MATLAB Assignment 5

Due Date: October 20, 2011

<u>Turning in your homework</u>: place all your files in a directory called "lastNameDate"; zip that directory and drop it in the mailbox at Learn@UW.

Problem 1. Consider the Example 3.3.4 at page 68 of the textbook. Generate a pair of input files sliderCrank.acf and sliderCrank.adm that will be used in a future assignment by your simEngine2D to perform 10 seconds worth of Kynematics analysis of the mechanism using a step size of $\Delta t = 0.001$ seconds. Specify in the acf file that the output should be saved at 500 intermediate points. Additionally, consider that a motion is applied to this mechanism as follows: $\phi_1(t) - t^2 = 0$.

A couple of remarks are in order. Consider three bodies for this model. There is no need to define a "ground" body (body 4 in the example 3.3.4). As far as the inertia properties of the bodies are concerned, assume that the mass of body 1 is 2.0 and $Jbar = 0.3$; for body 2 $Mass = 3$ and $Jbar = 0.2$; and for body 3 $Mass = 5$ and $Jbar = 0.5$. Moreover, for the initial configuration of the mechanism at $t = 0$, choose the setup that places the mechanism in an "all stretched out" configuration. That is, using the the dimensions shown in Figure 3.3.6, for body 1 $xZero = 2$ and $yZero = 0$; for body 2 $xZero = 7$ and $yZero = 0$; and for body 3 $xZero = 10$ and $yZero = 0$.

Secondly, please use the following set of constraints to model this mechanism: two absolute constraints for the motion of body 1 with respect to ground (`Constraint-1` and `Constraint-2`, one equation each); one revolute joint between bodies 1 and 2 (`Constraint-3`, note is has two equations); one revolute joint between body 2 and 3 (`Constraint-4`, it has two equations); two absolute constraints to capture the kinematics of body 3 (`Constraint-5` and `Constraint-6`, one equation each - the former is an absolute-y, the later is an absolute angle); and finally, one absolute angle driving constraint, `Constraint-7`: $\phi_1(t) - t^2 = 0$.

Problem 2. Recall the discussion in class about the five stages we have to go through in order to handle a certain physical joint/constraint: (Stage 1) understand what it does; (Stage 2) generate constraint *equations* $\mathbf{\Phi}(\mathbf{q}, t)$ that capture mathematically the behavior of the physical joint; (Stage 3) generate the Jacobian $\mathbf{\Phi}_{\mathbf{q}_i}$ and $\mathbf{\Phi}_{\mathbf{q}_j}$; (Stage 4) generate $\nu(\mathbf{q}, t)$; (Stage 5) generate $\gamma(\dot{\mathbf{q}}, \mathbf{q}, t)$, the right hand side of the acceleration equation.

For this problem you'll have to do two things:

a) Generate a MATLAB function that evaluates $\mathbf{\Phi}(\mathbf{q}_i, t)$, $\mathbf{\Phi}_{\mathbf{q}_i}$, $\nu(\mathbf{q}_i, t)$, and $\gamma(\dot{\mathbf{q}}_i, \mathbf{q}_i, t)$ for the absolute$-x$ constraint discussed in class.

Below I'm providing a possible function implementation. It draws on constraint attributes that you will have parsed (see for instance Problem 3 of MATLAB Assignment 2). Note that you don't have to follow my suggestion below, which while simple is likely suboptimal. It all depends on your data structures and how you choose to store the attributes of the constraint such as xPprimeI,yPprimeI,xPground, etc.

```
%-----------------------------------------------------------------------
function[myPhi, myPhi_q, myNu, myGamma] = absXconstraint(t,q_i,qd_i,...
    xPprimeI,yPprimeI,xPground,cHandle,cDotHandle,cDDotHandle,flag)
% Computes all the information associated with an absolute x constraint.
% The quantities that are computed are based on the entries in
% the array "flag".

myPhi_q = zeros(3,1);

if flag(1,1)==1
    %evaluate Phi (Stage 2)
    myPhi = q_i(1,1) + xPprimeI*cos(q_i(3,1)) - yPprimeI*sin(q_i(3,1)) ...
        - xPground - cHandle(t);
end

if flag(2,1)==1
    %evaluate Phi_q (Stage 3)
    myPhi_q(1,1) = 1;
    myPhi_q(2,1) = 0;
    myPhi_q(3,1) = -xPprimeI*sin(q_i(3,1)) - yPprimeI*cos(q_i(3,1));
end

if flag(3,1)==1
    %evaluate nu (Stage 4)
    myNu = cDotHandle(t);
end

if flag(4,1)==1
    %evaluate gamma (Stage 5)
    myGamma = (xPprimeI*cos(q_i(3,1)) - yPprimeI*sin(q_i(3,1)))*...
        qd_i(3,1)*qd_i(3,1)+ cDDotHandle(t);
end

return;
%-----------------------------------------------------------------------
```

b) Parse the file sliderCrank.adm from Problem 1 and then invoke your function from point a) above to compute $\mathbf{\Phi}(\mathbf{q}_i, t)$, $\mathbf{\Phi}_{\mathbf{q}_i}$, $\nu(\mathbf{q}_i, t)$, and $\gamma(\dot{\mathbf{q}}_i, \mathbf{q}_i, t)$ associated with the `Constraint-1` of Problem 1. Once you compute these quantities, post this information (for Stage 2 through Stage 5) on the class forum to check against results obtained by your colleagues. When computing γ (Stage 5) assume that all entries in `qd_i` are zero.

Problem 3. This problem is very similar to Problem 2, except that it will ask you to deal with an absolute$-y$ constraint. Thus,
a) Generate a MATLAB function that evaluates $\mathbf{\Phi}(\mathbf{q}_i, t)$, $\mathbf{\Phi}_{\mathbf{q}_i}$, $\nu(\mathbf{q}_i, t)$, and $\gamma(\dot{\mathbf{q}}_i, \mathbf{q}_i, t)$ for the absolute$-y$ constraint discussed in class.
b) Parse the file sliderCrank.adm from Problem 1 and then invoke your function from point a) above to compute $\mathbf{\Phi}(\mathbf{q}_i, t)$, $\mathbf{\Phi}_{\mathbf{q}_i}$, $\nu(\mathbf{q}_i, t)$, and $\gamma(\dot{\mathbf{q}}_i, \mathbf{q}_i, t)$ associated with the `Constraint-2` of Problem 1. Once you compute these quantities, post this information (for Stage 2 through Stage 5) on the class forum to check against results obtained by your colleagues. When computing γ (Stage 5) assume that all entries in `qd_i` are zero.
c) Parse the file sliderCrank.adm from Problem 1 and then invoke your function from point a) above to compute $\mathbf{\Phi}(\mathbf{q}_i, t)$, $\mathbf{\Phi}_{\mathbf{q}_i}$, $\nu(\mathbf{q}_i, t)$, and $\gamma(\dot{\mathbf{q}}_i, \mathbf{q}_i, t)$ associated with the `Constraint-5` of Problem 1. Once you compute these quantities, post this information (for Stage 2 through Stage 5) on the class forum to check against results obtained by your colleagues.

Problem 4. This problem is very similar to Problem 2, except that it will ask you to deal with an absolute$-\phi$ constraint. Thus,
a) Generate a MATLAB function that evaluates $\mathbf{\Phi}(\mathbf{q}_i, t)$, $\mathbf{\Phi}_{\mathbf{q}_i}$, $\nu(\mathbf{q}_i, t)$, and $\gamma(\dot{\mathbf{q}}_i, \mathbf{q}_i, t)$ for the absolute$-phi$ constraint discussed in class.
b) Parse the file sliderCrank.adm from Problem 1 and then invoke your function from point a) above to compute $\mathbf{\Phi}(\mathbf{q}_i, t)$, $\mathbf{\Phi}_{\mathbf{q}_i}$, $\nu(\mathbf{q}_i, t)$, and $\gamma(\dot{\mathbf{q}}_i, \mathbf{q}_i, t)$ associated with the `Constraint-6` of Problem 1. Once you compute these quantities, post this information (for Stage 2 through Stage 5) on the class forum to check against results obtained by your colleagues. When computing γ (Stage 5) assume that all entries in `qd_i` are zero.
c) Parse the file sliderCrank.adm from Problem 1 and then invoke your function from point a) above to compute $\mathbf{\Phi}(\mathbf{q}_i, t)$, $\mathbf{\Phi}_{\mathbf{q}_i}$, $\nu(\mathbf{q}_i, t)$, and $\gamma(\dot{\mathbf{q}}_i, \mathbf{q}_i, t)$ associated with the `Constraint-7` of Problem 1. Once you compute these quantities, post this information (for Stage 2 through Stage 5) on the class forum to check against results obtained by your colleagues. **NOTE:** Please keep in mind that the time for which you evaluate the motion is $t = 0$.

Problem 5. This problem is very similar to Problem 2, except that it will ask you to deal with a revolute joint. Thus,
a) Generate a MATLAB function that evaluates $\mathbf{\Phi}(\mathbf{q}_i, \mathbf{q}_j, t)$, $\mathbf{\Phi}_{\mathbf{q}_i}$, $\mathbf{\Phi}_{\mathbf{q}_j}$, $\nu(\mathbf{q}_i, \mathbf{q}_j, t)$, and $\gamma(\dot{\mathbf{q}}_i, \mathbf{q}_i, \dot{\mathbf{q}}_j, \mathbf{q}_j, t)$ for the revolute joint discussed in class.
b) Parse the file sliderCrank.adm from Problem 1 and then invoke your function from point a) above to compute the information in Stages 2 through 5 associated with the `Constraint-3` of Problem 1.

Once you compute these quantities, post this information (for Stage 2 through Stage 5) on the class forum to check against results obtained by your colleagues. When computing γ (Stage 5) assume that all entries in `qd_i` and `qd_j` are zero.

c) Parse the file sliderCrank.adm from Problem 1 and then invoke your function from point a) above to compute the information in Stages 2 through 5 associated with the `Constraint-4` of Problem 1. Once you compute these quantities, post this information (for Stage 2 through Stage 5) on the class forum to check against results obtained by your colleagues. When computing γ (Stage 5) assume that all entries in `qd_i` and `qd_j` are zero.