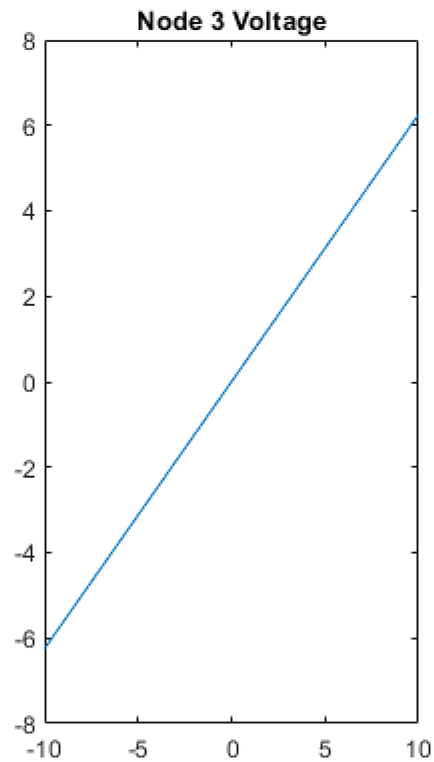
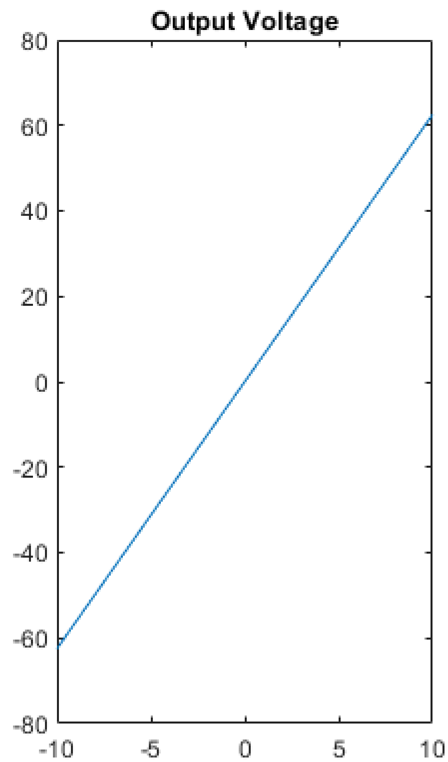
Section 1 - PA 9

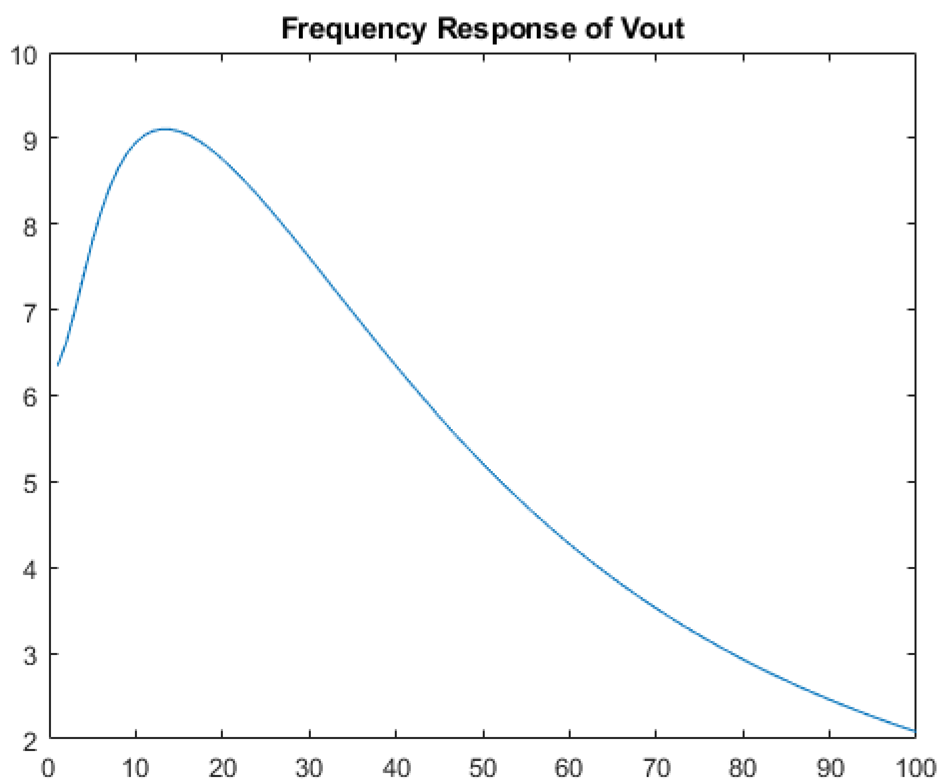
In this section we analyze a complex circuit using MNA techniques. Starting with the equations obtained from applying Kirchoff's Current Law, a G-matrix can be formed to model the conductivities present in the circuit, and a C-matrix for the reactive elements in the circuit. A DC analysis (w=0) is done to sweep a voltage range for the circuit to obtain a DC gain value of roughly 10. An AC sweep is done to obtain the frequency response of the circuit, revealing its low pass filter/amplifier characteristics. Finally, a histogram is created to record the gain response to a series of perturbations which have their frequency normally distributed about pi with a standard deviation of 0.05. The resulting histogram is not surprisingly, also gaussian in shape, centered about 8.5dB.

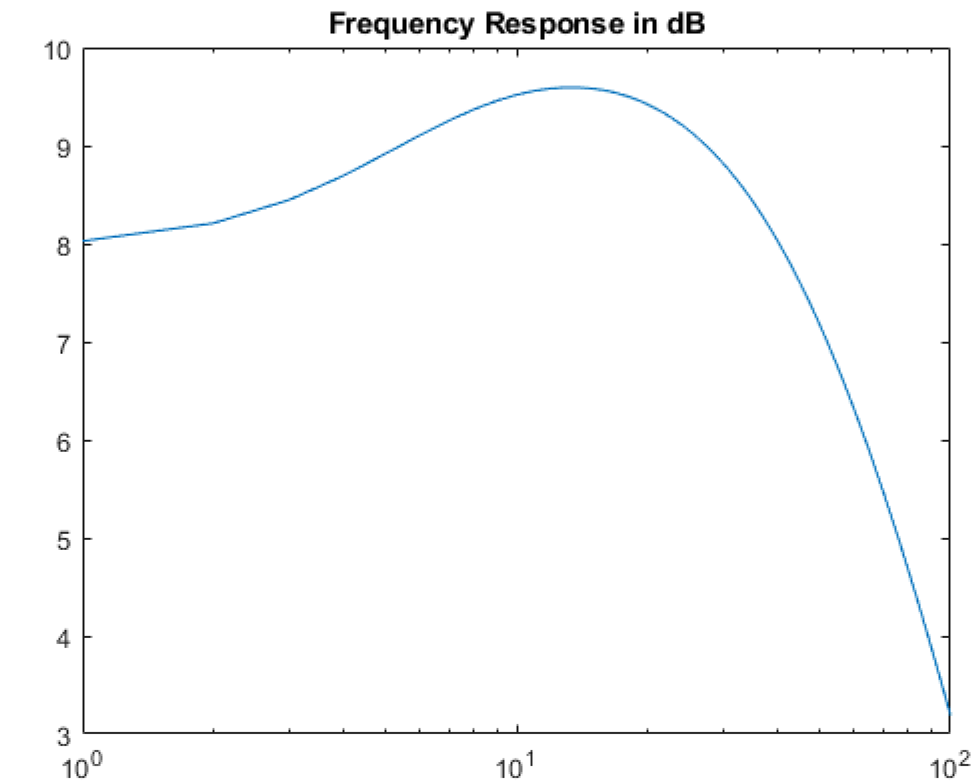
```
%defining component values
R1 = 1;
R2 = 2;
R3 = 10;
R4 = 0.1;
R5 = 1000;
g1 = 1/R1;
g2 = 1/R2;
g3 = 1/R3;
g4 = 1/R4;
g5 = 1/R5;
L = 0.2;
C = .25;
a = 100;

%Capacitive matrix
C = [C -C 0 0 0 0 0;
    -C C 0 0 0 0 0;
     0 0 0 0 0 0 0;
     0 0 0 0 0 0 0;
     0 0 0 0 0 0 0;
     0 0 0 0 0 0 -L;
     0 0 0 0 0 0 0];
%Conductivity matrix
 G =   [g1 -g1 0 0 0 1 0 ;
        -g1 g1+g2 0 0 0 0 1;
        0 0 g3 0 0 0 -1;
        0 0 0 1 0 0 -a;
        0 0 0 -g4 g4+g5 0 0 ;
        0 1 -1 0 0 0 0 ;
        1 0 0 0 0 0 0];
    Vmid = zeros(1,21);
    Vout = zeros(1,21);

    %input DC voltage sweep
    for Vin = -10:10
    F = [0 0 0 0 0 0 Vin]';
    V = G\F;
    Vmid(Vin+11) = V(3);
    Vout(Vin+11) = V(5);
    end
```

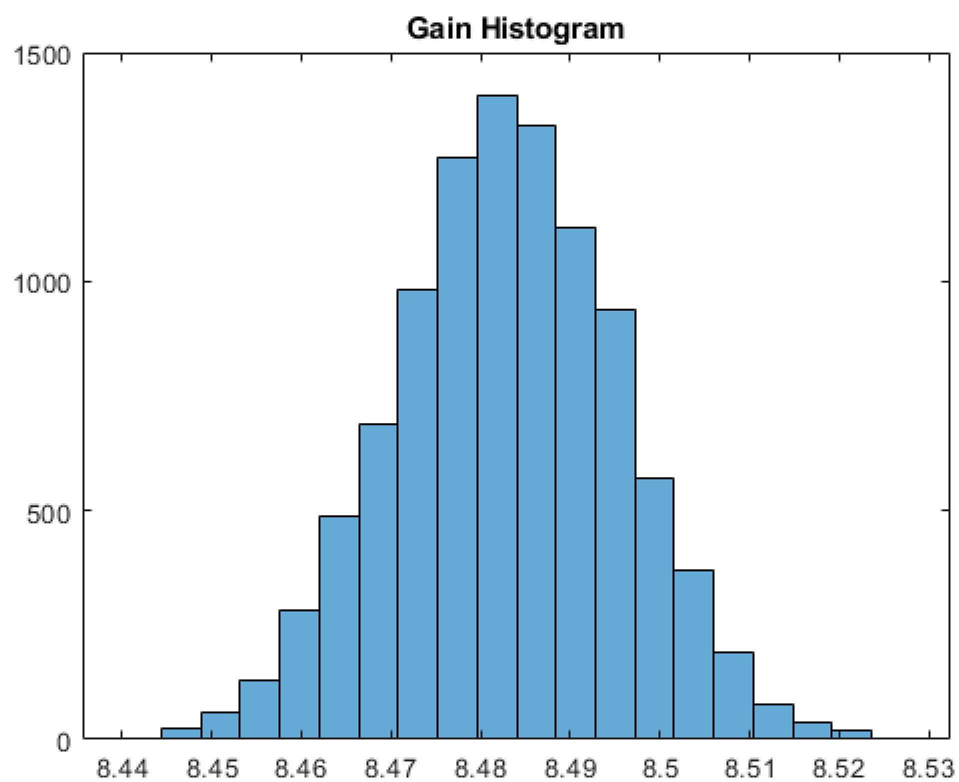
```
    %plot DC voltage characteristic across gain stage
    Vin = -10:10;
    figure(1)
    subplot(1,2,1);
    plot(Vin,real(Vmid))
    title('Node 3 Voltage')
    subplot(1,2,2);
    plot(Vin,real(Vout))
    title('Output Voltage')

    Vout = zeros(1,100);
    %frequency domain sweep
    Vin=1;
for w=1:100
    %sweep from 1 to 100 rad/s
    %F stays at constant, Vin = 1V
    F = [0 0 0 0 0 0 Vin]';
    V = (G+1i*w*C)\F;
    %storing values of Vout for each iteration of frequency
    Vout(w) = V(5);
end
w=1:100;
figure(2)
%plot frequency response of Vout
plot(w,real(Vout))
title('Frequency Response of Vout')
figure(3)
%plotting gain in dB's
semilogx(w,10*log10(real(Vout)/Vin))
title('Frequency Response in dB')

w = pi+randn(1,10000)*0.05;
Vout = zeros(1,10000);
for n = 1:10000
    F = [0 0 0 0 0 0 Vin]';
    V = (G+1i*w(n)*C)\F;
    Vout(n) = V(5);
end
gain = 10*log10(real(Vout)/Vin);
%plotting frequency response to gaussian perturbations
figure(4)
histogram(gain,20)
title('Gain Histogram')
```


Node 3 Voltage
8
6
4
2
0
-2
-4
-6
-8
-10
-5
0
5
10


Output Voltage
80
60
40
20
0
-20
-40
-60
-80
-10
-5
0
5
10


Frequency Response of Vout
10
9
8
7
6
5
4
3
2
0
10
20
30
40
50
60
70
80
90
100


Frequency Response in dB
10
9
8
7
6
5
4
3
10^0
10^1
10^2


Gain Histogram
1500
1000
500
0
8.44
8.45
8.46
8.47
8.48
8.49
8.5
8.51
8.52
8.53

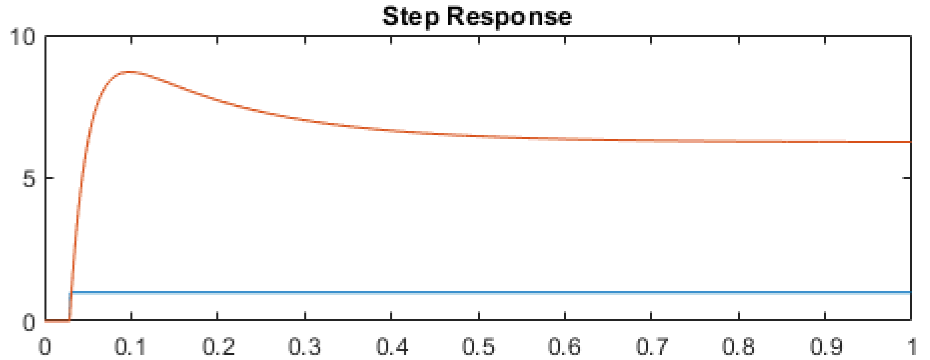
Section 2 - Transient Circuit Simulation

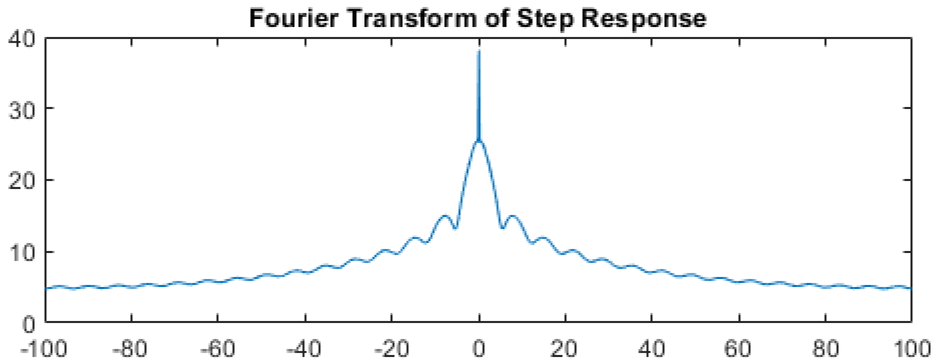
In this section, we use a finite difference approach to numerically simulate the time domain response of the circuit analyzed in the previous section. The circuit is a low pass filter/amplifier. It comes with an expected frequency response of high gain for low frequency inputs, and a rolloff in gain for frequencies above its corner frequency. Various input types are analyzed in the time and frequency domain. A step, sine, and gaussian input are fed into the system, and plots of input/output voltages are produced. In addition, the fft and fftshift techniques are used to expose the frequencies present in these signals. The fft plots show a two sided frequency spectrum, although the negative frequencies don't really have much physical meaning in this analysis.

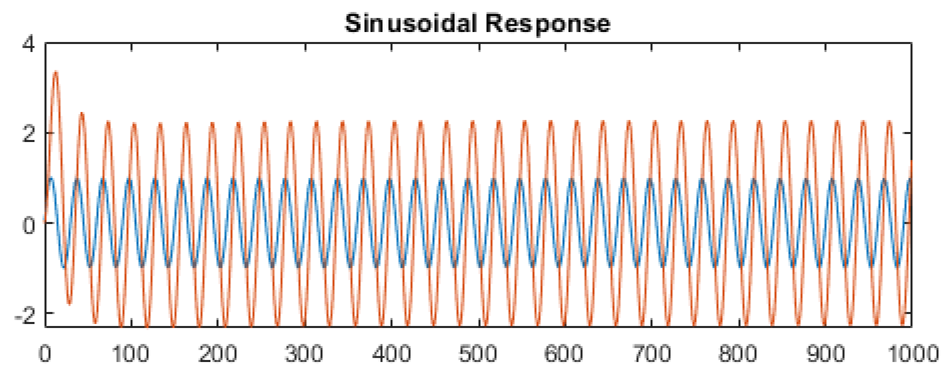
```
%defining component values
R1 = 1;
R2 = 2;
R3 = 10;
R4 = 0.1;
R5 = 1000;
g1 = 1/R1;
g2 = 1/R2;
g3 = 1/R3;
g4 = 1/R4;
g5 = 1/R5;
L = 0.2;
C = .25;
a = 100;
t=0;
dt=1e-3;
Tstop = 1000*dt;
%Capacitive matrix
C = [C -C 0 0 0 0 0;
    -C C 0 0 0 0 0;
     0 0 0 0 0 0 0;
     0 0 0 0 0 0 0;
     0 0 0 0 0 0 0;
     0 0 0 0 0 0 -L;
     0 0 0 0 0 0 0];
%Conductivity matrix
 G =   [g1 -g1 0 0 0 1 0 ;
        -g1 g1+g2 0 0 0 0 1;
        0 0 g3 0 0 0 -1;
        0 0 0 1 0 0 -a;
        0 0 0 -g4 g4+g5 0 0 ;
        0 1 -1 0 0 0 0 ;
        1 0 0 0 0 0 0];

 A = (1/dt)*C +G;
 Vin = zeros(1,1000);
 Vout = zeros(1,1000);
 Vold = zeros(7,1);
 vin = 0;
 iteration=1;
 while t<Tstop
```

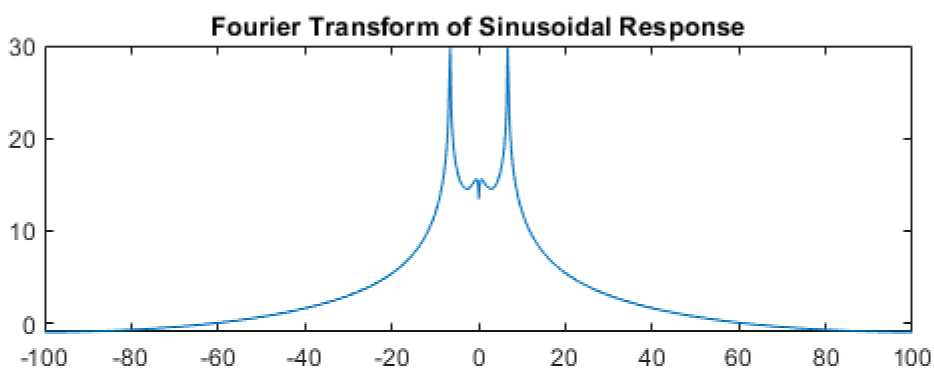
```
    %defining input function
    if (t>30*dt)
        vin = 1;
    end
    F = [0 0 0 0 0 0 vin]';
    V = A\(C*Vold/dt + F);
    Vin(iteration) = vin;
    Vout(iteration) = V(5);
    Vold = V;
    t = t+dt;
    iteration=iteration+1;
end
t = 0:dt:999*dt;
Frequency=linspace(-100,100,1000);
%plotting step response
 figure(5)
 subplot(2,1,1);
 plot(t,Vin)
 hold on
 plot(t,Vout)
 title('Step Response')
 subplot(2,1,2);
 plot(Frequency,10*log10(abs(fftshift(fft(Vout)))))
 title('Fourier Transform of Step Response')

 %resetting variables
Vin = zeros(1,1000);
Vout = zeros(1,1000);
Vold = zeros(7,1);
vin = 0;
t=0;
f = 1/0.03;
iteration=1;
while t<Tstop
    vin = sin(2*pi*f*t);
    F = [0 0 0 0 0 0 vin]';
    V = A\(C*Vold/dt + F);
    Vin(iteration) = vin;
    Vout(iteration) = V(5);
    Vold = V;
    iteration=iteration+1;
    t = t+dt;
end
t = 0:1:999;
%plotting sinusoidal response
 figure(6)
 subplot(2,1,1);
 plot(t,Vin)
 hold on
 plot(t,Vout)
 title('Sinusoidal Response')
 subplot(2,1,2);
 plot(Frequency,10*log10(abs(fftshift(fft(Vout)))))
 title('Fourier Transform of Sinusoidal Response')
```

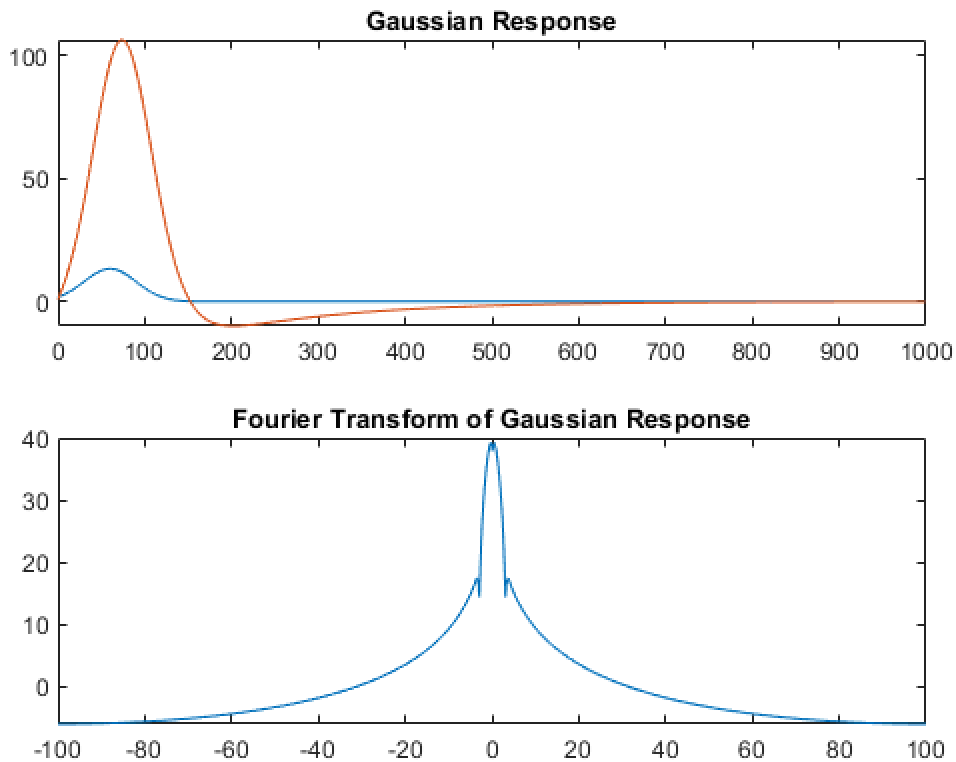
```
 %resetting variables
Vin = zeros(1,1000);
Vout = zeros(1,1000);
Vold = zeros(7,1);
vin = 0;
t=0;
iteration=1;
delay=0.06;
sigma=0.03;
while t<Tstop
   %set up gaussian pulse
    vin = (1/(sqrt(2*pi)*sigma))*exp(-((delay-t)^2)/(2*sigma^2));
    F = [0 0 0 0 0 0 vin]';
    V = A\(C*Vold/dt + F);
    Vin(iteration) = vin;
    Vout(iteration) = V(5);
    Vold = V;
    iteration=iteration+1;
    t = t+dt;
end
t = 0:1:999;
%plotting gaussian response
 figure(7)
 subplot(2,1,1);
 plot(t,Vin)
 hold on
 plot(t,Vout)
 title('Gaussian Response')
 subplot(2,1,2);
 plot(Frequency,10*log10(abs(fftshift(fft(Vout)))))
 title('Fourier Transform of Gaussian Response')
```


Step Response
10
5
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


Fourier Transform of Step Response
40
30
20
10
0
-100
-80
-60
-40
-20
0
20
40
60
80
100


Sinusoidal Response
4
2
0
-2
0
100
200
300
400
500
600
700
800
900
1000


Fourier Transform of Sinusoidal Response
30
20
10
0
-100
-80
-60
-40
-20
0
20
40
60
80
100



Published with MATLAB® R2018a

Section 3 - Circuit with Noise

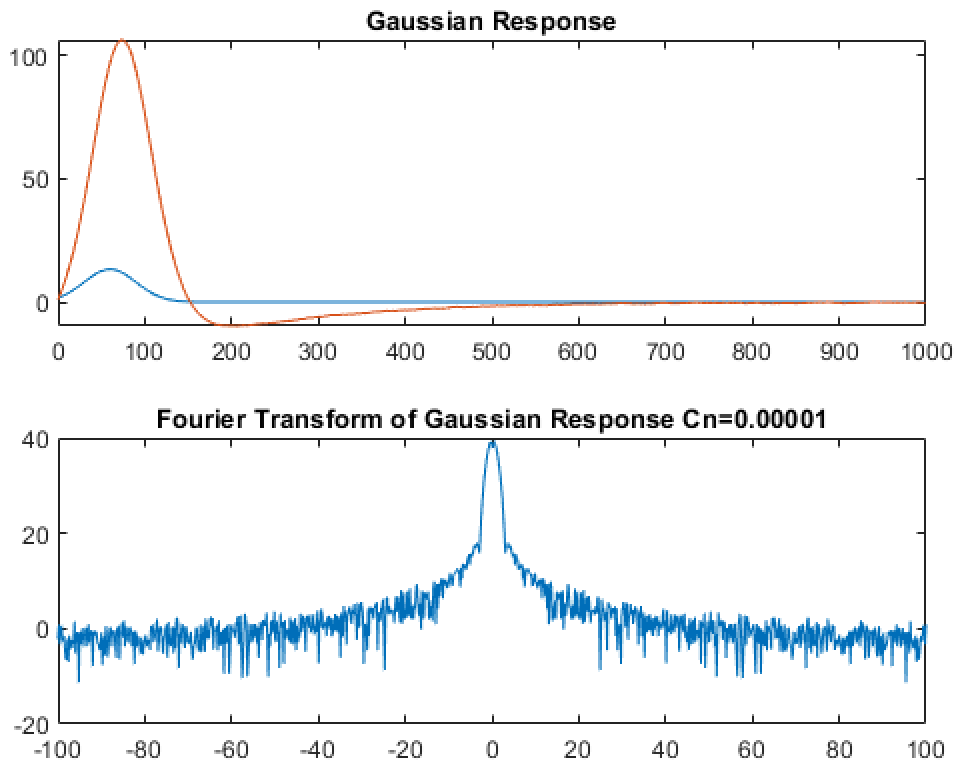
Here we are modelling a variation to the previous circuit, attempting to model the thermal noise generated in a resistor. We need to take two new components into our MNA equations, a current source to model the noise, and a capacitor to bandwidth limit the circuit. The new capacitor Cn will be placed in position (3,3) in the C matrix, and In will be placed in (3,1) in the F matrix. F is randomly generated from a normal distribution with each iteration in time. Section 4 involves implementing a more complex nonlinear transconductance to our system, which involves using the B vector in our solution. We would need to implement a Jacobian which involves a bunch of computation that I don't really want to do. Other than the introduction to the B term whenever the system of equations is solved, most of the rest of the code can be reused.

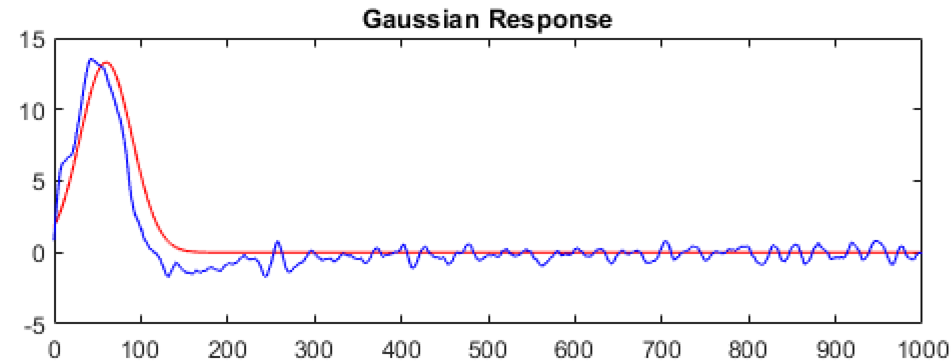
```
%component values
R1 = 1;
R2 = 2;
R3 = 10;
R4 = 0.1;
R5 = 1000;
g1 = 1/R1;
g2 = 1/R2;
g3 = 1/R3;
g4 = 1/R4;
g5 = 1/R5;
L = 0.2;
C1 = .25;
Cn = 0.00001;
a = 100;
t=0;
dt=1e-3;
Tstop = 1000*dt;
%Capacitive matrix
C = [C1 -C1 0 0 0 0 0;
    -C1 C1 0 0 0 0 0;
     0 0 Cn 0 0 0 0;
     0 0 0 0 0 0 0;
     0 0 0 0 0 0 0;
     0 0 0 0 0 0 -L;
     0 0 0 0 0 0 0];
%Conductivity matrix
 G =   [g1 -g1 0 0 0 1 0 ;
        -g1 g1+g2 0 0 0 0 1;
        0 0 g3 0 0 0 -1;
        0 0 0 1 0 0 -a;
        0 0 0 -g4 g4+g5 0 0 ;
        0 1 -1 0 0 0 0 ;
        1 0 0 0 0 0 0];
 A = (1/dt)*C+G;
   %resetting variables
 Vin = zeros(1,1000);
 Vout = zeros(1,1000);
 Vold = zeros(7,1);
 vin = 0;
```

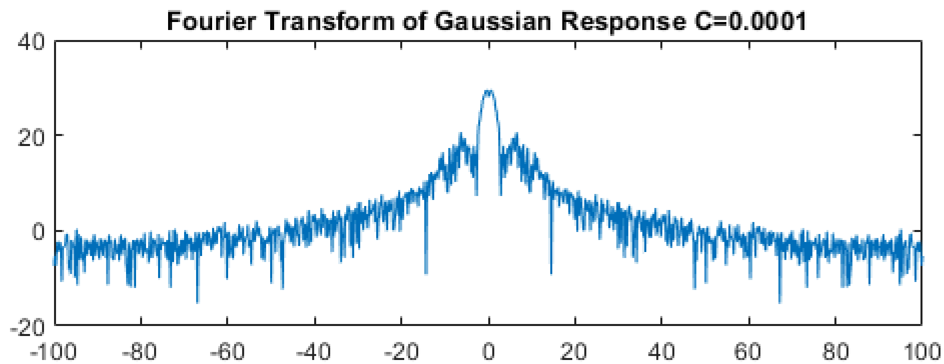
```
t=0;
iteration=1;
delay=0.06;
sigma=0.03;
while t<Tstop
   %set up gaussian pulse
    vin = (1/(sqrt(2*pi)*sigma))*exp(-((delay-t)^2)/(2*sigma^2));
    In = randn*0.01;
    F = [0 0 In 0 0 0 vin]';
    V = A\(C*Vold/dt + F);
    Vin(iteration) = vin;
    Vout(iteration) = V(5);
    Vold = V;
    iteration=iteration+1;
    t = t+dt;
end
t = 0:1:999;
%plotting gaussian response
 figure(8)
 subplot(2,1,1);
 plot(t,Vin)
 hold on
 plot(t,Vout)
 title('Gaussian Response')
 subplot(2,1,2);
 plot(Frequency,10*log10(abs(fftshift(fft(Vout)))))
 title('Fourier Transform of Gaussian Response Cn=0.00001')

 %second plot of noise analysis with C = 0.001 instead of 0.00001
 Cn= 0.0001;
 C = [C1 -C1 0 0 0 0 0;
   -C1 C1 0 0 0 0 0;
    0 0 Cn 0 0 0 0;
    0 0 0 0 0 0 0;
    0 0 0 0 0 0 0;
    0 0 0 0 0 0 -L;
    0 0 0 0 0 0 0];
  Vin = zeros(1,1000);
Vout = zeros(1,1000);
Vold = zeros(7,1);
vin = 0;
t=0;
iteration=1;
delay=0.06;
sigma=0.03;
while t<Tstop
   %set up gaussian pulse
    vin = (1/(sqrt(2*pi)*sigma))*exp(-((delay-t)^2)/(2*sigma^2));
    In = randn*0.01;
    F = [0 0 In 0 0 0 vin]';
    V = A\(C*Vold/dt + F);
    Vin(iteration) = vin;
    Vout(iteration) = V(5);
    Vold = V;
```

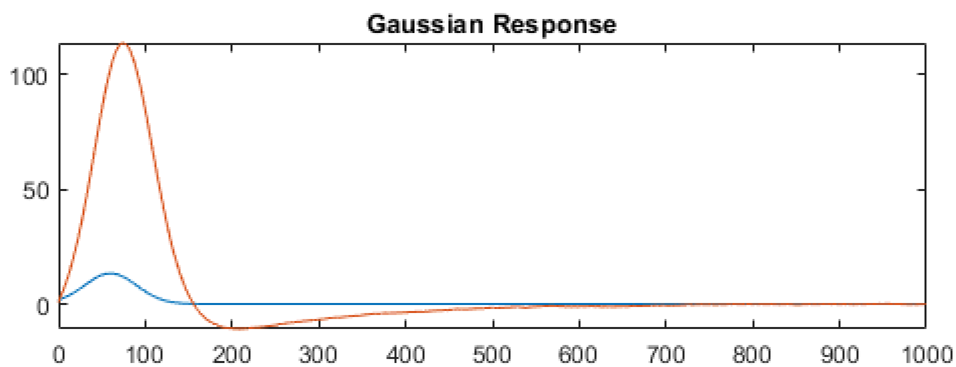
```
    iteration=iteration+1;
    t = t+dt;
end
t = 0:1:999;
%plotting gaussian response
 figure(9)
 subplot(2,1,1);
 plot(t,Vin,'r')
 hold on
 plot(t,Vout,'b')
 title('Gaussian Response')
 subplot(2,1,2);
 plot(Frequency,10*log10(abs(fftshift(fft(Vout)))))
 title('Fourier Transform of Gaussian Response C=0.0001')

 %third plot of noise analysis with C=0.000001
 Cn= 0.000001;
 C = [C1 -C1 0 0 0 0 0;
   -C1 C1 0 0 0 0 0;
    0 0 Cn 0 0 0 0;
    0 0 0 0 0 0 0;
    0 0 0 0 0 0 0;
    0 0 0 0 0 0 -L;
    0 0 0 0 0 0 0];

Vin = zeros(1,1000);
Vout = zeros(1,1000);
Vold = zeros(7,1);
vin = 0;
t=0;
iteration=1;
delay=0.06;
sigma=0.03;
while t<Tstop
   %set up gaussian pulse
    vin = (1/(sqrt(2*pi)*sigma))*exp(-((delay-t)^2)/(2*sigma^2));
    In = randn*0.01;
    F = [0 0 In 0 0 0 vin]';
    V = A\(C*Vold/dt + F);
    Vin(iteration) = vin;
    Vout(iteration) = V(5);
    Vold = V;
    iteration=iteration+1;
    t = t+dt;
end
t = 0:1:999;
%plotting gaussian response
 figure(10)
 subplot(2,1,1);
 plot(t,Vin)
 hold on
 plot(t,Vout)
 title('Gaussian Response')
 subplot(2,1,2);
```

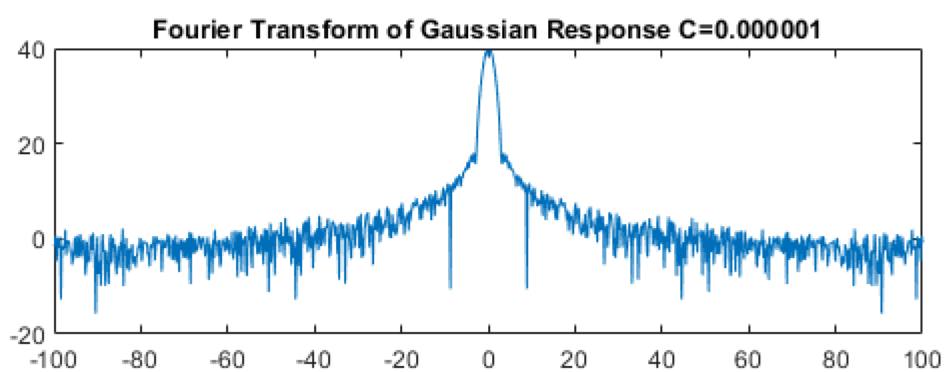
```
plot(Frequency,10*log10(abs(fftshift(fft(Vout)))))
title('Fourier Transform of Gaussian Response C=0.000001')
```




Gaussian Response
15
10
5
0
-5
0 100 200 300 400 500 600 700 800 900 1000


Fourier Transform of Gaussian Response C=0.0001
40
20
0
-20
-100 -80 -60 -40 -20 0 20 40 60 80 100


Gaussian Response
100
50
0
0 100 200 300 400 500 600 700 800 900 1000


Fourier Transform of Gaussian Response C=0.000001
40
20
0
-20
-100 -80 -60 -40 -20 0 20 40 60 80 100

Published with MATLAB® R2018a