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
```