0;
f_s = 0;
tau_s = 0;
```

operator input function

```
input_force = @(t) ( 5-5*cos(4*pi*t));
```

--simulation--

```
dt = 0.001;
sim_time = 4;
t = linspace(0, sim_time, sim_time/dt);

x_m_log = zeros(size(t));
x_s_log = zeros(size(t));
f_m_log = zeros(size(t));
f_s_log = zeros(size(t));

% iteration
% for i = 1:length(t)
```

operator input force

```
    tau_op = input_force(t(i));
```

```
        Subscript indices must either be real positive integers or logicals.

        Error in teleop_simulation (line 42)
            tau_op = input_force(t(i));
```

----------------operator_dynamics => master_impedance----------------

% operator dynmacis => master impedance doesn't work at every cases % master impedance => operator dynamics doesn't work at every cases

```
           % master impedance model
           f_m = m_m * xdd_m + b_m * xd_m - tau_m;

           % operator dynamics
           xdd_m = (tau_op - f_m - b_op * xd_m - c_op * x_m) / m_op;
           xd_m = xd_m + xdd_m * dt;
           x_m = x_m + xd_m * dt;

% ----------------------------------------------------------------------
```

----------------master dynamics, operator impedance----------------

operator impedance => master dynamics works at every cases master dynamics => operator impedance works except 1st case.

```
    % operator impedance model(eq4)
    f_m = tau_op - (m_op * xdd_m + b_op * xd_m + c_op * x_m);
    % master dynamics(eq1)
    xdd_m = (tau_m + f_m - b_m * xd_m) / m_m;
    xd_m = xd_m + xdd_m * dt;
    x_m = x_m + xd_m * dt;

% -----------------------------------------------------------------------
```

master controller(eq5)

```
    tau_m = master_controller(x_m, xd_m, xdd_m, f_m, x_s, xd_s, xdd_s, f_s);
```

slave controller(eq6)

```
    tau_s = slave_controller(x_m, xd_m, xdd_m, f_m, x_s, xd_s, xdd_s, f_s);
```

slave dynamics(eq2)

```
    xdd_s = (tau_s - f_s - b_s * xd_s) / m_s;
    xd_s = xd_s + xdd_s * dt;
    x_s = x_s + xd_s * dt;
```

object impedance model(eq3)

```
    f_s = m_w * xdd_s + b_w * xd_s + c_w * x_s;
```

logging

```
    x_m_log(i) = x_m;
    x_s_log(i) = x_s;
    f_m_log(i) = f_m;
    f_s_log(i) = f_s;
% end
% figure(1);
% plot(t, x_m_log, 'r--','linewidth',2);
% grid on;
% hold on;
% plot(t, x_s_log, 'b','linewidth',2);
% legend('master', 'slave');
% title('position response');
%
% figure(2);
% plot(t, f_m_log, 'r--','linewidth',2);
% grid on;
% hold on;
% plot(t, f_s_log, 'b','linewidth',2);
% legend('master', 'slave');
% title('force response');

% autoArrangeFigures(1,2)
```

Published with MATLAB® R2013b