HW #1 (MATLAB Programming)

1. Transverse relaxation is an exponential decay process of the x and y components of magnetization. Mathematically this is expressed by $M_{xy}(t) = M_{xy}(0)\exp(-t/T2)$. Assume M consists of only an x component, and T2 = 100 ms. Ignoring other effects, what is the magnetization vector due to T2-decay after 50 ms? (Let's call this magnetization M1).

T2 = 100ms
t = 50ms

$$M_{xy}(50) = M_{xy}(0) \cdot \mathrm{e}^{-50/100}$$

$$M_{xy}(50) = M_{xy}(0) \cdot \mathrm{e}^{-1/2}$$

$$M_1 = M_x(0) \cdot \mathrm{e}^{-0.5*}\ \bar{\imath}$$

$$\boxed{M_{xy}(t) = M_{xy}(0)\exp(-t/T2)}$$

So, after 50ms of T2 decay , the magnetization vector M1 is approximately 0.6065 times the initial magnetization $M_{xy}(0)$.

2. Let's express M1=A*M, where A is a 3x3 matrix. Write down the elements of A, and test it with three different starting vectors, [1,0,0] T , [0,1,0] T , and [0,0,1] T .

$$A=\begin{vmatrix} e^{-t/T2} & 0 & 0 \\ 0 & e^{-t/T2} & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

$$A=\begin{vmatrix} e^{-50/100} & 0 & 0 \\ 0 & e^{-50/100} & 0 \\ 0 & 0 & 1 \end{vmatrix} \longrightarrow \begin{vmatrix} e^{-0.5} & 0 & 0 \\ 0 & e^{-0.5} & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

Case 1:

$$M = \begin{vmatrix} 1 \\ 0 \\ 0 \end{vmatrix} \qquad M1=A*M \qquad M1 = \begin{vmatrix} e^{-0.5} \\ 0 \\ 0 \end{vmatrix}$$

These results demonstrate the effect of the rotation matrix A. The vectors $[1,0,0]T$ and $[0,1,0]T$are rotated in the xy-plane. case 1 implies that the vector is positioned along the x-axis with an x-component of e−0.5. case 1 implies that the vector is positioned along the y-axis with an y-component of e−0.5.

Case 2:

$$M = \begin{vmatrix} 0 \\ 1 \\ 0 \end{vmatrix} \qquad M1=A*M \qquad M1 = \begin{vmatrix} 0 \\ e^{-0.5} \\ 0 \end{vmatrix}$$

Case 3: while the vector $[0,0,1]T$ remains unchanged because the rotation is about the z-axis

$$M = \begin{vmatrix} 0 \\ 0 \\ 1 \end{vmatrix} \qquad M1=A*M \qquad M1 = \begin{vmatrix} 0 \\ 0 \\ 1 \end{vmatrix}$$

3. Longitudinal (z) relaxation is a bit more complicated than transverse relaxation. The magnetization recovers exponentially with a time constant T1, to a non-zero value, often called M0. Mathematically, $M_z(t) = M_z(0)\exp\left(-\frac{t}{T1}\right) + \left(1 - \exp\left(-\frac{t}{T1}\right)\right)M0$. Note that in this project, we assume that M0 = 1.
In problem 2, we neglected T1-relaxation. You can express T1-relaxation in a matrix-vector form as M1 = A*M + B. Write down the elements of A and B (neglect T2-relaxation at this moment). What is M1 at t = 50 ms for T1 = 600 ms?

$$A = \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & e^{-t/T1} \end{vmatrix}$$

The matrix A reflects that the x and y components of the magnetization vector remain unchanged (since we are neglecting $T2$-relaxation), and only the z component decays exponentially.

$$B = \begin{vmatrix} 0 \\ 0 \\ 1 - \exp\left(-\frac{t}{T1}\right) \end{vmatrix}$$

The matrix B accounts for the recovery term of Mz .

M1 = A*M + B

M1 at t = 50 ms for T1 = 600 ms

$$A = \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & e^{-50/600} \end{vmatrix} = \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0.920 \end{vmatrix} \qquad B = \begin{vmatrix} 0 \\ 0 \\ 0.079 \end{vmatrix} \qquad M = \begin{vmatrix} Mx \\ My \\ Mz \end{vmatrix}$$

$$\text{M1} = \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0.920 \end{vmatrix} * \begin{matrix} Mx \\ My \\ Mz \end{matrix} + \begin{vmatrix} 0 \\ 0 \\ 0.079 \end{vmatrix} \longrightarrow \text{M1} = \begin{vmatrix} Mx \\ My \\ 0.92Mz + 0.079 \end{vmatrix}$$

This shows how the longitudinal relaxation process affects the Mz component over time, while the Mx and My components remain unchanged in the absence of $T2$ relaxation effects.

4. T1 and T2 relaxation effects happen independently. T1 relaxation affects only longitudinal magnetization while T2 relaxation only affects transverse magnetization. Now, consider both T1 and T2 relaxation and fill out A and B in the form M1=A*M+B. Compute M1 at t = 50 ms for T1 = 600 ms and T2 = 100 ms.

The matrix A reflects that the x and y components of the transverse decay (since we are taking $T2$-relaxation), and the z component decays exponentially.

$$A=\begin{vmatrix} e^{-t/T2} & 0 & 0 \\ 0 & e^{-t/T2} & 0 \\ 0 & 0 & e^{-t/T1} \end{vmatrix} \longrightarrow A=\begin{vmatrix} 0.606 & 0 & 0 \\ 0 & 0.606 & 0 \\ 0 & 0 & 0.920 \end{vmatrix}$$

The matrix B accounts for the recovery term of Mz

$$B = \begin{vmatrix} 0 \\ 0 \\ 1 - \exp\left(-\frac{t}{T1}\right) \end{vmatrix} = \begin{matrix} 0 \\ 0 \\ 0.079 \end{matrix}$$

M1 = A*M + B

M1 at t = 50 ms for T1 = 600 ms and T2 = 100 ms.

$$M1 = \begin{vmatrix} 0.606 & 0 & 0 \\ 0 & 0.606 & 0 \\ 0 & 0 & 0.920 \end{vmatrix} * \begin{vmatrix} Mx \\ My \\ Mz \end{vmatrix} + \begin{vmatrix} 0 \\ 0 \\ 0.079 \end{vmatrix}$$

$$M1 = \begin{vmatrix} 0.606\,Mx \\ 0.606My \\ 0.920Mz + 0.079 \end{vmatrix}$$

At t=50ms with $T1$=600 ms and $T2$=100 ms, the new magnetization vector $M1$ reflects the effects of both longitudinal ($T1$) and transverse ($T2$) relaxations:

- The transverse components Mx and My decay to 60 % of their original values.
- The longitudinal component Mz decays to 92% of its original value and recovers by 8% towards the equilibrium value $M0$=1.

5. Precession is a rotation about the z-axis. With a matrix, we can express this in the form M1=Rz*M, where Rz is a 3x3 matrix. Write a Matlab function ("zrot.m") with the syntax: function Rz = zrot(phi) . Here, phi is the rotation angle about the z-axis.

```
function Rz = zrot(phi)

   % Convert angle to radians
   phi = deg2rad(phi);
   % Compute cosine and sine of the angle
   x = cos(phi);
   y = sin(phi);
   % Construct the rotation matrix
   Rz = [x, y, 0;
         -y,  x, 0;
         0,   0, 1];
end


% the rotation angle (in degrees)
phi = 45;
% Generate the rotation matrix
Rz = zrot(phi);
% Display the rotation matrix
disp('Rotation matrix about the z-axis:');
disp(Rz);
```

Output :

```
Rotation matrix about the z-axis:
    0.7071    0.7071         0
   -0.7071    0.7071         0
         0         0    1.0000
```

6. Excitation is also a rotation. For now, let's assume that it is a rotation about the x- or y-axis. Write the functions "xrot.m" and "yrot.m", in a similar fashion to what you did in problem 5.

```
% xrot.m
function Rx = xrot(phi)
   % Convert angle to radians
   phi = deg2rad(phi);
   % cosine and sine of the angle
   x = cos(phi);
   y = sin(phi);
   % rotation matrix about the x-axis
   Rx = [1, 0, 0; 0, x, -y; 0, y, x];
end
```

Output :

```
>> Rx = xrot(45)

Rx =

    1.0000         0         0
         0    0.7071   -0.7071
         0    0.7071    0.7071
```

```
% yrot.m
function Ry = yrot(phi)
   % Convert angle to radians
   phi = deg2rad(phi);
   % cosine and sine of the angle
   x = cos(phi);
   y = sin(phi);
   % rotation matrix about the y-axis
   Ry = [x, 0, y; 0, 1, 0; -y, 0, x];
end
```

Output :

```
>> Ry = yrot(45)

Ry =

    0.7071         0    0.7071
         0    1.0000         0
   -0.7071         0    0.7071
```

7. What if the rotation is about a transverse axis other than x or y? Write a function "throt.m" with the syntax *function Rth = throt(phi,theta)* that returns the rotation matrix for a rotation of phi about the axis defined by y=x*tan(theta). Hint: you can think of this rotation as Rth(phi,theta)=Rz(theta)*Rx(phi)*Rz(-theta).

```
% throt.m
function Rth = throt(phi, theta)
% Convert angles to radians
phi = deg2rad(phi);
theta = deg2rad(theta);
% cosine and sine of the angles
c_phi = cos(phi);
s_phi = sin(phi);
c_theta = cos(theta);
s_theta = sin(theta);
% rotation matrices
Rz_theta = [c_theta, -s_theta, 0; s_theta, c_theta, 0; 0, 0, 1];
Rx_phi = [1, 0, 0; 0, c_phi, -s_phi; 0, s_phi, c_phi];
Rz_minus_theta = [c_theta, s_theta, 0; -s_theta, c_theta, 0; 0, 0, 1];
% Combining  the rotation matrices
Rth = Rz_theta * Rx_phi * Rz_minus_theta;
end
```

Output :

```
>> disp('y = x * tan(theta):');
disp(Rth);
y = x * tan(theta):
    0.9268    0.1268    0.3536
    0.1268    0.7803   -0.6124
   -0.3536    0.6124    0.7071
```

8. let's combine the matrices for relaxation and precession into one function of system. Note that the matrices A in problem 4 and Rz in problem 5 commute (i.e., A*Rz = Rz*A). Write a matlab function "freeprecess.m" with the syntax *function [Afp,Bfp] = freeprecess(T,T1,T2,df)* that returns the matrices such that M1=Afp*M+Bfp. Here, T is the duration of the free precession (in ms), T1 and T2 are the relaxation times (in ms), and df is the off-resonance frequency (in Hz).

```
% freeprecess.m
function [Afp, Bfp] = freeprecess(T, T1, T2, df)
% rotation angle for precession
omega = 2 * pi * df * T / 1000; % Convert df from Hz to kHz
phi = omega * T; % total rotation angle experienced by spins during the duration of free precession

% relaxation matrix A
A = diag([exp(-T/T2), exp(-T/T2), exp(-T/T1)]);
% precession matrix Rz
Rz = [cos(phi), -sin(phi), 0; sin(phi), cos(phi), 0; 0, 0, 1];
% combined matrix Afp
Afp = A * Rz;
% Bfp
Bfp = [0; 0; (1 - exp(-T/T1))];
end
```

Output :

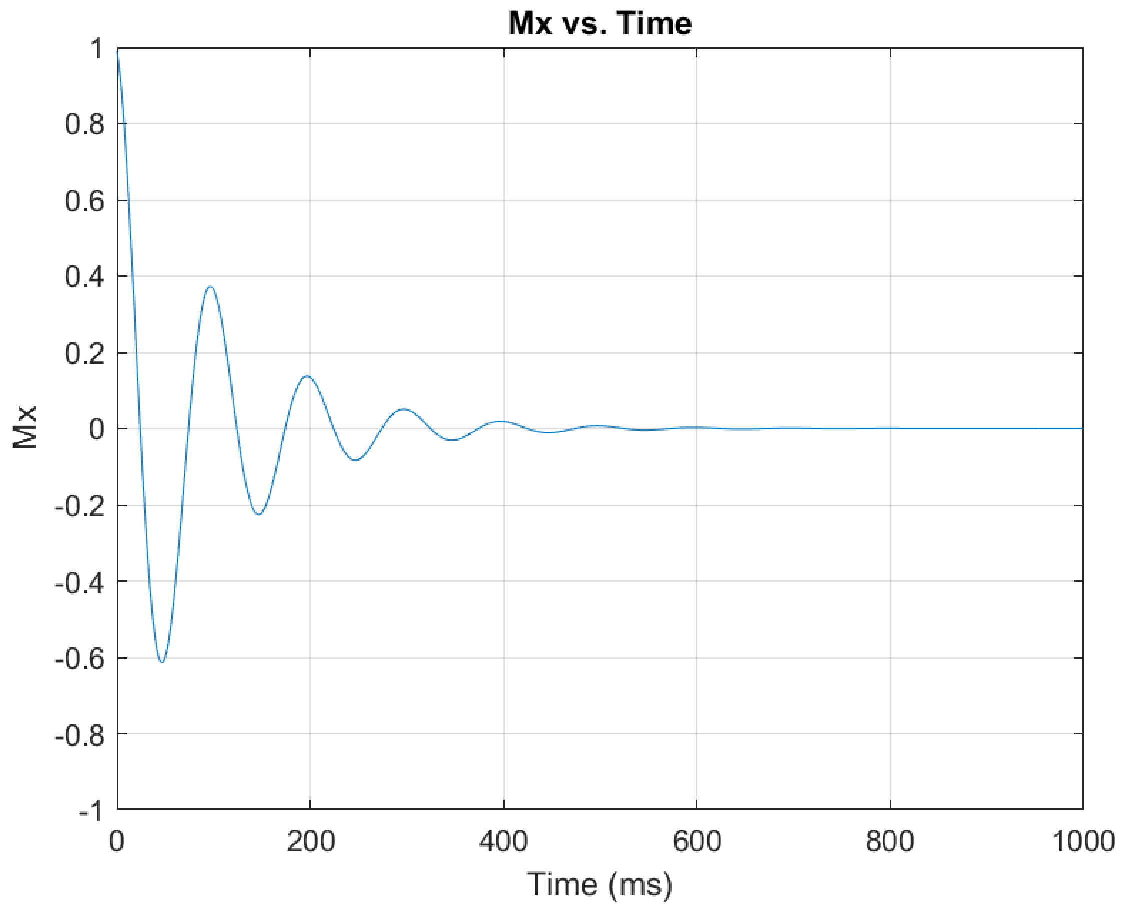
```
>> % usage of freeprecess function
T = 50;     % Duration of free precession (ms)
T1 = 600;   % Longitudinal relaxation time (ms)
T2 = 100;   % Transverse relaxation time (ms)
df = 100;   % Off-resonance frequency (Hz)

% Compute matrices for relaxation and precession
[Afp, Bfp] = freeprecess(T, T1, T2, df)

Afp =

    0.6065    0.0000         0
   -0.0000    0.6065         0
         0         0    0.9200


Bfp =

         0
         0
    0.0800
```

9. For T1=600 ms, T2 = 100 ms, and df = 10 Hz, use your function “freeprecess.m” to plot Mx vs. time, My vs. time, and Mz vs. time starting from M = $[1,0,0]^T$. Use a time step of 1 ms, and plot the response for 1000 ms. You should only need to call “freeprecess.m” once. You have just plotted the free-induction-decay (FID) for a single species.

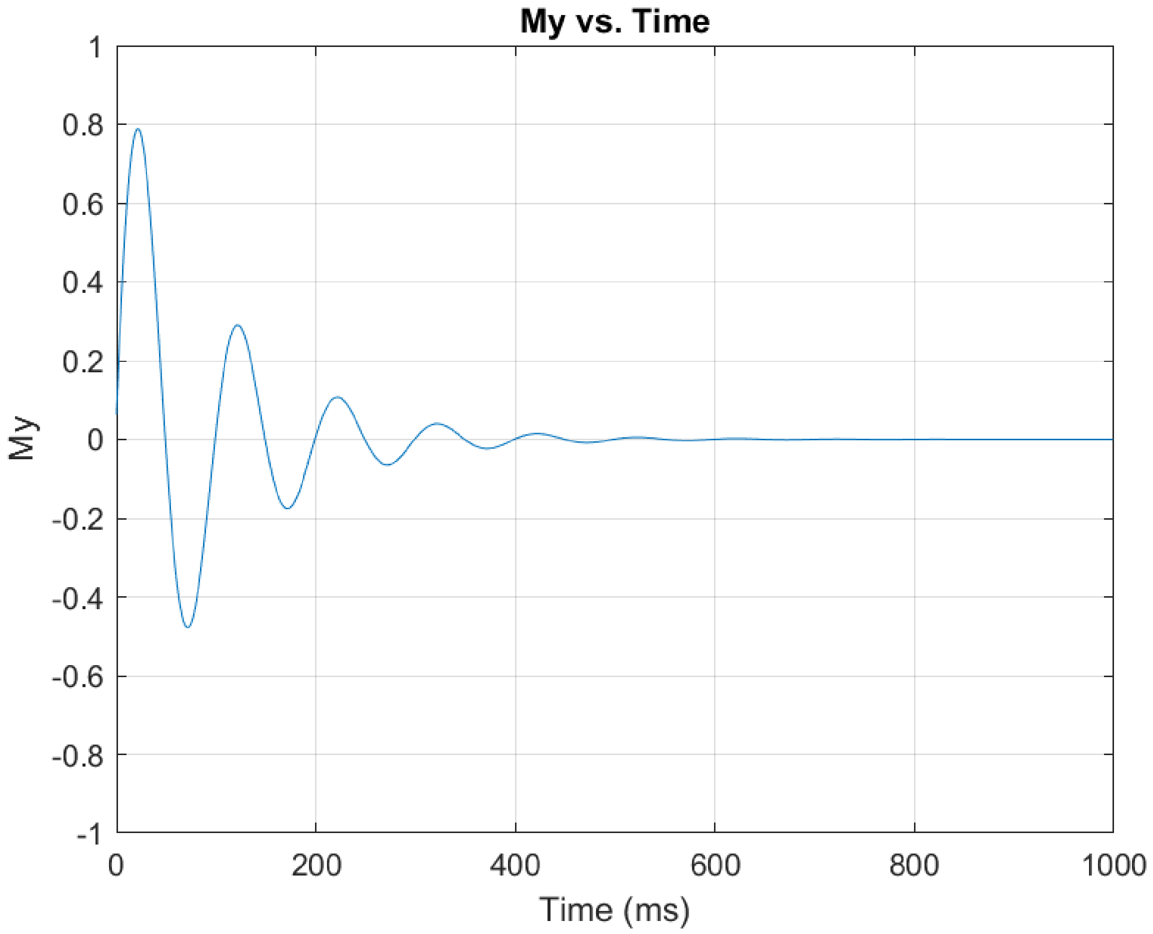
```
%parameters
T1 = 600;
T2 = 100;
df = 10;
T = 1000;
dt = 1;
M = [1; 0; 0];
steps = T / dt;
Mx = zeros(steps, 1);
My = zeros(steps, 1);
Mz = zeros(steps, 1);
%Computes the matrices for relaxation and precession using the freeprecess function

for i = 1:steps
[Afp, Bfp] = freeprecess(dt, T1, T2, df);
M = Afp * M + Bfp;
Mx(i) = M(1);
My(i) = M(2);
Mz(i) = M(3);
end
```

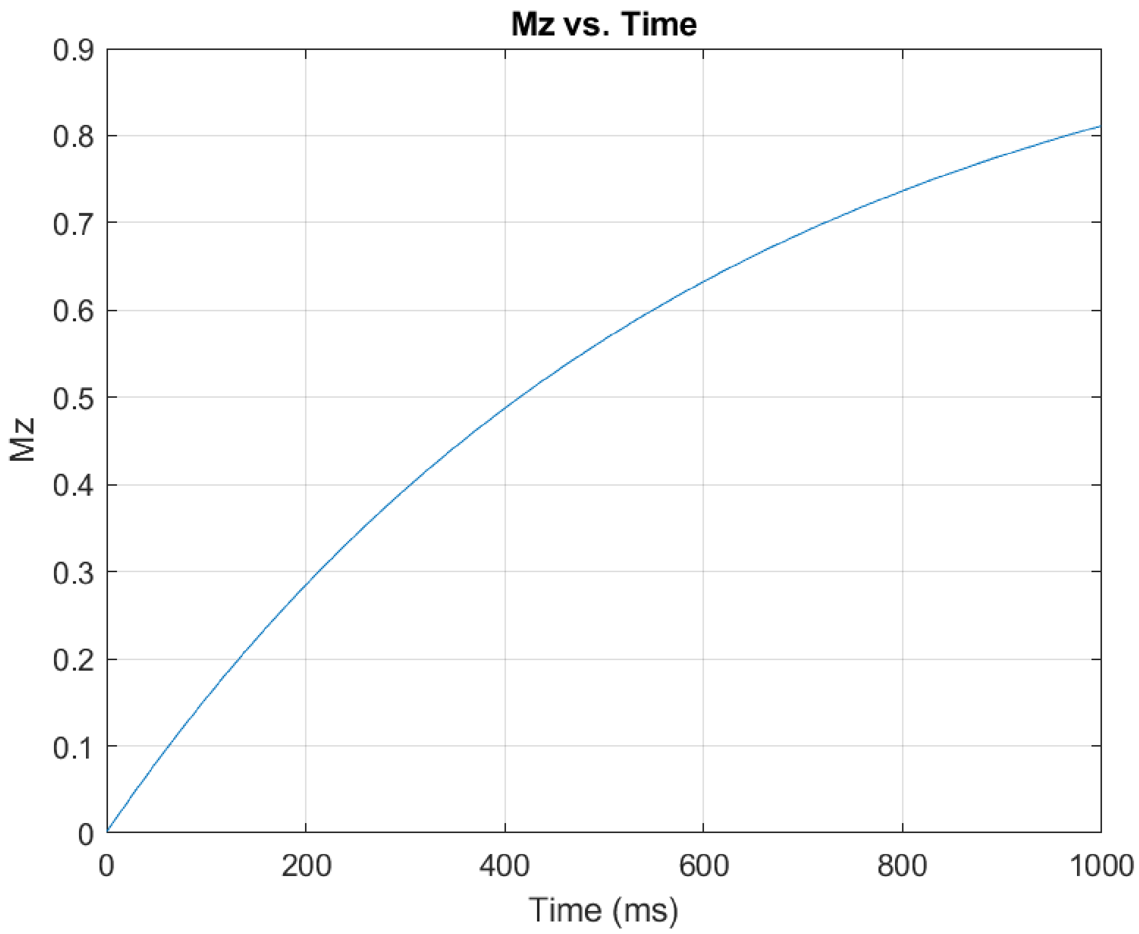
```
time = (0:dt:T-dt)';
figure;
plot(time, Mx);
xlabel('Time (ms)');
ylabel('Mx');
title('Mx vs. Time');
grid on;
ylim([-1 1]);
figure;
plot(time, My);
xlabel('Time (ms)');
ylabel('My');
title('My vs. Time');
grid on;
ylim([-1 1]);

figure;
plot(time, Mz);
xlabel('Time (ms)');
ylabel('Mz');
title('Mz vs. Time');
grid on;
```

Output :



Mx undergoes oscillations due to the precession effect and decays due to relaxation effects (T1 and T2).





- The plot of Mz vs. time shows an initial increase in Mz from 0 to its equilibrium value (1 in this case) due to longitudinal relaxation (T1 recovery).
- Mz recovers exponentially towards its equilibrium value with a time constant T1. This recovery is depicted by the gradual increase in Mz over time.

Observations:

- $T1$ (longitudinal or spin-lattice relaxation time) is the characteristic time it takes for the magnetization along the z-axis to reach its equilibrium value.

- $T2$ (transverse or spin-spin relaxation time) is the characteristic time it takes for the magnetization perpendicular to the z-axis to decay to 1/e of its initial value.

- The magnetic vector M precesses around the external magnetic field with a frequency determined by the offset frequency (df) and the strength of the external magnetic field.

- The precession is described by the Bloch equations, which are differential equations governing the behavior of magnetization in a magnetic field.

- The three plots show how each component of the magnetic vector M (Mx, My, and Mz) evolves over time under the influence of relaxation and precession.

- Each component starts at an initial value (in this case, Mx=1and My=Mz=0), and their values change over time due to relaxation and precession.

- The plots illustrate how the magnetization components oscillate and decay over time, reflecting the dynamics of relaxation and precession

10. Let’s consider the above example with >1 species, and there are lots of spins with different precession frequencies. Now, assume 1000 spins with frequency distribution following a Gaussian normal pdf. Further assume that the mean and standard deviation of the Gaussian distribution is 10 Hz and 1 Hz (Hint: use the built-in function “randn” to model the Gaussian distribution).

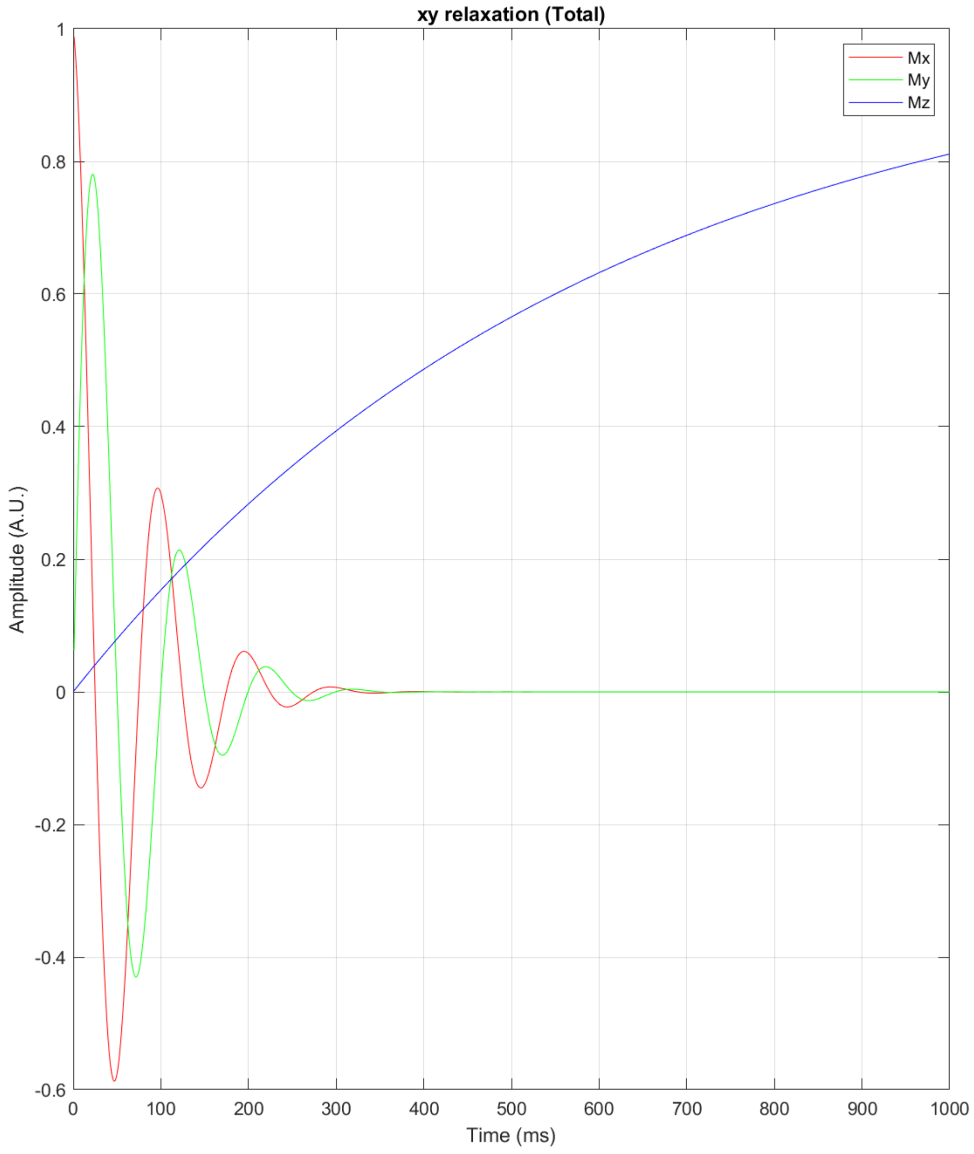
```
clc; clear; close all;

% Define parameters
T1 = 600; % Longitudinal relaxation time (ms)
T2 = 100; % Transverse relaxation time (ms)
mu = 10; % Mean of off-resonance frequency Gaussian distribution (Hz)
st_dev = input(' standard deviation (Hz): '); % Standard deviations of the
Gaussian pdf (Hz)
time = 1:1000; % time
dfs = mu + st_dev * randn(1, 1000);


% Initialize arrays to store magnetization components
Mx_total = zeros(1, 1000);
My_total = zeros(1, 1000);
Mz_total = zeros(1, 1000);
```

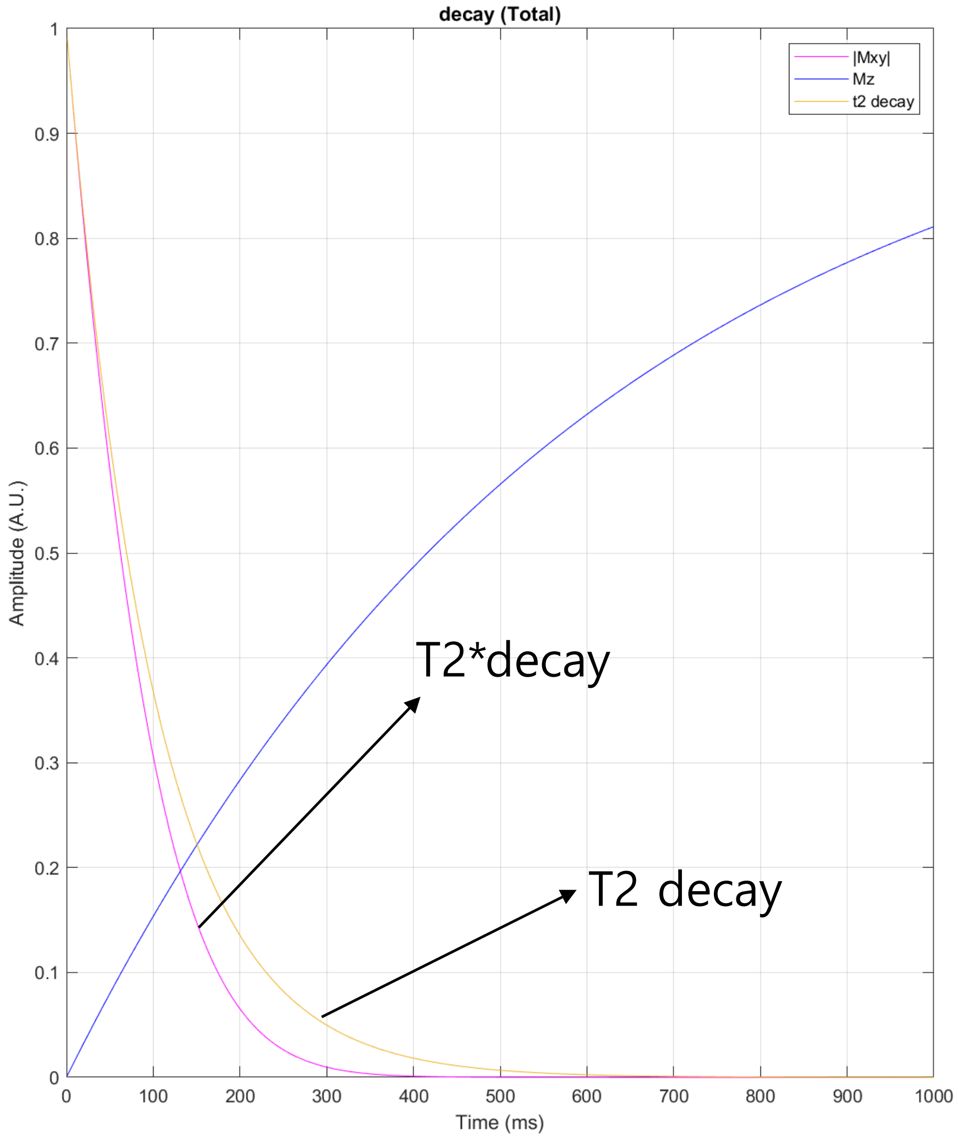
```
for i = 1:1000 % Loop over each off-resonance frequency
df = dfs(i); % Select current off-resonance frequencyt
M = [1; 0; 0];
% Initialize arrays to store magnetization components for the current standard
deviation
Mx = zeros(1, 1000);
My = zeros(1, 1000);
Mz = zeros(1, 1000);
% Loop over each time step
for t = 1:1000
% Calculate precession and relaxation
[Afp, Bfp] = freeprecess(1, T1, T2, df);
M = Afp * M + Bfp;
% Store magnetization components
Mx(t) = M(1);
My(t) = M(2);
Mz(t) = M(3);
end
% Accumulate the magnetization for all spins
Mx_total = Mx_total + Mx;
My_total = My_total + My;
Mz_total = Mz_total + Mz;
end
```

a) How does the signal sum differ from problem 9? This signal decay is called a T2* decay.
b) Plot Mx, My, and Mz for the following three frequency distributions: standard deviation of the Gaussian pdf = 1, 10, and 50 Hz. Discuss the reason behind the differences you have observed.

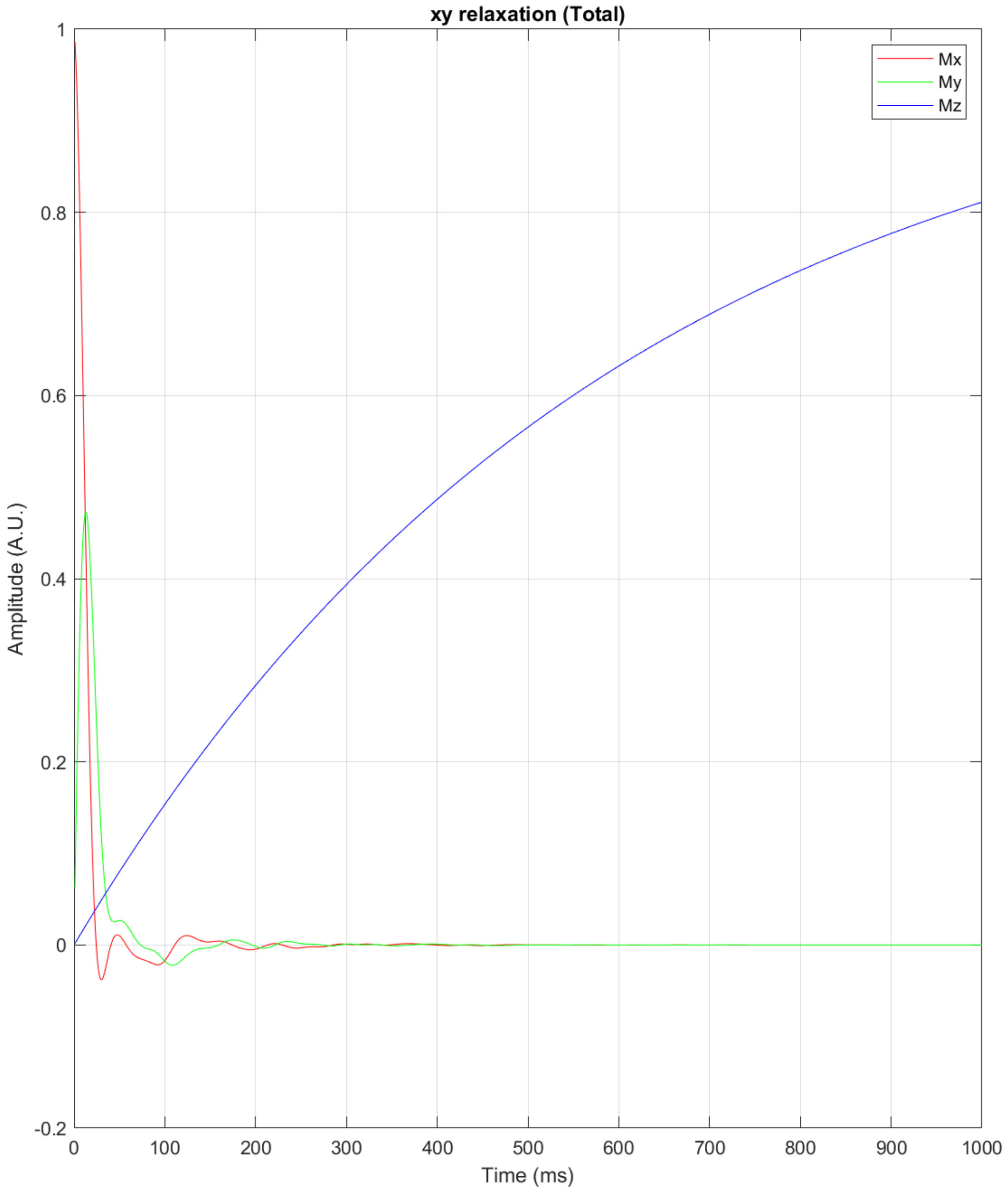
```
% Plot magnetization evolution
figure(2)
subplot(1,2,1);
plot(time, Mx_total/size(dfs,2), 'r', time, My_total/size(dfs,2), 'g', time, Mz_total/size(dfs,2), 'b');
xlabel('Time (ms)');
ylabel('Amplitude (A.U.)');
title('xy relaxation (Total)');
legend('Mx', 'My', 'Mz');
grid on;

% T2 decay
t2_decay = zeros(1,1000);
for i = 1:1000
t2_decay(i) = exp(-i/T2);
end
subplot(1,2,2);
Mxy_total = sqrt(Mx_total.^2 + My_total.^2);
plot(time, Mxy_total/size(dfs,2), 'm', time, Mz_total/size(dfs,2), 'b'); hold on;
plot(time, t2_decay);
xlabel('Time (ms)');
ylabel('Amplitude (A.U.)');
title('decay (Total)');
legend('|Mxy|', 'Mz');
grid on;
```

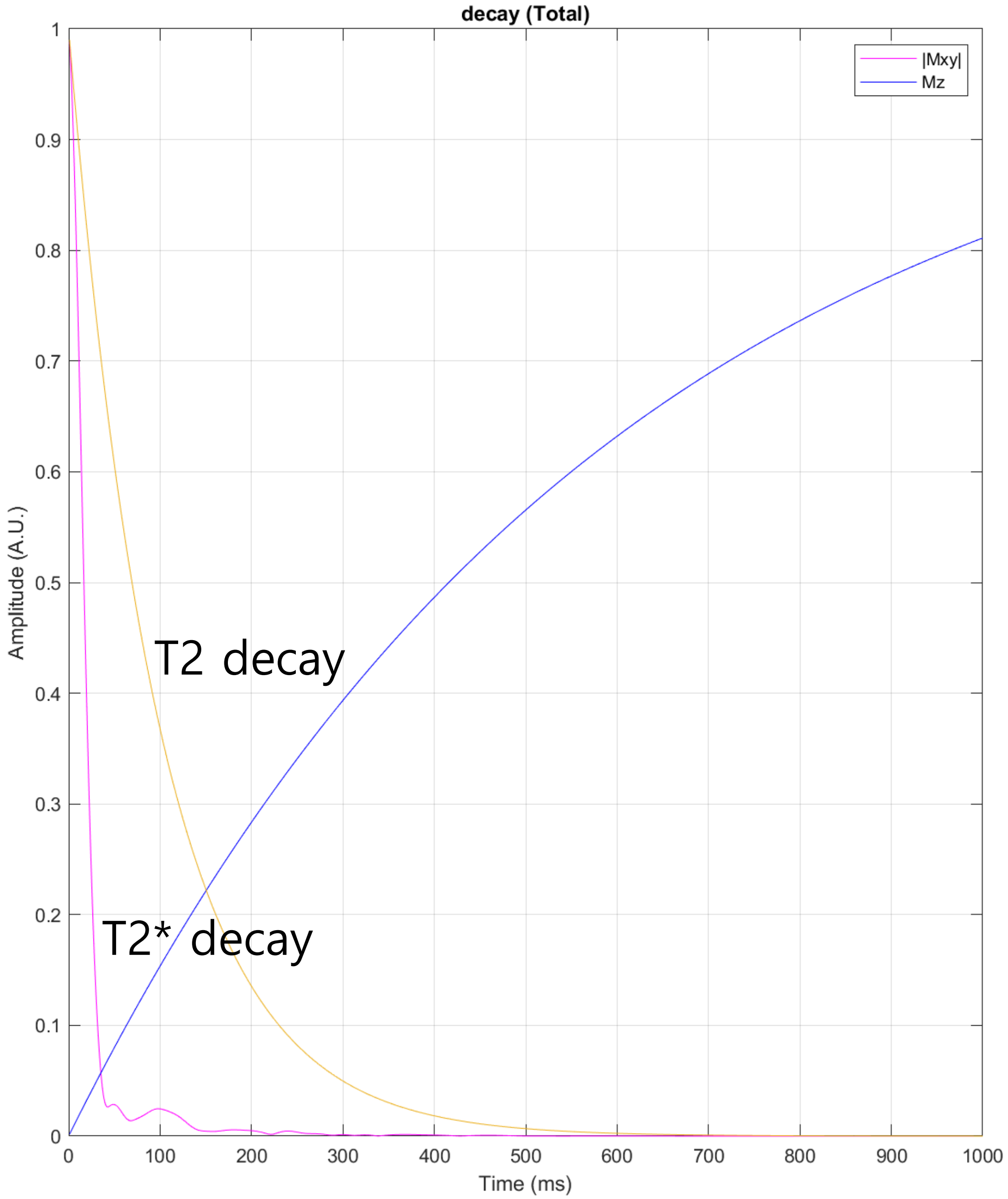
standard deviation (Hz): 1



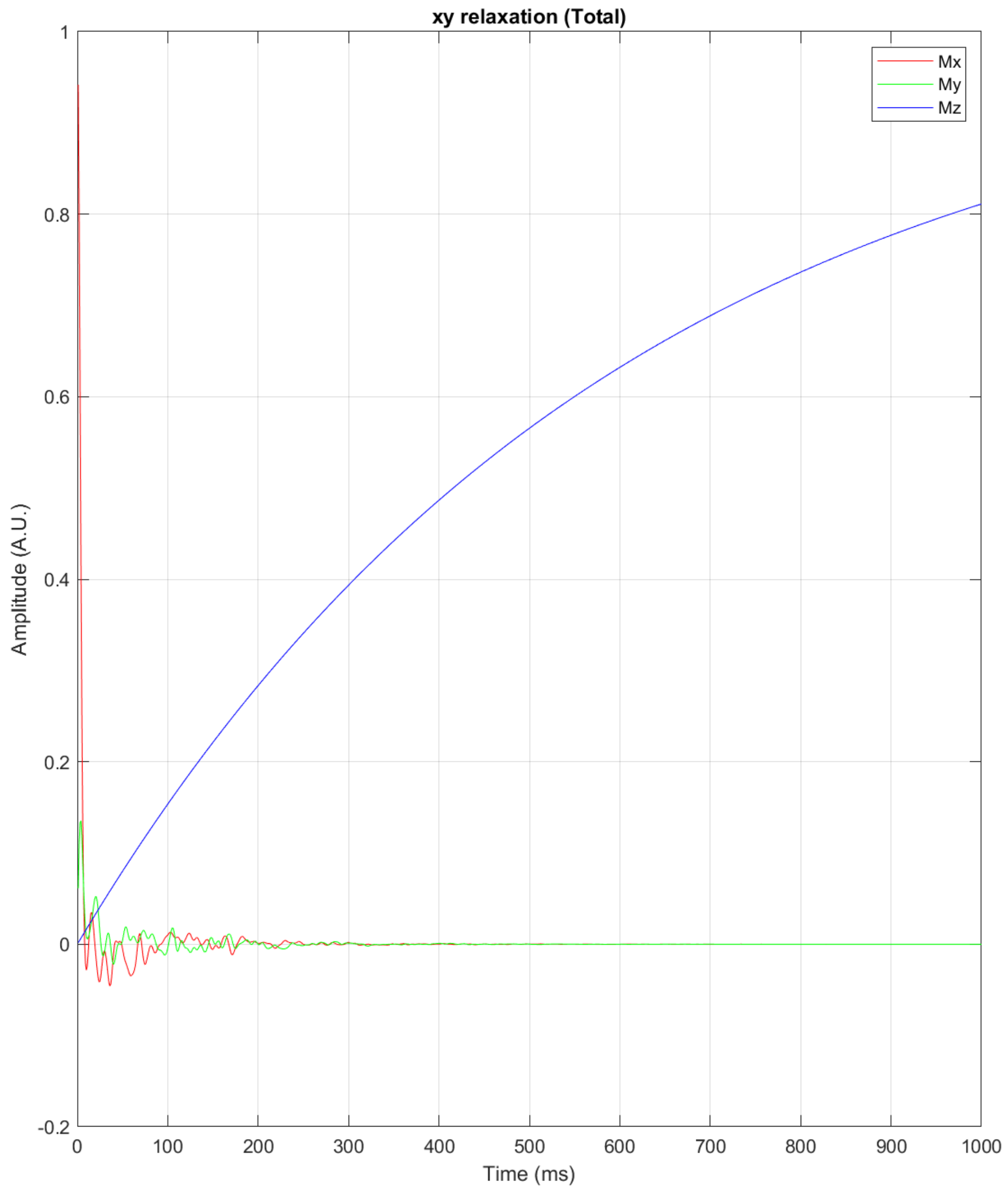


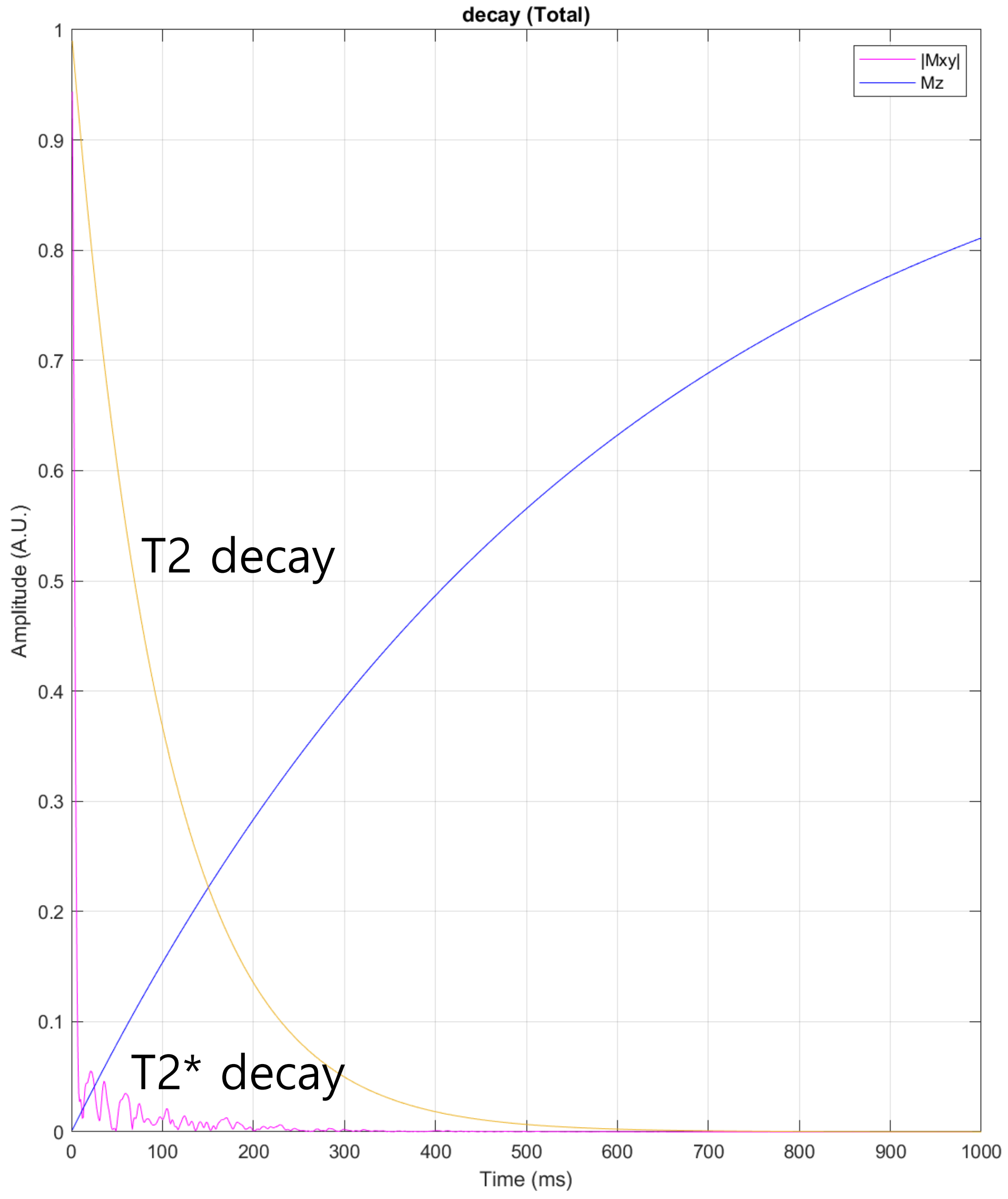
standard deviation (Hz): 10





standard deviation (Hz): 50





Observation:

a) In problem 9 , there was only one spin with single precession frequency. The signal decay observed there was due to T2 relaxation. However, in this scenario with multiple spins having different precession frequencies following a Gaussian distribution, the signal sum experiences T2* decay. T2* decay arises due to the variation in precession frequencies across spins, causing dephasing of the signal.

b) Let's discuss the plots for different standard deviations of the Gaussian distribution :

Standard Deviation = 1 Hz:
With a small standard deviation, the spread of precession frequencies around the mean is minimal. Hence, the spins experience similar magnetic field strengths, leading to a coherent signal evolution. Consequently, the magnetization components (Mx, My, and Mz) exhibit smooth curves with coherent behavior.

Standard Deviation = 10 Hz:
As the standard deviation increases, the spread of precession frequencies becomes wider. This results in more pronounced differences in the evolution of magnetization components. Some spins precess faster or slower than others due to the broader frequency distribution. Consequently, there's more variation in the behavior of Mx, My, and Mz over time.

Standard Deviation = 50 Hz:

With a larger standard deviation, the spread of precession frequencies becomes even wider. Now, the differences in precession frequencies among spins are significant. This leads to rapid dephasing of the spins, causing more erratic behavior in the magnetization components. The decay of the signal becomes faster due to the increased dephasing, resulting in a more pronounced T2* decay.

Overall, the differences in the standard deviation of the Gaussian distribution directly influence the spread of precession frequencies among spins, thereby affecting the coherence and decay of the signal. And in the plot we can observe the difference between t2 and t2* decay, In the plot, you can see that the decay of Mz is faster and less smooth compared to the decay of Mxy.
This faster decay is due to the additional dephasing caused by the variation in precession frequencies among spins, leading to a quicker loss of coherence in the signal.