Project 1 – ME451

Fall 2014

For Project 1 we will revisit the excavator example described in the textbook in Example 3.5.3.

a) Generate a pair of acf/adm files, excavator.acf/excavator.adm, that one would use to carry out Kinematics Analysis of the excavator in Example 3.5.3. The length of the simulation is 2 seconds, the time step used should be 0.05 seconds, and the number of output points should be 40; i.e., output data at each time step.
b) Use your simEngine2D and the two files at a) above to carry out a Kinematics Analysis of the mechanism. In four figures, plot the velocity and acceleration, both in the x and y directions, of the point on body 2 where the actuator with $C_{12}(t)$ is attached. Call this point Q.
c) Generate an ADAMS model of this mechanism, and confirm that the results obtained with your simEngine2D are correct. To this end, generate four ADAMS plots that show the velocity and acceleration of point Q.

What should you turn in?

In one zipped file,

- Turn in the acf/adm files that address a) above (excavator.acf/excavator.adm), the file containing the simulation results excavator.res, and all your simEngine2D MATLAB files that we need in order to run the Kinematics Analysis of the given mechanism. Please note that your code will not be debugged if issuing in MATLAB the command

  `>> simEngine2D('excavator')`

  does not run the expected analysis.
- Provide the ADAMS model that was developed for c) above.
- Provide eight plots that were generated based on your MATLAB and ADAMS simulations.

Final Remarks:

- Note that simEngine2D is expected to work with three degrees of freedom for each body in the model. In other words, we'll not use the simplified representation used in the textbook to characterize the kinematics of the excavator.
- Teams of up to two students are allowed to work on this project. The solution should reflect the team's or the person's understanding of the subject, as well as your code design choices.
- It is ok to post questions and exchange ideas related to the Project on the class Forum.
- Use the solution provided in the textbook, say for $\dot{\varphi}_1$ and $\dot{\varphi}_2$, to debug your code. For instance, right after the position analysis, you should verify whether your solution and the textbook solution coincide. You can do this by plotting the trajectories and comparing the plots.
- The zip file should be uploaded at Learn@UW by November 11 at 11:59 PM.
- Post any follow up or clarification questions on the class Forum.