Hutomo Saleh
461980

CRES 2nd Assignment

Submitted by: Hutomo Saleh

1) Using the backward-Euler method, the impedance is: [1.0]

$$Z_{CbE} = W_{BE}(s) \cdot \frac{1}{C}$$

If we insert $W_{BE}(s)$ from previous assignment, we get: [1.1]

$$Z_{CbE} = \underbrace{\frac{1}{4 f_{NY} C}}_{R_{CbE}} + \frac{1}{j2\pi f} \cdot \frac{1}{C_{tr}(f)}$$

[1.2]

$$\Rightarrow R_{CbE} = \frac{1}{4 f_{NY} C}$$

2) The Trapezoidal Method is: [2.0]

$$\frac{x(k) - x(k-1)}{\tau} = \frac{\lambda}{2} \cdot \left(x(k) + x(k-1)\right)$$

The solution is: [2.1]

$$x(k) = K \cdot z^k$$

With $K = x(0) = x_0$, we get: [2.2]

$$x(k) = x_0 \cdot z^k$$

Substituting with [2.2] into [2.0] leads to: [2.3]

$$\frac{x_0 \cdot z^k - x_0 \cdot z^{k-1}}{\tau} = \frac{\lambda}{2} \left(x_0 \cdot z^k + x_0 \cdot z^{k-1} \right)$$

[2.4]

$$z^k - z^{k-1} = \frac{\tau \lambda}{2} \left(z^k + z^{k-1} \right)$$

[2.5]

$$1 - \frac{1}{z} = \frac{\tau \cdot \lambda}{2} \left(1 + \frac{1}{z} \right)$$

[2.6]

$$z = \frac{1 + \frac{\tau \lambda}{2}}{1 - \frac{\tau \lambda}{2}}$$

Hutomo Saleh
461980

* Substituting with [2.6] into [2.9] will result in:

$$x(t) = x_0 \cdot \left(\frac{1 + \frac{\tau\lambda}{2}}{1 - \frac{\tau\lambda}{2}} \right)^k \qquad [2.7]$$

The region of stability is therefore:

2a) $$|z| = \left| \frac{1 + \frac{\tau\lambda}{2}}{1 - \frac{\tau\lambda}{2}} \right| < 1 \qquad [2.8]$$

Therefore the value of λ must be < 1 in order to be stabel for any τ.

Proof $\Rightarrow$ Case 1 : $-\left(1 + \frac{\tau\lambda}{2}\right) < 1 - \frac{\tau\lambda}{2}$

$$1 + \frac{\lambda\tau}{2} > -1 + \frac{\tau\lambda}{2} \quad \times$$

Case 2 : $1 + \frac{\tau\lambda}{2} < 1 - \frac{\tau\lambda}{2}$

$$2 \cdot \frac{\tau\lambda}{2} < 0$$

$$\tau\lambda < 0 \Rightarrow$$ τ is always positive λ needs to be < 1 !

For it to NOT monotically decrease, we need:

$$z < 0 \qquad [2.9]$$

$$\frac{1 + \frac{\tau\lambda}{2}}{1 - \frac{\tau\lambda}{2}} < 0 \qquad [2.10]$$

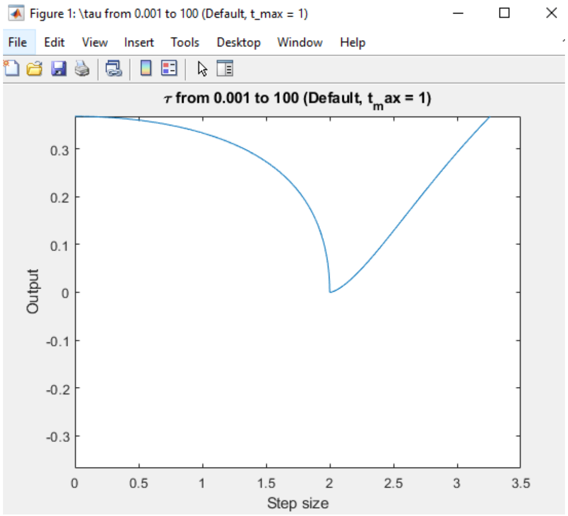
$$1 + \frac{\lambda\tau}{2} < 0$$

$$\frac{\lambda\tau}{2} < -1 \quad \text{with } \lambda = -1$$

$$-\tau < -2$$

$$\tau \geq 2 \Rightarrow$$ Time step size threshold !

2b)



As we can see in Figure 1, if we span the time step size from 0.001 to 100, there will be a change of direction. At τ = 2 the function does not decrease monotonically.

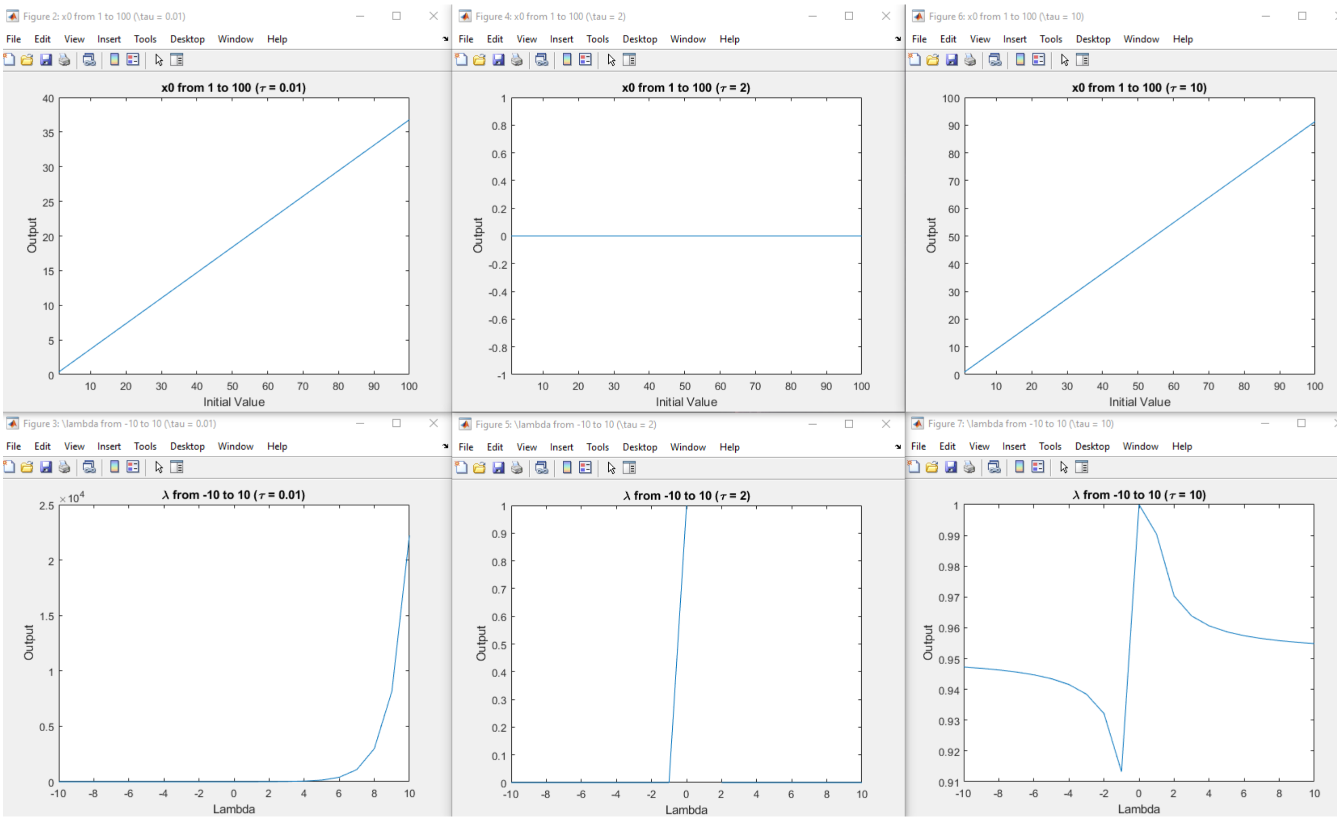


Figure 2 - 7: Curve of x0 and λ on different time step sizes

τ = 0.01 : Figure 2 shows a proportional increase with the increase of x0. This follows the relationship of x0 and the solution in [2.7].
Figure 3 shows that as lambda becomes positive, the output becomes exponentially large. The proportionality is also shown in [2.7]

τ = 2 : Figure 4 due to the output being 0 at exactly τ = 2. The output is always 0 regardless of x0.

τ = 10 : The proportionality of x0 and output is shown again in Figure 5.
Figure 7 shows an unusual behaviour due to the τ being larger than 2.

Hutomo Saleh
461980

3. LU Factorization seperates the values of j and v. The L and U Matrix also consists of nonzero elements which allows us to use methods to improve efficiency.

Code

Hutomo Saleh
461980

```
%% Loop through tau, initial & lambda values
change_tau_value(0.001, 100, 0.001, -10, 1, '\tau from 0.001 to 100 (Default, t_max = 1)');
change_initial_value(1, 100, 1, -1, 0.01, 'x0 from 1 to 100 (\tau = 0.01)');
change_lambda(-10, 10, 1, 1, 0.01, '\lambda from -10 to 10 (\tau = 0.01)');
change_initial_value(1, 100, 1, -1, 2, 'x0 from 1 to 100 (\tau = 2)');
change_lambda(-10, 10, 1, 1, 2, '\lambda from -10 to 10 (\tau = 2)');
change_initial_value(1, 100, 1, -1, 10, 'x0 from 1 to 100 (\tau = 10)');
change_lambda(-10, 10, 1, 1, 10, '\lambda from -10 to 10 (\tau = 10)');


function change_tau_value(tau_start, tau_end, increment, lambda, y0, name)
    %% Calculate
    result = zeros(1, tau_end / increment);  % For Y-Axis values
    x_value = result;  % For X-Axis values
    i = 1;  % Index for inserting into the matrix
    for h = tau_start:increment:tau_end
        result(i) = trapezoidal(h, lambda, y0);
        x_value(i) = h;  % Put tau value for X-Axis range
        i = i + 1;
    end

    %% Plot
    figure('name', name)
    plot(x_value, result(1, :));  % Plot it
    title(name); % Name the window
    y_limit = result(1, 1);  % Define the limits
    ylim([-y_limit y_limit]);  % Set the Y-Axis range
    ylabel("Output")  % Set Axis labels
    xlabel("Step size")
end
function change_initial_value(y0_start, y0_end, increment, lambda, tau, name)
    %% Calculate
    result = zeros(1, y0_end / increment);  % For Y-Axis values
    x_value = result;  % For X-Axis values
    i = 1;  % Index for inserting into the matrix
    for y0 = y0_start:increment:y0_end
        result(i) = trapezoidal(tau, lambda, y0);
        x_value(i) = y0;  % Put tau value for X-Axis range
        i = i + 1;
    end

    %% Plot
    figure('name', name)
    plot(x_value, result(1, :));  % Plot it
    title(name); % Name the window
    xlim([1 100]);  % Set the Y-Axis range
    ylabel("Output")  % Set Axis labels
    xlabel("Initial Value")
end

function change_lambda(lambda_start, lambda_end, increment, y0, tau, name)
    %% Calculate
    result = zeros(1, lambda_end / increment);  % For Y-Axis values
    x_value = result;  % For X-Axis values
    i = 1;  % Index for inserting into the matrix
    for lambda = lambda_start:increment:lambda_end
        result(i) = trapezoidal(tau, lambda, y0);
        x_value(i) = lambda;  % Put tau value for X-Axis range
        i = i + 1;
    end

    %% Plot
    figure('name', name)
    plot(x_value, result(1, :));  % Plot it
    title(name); % Name the window
    xlim([-10 10]);  % Set the Y-Axis range
    ylabel("Output")  % Set Axis labels
    xlabel("Lambda")
end

function y = trapezoidal(tau, lambda, y0)
    t_max = 1;
    k = t_max / tau;
    y = (1 + 0.5 * tau * lambda)^k / (1 - 0.5 * tau * lambda)^k * y0;
end
```