Homework 1

Erivelton Gualter dos Santos

1 Question

The circuit shown in the figure below contains a nonlinear inductor and is driven by a time-dependent current source. Suppose that the nonlinear inductor is described by $i_L = I_0 \sin(k\phi_L)$, where ϕ_L is the magnet flux of the inductor and I_0 and k are constants. Using ϕ_L and v_C as state variables, find the state equations.

Knowing the capacitor current is related to the capacitor voltage by

$$i_c(t) = C\frac{dv_c(t)}{dt} \tag{1}$$

and the inductor voltage is related to the inductor current by

$$v_l(t) = L\frac{di_l(t)}{dt} \tag{2}$$

and after choosing the state variables as v_c and ϕ_l, we can find the state space representation.
Replacing the nonlinear inductor current in equation 2, we have:

$$\begin{aligned} v_l(t) &= L\tfrac{d}{dt}(I_0 \sin(k\phi_l)) \\ &= LkI_0 \cos(k\phi_l)\dot{\phi}_l \end{aligned}$$

as $v_l = v_c = v_R$, we can write the following equation in terms of state variables:

$$v_c(t) = LkI_0 \cos(k\phi_l)\dot{\phi}_l \tag{3}$$

Applying Kirchhoff's current law (KCL) in the circuit, we hav e $i_c = i_s - i_l - i_R$. By replacing the respective quantities:

$$i_c = i_s - I_0 \sin(k\phi_l) - \frac{v_c}{R} \tag{4}$$

Therefore, replacing equation 4 in equation 1:

$$i_s - I_0 \sin(k\phi_l) - \frac{v_c}{R} = C\frac{dv_c(t)}{dt} \tag{5}$$

Finally, after reoordering the equations 3 and 5:

$$\begin{cases} \dot{v}_c = \frac{1}{C}\left(-\frac{v_c}{R} - I_0 \sin(k\phi_l) + i_s\right) \\ \dot{\phi}_l = \frac{v_c}{LkI_0 \cos(k\phi_l)} \end{cases} \tag{6}$$

2 Question

Use Matlab/Simulink to simulate the stable electronic oscillator in Example 8 in Lecture 1. Choose two sets of initial conditions that are different from the ones on pages 28-30 in this lecture, and produce the phase plane (or XY plane) plots and plot output responses with the various initial conditions. In your simulation, please choose $A = 1.5$, $V_1 = V_2 = 1$, $L = 1$, $C = 1F$, and $R = 0.1\Omega$.

The following figures represent two phase plane for distinct initial condition. The figure 1 start at $0, 30$, while figure 2 start at $0, 5$. Both cases contain a closed trajectory, also known as **limit cycle**. After several tests it was noted the state trajector end up in this limit cycle with radius of 11.5309.

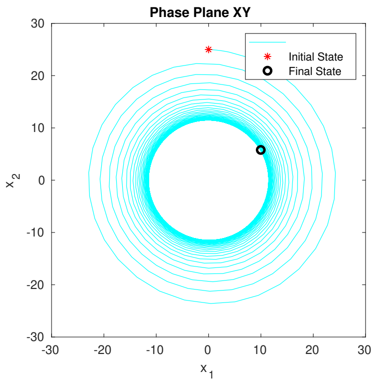

Figure 1: Phase plane for $(0, 30)$.

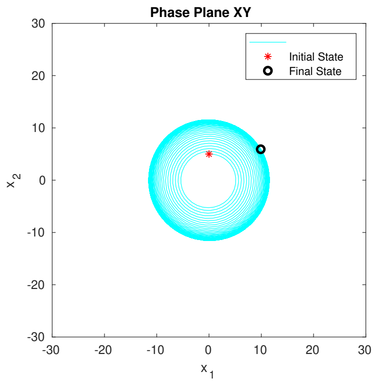

Figure 2: Phase plane for $(0, 5)$.

3 Question

For the following system, find the equilibrium points and determine the type of each isolated equilibrium point:

$$\begin{aligned} \dot{x}_1 &= 2x_1 - x_1 x_2 \\ \dot{x}_2 &= 2x_1^2 - x_2 \end{aligned}$$

By definition, the following equation must hold:

$$0 = 2x_1 - x_1 x_2 \tag{7}$$
$$0 = 2x_1^2 - x_2 \tag{8}$$

Replacing Equation 8 in Equation 7, we have:

$$\begin{aligned} 0 &= 2x_1 - 2x_1^3 \\ 0 &= 2x_1(1 - x_1^2) \end{aligned}$$

Then, there are three solutions for x_1 and x_2

$$\begin{bmatrix} x_{1e} \\ x_{2e} \end{bmatrix} = \begin{bmatrix} 0 & -1 & 1 \\ 0 & 2 & 2 \end{bmatrix} \tag{9}$$

Figure **??** illustrates the phase portrait for this plant. It contains several state trajectores from initial states varying from $-3 \leq x_1 \leq 3$ and $-3 \leq x_2 \leq 3$.

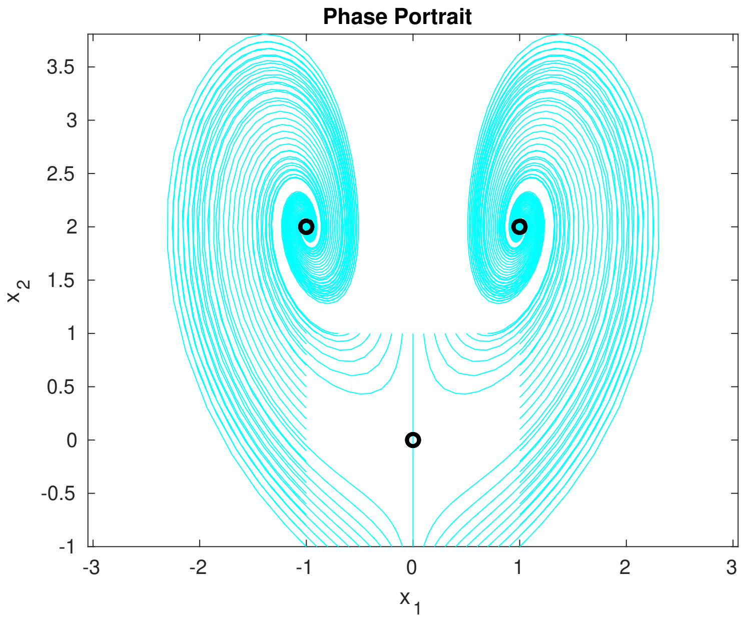

Figure 3: Phase Portrait.

In order to evaluate the type of each isolated equilibrium point for a nonlinear plant is necessary to examine the region around these equilibrium points. Therefore, a linearization must be performed. Applying a Taylor series about the any point, the following Jacobian matrix were obtained:

$$\begin{aligned} \left.\frac{\partial f}{\partial x}\right|_{x=xe_1} &= \begin{pmatrix} 2 & 0 \\ 0 & -1 \end{pmatrix}, \\ \left.\frac{\partial f}{\partial x}\right|_{x=xe_2} &= \begin{pmatrix} 0 & 1 \\ -4 & -1 \end{pmatrix}, \\ \left.\frac{\partial f}{\partial x}\right|_{x=xe_3} &= \begin{pmatrix} 0 & -1 \\ 4 & -1 \end{pmatrix} \end{aligned}$$

where the partial deriatives correspond to the Jacobian matrix of the three equilibrium points presented above. Additionally, the following correspond to the Jordan Canoninal form representation of those Jacobian matrix.

$$
\begin{aligned}
J_1 &= \begin{pmatrix} 2 & 0 \\ 0 & -1 \end{pmatrix}, \\
J_2 &= \begin{pmatrix} -\frac{1}{2} - \frac{\sqrt{15}\,1\mathrm{i}}{2} & 0 \\ 0 & -\frac{1}{2} + \frac{\sqrt{15}\,1\mathrm{i}}{2} \end{pmatrix}, \\
J_3 &= \begin{pmatrix} -\frac{1}{2} - \frac{\sqrt{15}\,1\mathrm{i}}{2} & 0 \\ 0 & -\frac{1}{2} + \frac{\sqrt{15}\,1\mathrm{i}}{2} \end{pmatrix}
\end{aligned}
$$

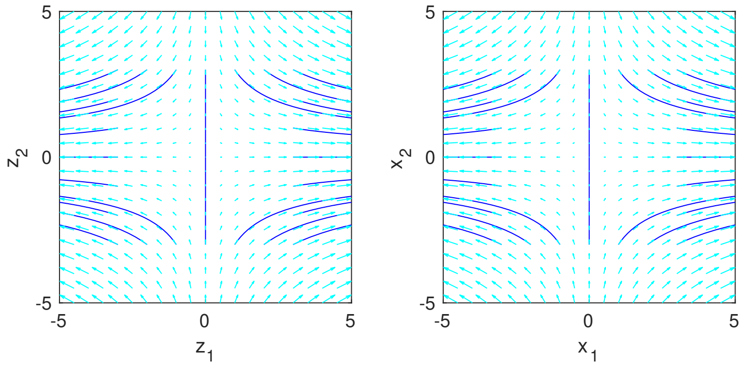

Figure 4: Phase Portrait for the first equilibrium point.

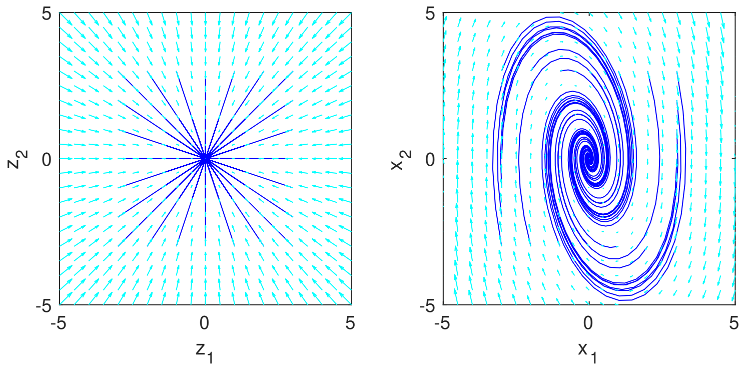

Figure 5: Phase Portrait for the second equilibrium point.

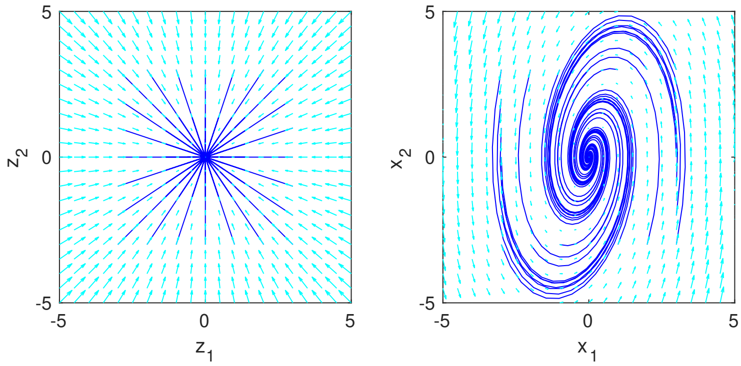

Figure 6: Phase Portrait for the third equilibrium point.

Therefore, by verifying the position of the eigenvalues or the phase portrait, it is clear that:

- 1^{st} eigenvalue is a **saddle point**;
- 2^{nd} and 3^{rd} eigenvalues are a **stable focus**;

4 Question

By plotting trajectories starting at different initial conditions, draw the phase portrait of the following LTI systems:

$$\dot{x}_1 = x_2 \tag{10}$$
$$\dot{x}_2 = -10x_1 - 10x_2 \tag{11}$$

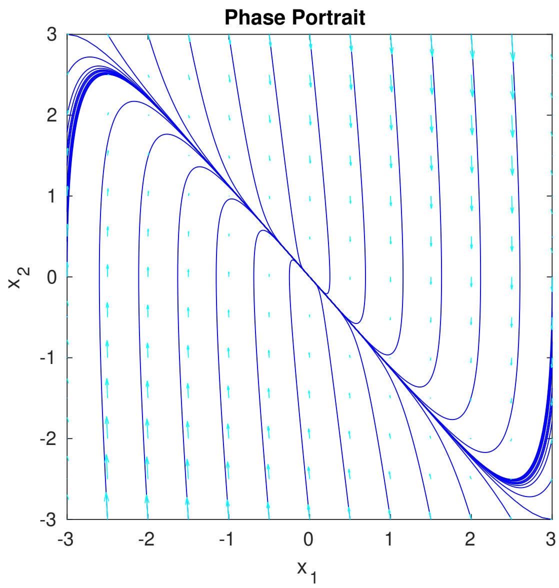

Figure 7: Phase plane for XY.

5 Question

The phase portrait (or phase-plane plot) of the following system is shown below. Mark the arrowheads and discuss the stability of each isolated equilibrium point

$$\dot{x}_1 = x_2 \tag{12}$$

$$\dot{x}_2 = x_1 - 2\tan^{-1}(x_1 + x_2) \tag{13}$$

The equilibria of the system above are $(0, 0)$, $(2.33, 0)$, and $(-2.33, 0)$.

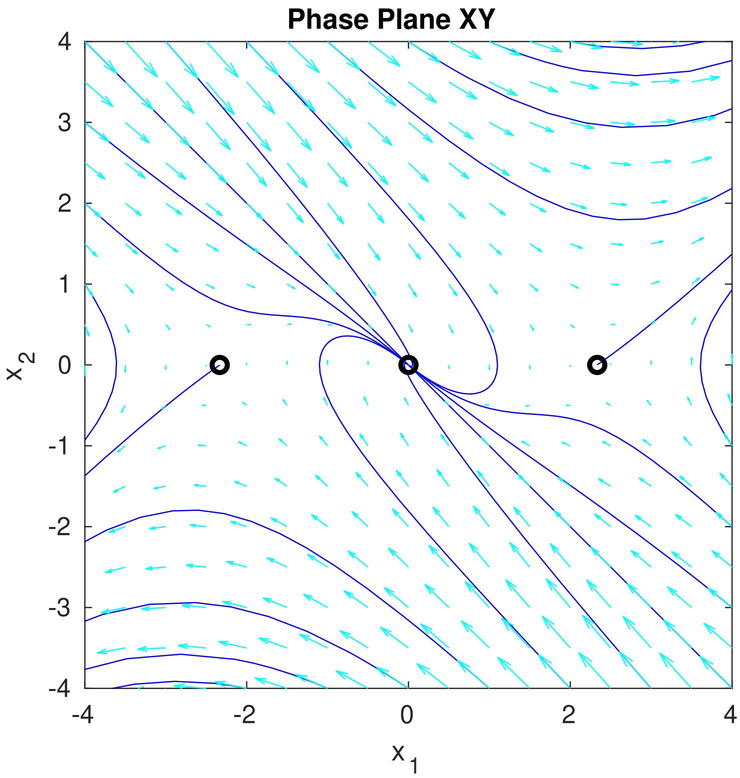

Figure 8: Phase portrait.

The Jacobian of the system is presented in the following for the respective equilibrium points.

$$\left.\frac{\partial f}{\partial x}\right|_{x=xe_1} = \begin{pmatrix} 0 & 1 \\ -1 & -2 \end{pmatrix}$$
$$\left.\frac{\partial f}{\partial x}\right|_{x=xe_2} = \begin{pmatrix} 0 & 1 \\ \frac{20}{29} & -\frac{9}{29} \end{pmatrix}$$
$$\left.\frac{\partial f}{\partial x}\right|_{x=xe_3} = \begin{pmatrix} 0 & 1 \\ \frac{20}{29} & -\frac{9}{29} \end{pmatrix}$$

Then, the Jordan canonical matrix form is:

$$J_1 = \begin{pmatrix} -1 & 1 \\ 0 & -1 \end{pmatrix}$$

$$J_2 = \begin{pmatrix} -1 & 0 \\ 0 & \frac{20}{29} \end{pmatrix}$$

$$J_3 = \begin{pmatrix} -1 & 0 \\ 0 & \frac{20}{29} \end{pmatrix}$$

Therefore, according to the location of the eigenvalues and also the direction of the arrowheads, it is well known that:

- 1^{st} eigenvalue is a **stable node**;
- 2^{nd} and 3^{rd} eigenvalues are a **saddle node**;

6 Matlab Codes

```
% Erivelton Gualter
% Nonlinear System - Homework 1 - Question 2

function question2()
    A = 1.5;
    V1 = 1;
    V2 = 1;
    L = 1;  % H
    C = 1;  % F
    R = .1; % Ohm

    param = [A V1 V2 L C R];
    fun = @(t,x) derStableOscilator(t,x, param);

    x0 = [0; 5];
    [t, X] =ode45(fun,[0 1000],x0);

    figure('Name','Question 2 - Phase Plane ...
        XY','NumberTitle','off');
    hold on;
    axis equal
    plot(X(:,1), X(:,2), 'c');
    plot(X(1,1), X(1,2), '*r');
    plot(X(end,1), X(end,2), 'ok', 'LineWidth', 2);

    title('Phase Plane XY');
    ylabel('x_2'); xlabel('x_1');
    legend('','Initial State','Final State');
    box on

    axis([-30 30 -30 30])

    print('question2b','-depsc')

    % Function
    function xDer = derStableOscilator(t, x, param)

        A = param(1);
```

```
        V1 = param(2);
        V2 = param(3);
        L = param(4);
        C = param(5);
        R = param(6);

        x1 = x(1);
        x2 = x(2);

        xd1 = inv(C)*x2;
        xd2 = inv(L)*(V1*atan(A*R*x2/V2)-x1-R*x2) ;

        xDer = [xd1; xd2];
    end
end
```

```
% Erivelton Gualter
% Nonlinear System - Homework 1 - Question 3

function question3()

    figure('Name','Question 3 - Phase ...
        Portrait','NumberTitle','off');
    hold on;
    axis equal

    xl = 1; yl = 1;
    for x = -xl:0.1:xl
        for y = -yl:0.1:yl
            if abs(x) == xl || abs(y) == yl
                x0 = [x; y];
                [t, X] =ode45(@derEquation3,[0 20],x0);
                plot(X(:,1), X(:,2), 'c');
                plot(X(end,1), X(end,2), 'ok', 'LineWidth', 2);
            end
        end
    end

    title('Phase Portrait');
    ylabel('x_2'); xlabel('x_1');
    box on
    print('question3','-depsc')

    J1 = jacob([0 0]);
    J2 = jacob([-1 2]);
    J3 = jacob([1 2]);

    j1 = jordan(J1);
    j2 = jordan(J2);
    j3 = jordan(J3);

    figure;
    subplot(121); plotZphase(j1); axis(5*[-1 1 -1 1]); ...
        ylabel('z_2'); xlabel('z_1'); box on
    subplot(122); plotZphase(J1); axis(5*[-1 1 -1 1]); ...
        ylabel('x_2'); xlabel('x_1'); box on
    print('question3a','-depsc')

    figure;
    subplot(121); plotZphase(j2); axis(5*[-1 1 -1 1]); ...
        ylabel('z_2'); xlabel('z_1'); box on
```

```
    subplot(122); plotZphase(J2); axis(5*[-1 1 -1 1]); ...
        ylabel('x_2'); xlabel('x_1'); box on
    print('question3b','-depsc')

    figure;
    subplot(121); plotZphase(j3); axis(5*[-1 1 -1 1]); ...
        ylabel('z_2'); xlabel('z_1'); box on
    subplot(122); plotZphase(J3); axis(5*[-1 1 -1 1]); ...
        ylabel('x_2'); xlabel('x_1'); box on
    print('question3c','-depsc')

    % Functions
    function xDer = derEquation3 (t, x)
        x1 = x(1);
        x2 = x(2);

        xd1 = 2*x1 - x1*x2;
        xd2 = 2*x1*x1 - x2;

        xDer = [xd1; xd2];
    end

    function J = jacob(xe)
        x1e = xe(1);
        x2e = xe(2);
        J = [2-x2e, -x1e; 4*x1e, -1];
    end

            xd1 = 2*x1 - x1*x2;
        xd2 = 2*x1*x1 - x2;
    function plotZphase(A)
        sys = ss(A,[0;0],eye(2),0);

        hold on;
        axis equal

        xl = 3; yl = 3;
        for x = -xl:1:xl
            for y = -yl:1:yl
                if abs(x) == xl || abs(y) == yl
                    x0 = [x; y];
                    [¬,T,X] = initial(sys, x0);
                    plot(X(:,1), X(:,2), 'b');
                end
            end
        end

        [x1, x2] = meshgrid(-5:0.5:5, -5:.5:5);
        x1dot = A(1,1).*x1 + A(1,2).*x2;
        x2dot = A(2,1).*x1 + A(2,2).*x2;
        quiver(x1,x2,x1dot, x2dot, 'color', 'cyan')

    end

end
```

```
% Erivelton Gualter
% Nonlinear System - Homework 1 - Question 4

function question4()
    A = [0 1; -10 -10];
```

```
    B = [0; 0];
    C = [1 0; 0 1];
    D = 0;

    sys = ss(A,B,C,D);

    figure('Name','Question 4 - Phase ...
        Portrait','NumberTitle','off');
    hold on;
    axis equal

    xl = 3; yl = 3;
    for x = -xl:0.5:xl
        for y = -yl:0.5:yl
            if abs(x) == xl || abs(y) == yl
                x0 = [x; y];
                [¬,T,X] = initial(sys, x0);
                plot(X(:,1), X(:,2), 'b');
            end
        end
    end

    [x1, x2] = meshgrid(-5:0.5:5, -5:.5:5);
    x1dot = A(1,1).*x1 + A(1,2).*x2;
    x2dot = A(2,1).*x1 + A(2,2).*x2;
    quiver(x1,x2,x1dot, x2dot, 'color', 'cyan')

    title('Phase Portrait');
    ylabel('x_2'); xlabel('x_1');
    axis([-xl xl -yl yl])
    box on

    print('question4','-depsc')
end
```

```
% Erivelton Gualter
% Nonlinear System - Homework 1 - Question 5

function question5()

    figure('Name','Question 5 - Phase ...
        Portrait','NumberTitle','off');
    hold on;
    axis equal

    xl = 4; yl = 4;
    for x = -xl:1:xl
        for y = -yl:1:yl
            if abs(x) == xl || abs(y) == yl
                x0 = [x; y];
                [t, X] =ode45(@derEquation5,[0 20],x0);
                plot(X(:,1), X(:,2), 'b');
            end
        end
    end

    xye = [0 7/3 -7/3; 0 0 0];
    for i=1:length(xye)
        [t, X] =ode45(@derEquation5,[0 20],xye(:,i));
        plot(X(:,1), X(:,2), 'b');
    end
```

```
    axis([-xl xl -yl yl])

    title('Phase Plane XY');
    ylabel('x_2'); xlabel('x_1');
    box on

    [x1, x2] = meshgrid(-4:0.5:4, -4:.5:4);
    x1dot = x2;
    x2dot = x1 - 2*atan(x1+x2);
    quiver(x1,x2,x1dot, x2dot, 'color', 'cyan')

    plot(xye(1,1), xye(2,1), 'ok', 'LineWidth', 2);
    plot(xye(1,2), xye(2,2), 'ok', 'LineWidth', 2);
    plot(xye(1,3), xye(2,3), 'ok', 'LineWidth', 2);

    print('question5','-depsc')

    J1 = jacob(xye(:, 1));
    J2 = jacob(xye(:, 2));
    J3 = jacob(xye(:, 3));

    j1 = jordan(J1);
    j2 = jordan(J2);
    j3 = jordan(J3);

    latex(sym(J1))
    latex(sym(J2))
    latex(sym(J3))
    latex(sym(j1))
    latex(sym(j2))
    latex(sym(j3))

    % Function
    function xDer = derEquation5 (t, x)
        x1 = x(1);
        x2 = x(2);

        xd1 = x2;
        xd2 = x1 - 2*atan(x1+x2);

        xDer = [xd1; xd2];
    end

    function J = jacob(xe)
        x1e = xe(1);
        x2e = xe(2);
        J = [0, 1; 1-2/(1+(x1e+x2e)^2), -2/(1+(x1e+x2e)^2)];
    end
end
```