ECE 3620 – Kade Howes, A02298066, 10/05/2022

Programming Assignment 1

1.

The program should be written in C or C++. Only libraries from the C/C++ standard libraries are permitted. No other external libraries should be used – you are writing the entire program from scratch. You must write all of the code necessary for the processing.

View Appendix A for C++ code.

2.

For the differential equation

$$(D + 2.5)y_0(t) = 0$$

with initial condition $y_0(0) = 3$*:*

2a

Find and plot the analytical (exact) solution to the differential equation for $0 \leq t \leq 10$*.*

From the characteristic equation:

$$\lambda + 2.5 = 0$$

We know $\lambda = -2.5$. So the solution for $y_0(t)$ is:

$$\begin{aligned} y_0(t) &= Ce^{-2.5t} \\ &\text{where} \\ C &= y_0(0) = 3 \\ \implies y_0(t) &= 3e^{-2.5t} \end{aligned}$$

This results in a plot shown below in Figure 2a.

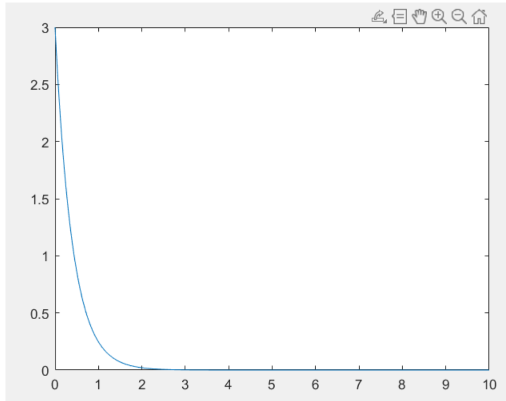


Figure 2a

2b

Write a program in C(++) to plot a numerical solution using (3). You may have to try several values of Δt *to get a good enough approximation.*

Putting the differential equation and initial condition into my program with a Δt of 0.001 results in the plot shown below in Figure 2b:

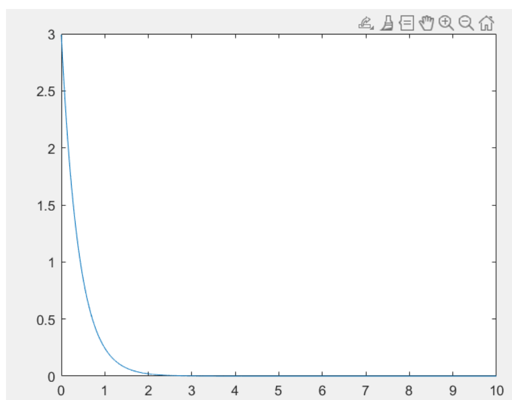


Figure 2b

2c

Compare the exact solution with the approximate solution.

As seen in the comparison of Figure 2a and Figure 2b, the plot are essentially identical. The approximate solution conformed well to the exact solution, with the almost exactly the same shape and rate of change.

3.

For the third-order differential equation

$$(D^3 + 0.6D^2 + 25.1125D + 2.5603)y_0(t) = 0$$

with initial conditions $y_0(0) = 1.5, \dot{y}_0(0) = 2, \ddot{y}_0(0) = -1$*:*

3a

Find and plot the analytical solution to the differential equation for $0 \leq t \leq 10$*. Identify the roots of the characteristic equation and plot them in the complex plane.*

Using an online cubic root calculator as approved results in the following roots: $\lambda = 0.1, -0.25 \pm 5j$.

This results in the form for the solution $y_0(t)$:

$$\begin{aligned}
y_0(t) &= Ce^{-0.1t} + Ae^{-0.25t}\cos(5t+\theta) \\
\implies \dot{y}_0(t) &= -0.1Ce^{-0.1t} - 0.25Ae^{-0.25t}\cos(5t+\theta) - 5Ae^{-0.25t}\cos(5t+\theta) \\
\implies \ddot{y}_0(t) &= 0.01Ce^{-0.1t} + 0.0625Ae^{-0.25t}\cos(5t+\theta) + 1.25Ae^{-0.25t}\sin(5t+\theta) + \\
&\quad 1.25Ae^{-0.25t}\sin(5t+\theta) - 25Ae^{-0.25t}\cos(5t+\theta)
\end{aligned}$$

Which gives us three equations to find the constants C, A, and θ:

$$\begin{aligned}
1.5 &= C + A\cos(\theta) \\
2 &= -0.1C - A\cos(\theta) - 5A\sin(\theta) \\
-1 &= 0.01C + 0.0625A\cos(\theta) + 1.25A\sin(\theta) + 1.25A\sin(\theta) - 25A\cos(\theta)
\end{aligned}$$

Using a calculator to solve for C, A, and θ:

$$\begin{aligned}
C &= 1.5 \\
A &= -0.43 \\
\theta &= \frac{\pi}{2}
\end{aligned}$$

Which holds since C must equal 1.5 and θ must equal $\frac{\pi}{2}$ by inspection of the numerical plot solution shown later in Figure 3c.

Thus the solution $y_0(t)$ is

$$y_0(t) = 1.5e^{-0.1t} - 0.43e^{-0.25t}\cos(5t + \frac{\pi}{2})$$

Plotting this function results in the graph shown below in Figure 3a.

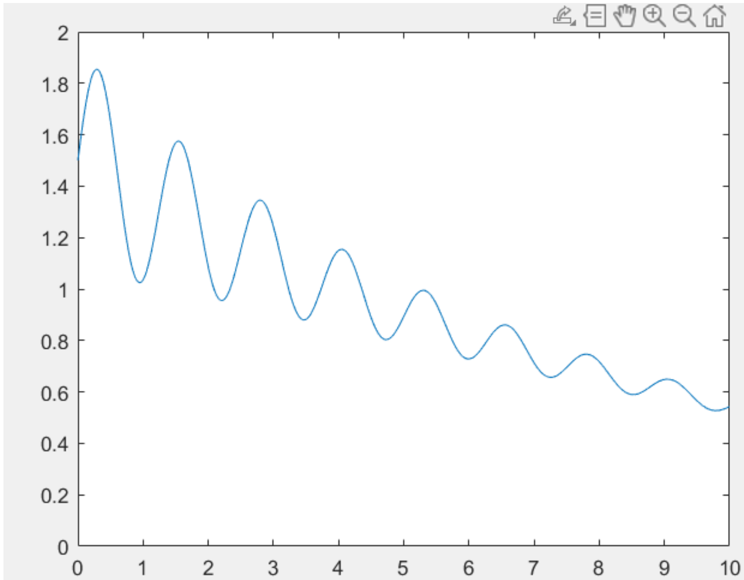


Figure 3a

Here is the plot of the roots of the characteristic equation on the complex plane (Figure 3b):

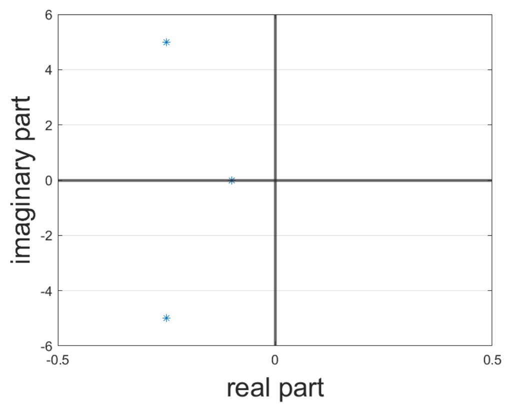


Figure 3b

3b

Put the third-order differential equation into state-space form.

let $x_1(t) = y_0(t)$, $x_2(t) = \dot{y}_0(t)$, and $x_3(t) = \ddot{y}_0(t)$.

State-space form should be $\dot{\mathbf{x}}(t) = A\mathbf{x}(t)$.

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -2.5063 & -25.1125 & -0.6 \end{bmatrix}$$

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -2.5063 & -25.1125 & -0.6 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix}$$

3c

Write a program in C(++) to plot an approximate solution using (8). You may have to try several values of Δt *to get a reasonable approximation.*

Putting the differential equation and initial conditions into my program with a Δt of 0.001 results in the plot shown below in Figure 3c:

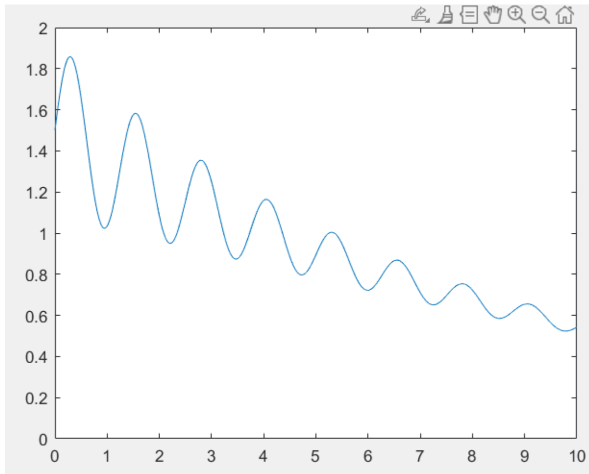


Figure 3c

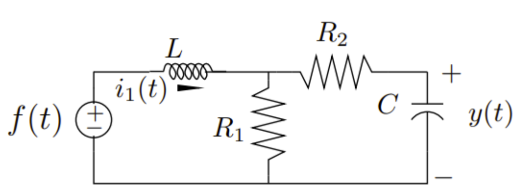
3d

Compare the exact solution with the approximate solution.

As can be seen in the visual comparison of the plots shown in Figure 3a and Figure 3c, the exact solution and approximate solution are very close. The approximate solution produced by the program was accurate and follows the form we would expect to see from the exact solution. This is a exponentially decreasing damped oscillating waveform.

4.

For the circuit shown here:



where $R_1 = 1\ k\Omega$, $R_2 = 22\ k\Omega$, $C = 10\ \mu F$, *and* $L = 5\ H$.

4a

Determine a differential equation relating the input $f(t)$ *to the output* $y(t)$.

Using mesh current loops $i_1(t)$ (clockwise in the left loop) and $i_2(t)$ (clockwise in the right loop) results in the equations:

$$\begin{aligned} f(t) &= LDi_1(t) + R_1(i_1(t) - i_2(t)) \\ 0 &= R_2(i_2(t)) + \frac{1}{CD} i_2(t) + R_1(i_2(t) - i_1(t)) \\ \implies f(t) &= (LD + R)i_1(t) - R_1 i_2(t) \\ 0 &= -R_1 i_1(t) + (\frac{1}{CD} + R_1 + R_2) i_2(t) \end{aligned}$$

converting to matrix form:

$$\begin{bmatrix} LD + R_1 & -R_1 \\ -R_1 & \frac{1}{CD} + R_1 + R_2 \end{bmatrix} \begin{bmatrix} i_1(t) \\ i_2(t) \end{bmatrix} = \begin{bmatrix} f(t) \\ 0 \end{bmatrix}$$

using Cramer's rule (more detail shown in handwork in Appendix B):

$$\begin{aligned} i_2(t) &= \frac{R_1 f(t)}{(LD + R_1)(\frac{1}{CD} + R_1 + R_2) - R_1^2} \\ \implies \left[D^2 + D\frac{R_1 R_2 C + L}{LR_1 C + LR_2 C} + \frac{R_1}{LR_1 C + LR_2 C} \right] y(t) &= R_1 f(t) \\ = \left[D^2 + 195.652D + 869.565 \right] y(t) &= 1000 f(t) \end{aligned}$$

4b

Determine the initial condition for $\dot{y}(t)$ *if* $i_1(0) = 0.2\ A$ *and* $y(0) = 5\ V$.

Using the same mesh current loops when $t = 0$ results in the loop equation:

$$\begin{aligned} R_2 i_2(0) + 5 + R_1(i_2(0) - 0.2) &= 0 \\ \implies i_2(0) = \frac{0.2R_1 - 5}{R_1 + R_2} &= 0.00848 \\ \dot{y}(0) = \frac{i_2(0)}{C} &= 847.826 \frac{V}{s} \end{aligned}$$

4c

Determine the analytical solution for the zero-input response of the system with these initial conditions

Starting with the characteristic equation

$$\left[D^2 + 195.652D + 869.565 \right] y_0(t) = 0$$

Then, using the quadratic formula (more details shown in Appendix B) results in the characteristic roots $\lambda = -4.5503, -191.1019$.

So, the form of the solution is $y_0(t) = C_1 e^{-4.5503t} + C_2 e^{-191.1019t}$.

Finding $\dot{y}_0(t)$ and using initial conditions results in the system of equations:

$$\begin{aligned} 5 &= C_1 + C_2 \\ 847.826 &= -4.5503C_1 - 191.1019C_2 \end{aligned}$$

Putting into matrix form:

$$\begin{bmatrix} 1 & 1 \\ -4.5503 & 191.1019 \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} = \begin{bmatrix} 5 \\ 847.826 \end{bmatrix}$$

Solving for $C_1, C_2 \implies C_1 = 9.\bar{6}$ and $C_2 = -4.\bar{6}$.

So the analytical solution for the zero-input response is:

$$y_0(t) = 9.\bar{6}e^{-4.5503t} + 4.\bar{6}e^{-191.1019t}$$

4d

Represent the differential equation for the circuit in state space form.

let $x_1(t) = y_0(t)$, and $x_2(t) = \dot{y}_0(t)$.
State-space form should be $\dot{\mathbf{x}}(t) = A\mathbf{x}(t)$.

$$\begin{aligned} A &= \begin{bmatrix} 0 & 1 \\ -869.565 & -195.652 \end{bmatrix} \\ \dot{\mathbf{x}}(t) &= \begin{bmatrix} 0 & 1 \\ -869.565 & -195.652 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} \end{aligned}$$

4e

Using your program, determine a numerical solution to the differential equation for the zero-input response.

Putting the characteristic equation into my program with the initial conditions for $y_0(0) = 5$ V and $\dot{y}_0(0) = 847.826\frac{V}{s}$ with a $\Delta t = .001$ s results in the plot shown in Figure 4b.

4f

Plot and compare the analytical and the numerical solution. Comment on your results.

Figure 4a below shows the plot of the analytical solution $y_0(t) = 9.\bar{6}e^{-4.5503t} + 4.\bar{6}e^{-191.1019t}$.

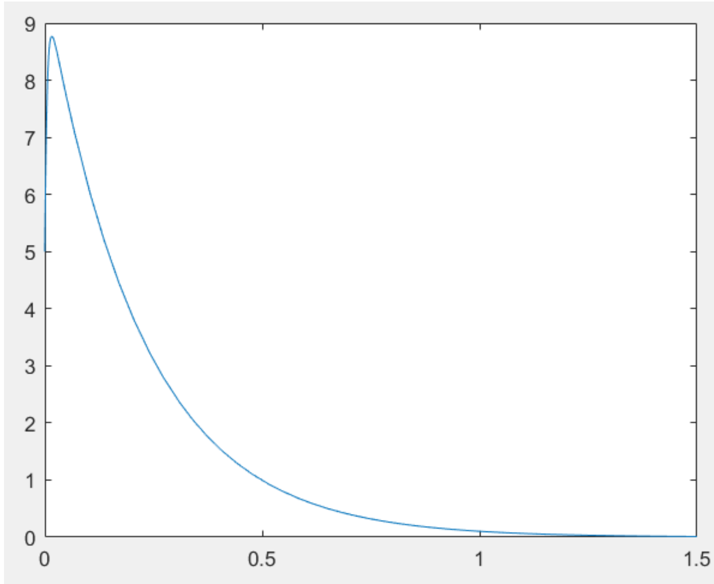


Figure 4a

Figure 4b below shows the plot of the numerical solution output by my program with $\Delta t = .001$ s.

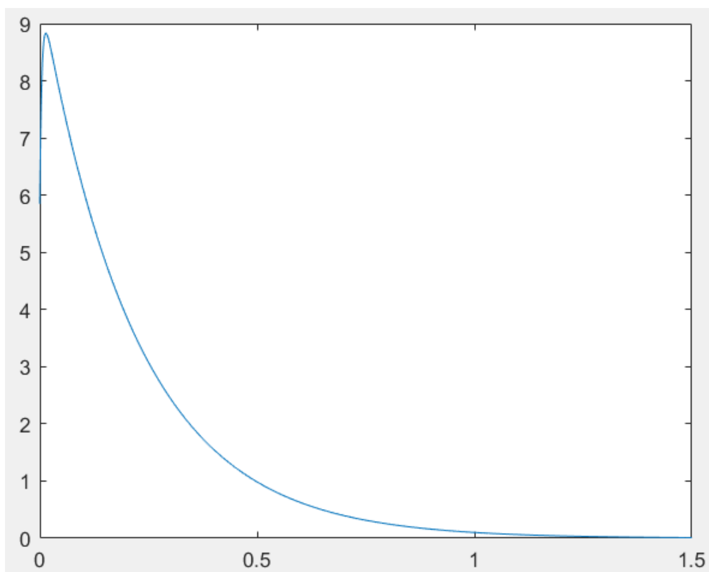


Figure 4b

As seen in the visual comparison of the Figure 4a and Figure 4b above, the analytical and numerical solutions are very close to identical. Visually, it is near impossible to tell a difference. Of course, when changing Δt to something larger than .001 (.005 for example) causes increased inaccuracies in the numerical solution. This error is propagated more intensely around points with large rates of change, such as the peak seen at the beginning of the waveform in either figure. Figure 4c below is the numerical solution with $\Delta t = 0.005$, which causes the two solutions to have more variation between them, as seen especially around the peak. This applies for all numerical solutions: **The larger the Δt, the less accurate the approximation will be.**

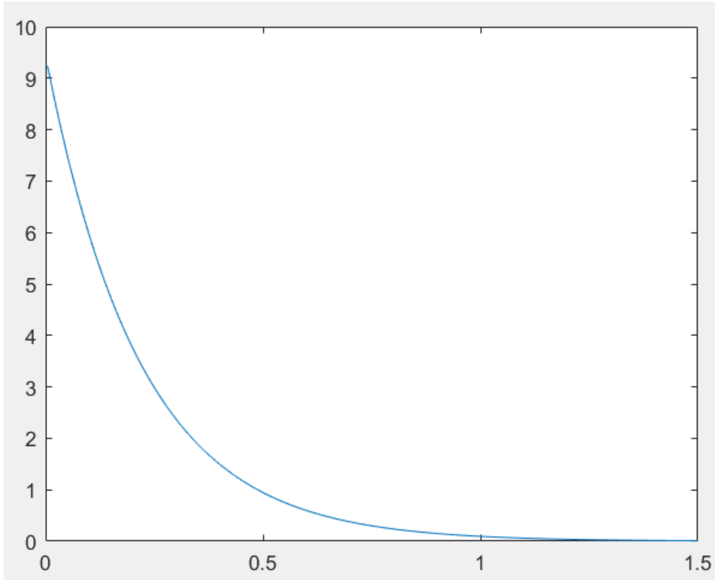


Figure 4c

4g

Suppose that the circuit had nonlinear element in it, such as dependent sources. Describe how the analytical solution and numerical solution would change.

First, an important phrase is *non-linear*. That meanse we could not use either of our standard methods for solving shown in this report (analytical or numerical methods), which are for linear systems only.

Secondly, since there are resistors in our original circuit, the output will eventually follow an exponential decay down to 0 with real world dependent sources. It is expected to follow an exponential decay, but not with the same rate as it would be without any dependent sources. The output will of course depend on the gain of the dependent sources, affecting the rate the output changes.

Appendix A

Here is my C++ code to dynamically approximate the solution to any order differential equation (odeSolve.cpp):

```
 #include <iostream>
#include <fstream>

using namespace std;
void getCoefficients(int order, double *coefficients);
void getInitials(int order, double *initials);
float getDeltaT();
float getStartTime();
float getEndTime();
void initializeX(int order, double *x, double *initials);
void initializeA(int order, double dt, double **A, double *coeffiecients);
void matrixVecMult(int order, double **A, double *x, double *newx);

int main() {
    // open file to write output to
    ofstream outfile("output.txt");
```

```
    // ask user for the order of differential equation and store in order
    int order = 1;
    cout << "Enter the order of the differential equation: ";
    cin >> order;

    // get user input (coefficients, intials, delta t, start time, and end time)
    double coefficients[order];
    getCoefficients(order, coefficients);
    double initials[order];
    getInitials(order, initials);
    float dt = getDeltaT();
    float startTime = getStartTime();
    float endTime = getEndTime();

    // create x vector (start with intial conditions)
    double x[order];
    initializeX(order, x, initials);
    // create matrix I + A(dt)
    double **A;
    A = new double* [order];
    for (int i = 0; i < order; i++) {
        A[i] = new double[order];
    }
    initializeA(order, dt, A, coefficients);

    // set up time and counting variables
    float runTime = startTime - endTime;
    float numPoints = runTime / dt;
    double newx[order];
    for (float t = startTime; t <= endTime; t += dt) {
        matrixVecMult(order, A, x, newx);
        // write newx's x1 to file with time t
        outfile << t << " " << newx[0] << endl;
        // change x = newx for next iteration
        for (int i = 0; i < order; i++) {
            x[i] = newx[i];
        }
    }

    // close output file
    outfile.close();
}

void matrixVecMult(int order, double **A, double *x, double *newx) {
    double sum;
    for (int i = 0; i < order; i++) {
        sum = 0;
        for (int j = 0; j < order; j++) {
            sum += A[i][j] * x[j];
        }
        newx[i] = sum;
    }
}

void initializeX(int order, double *x, double *initials) {
    cout << "x vector: \n";
    for (int i = 0; i < order; i++) {
        x[i] = initials[i];
```

```
        cout << x[i] << "\n";
    }

}

void initializeA(int order, double dt, double **A, double *coeffiecients) {
    for (int i = 0; i < order; i++) {
        for (int j = 0; j < order; j++) {
            if (j == (i + 1)) {
                A[i][j] = 1;
            }
            // last row
            if (i == (order - 1)) {
                A[i][j] = -1 * coeffiecients[j];
            }
            // multiply by dt
            A[i][j] = A[i][j] * dt;
            // add identity matrix now
            if (i == j) {
                A[i][j] = A[i][j] + 1;
            }
        }
    }

    // print A matrix
    cout << "Matrix I + A(dt):\n";
    for (int i = 0; i < order; i++) {
        for (int j = 0; j < order; j++) {
            cout << A[i][j] << " ";
        }
        cout << "\n";
    }
}


void getCoefficients(int order, double *coefficients) {
    // ask user for constant coeffiecients
    cout << "Enter coefficients from a_" << order-1 << " down to a_0:\n";
    int index = 0;
    for (int i = order-1; i >= 0; i--, index++) {
        cout << "coefficient a_" << i << ": ";
        cin >> coefficients[i];
    }
    // print out coeffients
    cout << "coefficients array: ";
    for (int i = 0; i < order; i++) {
        cout << coefficients[i] << " ";
    }
}

void getInitials(int order, double *initials) {
    cout << "\n\nEnter initial conditions:\n";
    for (int i = 0; i < order; i++) {
        // ask for specific y'(t) inital condition
        cout << "Intial condition for y";
        for (int j = 0; j < i; j++) {
            cout << "'";
        }
```

```
        cout << "(t): ";
        cin >> initials[i];
    }

    // print out initial conditions
    cout << "initials array: ";
    for (int i = 0; i < order; i++) {
        cout << initials[i] << " ";
    }
}

float getDeltaT() {
    cout << "\n\nEnter delta t (dt): ";
    float t;
    cin >> t;
    return t;
}

float getStartTime() {
    cout << "\n\nEnter start time: ";
    float t;
    cin >> t;
    return t;
}

float getEndTime() {
    cout << "\n\nEnter end time: ";
    float t;
    cin >> t;
    return t;
}
```

Appendix B: handwork attached below

2. For Differential Equation
$$(D + 2.5)\, y_0(t) = 0 \quad \text{with } y_0(0) = 3$$

a) Find and plot analytical solution for $0 \le t \le 10$

$$\lambda + 2.5 = 0 \qquad \lambda = -2.5$$

$$y_0(t) = Ce^{-2.5t}$$

initial

$$3 = C$$

$$\boxed{y_0(t) = 3e^{-2.5t}} \text{ for } t \ge 0$$

plot on matlab

b) program plot on matlab

c) Comparison

---

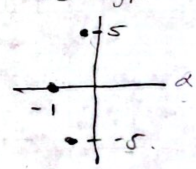
3. For Differential Equation
$$(D^3 + 0.6D^2 + 25.1125D + 2.5063)\, y_0(t) = 0$$
with initial conditions $y_0(0) = 1.5$, $\dot{y}_0(0) = 2$, $\ddot{y}_0(0) = -1$

a) Find and plot analytical solution

Using online cubic root calculator as approved:

$$\lambda = -0.1, \quad \lambda = -0.25 - 5j, \quad \lambda = -0.25 + 5j$$

roots:



$$y_0(t) = Ce^{-0.1t} + Ae^{-0.25t}\cos(5t + \theta)$$

$$\dot{y}_0(t) = -0.1Ce^{-0.1t} - 0.25Ae^{-0.25t}\cos(5t + \theta) - 5Ae^{-0.25t}\sin(5t + \theta)$$

$$\ddot{y}_0(t) = 0.01Ce^{-0.1t} + 0.0625Ae^{-0.25t}\cos(5t + \theta) + 1.25Ae^{-0.25t}\sin(5t + \theta)$$
$$+ 1.25Ae^{-0.25}\sin(5t + \theta) - 25Ae^{-25t}\cos(5t + \theta)$$

$$\Rightarrow 1.5 = C + A\cos(\theta)$$
$$2 = -0.1C - 0.25A\cos(\theta) - 5A\sin(\theta)$$
$$-1 = 0.01C + 0.0625A\cos(\theta) + 1.25A\sin(\theta) + 1.25A\sin(\theta) - 25A\cos(\theta)$$

Using waveform from program, I know $C = 1.5$ and $\theta = \frac{\pi}{2}$

so $2 = -0.1(1.5) - 5A \Rightarrow A = -0.43$

$$C = 1.5 \quad A = -0.43 \quad \theta = \frac{\pi}{2} \leftarrow$$ (checked with online calculator as well)

$$\boxed{y_0(t) = 1.5e^{-0.1t} - 0.43e^{-0.25t}\cos\left(5t + \tfrac{\pi}{2}\right)}$$

---

b) Put 3rd order Diff Eq into state-space form

let $x_1(t) = y_0(t)$, $x_2(t) = \dot{y}_0(t)$, $x_3(t) = \ddot{y}_0(t)$

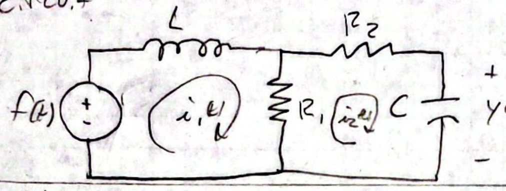
$$\dot{x}(t) = Ax(t) \qquad A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -2.5063 & -25.1125 & -0.6 \end{bmatrix}$$

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -2.5063 & -25.1125 & -0.6 \end{bmatrix} x(t)$$

c) Program solution and plot

d) Compare

4. For circuit



$R_1 = 1\ k\Omega$
$R_2 = 22\ k\Omega$
$C = 10\ \mu F$
$L = 5\ H$

a) Determine Diff Eq relating input $f(t)$ to output $y(t)$

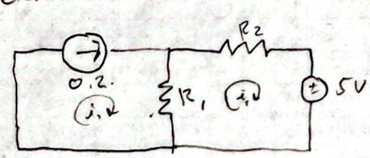
$$f(t) = L\,D i_1(t) + R_1(i_1(t) - i_2(t))$$
$$0 = R_2(i_2(t)) + \frac{1}{CD} i_2(t) + R_1(i_2(t) - i_1(t))$$
$$\Rightarrow f(t) = (LD + R)i_1(t) - R_1 i_2(t)$$
$$0 = -R_1 i_1(t) + \left(\frac{1}{CD} + R_1 + R_2\right) i_2(t)$$
$$\Rightarrow \begin{bmatrix} LD + R_1 & -R_1 \\ -R_1 & \frac{1}{CD} + R_1 + R_2 \end{bmatrix} \begin{bmatrix} i_1(t) \\ i_2(t) \end{bmatrix} = \begin{bmatrix} f(t) \\ 0 \end{bmatrix}$$

where $i_2(t) = CD\,y(t)$

$$\Rightarrow i_2(t) = \frac{\begin{vmatrix} LD + R_1 & f(t) \\ -R_1 & 0 \end{vmatrix}}{\begin{vmatrix} LD + R_1 & -R_1 \\ -R_1 & \frac{1}{CD} + R_1 + R_2 \end{vmatrix}} = \frac{R_1 f(t)}{(LD + R_1)(\frac{1}{CD} + R_1 + R_2) - R_1^2}$$

$$\Rightarrow \left[\left(\frac{1}{CD} + R_1 + R_2\right)(LD + R_1) - R_1^2\right] CD\,y(t) = R_1 f(t)$$
$$= \left[\frac{L}{C} + LDR_1 + LDR_2 + \frac{R_1}{CD} + \cancel{R_1^2} + R_1R_2 - \cancel{R_1^2}\right] CD\,y(t) = R_1 f(t)$$
$$= \left[D^2(LR_1C + LR_2C) + D(R_1R_2C + L) + R_1\right] y(t) = R_1 f(t)$$
$$= \left[D^2 + D\frac{(R_1R_2C + L)}{(LR_1C + LR_2C)} + \frac{(R_1)}{(LR_1C + LR_2C)}\right] y(t) = R_1 f(t)$$
$$= \boxed{\left[D^2 + 195.652D + 869.565\right] y(t) = R_1 f(t)}$$

b) Determine initial condition for $\dot{y}_0(t)$ if $i_1(0) = 0.2\ A$ and $y(0) = 5\ V$



$$i_2(0) = C\dot{y}(0)$$
$$R_2(i_2(0)) + 5 + R_1(i_2(0) - 0.2) = 0$$
$$i_2(0) = \frac{0.2R_1 - 5}{R_1 + R_2} = 0.00848$$
$$\boxed{\dot{y}(0) = \frac{i_2(0)}{C} = 847.826\ \frac{V}{s}}$$

4 cont.

c) Determine analytical solution for the zero-input response

$$[D^2 + 195.652D + 869.565]y(t) = 0 \qquad \frac{-195.652 \pm \sqrt{195.652^2 - 4(869.565)}}{2}$$

$$\lambda = -4.5503, -191.1019$$

$y_o(0) = 5$

$\dot{y}_o(0) = 847.826$

$$y_o(t) = C_1 e^{-4.5503t} + C_2 e^{-191.1019t}$$

$$\dot{y}_o(t) = -4.5503C_1 e^{-4.5503t} - 191.1019C_2 e^{-191.1019t}$$

$$\Rightarrow \begin{aligned} 5 &= C_1 + C_2 \\ 847.826 &= -4.5503C_1 - 191.1019C_2 \end{aligned} \Rightarrow \begin{bmatrix} 1 & 1 \\ -4.5503 & -191.1019 \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} = \begin{bmatrix} 5 \\ 847.826 \end{bmatrix}$$

$$\Rightarrow C_1 = 9.\bar{6} \qquad C_2 = -4.\bar{6}$$

$$\Rightarrow \boxed{y_o(t) = 9.\bar{6}e^{-4.5503t} - 4.\bar{6}e^{-191.1019t}}$$

d) Represent the differential equation in state space form

let $x_1(t) = y_o(t)$, $x_2(t) = \dot{y}_o(t)$

$$\dot{x}(t) = Ax(t) \qquad A = \begin{bmatrix} 0 & 1 \\ -869.565 & -195.652 \end{bmatrix}$$

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 \\ -869.565 & -195.652 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}$$

e) Program solution and analytical solution plots

f) Compare analytical and numerical

g) Suppose the circuit has non-linear element, such as dependent source. Describe how analytical and numerical solution would change

First the important part is "non-linear" element that means we could not use either of our standard methods for solving. Because there are resistors in our original circuit, the output will eventually decay down to 0. It is expected to follow an exponential decay, but not with the same rate as it would be without any real world dependent sources.