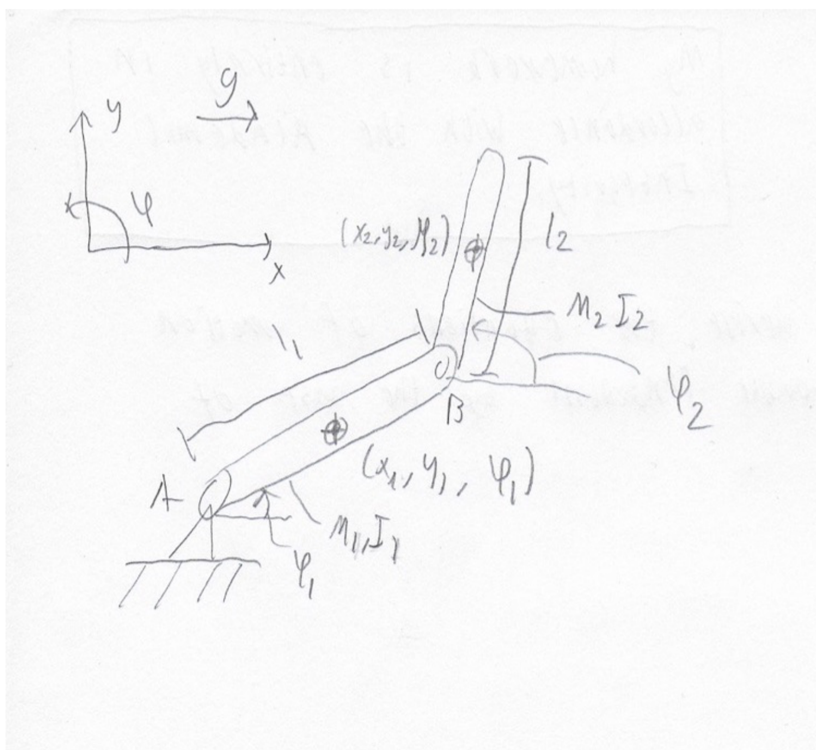
Multibody Dynamics B - Assignment 4

July 4, 2019

a)

In this assignment we are again exploring a double pendulum. We derive the equations of motion using the Lagrange method where we make use of the concept of energy. Using the Lagrange method we derive the potential and the kinetic energy in terms of the generalized coordinates and use that to define our equation of motion. Below we can see the system we are exploring.



Where

$$\rho = 670 kg/m^3 \quad l = 0.6m \quad w = 50mm \quad t = 3mm \quad and \quad g = 9.81 m/s^2$$

Now using the Lagrange method we use this equation to define our equation of motion

$$Q_j = \frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_j}\right) - \frac{\partial T}{\partial q_j} + \frac{\partial V}{\partial q_j} \tag{1}$$

where T is the kinetic energy

$$T = 1/2 * \dot{x}_i * M_{ij} * \dot{x}_j$$

V is the potential energy,

$$-\frac{\partial V}{\partial x_i} + F_i = \frac{d}{dt}\left(\frac{\partial T}{\partial \dot{x_i}}\right)$$

Q_j are the generalized forces and q_j are the generalized coordinates where the CoM coordinates x_i for the rigid bodies can be written as $x_i = x_i(q_j)$. Now our double pendulum system is a 2 DOF system, n=2 and m=2 which gives $n*3 - m*2 = 6 - 4 = 2DOF$. Thus we need to define two generalized coordinates. We choose φ_1 and φ_2, see figure 1, as our generalized coordinates. The choice for these generalized coordinates are chosen for they can describe the configuration of the system completely. The kinematics problem is highly nonlinear and thus the generalized coordinates chosen need to describe the configuration uniquely in all positions, i.e. no two positions can be described with the same angles of φ_1 and φ_2.

b)

Now we need to derive our equation of motion using the generalized coordinates we have chosen as well as using equation (1) above. First we need to calculate the CoM coordinates and velocities for the two bodies to find the kinetic and potential energy of the system. For body 1 we have

$$x_1 = l_1/2 * cos(\varphi_1) \quad y_1 = l_1/2 * sin(\varphi_1)$$

and for body 2 we have

$$x_2 = l_1 * cos(\varphi_1) + l_2/2 * cos(\varphi_2) \quad y_2 = l_1 * sin(\varphi_1) + l_2/2 * sin(\varphi_2)$$

and their velocities, for body 1

$$\dot{x_1} = -1/2 * l_1 * sin(\varphi_1) * \dot{\varphi_1} \quad \dot{y_1} = 1/2 * l_1 * cos(\varphi_1) * \dot{\varphi_1}$$

and for body 2

$$\dot{x_2} = -l_1 * sin(\varphi_1) * \dot{\varphi_1} - 1/2 * l_2 * sin(\varphi_2) * \dot{\varphi_2} \quad \dot{y_2} = l_1 * cos(\varphi_1) * \dot{\varphi_1} 1/2 * l_2 * cos(\varphi_2) * \dot{\varphi_2}$$

This allowed us to calculate the kinetic energy,

$$T = 1/2 * m_1 * (\dot{x_1}^2 + \dot{y_1}^2) + 1/2 * m_2 * (\dot{x_2}^2 + \dot{y_2}^2)$$

and the potential energy

$$V = -m_1 * g * x_1 - m_2 * g * (x_2) \tag{2}$$

Now we have our kinetic and potential energies, allowing us to compute the equation of motion using equation (1). We compute $\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q_j}}\right)$, $\frac{\partial T}{\partial q_j}$ and $\frac{\partial V}{\partial q_j}$ using Matlab. Now we know that $Q_j = 0$ since we assume all conservative forces are assumed collected in V, and thus we hhave our equation of motion which can be expressed in matrix form as

$$\mathbf{A}\begin{pmatrix}\ddot{\varphi_1} \\ \ddot{\varphi_2}\end{pmatrix} = \mathbf{B}$$

where matrix **A** is

$$\mathbf{A} = \begin{pmatrix} (l_1^2 * m_1 + 4 * l_1^2 * m_2)/4 & 2 * l_2 * m_2 * cos(\varphi_1 - \varphi_2) * l_1/4 \\ l_2 * m_2 * 2 * l_1 * cos(\varphi_1 - \varphi_2)/4 & l_2^2 * m_2/4 \end{pmatrix}$$

and **B** is

$$\mathbf{B} = -\frac{1}{4}\begin{pmatrix} l_1 * (2 * l_2 * m_2 * sin(\varphi_1 - \varphi_2) * \dot{\varphi_2}^2 + 2 * g * m_2 * sin(\varphi_1)) \\ l_2 * m_2 * (-2 * l_1 * sin(\varphi_1 - \varphi_2) * \dot{\varphi_1}^2 + 2 * g * sin(\varphi_2)) \end{pmatrix}$$

Now we solve for $\begin{pmatrix} \ddot{\varphi}_1 \\ \ddot{\varphi}_2 \end{pmatrix}$.

C) i)

For initial positions as $\varphi_1 = \varphi_1 = \pi/2$ and initial speeds $\dot{\varphi}_1 = \dot{\varphi}_1 = 0$ we get

$$\begin{pmatrix} \ddot{\varphi}_1 \\ \ddot{\varphi}_2 \end{pmatrix} = \begin{pmatrix} -20.99 & rad/s^2 \\ 6.9483 & rad/s^2 \end{pmatrix}$$

Now we can calculate the accelerations of body and body 2 in CoM coordinates, which gives us

$$\ddot{x}_1 = -1/2 * l_1 * (cos(\varphi_1 1) * \dot{\varphi}_1^2 + sin(\varphi_1) * \ddot{\varphi}_1) \quad \ddot{y}_1 = 1/2 * l_1 * (-sin(\varphi_1 1) * \dot{\varphi}_1^2 + cos(\varphi_1) * \ddot{\varphi}_1)$$

$$\ddot{x}_2 = -l_1 * (cos(\varphi_1) * \dot{\varphi}_1^2 + sin(\varphi_1) * \ddot{\varphi}_1) - l_2/2 * (cos(\varphi_2) * \dot{\varphi}_2^2 + sin(\varphi_2) * \ddot{\varphi}_2) \quad \ddot{y}_2 = -l_1 * (sin(\varphi_1) * \dot{\varphi}_1^2 - cos(\varphi_1) * \ddot{\varphi}_1) + l_2/2 * ($$

This gives us, where $\ddot{x}_j = [\ddot{x}_1 \ddot{y}_1 \ddot{\varphi}_1 \ddot{x}_2 \ddot{y}_2 \ddot{\varphi}_2]$

$$\ddot{x}_j = \begin{pmatrix} 6.2971 & m/s^2 \\ 0 & m/s^2 \\ -20.99 & rad/s^2 \\ 10.5097 & m/s^2 \\ 0 & m/s^2 \\ 6.9483 & rad/s^2 \end{pmatrix}$$

Which confers with what we got in HW1 b).

ii) For initial positions as $\varphi_1 = \varphi_1 = 0$ and initial speeds $\dot{\varphi}_1 = \dot{\varphi}_1 = 0$ we get, using same equations as above

$$\ddot{x}_j = \begin{pmatrix} 0 & m/s^2 \\ 0 & m/s^2 \\ 0 & rad/s^2 \\ 0 & m/s^2 \\ 0 & m/s^2 \\ 0 & rad/s^2 \end{pmatrix}$$

Which compares to what we got in HW1 c).

iii) For initial positions as $\varphi_1 = \varphi_1 = 0$ and initial speeds $\dot{\varphi}_1 = \dot{\varphi}_1 = \pi$ we get, using same equations as above

$$\ddot{x}_j = \begin{pmatrix} -2.9609 & m/s^2 \\ 0 & m/s^2 \\ 0 & rad/s^2 \\ -8.8826 & m/s^2 \\ 0 & m/s^2 \\ 0 & rad/s^2 \end{pmatrix}$$

Which compares to what we got in HW1 d).

d)

Now we have a spring element connected to the pendulum. For a passive element we can rewrite Lagrange equation as

$$Q_j = \frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_j}\right) - \frac{\partial T}{\partial q_j} + \frac{\partial V}{\partial q_j} + C_{l,j} * \sigma_l$$

where $\sigma_l = K_s * Cs$ and

$$C_s = \sqrt{(x_1 + 1/6 * l_1 * cos(\varphi_1) - (-l/2 * l_1))^2 + (y1 + 1/6 * l1 * sin(\varphi_1) - 0)^2} - l_0$$

and $C_{l,j}$ is found using Matlab, by taking the jacobian with regard to $\varphi_1 and \varphi_2$. This is added to the **B** matrix so it becomes

$$\mathbf{B} = -\frac{1}{4}\begin{pmatrix} l_1 * (2 * l_2 * m_2 * sin(\varphi_1 - \varphi_2) * \dot{\varphi_2}^2 + 2 * g * m_2 * sin(\varphi_1)) + C_{l,j} * \sigma_l * 4 \\ l_2 * m_2 * (-2 * l_1 * sin(\varphi_1 - \varphi_2) * \dot{\varphi_1}^2 + 2 * g * sin(\varphi_2)) \end{pmatrix}$$

For initial positions as $\varphi_1 = \varphi_1 = \pi/2$ and initial speeds $\dot{\varphi}_1 = \dot{\varphi}_1 = 0$ we get, using same equations as above for deriving accelerations

$$\ddot{x}_j = \begin{pmatrix} 2.1063 & m/s^2 \\ 0 & m/s^2 \\ -7.021 & rad/s^2 \\ 8.4034 & m/s^2 \\ 0 & m/s^2 \\ -13.9693 & rad/s^2 \end{pmatrix}$$

Which compares to what we got in HW3 a).

e)

Now we have a motor connected in position A, but no spring. This can be solved by adding a constrain in the Lagrange equation. This is of the form

$$Q_i - \frac{\partial V}{\partial q_j} = \frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_j}\right) - \frac{\partial T}{\partial q_j} + C_{k,i}\lambda_k \tag{3}$$

where we can write the generalized inertia forces as

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_j}\right) - \frac{\partial T}{\partial q_j} = \bar{M}_{ij}\ddot{q}_j + h_i(q_j, \dot{q}_j)$$

where

$$h_i(q_j, \dot{q}_j) = \left(\bar{M}_{ij,k} - 1/2 * \bar{M}_{jk,i}\right)\dot{q}_j\dot{q}_k$$

and

$$\bar{M}_{ij} = x_{i,k} M_{ij} x_{j,l}$$

This can be written in matrix form as

$$\begin{pmatrix} \bar{M}_{ij} & C_{k,i} \\ C_{k,j} & 0_{kk} \end{pmatrix}\begin{pmatrix} \ddot{q}_j \\ \lambda_k \end{pmatrix} = \begin{pmatrix} Q_i - \dfrac{\partial V}{\partial q_j} - h_i \\ -C_{k,ij}\dot{q}_i\dot{q}_j \end{pmatrix}$$

Now the motor constraint is $C_1 = \varphi 1 - \omega t$, and we compute $C_{k,i}$, where $C_{k,j} = C_{k,i}^T$, and $-C_{k,ij}\dot{q}_i\dot{q}_j$ using Matlab. Solving this matrix and calculating the CoM accelerations like before we get

$$\ddot{x}_j = \begin{pmatrix} 0 & m/s^2 \\ -11.8435 & m/s^2 \\ 0 & rad/s^2 \\ 7.3575 & m/s^2 \\ -35.5306 & m/s^2 \\ -24.5250 & rad/s^2 \end{pmatrix}$$

and

$$\lambda_k = \begin{pmatrix} -0.2662 & N * m \end{pmatrix}$$

Now the torque that the motor exerts is of course, following the prescribed motion laws;

$$\lambda_5 = \tau = -0.2662 N * m$$

and then the power is

$$P = \tau * \omega = (-0.2662) * (-2 * \pi) = 1.6725 W$$

f)

No solution.

g)

This can be achieved by adding to the system a higher order of DOF. Then you can implement those DOF, extra generalized coordinates, as constraint equations, in A and B, and solve them by using the Lagrange method for adding constraints, that is via the Lagrange multiplier method in the virtual power equation.

Appendix A

Matlab code

```
%% Set up EoM
% Set up variables
syms phi1 phi2
syms phid1 phid2 % for dx/dt I use xd etc
syms phidd1 phidd2
syms l1 l2 m1 m2 g

x=[phi1, phi2];
xd=[phid1, phid2];

x1=1/2*l1*cos(phi1);
y1=1/2*l1*sin(phi1);
x2=l1*cos(phi1)+l2/2*cos(phi2+phi1);
y2=l1*sin(phi1)+l2/2*sin(phi2+phi1);

xd1=-1/2*l1*sin(phi1)*phid1;
yd1=1/2*l1*cos(phi1)*phid1;
xd2=-l1*sin(phi1)*phid1-l2/2*sin(phi2)*(phid2);
yd2=l1*cos(phi1)*phid1+l2/2*cos(phi2)*(phid2);

xdd1=-1/2*l1*(cos(phi1)*phid1^2+sin(phi1)*phidd1);
ydd1=1/2*l1*(-sin(phi1)*phid1^2+cos(phi1)*phidd1);
xdd2=-l1*cos(phi1)*phid1^2-l1*sin(phi1)*phidd1
-l2/2*(cos(phi2)*phid2^2+sin(phi2)*phidd2);
ydd2=-l1*sin(phi1)*phid1^2+l1*cos(phi1)*phidd1
+l2/2*(-sin(phi2)*phid2^2+cos(phi2)*phidd2);

T=1/2*m1*(xd1^2+yd1^2)+1/2*m2*(xd2^2+yd2^2);
V=-m1*g*x1-m2*g*(x2);

Tj=jacobian(T,x);
Tj=simplify(Tj);

Tdj=jacobian(T,xd);
Tdj=simplify(Tdj);

Vj=jacobian(V,x);
Vj=simplify(Vj);

Tdjt=[(l1^2*m1*phidd1)/4+l1^2*m2*phidd1+(l1*l2*m2*phidd2*cos(phi1-phi2)
-l1*l2*m2*phid2*sin(phi1-phi2)*(phid1-phid2))/2,...
    (l2*m2*(l2*phidd2+2*l1*phidd1*cos(phi1-phi2)
    -2*l1*phid1*sin(phi1-phi2)*(phid1-phid2)))/4];

Tdjt= simplify(Tdjt);

Q = Tdjt - Tj + Vj;
Q=simplify(Q);

% Add the virtual power of the spring
Cs=sqrt((2*l1*cos(phi1)/3 + l1/2)^2 + (2*l1/3*sin(phi1))^2)-l0; % Δ l
```

```
Csj=jacobian(Cs,x);

% Constraint for motor
C1 = phi1-omega

Cx = jacobian(C1,x);
Cx = simplify(Cx)

Cd = Cx*xd'; % this is dC/dt=dC/dx*xd

% and next the convective terms d(dC/dt)/dx*xd
C2 = jacobian(Cd,x)*xd';
C2 = simplify(C2)

%% Initilize parameters
rho=670;
l1=0.6;
l2=l1;
Vol=l1*0.05*0.003;
m1=rho*Vol;
m2=m1;
g=9.81;
ks=(15/2)*(m1*g/l1);
l0=2/3*l1;
I1=1/12*m1*l1^2;
I2=1/12*m2*l2^2;
%% 1 b) Both bars vertical up and zero speed
phi1=pi/2;
phi2=pi/2;
phid1=0;
phid2=0;
%% Calculations
A=[(l1^2*m1+4*l1^2*m2)/4, 2*l2*m2*cos(phi1-phi2)*l1/4;
    l2*m2/4*2*l1*cos(phi1-phi2), l2^2*m2/4];
B= -[(l1*(2*l2*m2*sin(phi1-phi2)*phid2^2+2*g*m1*sin(phi1)+4*g*m2*sin(phi1)))/4;
     (l2*m2*(-2*l1*sin(phi1-phi2)*phid1^2+2*g*sin(phi2)))/4];

%  A=[(l1^2*m1+l2^2*m2)/4+l1^2*m2, (l2^2*m2)/4 + l1*l2*m2*cos(phi2)/2;
%      l2^2*m2/4+2*l1*l2*m2*cos(phi2)/4, l2^2*m2/4];
%  B= -[g*m2*((l2*sin(phi1 + phi2))/2 + l1*sin(phi1)) + ...
    (g*l1*m1*sin(phi1))/2 - (l1*l2*m2*phid2^2*sin(phi2))/2 - ...
    l1*l2*m2*phid1*phid2*sin(phi2);
%      (l2*m2*(2*l1*sin(phi2)*phid1^2 + 2*g*sin(phi1 + phi2)))/4];


ac1=inv(A)*B

phidd1=ac1(1)
phidd2=ac1(2)
xdd1=-1/2*l1*(cos(phi1)*phid1^2+sin(phi1)*phidd1)
ydd1=1/2*l1*(-sin(phi1)*phid1^2+cos(phi1)*phidd1)
xdd2=-l1*cos(phi1)*phid1^2-l1*sin(phi1)*phidd1-l2/2*(cos(phi2)*phid2^2+sin(phi2)*phidd2)
ydd2=-l1*sin(phi1)*phid1^2+l1*cos(phi1)*phidd1+l2/2*(-sin(phi2)*phid2^2+cos(phi2)*phidd2)

%% 1 c) Both bars horizontal to the right and zero speed
phi1=0;
```

```
phi2=0;
phid1=0;
phid2=0;
%% Calculations
A=[(l1^2*m1+4*l1^2*m2)/4, 2*l2*m2*cos(phi1-phi2)*l1/4;
    l2*m2/4*2*l1*cos(phi1-phi2), l2^2*m2/4];
B= -[(l1*(2*l2*m2*sin(phi1-phi2)*phid2^2+2*g*m1*sin(phi1)+4*g*m2*sin(phi1)))/4;
    (l2*m2*(-2*l1*sin(phi1-phi2)*phid1^2+2*g*sin(phi2)))/4];


ac2=inv(A)*B

phidd1=ac2(1)
phidd2=ac2(2)
xdd1=-1/2*l1*(cos(phi1)*phid1^2+sin(phi1)*phidd1)
ydd1=1/2*l1*(-sin(phi1)*phid1^2+cos(phi1)*phidd1)
xdd2=-l1*cos(phi1)*phid1^2-l1*sin(phi1)*phidd1-l2/2*(cos(phi2)*phid2^2+sin(phi2)*phidd2)
ydd2=-l1*sin(phi1)*phid1^2+l1*cos(phi1)*phidd1+l2/2*(-sin(phi2)*phid2^2+cos(phi2)*phidd2)

%% 1 d) Both bars horizontal to the right and speed of 30 RPM
phi1=0;
phi2=0;
phid1=30*2*pi/60;
phid2=30*2*pi/60;
%% Calculations
A=[(l1^2*m1+4*l1^2*m2)/4, 2*l2*m2*cos(phi1-phi2)*l1/4;
    l2*m2/4*2*l1*cos(phi1-phi2), l2^2*m2/4];
B= -[(l1*(2*l2*m2*sin(phi1-phi2)*phid2^2+2*g*m1*sin(phi1)+4*g*m2*sin(phi1)))/4;
    (l2*m2*(-2*l1*sin(phi1-phi2)*phid1^2+2*g*sin(phi2)))/4];


ac3=inv(A)*B

phidd1=ac3(1)
phidd2=ac3(2)
xdd1=-1/2*l1*(cos(phi1)*phid1^2+sin(phi1)*phidd1)
ydd1=1/2*l1*(-sin(phi1)*phid1^2+cos(phi1)*phidd1)
xdd2=-l1*cos(phi1)*phid1^2-l1*sin(phi1)*phidd1-l2/2*(cos(phi2)*phid2^2+sin(phi2)*phidd2)
ydd2=-l1*sin(phi1)*phid1^2+l1*cos(phi1)*phidd1+l2/2*(-sin(phi2)*phid2^2+cos(phi2)*phidd2)

%% 3 a) Both bars vertical up and zero speed

phi1=pi/2;
phi2=pi/2;
phid1=0;
phid2=0;

%% Calculations

Cs=sqrt((2*l1*cos(phi1)/3 + l1/2)^2 + (2*l1/3*sin(phi1))^2)-l0;
% New B matrix
A=[(l1^2*m1+4*l1^2*m2)/4, 2*l2*m2*cos(phi1-phi2)*l1/4;
    l2*m2/4*2*l1*cos(phi1-phi2), l2^2*m2/4];
% B= ...
    -[(l1*(2*l2*m2*sin(phi1-phi2)*phid2^2+2*g*m1*sin(phi1)+4*g*m2*sin(phi1)))/4 ...
    - ((4*l1*sin(phi1)*(l1/2 + (2*l1*cos(phi1))/3))/3 - ...
```

```
    (8*l1^2*cos(phi1)*sin(phi1)/9)/(2*((l1/2 + ...
    (2*l1*cos(phi1))/3)^2+(4*l1^2*sin(phi1)^2)/9)^(1/2)))*ks*Cs;
%      (l2*m2*(-2*l1*sin(phi1-phi2)*phid1^2+2*g*sin(phi2)))/4];
B= ...
    -[(l1*(2*l2*m2*sin(phi1-phi2)*phid2^2+2*g*m1*sin(phi1)+4*g*m2*sin(phi1)))/4 ...
    - ((4*l1*sin(phi1)*(l1/2 + (2*l1*cos(phi1))/3))/3 - ...
    (8*l1^2*cos(phi1)*sin(phi1)/9)/(2*((l1/2 + ...
    (2*l1*cos(phi1))/3)^2+(4*l1^2*sin(phi1)^2)/9)^(1/2)))*ks*Cs;
     (l2*m2*(-2*l1*sin(phi1-phi2)*phid1^2+2*g*sin(phi2)))/4];


ac4=inv(A)*B

phidd1=ac4(1)
phidd2=ac4(2)
xdd1=-1/2*l1*(cos(phi1)*phid1^2+sin(phi1)*phidd1)
ydd1=1/2*l1*(-sin(phi1)*phid1^2+cos(phi1)*phidd1)
xdd2=-l1*cos(phi1)*phid1^2-l1*sin(phi1)*phidd1-l2/2*(cos(phi2)*phid2^2+sin(phi2)*phidd2)
ydd2=-l1*sin(phi1)*phid1^2+l1*cos(phi1)*phidd1+l2/2*(-sin(phi2)*phid2^2+cos(phi2)*phidd2)

%% 3 b) both bars vertical up and speed of omega=60 RPM

phi1=pi/2;
phi2=pi/2;
phid1=2*pi;
phid2=2*pi;

%% Calculations
xi=[x1; y1; phi1; x2; y2; phi2];
xi_k=jacobian(xi,x);

m=[m1 m1 I1 m2 m2 I2];
M=diag(m);

M_gen=xi_k'*M*xi_k;

h_i=Tdjt-Tj;


Mat=[M_gen Cx'; Cx zeros(1)];
Y=[-Vj';C2]
Sol=inv(Mat)*Y


phidd1=Sol(1)
phidd2=Sol(2)
xdd1=-1/2*l1*(cos(phi1)*phid1^2+sin(phi1)*phidd1)
ydd1=1/2*l1*(-sin(phi1)*phid1^2+cos(phi1)*phidd1)
xdd2=-l1*cos(phi1)*phid1^2-l1*sin(phi1)*phidd1-l2/2*(cos(phi2)*phid2^2+sin(phi2)*phidd2)
ydd2=-l1*sin(phi1)*phid1^2+l1*cos(phi1)*phidd1+l2/2*(-sin(phi2)*phid2^2+cos(phi2)*phidd2)
```