Y parameters

Name: Mohammed AL Shuaili

Date: 8/2/2025

Objectives:

1. Implement Y parameters
2. Generate Y parameters from TL model.
3. Code Y parameters with different frequencies.
4. Final FYP report

Let's consider the following example to code this in MATLAB, assume the rational approximation for Y_{21} and Y_{22} (denominators must match):

$$Y_{21}(s) = \frac{2s+3}{s^2+4s+5}\ , \qquad Y_{22}(s) = \frac{s+6}{s^2+4s+5}.$$

So,

$$\frac{V_R}{V_s} = -\frac{Y_{21}(s)}{Y_{22}(s)} = -\frac{\frac{2s+3}{s^2+4s+5}}{\frac{s+6}{s^2+4s+5}} = -\frac{(2s+3)}{(s+6)}$$

$$\frac{V_R}{V_s} = \frac{-(2s+3)}{s+6}$$

To make it proper (numerator degree < denominator degree) we can say:

$$\frac{V_R}{V_s} = -2 + \frac{9}{s+6}$$

To make this proper we can use the following method:

$$H(s) = \frac{N(s)}{D(s)}$$

If deg(N) ≥ deg(D), perform polynomial division:

$$N(s) = Q(s)D(s) + R(s)$$

Then the proper form is:

$$H(s) = Q(s) + \frac{R(s)}{D(s)}$$

Where deg(R) < deg(D). In MATLAB, this can easily be implemented as follows using deconvolution

```
clear all
clc
N = [-2 -3];% Define the numerator coeff
D = [1 6]; % Define the den coeff
[Q,R]=deconv(N,D);% Perform polynomial division
R = tf(R,D);
```

Then, the state space model is as follows:

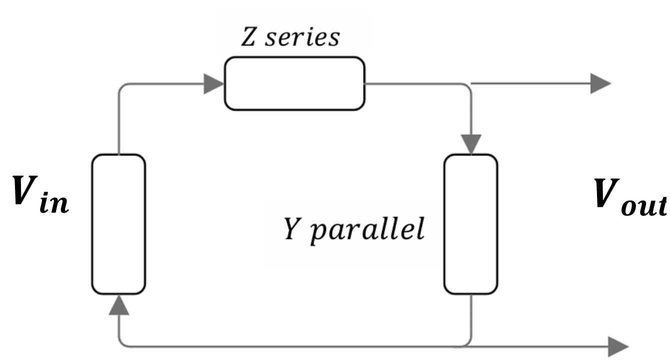
$$A = [-6], \quad B = [1], \quad C = [9], \quad D = -2$$

Then, AWE can be implemented to get the response and compare it to the actual one.

The following code will generate state space representation with Y21 and Y22:

```
clear all
clc
% Define numerator coefficients of Y21 and Y22, and common denominator
num_Y21 = [2 3];
num_Y22 = [1 6];
den = [1 4 5];
N = -num_Y21;
D = num_Y22;
% Perform polynomial division to make it strictly proper
[Q, R] = deconv(N, D);
% check leading coefficient (assumed to be 1)
if D(1) ~= 1
    D=D/D(1);
    N = N/D(1);
    [Q, R] = deconv(N, D);
end
% Extract coefficients for state-space representation
g = D(2:end);  % Exclude leading coefficient ( g terms )
% f terms
if R(1) ==0
    f=R(2:end);
else
    f = R;
end
% Construct state-space matrices
n = length(g); % Order of system
A = [zeros(n-1,1), eye(n-1); -flip(g)];
B = [zeros(n-1,1); 1];
C = flip(f);
D = Q;
```

- Generate Y parameters from TL model and compare it to the actual one.



From the exact solution, we can say that:

$$I_s = \frac{1}{Z_{series}} \, (V_s - 0) + Y_{parallel} \, V_s$$

$$so, \frac{I_s}{V_s} = \frac{1}{Z_{series}} + Y_{parallel}$$

From the Y parameters if $I_R = 0$:

$$\begin{bmatrix} I_s \\ -I_R \end{bmatrix} = \begin{bmatrix} Y_{11} & Y_{12} \\ Y_{21} & Y_{22} \end{bmatrix} \begin{bmatrix} V_s \\ V_R \end{bmatrix}$$

$$\frac{I_s}{V_s} = Y_{11}$$

$$Y_{11} = \frac{1 + Y_{parallel} Z_{series}}{Z_{series}}$$

Where,

$$Z_{series} = Z_o \sinh(\gamma l)$$

$$Z_o = \sqrt{\frac{z}{y}} \, , \qquad Y = G + sC, \qquad Z = (R + sL), \quad \gamma = \sqrt{ZY}$$

$$Y_{parallel} = Y_o \tanh\left(\frac{\gamma l}{2}\right), \quad Y_o = \frac{1}{Zo} \, , \quad Z_{parallel} = \frac{1}{Y_{parallel}}$$

So,

$$Y_{11} = \frac{1 + Y_o \tanh\left(\frac{\gamma l}{2}\right) Z_o \sinh(\gamma l)}{Z_o \sinh(\gamma l)}$$

$$Y_{11} = \frac{1 + \tanh\left(\frac{\gamma l}{2}\right) \sinh(\gamma l)}{Z_o \sinh(\gamma l)}$$

$$Y_{11} = \frac{\cosh\,(\gamma l)}{Z_o \, \sinh(\gamma l)}$$

$$Y_{11} = \frac{\cosh\,(\sqrt{(R + sL)(G + sC)} l)}{\sqrt{\frac{(R + sL)}{G + sC}} \, \sinh\left(\sqrt{(R + sL)(G + sC)} \, l\right)}$$

At s=iw

$$Y_{11}(s = iw) = \frac{\cosh\left(\sqrt{(R + iwL)(G + iwC)}l\right)}{\sqrt{\frac{(R + iwL)}{G + iwC}}\ \sinh\left(\sqrt{(R + iwL)(G + iwC)}\ l\right)}$$

Let's plot this at different values for w using the following code in MATLAB:

```
clear
clc
l = 400;
R = 0.01;
L = 2.5e-7;
C = 1e-10;
G=0;
f=0:1:5000;
w = 2*pi*f;
s=1i*w;
Z = (R+s*L);
Y = G+s*C;
Zo = sqrt(Z./Y);
gama = sqrt(Z.*Y);
Y11= cosh(gama .* l)./(Zo.*sinh(gama*l));
```

w	Y_{11}
2π	0.25 - 0.0000i
100π	0.2500 - 0.0020i
1000π	0.2485 - 0.0195i
5000π	0.2166 - 0.0848i
10000π	0.1546 - 0.1210i

- Generate Y parameters out of these values:

$$Y_{11} = \frac{(a_{nii}s^{n-1} + .. + a_{011})}{s^n + .. + b_{011}}$$

Consider n=2:

$$Y_R + Y_i = \frac{a_1 s + a_0}{s^2 + b_1 s + b_0}$$

At $s = jw$

$$(Y_R + jY_i)(-w^2 + b_1 jw + b_0) = a_1 jw + a_0$$

$$(-w^2 Y_R + b_1 jwY_R + b_0 Y_R - w^2 jY_i - b_1 wY_i + jb_0 Y_i) = a_1 jw + a_0$$

Re-write this as:

$$(b_1 jwY_R + b_0 Y_R - b_1 wY_i + jb_0 Y_i) - a_1 jw - a_0 = w^2 Y_R + jw^2 Y_i$$

To find the coefficients we can use the following:

Say B matrix contains the coefficients as follows:

$$B = \begin{bmatrix} a_0 \\ b_0 \\ a_1 \\ b_1 \end{bmatrix}$$

So,

$$A = \begin{bmatrix} -1 & Y_R^1 & 0 & -Y_i^1 w \\ 0 & Y_i^1 & -jw^1 & jwY_R^1 \\ \dots & .. & .. & .. \\ -1 & Y_R^N & 0 & -Y_i^N w \\ 0 & Y_i^N & -jw^N & -w^2 Y_i^N \end{bmatrix} \; and \; C = \begin{bmatrix} Y_R^1 w^2 \\ jY_i^1 w^2 \\ \dots \\ Y_R^N w^2 \\ jY_i^N w^2 \end{bmatrix} \; where \; A\,B = C \; and \; B = A \backslash B.$$

Where N is the number of data points (i.e w). the first 2 rows are repeated depending on how many data points we are considering.

For example, let's say we are considering the last 3 points from the table above, then A will have 4 by 6 rows (2*N).

$$A = \begin{bmatrix} -1 & Y_R^{(1)} & 0 & -Y_i^{(1)} w \\ 0 & Y_i^{(1)} & -jw^{(1)} & jwY_R^{(1)} \\ -1 & Y_R^{(2)} & 0 & -Y_i^{(2)} w \\ 0 & Y_i^{(2)} & -jw^{(2)} & jwY_R^2 \\ -1 & Y_R^{(3)} & 0 & -Y_i^{(3)} w \\ 0 & Y_i^{(3)} & -jw^{(3)} & jwY_R^{(3)} \end{bmatrix} \; and \; C = \begin{bmatrix} Y_R^1 w^2 \\ jY_i^1 w^2 \\ Y_R^{(2)} w^2 \\ jY_i^{(2)} w^2 \\ Y_R^{(3)} w^2 \\ jY_i^{(3)} w^2 \end{bmatrix}$$

In MATLAB:

```
clear all
clc
% Given the last 3 points
Yr = [0.2485, 0.2166, 0.1546];  % Real part of Y11
Yi = [-0.0195, -0.0848, -0.1210]; % Imaginary part of Y11
w = [1000*pi, 5000*pi, 10000*pi];
A = [];
C = [];
% Loop through each frequency point to construct A and C
for k = 1:length(w)
    wk = w(k);
    Yr_k = Yr(k);
    Yi_k = Yi(k);
    % Construct rows for A and C
    A_row1 = [-1, Yr_k, 0, -wk*Yi_k]; % Real part
    A_row2 = [0, Yi_k, -wk, wk*Yr_k]; % Imaginary part
```

```
    % Append to A
    A = [A; A_row1; A_row2];
    % C
    C_row1 = wk^2 * Yr_k; % Real part
    C_row2 = wk^2 * Yi_k; % Imaginary part
    % Append to C
    C = [C; C_row1; C_row2];
end

% Solve for B = [a0; b0; a1; b1]
B = A \ C;
% get cof
a0 = B(1);
b0 = B(2);
a1 = B(3);
b1 = B(4);
f = 0:100:10000;
w = 2*pi*f;
s = i*w;
% generated H
H = (a1*s+a0)./(s.^2+b1*s+b0);
l = 400;
R = 0.01;
L = 2.5e-7;
C = 1e-10;
G=0;
Z = (R+s.*L);
Y = G+s.*C;
Zo = sqrt(Z./Y);
gama = sqrt(Z.*Y);
Y11= cosh(gama.* l)./(Zo.*sinh(gama.*l));
% find the error
n =length(H)-1;
Error = abs(H-Y11);
Error = Error(2:end);
Error = sum(Error)/n;
% plot both in the same figure
figure(1)
plot(f,H,f,Y11,'r *');
legend('generated Y' ,'Exact Y')
grid on
```

The following figure shows the generated Y parameter compared to the exact one with Mean Absolute Error of 9.3115e-05.