Computational MRI

Coursework 1

Runyu Zhang

20069941

**Problem2.m**

clear;

clc;

t\_pulse = 1; % ms

delta\_theta = pi/2; % 90 deg

w1 = delta\_theta/t\_pulse;

t = linspace(0,t\_pulse,20);

fps = 10;

M0 = [0 0 1]';

M = forced\_procession\_rot(t,M0,w1);

fig = figure();

set(gcf, 'color', [1 1 1])

ax = axes();

F = [];

**for** ii = 1:length(t)

draw\_frame(ax,**struct**('M0',M0,...

'Mt',M(:,ii),...

'M',M(:,1:ii),...

't',t(ii)),["X'","Y'","Z'"]);

% draw B\_eff

draw\_vector(ax,[0,0,0],[1,0,0],'B\_{eff}');

F =[F,getframe(fig)];

end

save\_pdf(fig,'figure2.4.pdf');

save\_video('video2.4.avi',F,fps);

**Problem3\_1.m**

clear;

clc;

t\_pulse = 1; % ms

delta\_theta = pi/2; % 90 deg

w1 = delta\_theta/t\_pulse;

t = linspace(0,t\_pulse,50);

fps = 10;

w0 = 4\*pi; % KHz

M0 = [0 0 1]';

M = forced\_procession\_lab(t,M0,w0,w1);

fig = figure();

set(gcf, 'color', [1 1 1])

ax = axes();

F = [];

**for** ii = 1:length(t)

draw\_frame(ax,**struct**('M0',M0,...

'Mt',M(:,ii),...

'M',M(:,1:ii),...

't',t(ii)),["X","Y","Z"]);

F =[F,getframe(fig)];

end

save\_pdf(fig,'figure3.1.pdf');

save\_video('video3.1.avi',F,fps);

**Problem3\_2.m**

clear;

clc;

t\_max = 25; % ms

T1 = 10; % ms

T2 = 5; % ms

t = linspace(0,t\_max,500);

fps = 10;

w0 = 4\*pi; % KHz

M0 = [0 1 0]';

M = free\_procession\_euler(t,M0,T1,T2,w0);

fig = figure();

set(gcf, 'color', [1 1 1])

ax = axes();

F = [];

**for** ii = 1:length(t)

draw\_frame(ax,**struct**('M0',M0,...

'Mt',M(:,ii),...

'M',M(:,1:ii),...

't',t(ii)),["X","Y","Z"]);

F =[F,getframe(fig)];

end

save\_pdf(fig,'figure3.2.pdf');

save\_video('video3.2.avi',F,fps);

**Problem3\_3.m**

clear;

clc;

t\_max = 25; % ms

T1 = 10; % ms

T2 = 5; % ms

t = linspace(0,t\_max,50);

fps = 10;

M0 = [0 1 0]';

M = free\_procession\_euler(t,M0,T1,T2,0);

fig = figure();

set(gcf, 'color', [1 1 1])

ax = axes();

F = [];

**for** ii = 1:length(t)

draw\_frame(ax,**struct**('M0',M0,...

'Mt',M(:,ii),...

'M',M(:,1:ii),...

't',t(ii)),["X'","Y'","Z'"]);

F =[F,getframe(fig)];

end

save\_pdf(fig,'figure3.3.pdf');

save\_video('video3.3.avi',F,fps);

**Problem3\_4.m**

clear;

clc;

t\_max = 25; % ms

T1 = 10; % ms

T2 = 5; % ms

t = linspace(0,t\_max,100);

fps = 10;

delta\_w = 0.05\*4\*pi; % KHz

M0 = [0 1 0]';

M = free\_procession\_euler(t,M0,T1,T2,delta\_w);

fig = figure();

set(gcf, 'color', [1 1 1])

ax = axes();

F = [];

**for** ii = 1:length(t)

draw\_frame(ax,**struct**('M0',M0,...

'Mt',M(:,ii),...

'M',M(:,1:ii),...

't',t(ii)),["X'","Y'","Z'"]);

F =[F,getframe(fig)];

end

save\_pdf(fig,'figure3.4.pdf');

save\_video('video3.4.avi',F,fps);

**Problem3\_5.m**

clear;

clc;

t\_max = 25; % ms

T1 = 10; % ms

T2 = 5; % ms

t = linspace(0,t\_max,100);

fps = 10;

delta\_w = -0.05\*4\*pi; % KHz

M0 = [0 1 0]';

M = free\_procession\_euler(t,M0,T1,T2,delta\_w);

fig = figure();

set(gcf, 'color', [1 1 1])

ax = axes();

F = [];

**for** ii = 1:length(t)

draw\_frame(ax,**struct**('M0',M0,...

'Mt',M(:,ii),...

'M',M(:,1:ii),...

't',t(ii)),["X'","Y'","Z'"]);

F =[F,getframe(fig)];

end

save\_pdf(fig,'figure3.5.pdf');

save\_video('video3.5.avi',F,fps);

**Problem4\_1.m**

clear;

clc;

N = 10;

t\_max = 25; % ms

t\_pulse = 1; % ms

T1 = 10; % ms

T2 = 5; % ms

delta\_theta = pi/2; % 90 deg

w1 = delta\_theta/t\_pulse;

t1 = linspace(0,t\_pulse,100);

t2 = linspace(0,t\_max,2501);

w0 = 4\*pi; % KHz

M0 = [0 0 1]';

M1 = forced\_procession\_lab(t1,M0,w0,w1);

M2 = free\_procession\_euler(t2,M1(:,end),T1,T2,w0);

dummy = zeros(size(t1));

emf = [dummy,diff(M2(1,:))/(t2(2)-t2(1))];

**for** i = 1:N-1

M1 = forced\_procession\_lab(t1,M2(:,end),w0,w1);

M2 = free\_procession\_euler(t2,M1(:,end),T1,T2,w0);

emf = [emf,dummy,diff(M2(1,:))/(t2(2)-t2(1))];

end

t=linspace(0,(t\_pulse+t\_max)\*N,length(emf));

fig = figure;plot(t,emf);

xlabel('Time (ms)');

ylabel('Signal (a.u.)');

axis tight;

save\_pdf(fig,'figure4.1.pdf');

**Problem4\_2.m**

clear;

clc;

N = 5;

T1 = 10; % ms

TI = linspace(0,4,41)\*T1\*log(2); % Invertion time

emf\_p = [];

**for** ti = TI

[emf\_TI,~,~] = IR\_sequence(N,ti);

emf\_p = [emf\_p, emf\_TI];

end

fig = figure;plot(TI/(T1\*log(2)),emf\_p);

xlabel('T\_I (T\_1ln2)');

ylabel('Signal(T\_I) (a.u.)');

axis tight;

save\_pdf(fig,'figure4.2.T1est.pdf');

TI\_plot = [0.5\*T1\*log(2),1\*T1\*log(2),1.5\*T1\*log(2)];

**for** ii = 1:length(TI\_plot)

[emf\_TI,t,emf] = IR\_sequence(N,TI\_plot(ii));

fig = figure;plot(t,emf);

xlabel('Time (ms)');

ylabel('Signal (a.u.)');

axis tight;

save\_pdf(fig,sprintf('figure4.2.%d.pdf',ii));

end

function [emf\_TI,t,emf] = IR\_sequence(N,TI)

t\_max = 25; % ms

t\_pulse = 1; % ms

T1 = 10; % ms

T2 = 5; % ms

delta\_theta1 = pi; % 180 deg

delta\_theta2 = pi/2; % 90 deg

t1 = linspace(0,t\_pulse,100);

t2 = linspace(0,TI,500);

t3 = t1;

t4 = linspace(0,t\_max,2501);

w0 = 4\*pi; % KHz

M0 = [0 0 1]';

M1 = forced\_procession(t1,M0,w0,t\_pulse,delta\_theta1);

M2 = free\_procession\_euler(t2,M1(:,end),T1,T2,w0);

M3 = forced\_procession(t3,M2(:,end),w0,t\_pulse,delta\_theta2);

M4 = free\_procession\_euler(t4,M3(:,end),T1,T2,w0);

dummy = [zeros(size(t1)),zeros(size(t2)),zeros(size(t3))];

emf = [dummy,diff(M4(1,:))/(t4(2)-t4(1))];

**for** i = 1:N-1

M1 = forced\_procession(t1,M4(:,end),w0,t\_pulse,delta\_theta1);

M2 = free\_procession\_euler(t2,M1(:,end),T1,T2,w0);

M3 = forced\_procession(t3,M2(:,end),w0,t\_pulse,delta\_theta2);

M4 = free\_procession\_euler(t4,M3(:,end),T1,T2,w0);

emf = [emf,dummy,diff(M4(1,:))/(t4(2)-t4(1))];

end

t=linspace(0,(t\_pulse+TI+t\_pulse+t\_max)\*N,length(emf));

emf\_TI = max(emf);

end

function [M] = forced\_procession(t,M0,w0,t\_pulse,delta\_theta)

w1 = delta\_theta/t\_pulse;

M = forced\_procession\_lab(t,M0,w0,w1);

end

**Problem5\_1.m**

clear;

clc;

N = 3;

M0 = [0 0 1]';

TE = 15; % ms

t\_pulse = 1; % ms

T1 = 20; % ms

T2 = 15; % ms

% generate a collection of isochromats, initially aligns and sums up tp M0

N\_i = 1000;

m0 = M0/N\_i;

w0 = 4\*pi; % KHz

delta\_w0 = 0.05\*w0; % KHz

W0 = w0+delta\_w0\*2\*(rand(1,N\_i)-0.5);

emf\_sum = 0;

**for** w = W0

[t,emf] = spin\_echo\_sequence(N,m0,TE,w,t\_pulse,T1,T2);

emf\_sum = emf\_sum + emf;

end

fig=figure();plot(t,emf\_sum);

xlabel('Time (ms)');

ylabel('Signal (a.u.)');

axis tight;

save\_pdf(fig,'figure5.1.pdf');

**Problem5\_2.m**

clear;

clc;

N = 3;

M0 = [0 0 1]';

TE = 15; % ms

t\_pulse = 1; % ms

T1 = 20; % ms

T2 = 15; % ms

% generate a collecDon of isochromats, initially sums up tp M0

N\_i = 1000;

m0 = M0/N\_i;

w0 = 4\*pi; % KHz

delta\_w0 = 0.05\*w0; % KHz

Delta = 0.1\*pi;

W0 = lorentizian\_rand(N\_i,w0,Delta,delta\_w0);

emf\_sum = 0;

**for** w = W0

[t,emf] = spin\_echo\_sequence(N,m0,TE,w,t\_pulse,T1,T2);

emf\_sum = emf\_sum + emf;

end

fig=figure();plot(t,emf\_sum);

xlabel('Time (ms)');

ylabel('Signal (a.u.)');

axis tight;

save\_pdf(fig,'figure5.2.pdf');

**Problem5\_4.m**

clear;

clc;

N = 3;

M0 = [0 0 1]';

TE = 15; % ms

t\_pulse = 1; % ms

T1 = 20; % ms

T2 = 15; % ms

% generate a collecDon of isochromats, initially sums up tp M0

N\_i = 1000;

m0 = M0/N\_i;

w0 = 4\*pi; % KHz

delta\_w0 = 0.05\*w0; % KHz

Delta = 0.1\*pi;

W0 = lorentizian\_rand(N\_i,w0,Delta,delta\_w0);

emf\_sum = 0;

**for** w = W0

[t,emf] = spin\_echo\_sequence(N,m0,TE,w,t\_pulse,T1,T2,"H");

emf\_sum = emf\_sum + emf;

end

fig=figure();plot(t,emf\_sum);

xlabel('Time (ms)');

ylabel('Signal (a.u.)');

axis tight;

save\_pdf(fig,'figure5.4.pdf');

There are some functions I used in these codes.

function [M] = forced\_procession\_lab(t,M0,w0,w1,dir)

% lab frame

M = [];

**if** nargin == 4

dir = 'x';

end

**for** tt = t

% update M

theta\_z = w0\*tt; % CW

theta = w1\*tt; % CW

**if** dir == 'x'

R = rotx(theta);

elseif dir == 'y'

R = roty(theta);

end

Mt = rotz(theta\_z)\*R\*M0;

M = [M,Mt];

end

end

function [M] = forced\_procession\_rot(t,M0,w1)

% rotation frame

M = [];

**for** tt = t

% update M

theta = w1\*tt; % CW

M = [M,rotx(theta)\*M0];

end

end

function [M] = free\_procession\_euler(t,M0,T1,T2,w0)

M\_temp = M0;

M = [M0];

dt = t(2)-t(1);

rhs = @(M,T1,T2,M0)-M.\*[1;1;0]/T2+[0;0;(norm(M0)-M(3))/T1];

**for** ii = 2:length(t)

% update M using Euler method

M\_temp = M\_temp + dt\*rhs(M\_temp,T1,T2,M0);

theta\_z = w0\*t(ii); % CW

M(:,ii) = rotz(theta\_z)\*M\_temp;

end

end

function [A] = lorentizian\_rand(n,w0,Delta,delta\_w0)

M = 10;

A = [];

**for** ii = 1:n

**while**(1)

y = w0+delta\_w0\*2\*(rand(1)-0.5);

u = rand(1);

**if** u<(1/(pi\*Delta+pi\*(y-w0)^2/Delta))/M/y

A = [A, y];

**break**;

**else**

**continue**;

end

end

end

end

function [R] = rotx(theta)

c = cos(theta);

s = -sin(theta);

R = [1 0 0;

0 c -s;

0 s c];

end

function [R] = roty(theta)

c = cos(theta);

s = -sin(theta);

R = [c 0 s;

0 1 0;

-s 0 c];

End

function [R] = rotz(theta)

c = cos(theta);

s = -sin(theta);

R = [c -s 0;

s c 0;

0 0 1];

end

function [ax] = draw\_axis(ax,O,X,Y,Z,labels)

hold(ax,'on');

pbaspect manual;

arrow3(O,X,'r',0,0);

text(X(1),X(2),X(3),labels(1));

arrow3(O,Y,'g',0,0);

text(Y(1),Y(2),Y(3),labels(2));

arrow3(O,Z,'b',0,0);

text(Z(1),Z(2),Z(3),labels(3));

hold(ax,'off');

end

function [ax] = draw\_frame(ax, context, axis\_label)

O = [0 0 0]';

X = [2 0 0]';

Y = [0 2 0]';

Z = [0 0 2]';

% clear axes

cla(ax);

% redraw

draw\_axis(ax,O',X',Y',Z',axis\_label);

% draw M0

draw\_vector(ax,O',context.M0','M\_0');

% draw Mt

draw\_vector(ax,O',context.Mt','M\_t');

% draw Trace of Mt

hold(ax,'on');

M = context.M;

plot3(ax,M(1,:),M(2,:),M(3,:));

hold(ax,'off');

axis([-1,2,-1,2,0,2]);

axis off;

view([20 25 5]);

title(sprintf('t = %.2f ms',context.t));

ax.Position = [0.05 0 0.9 1];

end

function [ax] = draw\_vector(ax,O,V,label)

hold(ax,'on');

arrow3(O,V);

text(V(1),V(2),V(3),label);

hold(ax,'off');

end

function [] = save\_pdf(fig,name)

filepath = fileparts(mfilename('fullpath'));

fig.Renderer='Painters';

p = fig.PaperPosition;

set(fig,'PaperSize',[1.1\*p(3) 1.1\*p(4)]);

print(fig,strcat(filepath,[filesep '..' filesep '..' filesep 'figures' filesep],name),'-dpdf','-r0')

end

function [] = save\_video(name,frames,fps)

filepath = fileparts(mfilename('fullpath'));

writer = VideoWriter(strcat(filepath,[filesep '..' filesep '..' filesep 'videos' filesep],name));

writer.FrameRate = fps;

writer.Quality = 100;

open(writer);

**for** f = frames

writer.writeVideo(f);

end

close(writer);

end

function [t,emf] = spin\_echo\_sequence(N,M0,TE,w0,t\_pulse,T1,T2,type)

**if** nargin == 7

type = "CP";

end

tau = TE/2; % ms

delta\_theta1 = pi/2; % 90 deg

**if** type == "CP"

delta\_theta2 = pi; % 180 deg

elseif type =="H"

delta\_theta2 = pi/2; % 90 deg

end

t1 = linspace(0,t\_pulse,100);

t2 = linspace(0,tau,1001);

t3 = t1;

t4 = linspace(0,TE,2001);

M1 = forced\_procession(t1,M0,w0,t\_pulse,delta\_theta1,'x');

M2 = free\_procession\_euler(t2,M1(:,end),T1,T2,w0);

M3 = forced\_procession(t3,M2(:,end),w0,t\_pulse,delta\_theta2,'y');

M4 = free\_procession\_euler(t4,M3(:,end),T1,T2,w0);

dummy = zeros(size(t1));

emf = [dummy,diff(M2(1,:))/(t2(2)-t2(1)),dummy,diff(M4(1,:))/(t4(2)-t4(1))];

**for** i = 1:N-1

M3 = forced\_procession(t3,M4(:,end),w0,t\_pulse,delta\_theta2,'y');

M4 = free\_procession\_euler(t4,M3(:,end),T1,T2,w0);

emf = [emf,dummy,diff(M4(1,:))/(t4(2)-t4(1))];

end

t=linspace(0,t\_pulse+tau+(t\_pulse+TE)\*N,length(emf));

function [M] = forced\_procession(t,M0,w0,t\_pulse,delta\_theta,dir)

w1 = delta\_theta/t\_pulse;

M = forced\_procession\_lab(t,M0,w0,w1,dir);

end

end

function [s] = spin\_excess(B,T,gamma)

%Calculate spin excess

h = 6.626e-24;

k = 1.3805e-23;

s = exp(-h\*gamma.\*B/k./T);

end