Numerical Methods in Engineering and Applied Science. Assignment 4.

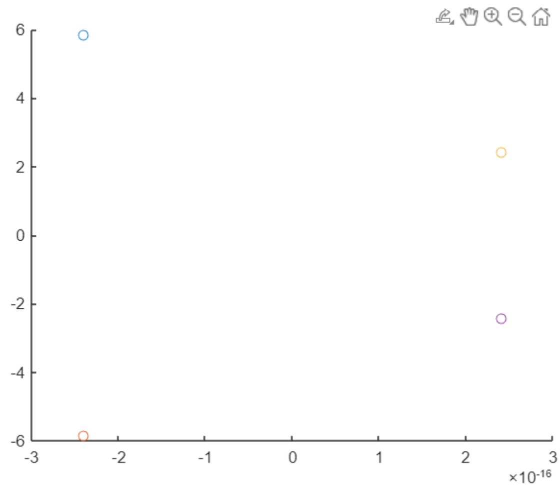
(1). Linearize the problem, analyze the eigenvalues of the linearized r.h.s. operator concerning the parameters of the physical model and properties of numerical schemes.
Suppose θ are small, then obtain the linear equation.

$$U' = AU$$

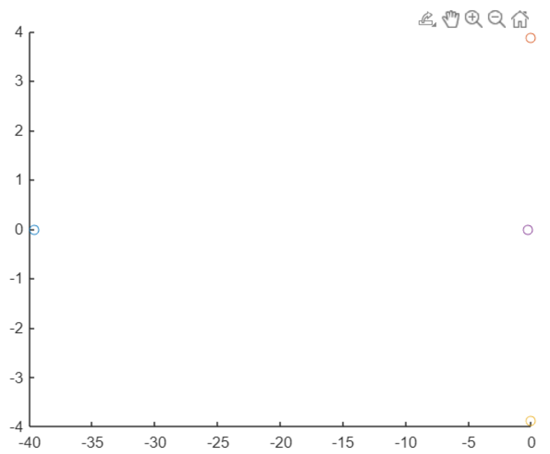
$$U = [\theta_1, \theta_2, \omega_1, \omega_2,]^T$$

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\dfrac{g(m_1+m_2)}{L_1 m_1} & \dfrac{g m_2}{L_1 m_1} & -c & 0 \\ \dfrac{g(m_1+m_2)}{L_2 m_1} & -\dfrac{g(m_1+m_2)}{L_2 m_1} & 0 & 0 \end{bmatrix}$$

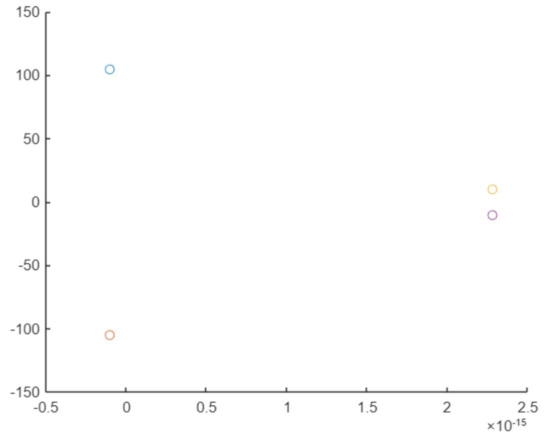
Suppose $c = 0$, damping factor is 0. Then we can see that $Re(\lambda) \approx 0$
It means the Leapfrog method is the best, considering its stability region and that it doesn't smooth small perturbations.



Let's increase c up to let's say 40. You can see one Real $\lambda = -40$. Euler(implicit), Trapezoidal (implicit), Adams schemes are suitable. Euler(explicit) can't hold right 3 e-values in a circle, so Euler(explicit) is not suitable. BDF can contain the e-vals so it is not suitable. And we can use some Runge-Kutta schemes.



For cases when g is large and L_1, L_2 are small there are large imaginary e_vals appear. There are hard to choose enough small h to contain e-vals into Adams or RK areas of stability.



We can't get cases when $Re(\lambda) > 0$ enough, because it would be the case of exponential growth of values when they must decay. So, Euler(implicit) and Trapezoidal are good in all initial conditions.

(2). Implement a numerical solution of the nonlinear or partly linearized problem using a fully implicit scheme or a 2nd order splitting scheme or the method of integrating factors. Motivate your choice of the numerical schemes with a suitable choice of the physical parameters (m1, m2, L1, L2, c).

I will use the first-order Euler implicit. Because I hope the real part of e-vals must be less than 0. Next u_{n+1} are founded with linearization around u_n because it makes possible to use any angle instead of linearization of small angles.

Scheme:

$$u' = F(u_{n+1})$$

$$u' = F(u_n) + \frac{\partial F}{\partial u}(u_n) * (u_{n+1} - u_n)$$

$$\dots$$

$$(u_{n+1} - u_n) = E = \left(I - \frac{\partial F}{\partial u}(u_n)\right)^{-1} hF(u_n)$$

$$u_{n+1} = u_n + E$$

Almost Newton-Raphson but with 1 iteration.
I've got strange e-vals.
But the most e-vals are still on the left side of the picture.
Anyway, it converges. I used any (m1, m2, L1, L2, c) still good.
(you can see movie in file)