Homework 3 - Problem 3

Coded by Michael White

```
clear;clc;
% Dynamic equations using Newton-Euler formulation
syms d_2 dDot_2 dDotDot_2 theta1 thetaDot_1 thetaDotDot_1 theta2 thetaDot_2 thetaDotDot_2 g Ixx_1 Iyy_1 Izz_1;
linkTable = [0 0 0 theta1; -pi/2 0 d_2 0];

T01 = functions.links.Link2Transform(linkTable(1,:));
T12 = functions.links.Link2Transform(linkTable(2,:));

% Pull and define rotations from transforms
R01 = functions.transform.rotationFromTransform(T01);
R12 = functions.transform.rotationFromTransform(T12);

syms dDot_2 dDotDot_2 thetaDot_1 thetaDotDot_1 thetaDot_2 thetaDotDot_2 g;
syms c1 c2 s1 s2 m1 m2;

v0_dot = [0 0 g].';

% Define initial conditions
P_01 = functions.transform.positionFromTransform(T01);
P_12 = functions.transform.positionFromTransform(T12);
Pc_11 = [0 0 0].';
Pc_22 = [0 0 0].';

syms Ixx_1 Iyy_1 Izz_1;
Ic_11 = [Ixx_1 0 0; 0 Iyy_1 0; 0 0 Izz_1];
Ic_22 = 0;
w_0 = [0 0 0].';
wDot_0 = 0;
v_11 = [0 0 0].';

% Functions needed for v_22
w_11 = functions.dynamics.omega_ip1ip1(R01,w_0,thetaDot_1);
v_22 = functions.dynamics.v_ip1ip1_prism(R12,v_11,w_11,P_12,dDot_2);

% Solve for J_2
v_22 = v_22([1,3],:);
J_2 = v_22./[thetaDot_1;dDot_2];
J_2 = J_2.*eye(2);

% Solve for JDot_2
JDot_2 = subs(diff(J_2,d_2),-1,-dDot_2);

% Pulling from previous problem
M = [Izz_1+m2*d_2^2 0; 0 m2];
V = [2*d_2*m2*dDot_2*thetaDot_1; -m2*d_2*thetaDot_1^2];
G = [0; 0];

M = M.*eye(2);
V = V.*eye(2);
G = G.*eye(2);

% Solve for Mx
Mx = simplify(((J_2^-1).').*M.*(J_2^-1));
Vx = simplify(((J_2^-1).')*(V-M*(J_2^-1)*JDot_2*thetaDot_1));
Gx = simplify(((J_2^-1).')*G);
```

```
% Simplify answers to column vectors
Mx = [Mx(1,1); Mx(2,2)];
Vx = [Vx(1,1); Vx(2,2)];
Gx = [Gx(1,1); Gx(2,2)];

% Display answers
disp('Mx = ');disp(Mx);
disp('Vx = ');disp(Vx);
disp('Gx = ');disp(Gx);
```

```
Mx =
(m2*d_2^2 + Izz_1)/d_2^2
                     m2

Vx =
(dDot_2*thetaDot_1*(- m2*d_2^2 + Izz_1))/d_2^2
                          -d_2*m2*thetaDot_1^2

Gx =
0
0
```

Published with MATLAB® R2020b