Table of Contents


```
global R1 C1 L1 L2
R1 = 1; %Ohm
C1 = 1; %Farad
L1 = 1; %Henry
L2 = 1; %Henry
```

Part a

```
% Write the state space model dx = f(x,V) using Kirchoff equations.
% Choose the inductors currents and the capacitor's voltage as state
 variables
% x1 = i_L1
% x2 = i_L2
% x3 = v_c
% y = x1

% Circuit
%  -- C -------- R ----
%  |        |         |
%  +        |         |
%  V        L2        L1
%  -        |         |
%  |        |         |
%  --------------------

% Note: Resistor voltage is given by
% V(I) = RI-RI^3

% L1dx1/dt = v_l1
% L2dx2/dt = v_l2
% C dx3/dt = i_c

% From Kirchoff's Voltage laws we get 2 equations
% V - v_c - v_l2 = 0
% V - v_c - v_r - v_l1 = 0
% From Kirchoff's Current laws we get 2 more equations
% i_l1 + i_l2  = i_c

%Combining everything we get
% L1dx1/dt = V - v_c - v_r = V -x3 - (R(x1) - R(x1)^3)
% L2dx2/dt = V - v_c = V - x3
```

```
% C dx3/dt = i_l1 + i_l2 = x1 + x2

% Isolating the derivatives we get
% dx1/dt = (1/L1) * (V -x3 - R(x1) + R(x1)^3)
% dx2/dt = (1/L2) * (V - x3)
% dx3/dt = (1/C)  * (x1 + x2)
```

Part b

```
% dx1/dt = (1/L1) * (V -x3 - R(x1) + R(x1)^3)
% dx2/dt = (1/L2) * (V - x3)
% dx3/dt = (1/C)  * (x1 + x2)

% Setting  all derivatives equal to zero we get
% (V -x3 - R(x1) + R(x1)^3) = 0
% (V - x3) = 0
% (x1 + x2) = 0

% Solving for x3 with the second equation we get
% x3 = V
% Solving x1 for the first equation we get
% R(x1) - R(x1)^3  = 0
% x1 = 0 OR x1 = 1
% Finally solving for x2 we get
% x2 = x1
% x2 = 0 OR x2 = 1

% So for each  voltage value we have 2 equilibrium points
```

Part c

```
% We expect that the system will be stable when the capacitor's
 voltage is equal to the battery and there is no more current flow
% If there is current flow it will cause the capacitor's voltage to
 change relative to the battery causing the system to change
% we can confirm this by looking at the linearized systems and their
 eigenvalues

% Linearizing the system around V we get our matricies as

% A = [ R(-1 + 3x1^2)/L1  0  -1/L1;
%            0            0  -1/L2;
%           1/C          1/C   0 ];
% B = [1;
%      1;
%      0];
% C = [1 0 0];
% D = 0

% Where x1 is the value for the equilibrium you are considering

% Since it can only take two values (x1 = 0 and x1 = 1) we can look at
 the two possible sets of eigenvalues for x1
```

```
equilibirums = equilibirum_constant(1);
% This is x1 = 0 (Expect it to be stable)
[A_0, B_0, C_0, D_0] = linearize(equilibirums(:,1));
% This is x1 = 1 (Expect it to be unstable)
[A_1, B_1, C_1, D_1] = linearize(equilibirums(:,2));

eig(A_0)
% -0.5698 + 0.0000i
% -0.2151 + 1.3071i
% -0.2151 - 1.3071i

% All the real parts of the system's eigenvalues are negative so the
 equilibria is stable

eig(A_1)
% 1.5437 + 0.0000i
% 0.2282 + 1.1151i
% 0.2282 - 1.1151i

% At least one real part of the system has a positive eigenvalue so
 this is not a stable equilibirum


ans =

  -0.5698 + 0.0000i
  -0.2151 + 1.3071i
  -0.2151 - 1.3071i


ans =

   1.5437 + 0.0000i
   0.2282 + 1.1151i
   0.2282 - 1.1151i
```

Part d & e

```
% Simulate and compare the linearized and the nonlinear system using
 MATLAB

constant_voltage = 1;
tmax = 20;

[ts, x_ode] = ode45(@(t,x) dx(x, @(x) constant_voltage, t), [0 tmax],
 [0; 0; 0]);

t = 0:0.1:tmax;
u = zeros(length(t),1);
u(:,1) = constant_voltage;
% Centering around the equilibrium point
[~, ~, x_stable] = lsim(ss(A_0,B_0,C_0,D_0), u, t, [0;0;0]);
```

```
[~, ~, x_unstable] = lsim(ss(A_1,B_1,C_1,D_1), u, t, [0;0;0]);
legend_strings = ["ode45", "Linear Stable", "Linear Unstable"];

subplot(3,1,1);
plot(ts, x_ode(:,1));
hold on;
plot(t, x_stable(:,1));
plot(t, x_unstable(:,1));
ylim([-1 3]);
legend(legend_strings);
xlabel("Time (s)");
ylabel("Inductor 1 (A)");
hold off;


subplot(3,1,2);
plot(ts, x_ode(:,2));
hold on;
plot(t, x_stable(:,2));
plot(t, x_unstable(:,2));
ylim([-1 3]);
legend(legend_strings);
xlabel("Time (s)");
ylabel("Inductor 2 (A)");
hold off;



subplot(3,1,3);
plot(ts, x_ode(:,3));
hold on;
plot(t, x_stable(:,3));
plot(t, x_unstable(:,3));
ylim([-1 3]);
legend(legend_strings);
xlabel("Time (s)");
ylabel("Capacitor (V)");
hold off;

sgtitle("Comparing nonlinear model to linear models with 1 (V) and
 initially at rest");
saveas(gcf, "images/ode_vs_linear_models_1V_rest.png");
snapnow
% Testing close to the unstable equilibirum point
constant_voltage = 1;
%tmax = 15;

[ts, x_ode] = ode45(@(t,x) dx(x, @(x) constant_voltage, t), [0 tmax],
 [.99; .99; 1]);

t = 0:0.1:tmax;
u = zeros(length(t),1);
u(:,1) = constant_voltage;
% Centering around the equilibrium point
```

```
[~, ~, x_stable] = lsim(ss(A_0,B_0,C_0,D_0), u, t, [.99;.99;1]);
[~, ~, x_unstable] = lsim(ss(A_1,B_1,C_1,D_1), u, t, [.99;.99;1]);
legend_strings = ["ode45", "Linear Stable", "Linear Unstable"];

subplot(3,1,1);
plot(ts, x_ode(:,1));
hold on;
plot(t, x_stable(:,1));
plot(t, x_unstable(:,1));
ylim([-1 3]);
legend(legend_strings);
xlabel("Time (s)");
ylabel("Inductor 1 (A)");
hold off;


subplot(3,1,2);
plot(ts, x_ode(:,2));
hold on;
plot(t, x_stable(:,2));
plot(t, x_unstable(:,2));
ylim([-1 3]);
legend(legend_strings);
xlabel("Time (s)");
ylabel("Inductor 2 (A)");
hold off;



subplot(3,1,3);
plot(ts, x_ode(:,3));
hold on;
plot(t, x_stable(:,3));
plot(t, x_unstable(:,3));
ylim([-1 3]);
legend(legend_strings);
xlabel("Time (s)");
ylabel("Capacitor (V)");
hold off;

sgtitle("Comparing nonlinear model to linear models with 1 (V) and
 close to unstable solution");
saveas(gcf, "images/ode_vs_linear_models_1V_near_unstable.png");
snapnow;
% As we can see the unstable linearization is always inaccurate
% A small deviation from the nonstable equilibrium causes the system
 to converge to the stable equilibirum.
```

Part f

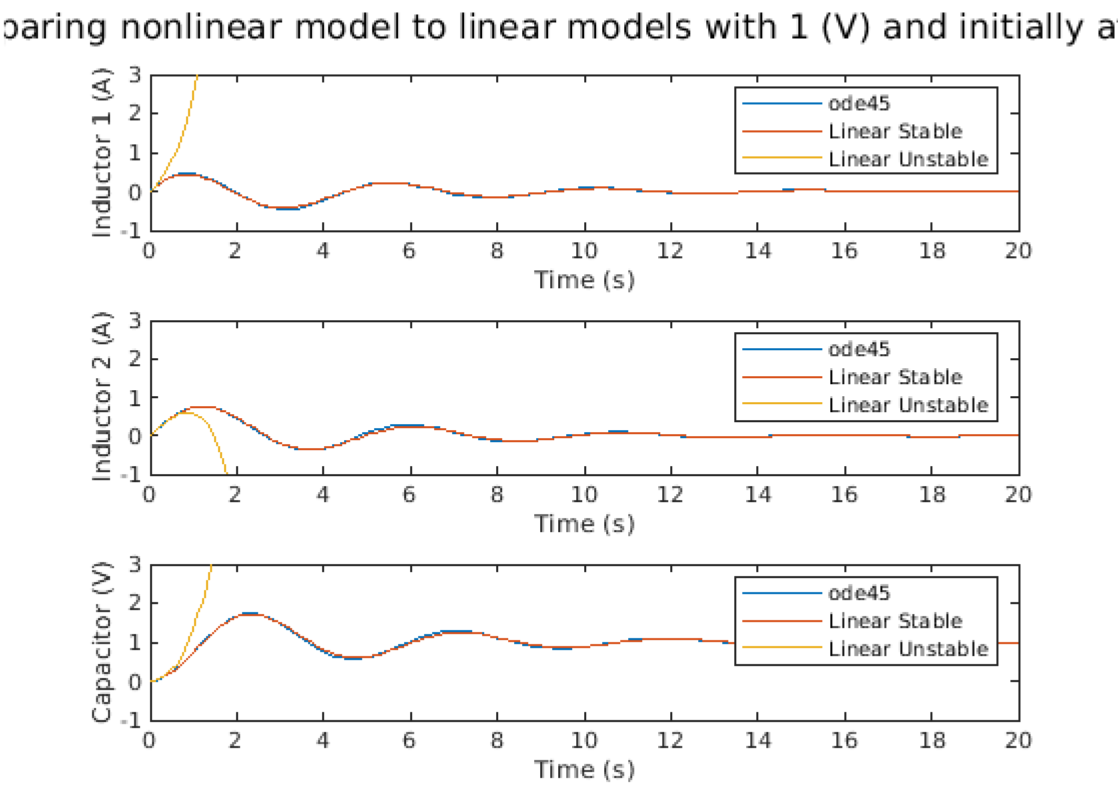
```
% We expect a circuit like this to be stable when there is no voltage
 difference across any component so when the voltage across the
 capacitor is equal to the voltage across the battery.
```

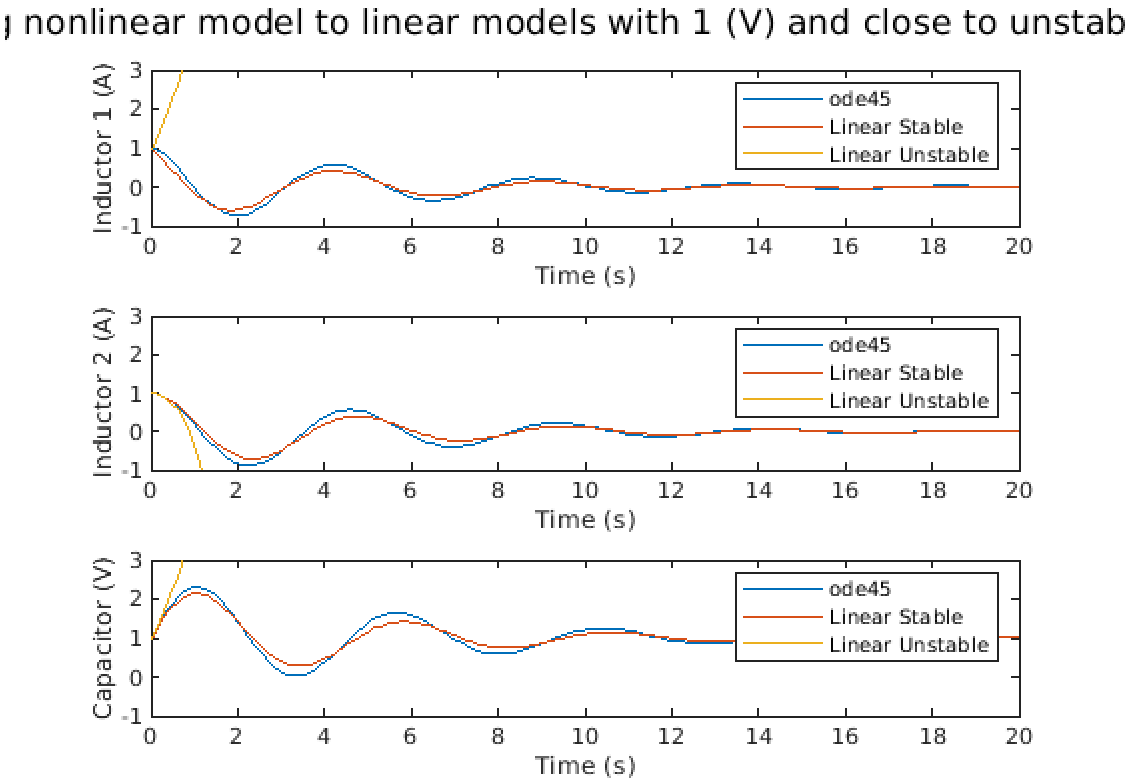
```
% For the inductors we expect the current across them to be zero.
% If the capacitor was at equilibrium but the inductor wasn't at
 equilibrium then the capacitor would charge past the battery's
 voltage.
% This would cause the a voltage difference across the battery and
 capacitor which will cause current to flow backwards.
% This would repeat forever except that the resistor is going to
 dissapate energy until we reach steady state.
% We see this behavior, the inductors oscillate around their
 equilibirum (0 Amps) and the capacitor oscillating around its
 equilibirum point (battery voltage)
```

Functions

```
function voltage = resistor_voltage(current)
    global R1
    voltage = R1*current - R1*current^3;
end

function x_dot = dx(x, func_input, t)
    % dx1/dt = (1/L1) * (V -x3 - R(x1) + R(x1)^3)
    % dx2/dt = (1/L2) * (V - x3)
    % dx3/dt = (1/C)  * (x1 + x2)
    global R1 C1 L1 L2
    x_dot = [
        (1/L1) * (func_input(t) - x(3) - resistor_voltage(x(1)));
        (1/L2) * (func_input(t) - x(3));
        (1/C1) * (x(1) + x(2))
        ];
end

function eq = equilibirum_constant(voltage)
    % There are two equilibrium points (See part b)
    % x1 = 0
    % x2 = 0
    % x3 = voltage
    % and
    % x1 = 1
    % x2 = 1
    % x3 = voltage
    eq = [
        0 1;
        0 1;
        voltage voltage;
    ];
end

function [A, B, C, D] = linearize(equilibirum)
    global R1 C1 L1 L2
    A = [ (R1/L1)*(-1 + 3*equilibirum(1)^2)  0  -1/L1;
               0                            0  -1/L2;
              1/C1                        1/C1   0 ];
    B = [1;
```

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





Published with MATLAB® R2020b