

Take Home Exam (40% of Exam 1 Score)

(Based on the simple pendulum of the in-class part of the exam)

Due: Tuesday, November 9, 11:59 PM (to keep this fair, no late submission will be accepted)

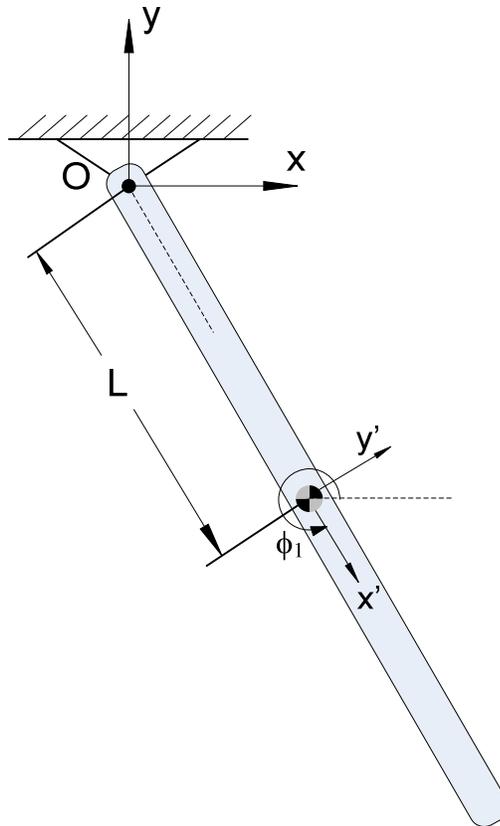


Figure 1. Simple pendulum

For the simple pendulum in Figure 1 consider that a motion is prescribed as $\phi_1 = \frac{3\pi}{2} + \sin(4t)$.

The center of mass (CM) is located in the middle of the symmetric link, as shown in the figure. For this problem, consider the following set of generalized coordinates: $\mathbf{q} = [x_1, y_1, \phi_1]^T$, where x_1, y_1 are the coordinates of the CM of body 1 in the global fixed reference frame OXY , and the angle ϕ_1 provides the orientation of the body with respect to the same global fixed reference frame. The length $L = 1$, all units are SI.

(continued, next page)

- a) [10 points] Write a MATLAB code, called `simEngine2D` that parses two files, `pendulum.acf` and `pendulum.adm`, emailed to you separately. They specify the type of analysis that needs to be performed and describe the simple pendulum model that you have to handle.
- b) [15 points] Write additional MATLAB code that augments `simEngine2D` so that it is capable of carrying out Position Analysis of an arbitrary 2D mechanism. In particular, here you'll use the simple pendulum to demonstrate your `simEngine2D`. To demonstrate the correctness of your answer, generate two separate plots that show $x_1(t)$ and $\phi_1(t)$.
NOTE 1: implement and use the Newton-Raphson method discussed in class for the solution of the nonlinear system. NOTE 2: if you can't get the MATLAB code to parse the acf and adm file, simply hardcode the required values in your MATLAB code and proceed with doing b) through d).
- c) [5 points] Augment the MATLAB code `simEngine2D` you developed at b) with functionality that allows it to perform a Velocity Analysis of any mechanism. In particular, here you'll use the simple pendulum to demonstrate your `simEngine2D`. To demonstrate the correctness of your results, plot $\dot{x}_1(t)$ and $\dot{\phi}_1(t)$. NOTE: You will have to solve a linear system to find both velocities and accelerations. If you want to solve a linear system $\mathbf{Ax} = \mathbf{b}$ in MATLAB once you have the \mathbf{A} and \mathbf{b} quantities, you can easily find the solution \mathbf{x} by just typing "`x = A\b`".
- d) [10 points] Augment the `simEngine2D` code you implemented at point c) with functionality that allows you to carry out Acceleration Analysis of any mechanism. Plot $\ddot{x}_1(t)$ and $\ddot{\phi}_1(t)$ to prove the correctness of your results.

NOTE: the outcome of the simulation using `simEngine2D` should be a file called `pendulum.res`, which, based on information specified in the acf file, stores position, velocity and acceleration information at each time step of the simulation for each body of interest. At each time step, this file stores first all positions of interest, followed by all velocities of interest, and finally all accelerations of interest.

What you should turn in:

- In one zipped file, "takeHomeExamYourFirstName.zip", turn in **all** the MATLAB files necessary to answer questions a) through d) above. Make sure that there is a file called `simEngine2D.m` which can be called from the MATLAB command line and generates the `pendulum.res`. This directory should also include the files `pendulum.acf` and `pendulum.adm` that you used to generate `pendulum.res`.
- Include in your directory the plots that you obtained at b) through d) above (in png format or similar)

PLEASE NOTE: Keep all your files in a directory `takeHomeExamYourFirstName`. It is this directory that you should zip and email the grader (jcmadsen@wisc.edu). The grader will unzip the file, start MATLAB from the directory `takeHomeExamYourFirstName`, and type

```
>> simEngine2D pendulum
```

The TA will not debug your code if when issuing this command he does not get the results file `pendulum.res`. Email the TA or come to talk to me during office hours for MATLAB related questions.