EE208 EXPERIMENT 8

STABILITY AND INSTABILITY OF NONLINEAR SYSTEMS

Group no. 9
Aryan Bansal – 2020EEB1162
Hiya Kwatra – 2020EEB1173
Ishaan Sethi – 2020EEB1174

OBJECTIVE

To examine stability features of a given nonlinear system by observing movement of eigenvalues across the s-plane.

SYSTEM

The following differential equations represent a simplified model of an overhead crane:

$$[m_L + m_C] \cdot \ddot{x}_1(t) + m_L l \cdot \left[\ddot{x}_3(t) \cdot \cos x_3(t) - \dot{x}_3^2(t) \cdot \sin x_3(t)\right] = u(t)$$

$$m_L \cdot [\ddot{x}_1(t) \cdot \cos x_3(t) + l \cdot \ddot{x}_3(t)] = -m_L g \cdot \sin x_3(t)$$

The model will involve certain parameters that are constant, some that are constant but adjustable, and some that are constant and arbitrary.

The parameters are –

Parameters	Description	Value	Remarks
m_C	Mass of trolley	10 kg	Constant
m_L	Mass of hook and load	10 kg (hook) + load from 0 kg to several hundred kg	Constant for a particular crane operation
l	Length of rope	1 m or longer	Constant for a particular crane operation
g	Acc. due to gravity	9.8 m/s^2	Constant

Variables for the problem include –

Variables	Description	Type	Units
$\boldsymbol{u}$	Force applied to the trolley	Input	N
$\boldsymbol{y = x_1 + l * sin(x_3)}$	Position of load	Output	m
$\boldsymbol{x_1}$	Position of trolley	State	m
$\boldsymbol{x_2}$	Speed of trolley	State	m/s
$\boldsymbol{x_3}$	Rope angle	State	rad
$\boldsymbol{x_4}$	Angular speed of rope	State	rad/s

TASKS

- Find system equilibrium points.
- Examine the local and global stability/instability of equilibria by tracking the movement of eigenvalues as parameters and variables assume different values and ranges.
- The stability features as above should be examined for different values assigned to the parameters $\boldsymbol{m_L}$ and $\boldsymbol{l}$**,** since these can be "set" or "made to assume" different values as described above.

OVERVIEW

1. We begin by converting the pair of coupled nonlinear DEs into a system of first order DEs, i.e., the $\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, u)$ form.
2. Equilibrium points are found by setting $\dot{\boldsymbol{x}} = \boldsymbol{0}$.
3. The linearized system matrices (A, B, C, D) can be found by evaluating various the respective Jacobians at equilibrium points.
4. Eigenvalues can be obtained as solutions of $det(\mathrm{sI}_3 - \mathrm{A}) = 0$.
5. Stability of equilibrium points can be judged using the eigenvalues with some constraints.
6. The effect of variation of parameters m_L and l on eigenvalues and stability is examined.

OBSERVATIONS

SYSTEM OF 1ST ORDER EQUATIONS

We are given:

$$[m_L + m_C] \cdot \ddot{x}_1(t) + m_L l \cdot \left[\ddot{x}_3(t) \cdot \cos x_3(t) - \dot{x}_3^2(t) \cdot \sin x_3(t)\right] = u(t)$$

$$m_L \cdot [\ddot{x}_1(t) \cdot \cos x_3(t) + l \cdot \ddot{x}_3(t)] = -m_L g \cdot \sin x_3(t)$$

Judging from the equations, x_3, which is the rope angle, is measured counter-clockwise from the $-\hat{\jmath}$ direction.

We know that by definition of the states:

$$\dot{x_1} = x_2, \text{ and}$$
$$\dot{x_3} = x_4$$

Using this knowledge, we can convert our given pair of coupled ODEs into a system of four 1st order nonlinear ODEs by substitution and matrix inversion to obtain $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u})$–

F =

$$\begin{pmatrix} x_2 \\ \dfrac{l\, m_L \sin(x_3)\, x_4{}^2 + u + g\, m_L \cos(x_3) \sin(x_3)}{\sigma_1} \\ x_4 \\ -\dfrac{l\, m_L \cos(x_3) \sin(x_3)\, x_4{}^2 + u \cos(x_3) + g\, m_C \sin(x_3) + g\, m_L \sin(x_3)}{l\, \sigma_1} \end{pmatrix}$$

where

$$\sigma_1 = -m_L \cos(x_3)^2 + m_C + m_L$$

FINDING EQUILIBRIA

We put $\boldsymbol{f}(\boldsymbol{x}, u) = \mathbf{0}.$

This gives us –

1. $x_2 = 0$
2. $x_4 = 0$
3. $u + gm_L\, cos(x_3)\, sin(x_3) = 0$
4. $u\, cos(x_3) + g(m_L + m_C)\, sin(x_3) = 0$

Putting (3) in (4), we obtain –

$$-gm_L\, cos^2(x_3)\, sin(x_3) + g(m_L + m_C)\, sin(x_3) = 0$$

$$\Rightarrow m_L\, sin^3(x_3) + m_C\, sin(x_3) = 0$$

$$\Rightarrow (m_L\ \ sin^2(x_3) + m_C)\ \times\ sin(x_3) = 0$$

Now, $m_L\, sin^2(x_3) + m_C \ = \ 0 \ \Rightarrow \ sin^2(x_3) \ = \ \frac{-m_C}{m_L} \ < \ \mathbf{0}.$

This is not possible, thus $\sin(x_3) = 0$.

$$\therefore \boldsymbol{x_3} = \boldsymbol{n\pi}, n \in \mathbb{Z}$$

We see that x_1 does not show up in any of these equations. Hence $\dot{x_1} = 0 \Rightarrow$ $\boldsymbol{x_1}$ **= free variable**.

Also, for the input –

$$u + gm_L\, cos(x_3)\, sin(x_3) = 0 \ \Rightarrow u = \ 0$$

Thus, our equilibria are –

1. x_1= free variable, does not show up in dynamic equations
2. $x_2 = 0$
3. $x_3 = n\pi, n \in \mathbb{Z}$
4. $x_4 = 0$
5. $u = 0$

INTERPRETATION OF EQUILIBRIUM POINTS

1. $x_2 = 0$ and $x_4 = 0$ means that the system is at rest.
2. $u = 0$ means there is no force being applied on the crane.
3. $x_3 = n$ implies two possible positions - 0° and 180°.
4. x_1 being a free variable means that the equilibrium does not depend on the crane's position at all.

JACOBIAN MATRICES AND LINEARIZATION

We will use the `jacobian` function in MATLAB to obtain the Jacobian matrices, and the `subs` function to evaluate them at the equilibrium points.

Since x_1 has no effect on the equilibrium and does not show up in the dynamic equations, we may set it to 0 without losing any generality.

Thus, we obtain two equilibrium positions –

$$\begin{aligned} x_1 &= 0 \\ x_2 &= 0 \\ x_3 &= 0 \\ x_4 &= 0 \\ u &= 0 \end{aligned} \qquad \begin{aligned} x_1 &= 0 \\ x_2 &= 0 \\ x_3 &= \pi \\ x_4 &= 0 \\ u &= 0 \end{aligned}$$

With MATLAB, we obtain the following "A" matrices valid around respective equilibria –

A1 =

$$\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{49\, m_L}{50} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{\frac{49\, m_L}{5} + 98}{10\, l} & 0 \end{pmatrix}$$

A2 =

$$\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{49\, m_L}{50} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{\frac{49\, m_L}{5} + 98}{10\, l} & 0 \end{pmatrix}$$

EIGENVALUES

Using the `eig` command, we can obtain eigenvalues of A_1 and A_2 in order to judge their stability.

EQUILIBRIUM 1

The eigenvalues of A_1 are –

$$\begin{pmatrix} 0 \\ 0 \\ -0.98995\sqrt{-\frac{1.0\ (m_L + 10.0)}{l}} \\ 0.98995\sqrt{-\frac{1.0\ (m_L + 10.0)}{l}} \end{pmatrix}$$

There are 2 poles at the origin, and two conjugate imaginary poles.

EQUILIBRIUM 2

The eigenvalues of A_2 are –

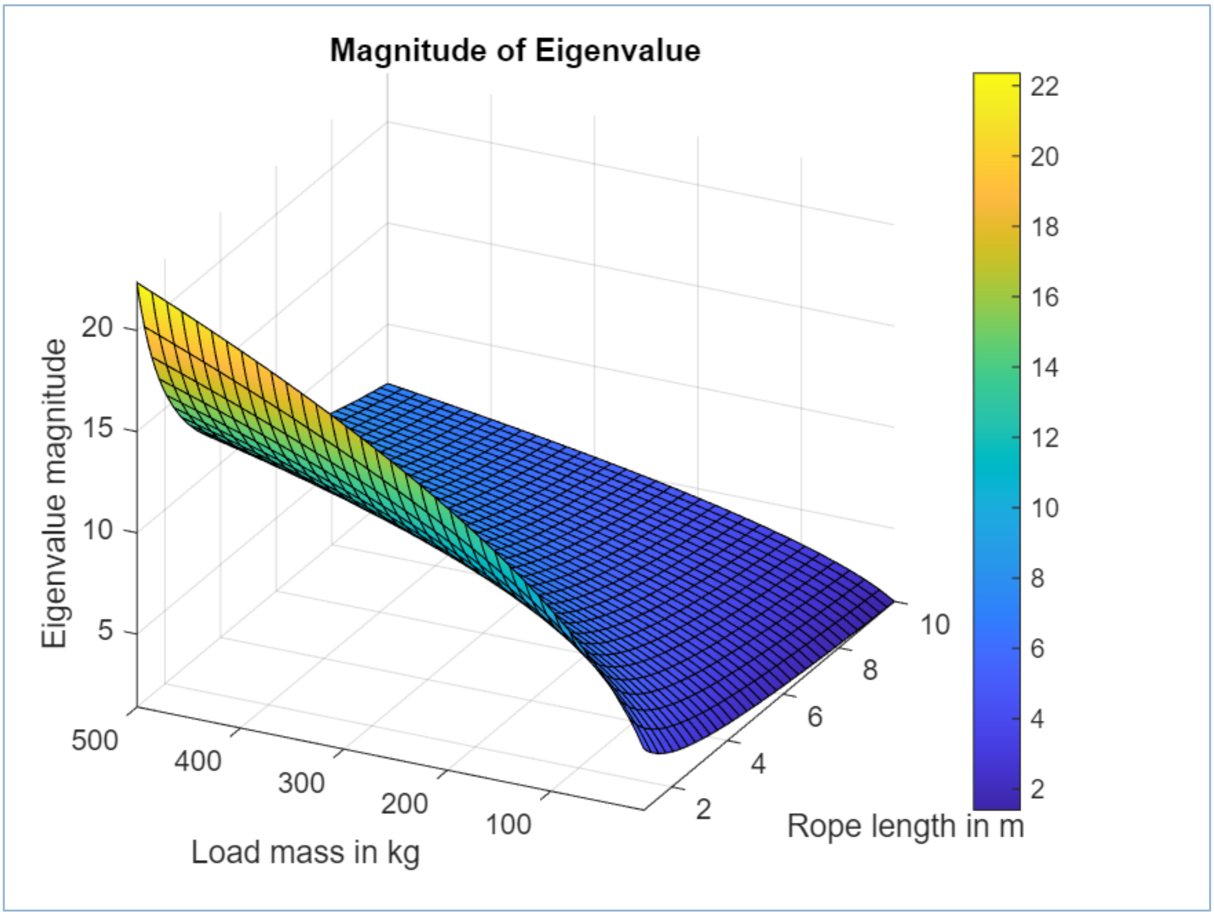
$$\begin{pmatrix} 0 \\ 0 \\ -0.98995\sqrt{\frac{m_L + 10.0}{l}} \\ 0.98995\sqrt{\frac{m_L + 10.0}{l}} \end{pmatrix}$$

There are 2 poles at the origin, one stable real pole, and one unstable real pole.

VARIATION OF PARAMETERS

We observe the change in eigenvalues as load mass and cable length is changed.

- We shall plot the eigenvalue $Z = 0.98995\sqrt{\frac{m_L+10}{l}}$, since the other non-zero eigenvalues have the same magnitude.
- The load mass is varied from 10 kg (incl. mass of hook) to 500 kg.
- The length of the rope is varied from 1m to 10m.



- Increased load mass increases eigenvalue magnitude – higher frequency poles (real or imaginary depending on equilibrium).
- Increased rope length decreases eigenvalue magnitude.
- These observations can be understood by simply observing the form of the eigenvalue as well - $\sqrt{\frac{x}{y}}$.

DISCUSSIONS

EQUILIBRIA

There are technically an infinite number of equilibria for this system, however they eventually boil down to only two –

1. Stationary Crane, Crane rod in vertically downward position ($x_3 = 0$)
2. Stationary Crane, Crane rod in vertically upward position ($x_3 = \pi$)

The first equilibrium is more relevant for the overhead crane problem, while the second equilibrium corresponds to the “inverted pendulum on a cart” problem.

The state x_1 has no effect on the nature of the equilibrium, it is a free variable. We can thus set it equal to 0 without losing any information about the equilibria.

LINEARIZATION

Our task is to obtain system equilibria and examine the local and global stability/instability of equilibria by tracking the movement of eigenvalues in the s-plane as parameters are varied.

The presence of “eigenvalues” in the problem statement implies linearization about equilibrium points. This can be done by evaluating the relevant Jacobians at equilibria to obtain a first order approximation.

We can thus obtain a linear state-space realization around the equilibrium points, and hence find eigenvalues to examine stability.

EIGENVALUES

Using the `eig` command we found the eigenvalues corresponding to the two equilibria.

EQB. 1

2 poles at the origin, a pair of conjugate imaginary poles.

Due to the two poles at the origin, this equilibrium should be *unstable*, since the geometric multiplicity (=1) of the 0 eigenvalue is not equal its algebraic multiplicity (=2).

Had friction or damping been included in the model, it is possible that the eigenvalues could've had negative real parts, making the equilibrium locally stable.

EQB. 2

2 poles at the origin, one -ve real pole, one +ve real pole.

This equilibrium point should also be *unstable* due to the presence of an eigenvalue with positive real part.

The existence of eigenvalues with zero real part makes this analysis inconclusive for the nonlinear case, as explained below.

MOVEMENT IN S-PLANE

We have seen the effect of changing m_L and l on the eigenvalues –

Parameter variation	Equilibrium 1	Equilibrium 2
Increasing m_L	Non-zero eigenvalues move away from the origin towards $\pm\infty$ along the real axis, increasing in magnitude.	Non-zero eigenvalues move away from the origin towards $\pm j\infty$ along the imaginary axis, increasing in magnitude.
Increasing l	Non-zero eigenvalues move towards the origin along the real axis, decreasing in magnitude.	Non-zero eigenvalues move towards the origin along the imaginary axis, decreasing in magnitude.

Although our task was to judge the change in stability as eigenvalues move in the s-plane, it turns out that variation of m_L and l, while affecting the magnitudes of non-zero eigenvalues, does not affect stability/instability of the equilibrium points, since the non-zero eigenvalues stay on the real/imaginary axes and the zero eigenvalues are unaffected.

STABILITY

The linearization technique used to judge stability of equilibrium points of nonlinear systems *fails* if some eigenvalues have zero real part (i.e., lie on imaginary axis). In this case higher-order terms need to be taken into account.

A statement regarding the geometric and algebraic multiplicities of repeated eigenvalues can be used to ascertain stability/instability in the case of linear systems, but this does not necessarily hold for nonlinear systems.

In this case, Lyapunov's Direct Method using a Lyapunov function may be more helpful, but that is beyond the scope of this experiment.

CONCLUSIONS

In this experiment we examined the given nonlinear system, found its equilibrium points, performed linearization using Jacobian matrices, found system eigenvalues at equilibria, and observed the effect of parameter variations on non-zero eigenvalues.

Our findings are as follows –

1. The system has infinite equilibrium points. All of them dissolve into two basic equilibria, as has been shown earlier in the report.
    a. They correspond to the crane at rest, with the crane's arm either vertically downwards or upwards.
    b. The crane's horizontal position is a free variable, and does not contribute to the system's nonlinear dynamics.
2. Due to the presence of eigenvalues with zero real parts, the method of linearization is inconclusive at judging stability and instability of equilibria for the nonlinear case.
3. If we look only at the linearized system, then both the equilibria are unstable, as explained in the previous section. This cannot be generalized to the nonlinear system however, even around the equilibrium.
4. We have also shown the effect of parameter variations on the eigenvalues – there is no effect on stability/instability. Only the magnitudes of eigenvalues change as they move along the real/imaginary axes.

REFERENCES

- 3.2, 3.3, Theorem 3.2 – *Nonlinear Control (Khalil, Global Ed)*
- Ex. W2.1, 9.2.1, 9.5.2 – *Feedback Control of Dynamic Systems (FPE, 8th Ed.)*
- Link 1
- Link 2
- Link 3
- Link 4
- Theorem 8.4.3, Link 5

MATLAB SCRIPT

Creating the system of 1st order odes.

```
syms m_L m_C g l;
syms u x1 x2 x3 x4;
syms dx2 dx4;

eqn1 = (m_L+m_C)*dx2 + (m_L*l)*(dx4*cos(x3) - (x4^2)*(sin(x3))) == u;
eqn2 = (m_L)*(dx2*cos(x3) + l*dx4) == -m_L*g*sin(x3);

Sols = solve([eqn1 eqn2],[dx2 dx4]);

f1 = x2;
f2 = collect(simplify(Sols.dx2));
f3 = x4;
f4 = collect(simplify(Sols.dx4));

F = [f1; f2; f3; f4];
```

Jacobian Matrices

```
A = jacobian(F,[x1 x2 x3 x4]);
B = jacobian(F,u);
C = jacobian(x1+l*sin(x3),[x1 x2 x3 x4]);
D = jacobian(x1+l*sin(x3),u);
```

Eigenvalues

```
A1 = subs(A,vars,eqb1);
A2 = subs(A,vars,eqb2);

A1_eig = vpa(simplify(eig(A1)),5);
A2_eig = vpa(simplify(eig(A2)),5);
```

Parameter Variation

```
fsurf(A2_eig(4),[1 10 10 500])
colorbar
title("Magnitude of Eigenvalue")
xlabel("Rope length in m")
```

```
ylabel("Load mass in kg")
zlabel("Eigenvalue magnitude")
xlim([1.00 10.00])
ylim([10 500])
zlim([1.4 22.4])
view([-63.7 28.7])
```

EIGENVECTORS

```
[V1,D1] = eig(A1)
[V2,D2] = eig(A2)
```

LINK TO THE SCRIPT

https://drive.google.com/file/d/1HCntqNA8lW5jJa7vWfXDLbrTP3jCxYB4/view?usp=sharing