שיטות נומריות למהנדסים (019003)

Homework 4

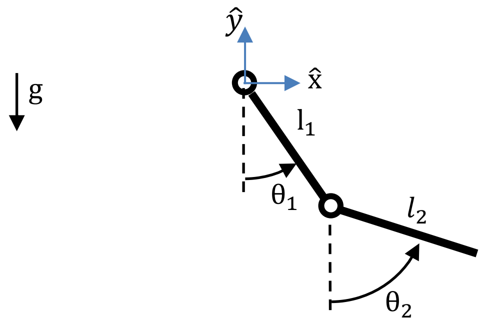


Figure 1 Schematic model of a double pendulum

The dynamics of the double pendulum is modeled by the following set of coupled ODE:

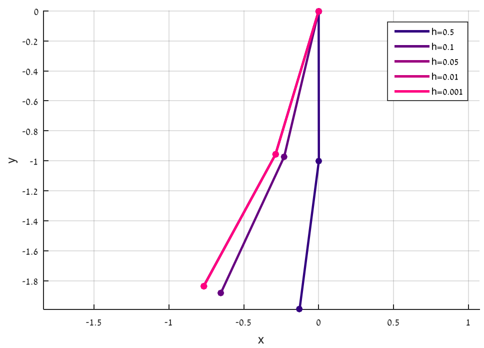
$$(1)\begin{cases}(m_1+m_2)l_1\ddot{\theta}_1+m_2l_2\ddot{\theta}_2\cos(\theta_2-\theta_1)=m_2l_2\dot{\theta}_2^2\sin(\theta_2-\theta_1)-(m_1+m_2)g\sin\theta_1\\ l_2\ddot{\theta}_2+l_1\ddot{\theta}_1\cos(\theta_2-\theta_1)=-l_1\dot{\theta}_1^2\sin(\theta_2-\theta_1)-g\sin\theta_2\end{cases}$$

The masses and the lengths of the pendulum links (“ODE parameters”) are:
$m_1=1,\ m_2=0.5,\ l_1=1,\ l_2=1,\ g=1$. The thickness of the links is assumed to be zero.

1. Implement your own ODE solver.
   a. Write your own Matlab function to solve an ODE using the Runge-Kutta 4 method, make sure you use valid coefficients. The function would follow the form,
      [t,y] = MY_RK4(fun_handle, step_size, time_span, initial_value)
      Inputs:
      fun_handle – a handle to a 1st order system of odes
      step_size – desired step size
      time_span – a two elements row vector of integration start and end times
      initial_value – a column vector of initial function values
      Outputs:
      t – column vector of solution time locations
      y – column vector of solution function values
   b. Perform order reduction of equation (1) to a system of first order ODEs. You may use Matlab’s symbolic toolbox. Write a Matlab function implementation of the ODE set. The ODE parameters should be hard coded in the function implementation. The function should follow the form:
      dy = My_DoublePendulum(t,y)
      Inputs:
      t – scalar, time
      y – column vector, variables values
      Outputs:

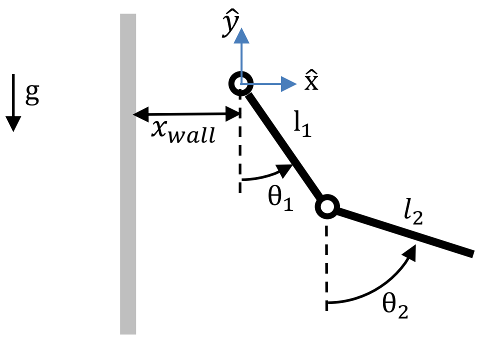
dy – column vector, variables derivatives

2. The pendulum is launched from the initial position $[\theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2] = \left[\frac{\pi}{4}, 1, 0, 0\right]$. You may not use MATLAB's built in ODE solvers (ode45 etc.).
   a. Solve the ODE problem for different step sizes *h=[0.01, 0.05, 0.1, 0.5,1]*. Draw, on the same plot, the two links at t=10 for the different step sizes, similar to the example below. Make sure the axes are equally scaled (use the command *axis equal*).



   b. Discuss the simulation results.
   c. Bonus 5 points: Make a 10 seconds video animation of the pendulum motion for one of the step sizes and add it to your submission.

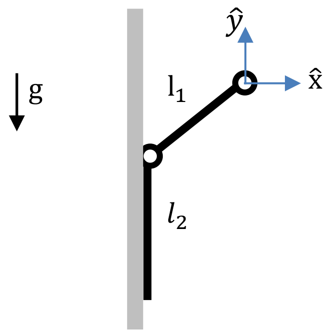
3. We now add a wall to our simulation



The wall's x position is x_{wall} = -0.5. If a link collides with the wall the simulation should stop.

   a. Use the function *MY_RK4* and add step halving and event detection (bisection) to find the pendulum's state as the pendulum horizontal distance from the wall is less than 10^{-8}. Save the new function as *MY_RK4_event.*
   b. Plot the x location of the links' edges vs time during the motion. Use dots, '.', to represent the edges with different colors for each edge. Discuss the results.

4. Now we want the pendulum's second link to collide flat with the wall (meaning both of its ends will collide with the wall simultaneously) see figure bellow. You will need to find one initial position of the pendulum $[\theta_{01}, \theta_{02}, 0, 0]$ causing such collision.



   a. Find the values for the angle θ_1 at the collision.
   b. Explain the shooting method in your own words.
   c. Explain the difficulties in using the shooting method in the case of a 4th order differential equation.
   d. Explain the difficulties in using the shooting method in the case of the requested collision, regardless of the difficulty in section 3.

5. Shooting method, implement using *fzero*, or write your root finding function.
   a. Find an initial state $[\theta_{01}, \theta_{02}, 0, 0]$ so that the pendulum's second link will collide flat with the wall.
   b. Find an initial state $[1, \theta_{0top}, 0,0]$ that will cause the top point of the second link to hit the wall.
   c. Find an initial state $[1, \theta_{0bot}, 0,0]$ that will cause the bottom point of the second link to hit the wall.
   d. Discuss the result. are the solutions unique?