Yevonnael Andrew

Assignment 2 of 2 – LUT University

Github URL: https://github.com/yevonnaelandrew/MATLAB_Assignment/tree/main/Code

The function of the system is

$$f(x) = \begin{bmatrix} a\cos\emptyset + b\cos\theta - d \\ a\sin\emptyset - b\sin\theta \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

We want to solve for $f(x,t) = 0$ for $x = \begin{bmatrix} \theta \\ d \end{bmatrix}$ for given $\emptyset$. To solve it, we will use Newton-Raphson's method. First, we will find the Jacobian of $f(x,t)$:

$$J = \frac{\partial f(x,t)}{\partial x} = \begin{bmatrix} -b\sin\theta & -1 \\ -b\cos\theta & 0 \end{bmatrix}$$

The given $\emptyset$ are the function of time where $\emptyset = \frac{\pi}{6} + \omega t$ where the time will be defined by an equally distributed array `t=linspace(0,1,101)`. During the iteration in each timestep, we will also solve $\dot{f}(x,t) = 0$ for $\dot{x} = \begin{bmatrix} \dot{\theta} \\ \dot{d} \end{bmatrix}$ The time derivative of the system function is:

$$\dot{f}(x,t) = \begin{bmatrix} -a\dot{\emptyset}\sin\emptyset + b\dot{\emptyset}\sin\theta - \dot{d} \\ a\dot{\emptyset}\cos\emptyset - b\dot{\emptyset}\cos\theta \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

The angular velocity of the crank is:

$$\omega = \dot{\emptyset} = -1\frac{rad}{s}$$

The Jacobian of $\dot{f}(x,t)$:

$$J = \frac{\partial \dot{f}(x,t)}{\partial x} = \begin{bmatrix} -b\sin\theta & -1 \\ -b\cos\theta & 0 \end{bmatrix}$$

In this system, the Jacobian of function and its time derivative is the same.

After all the mathematical formula is found, we will translate them into MATLAB code. For $f(x,t)$ I use the code presented in the book (no modification). For $\dot{f}(x,t)$, the code is modified a little bit to carry the information of θ that are needed for the calculation.

In each iteration of the timestep, the function will first find theta and distance. The found theta then will be used to find theta velocity and point velocity. We will store theta, distance,

theta velocity, and point velocity into an empty array at each iteration. After the iteration is finished, we will plot the result as follows.