

Contents

- Post-analysis for ELENA B-train data acquisition during both acceleration and deceleration
- Close and clear all previous data
- Constants
- Define cycle parameters
- Quality of B-train signals
- Marker signals
- Remove zero elements due to skipped files, and calculate statistics
- Averages over N cycles
- Plots
- Output indicators

Post-analysis for ELENA B-train data acquisition during both acceleration and deceleration

by Christian Grech (TE-MSC-MM) v2.0 28.09.2018 The script recognizes .mat files recorded using MATLAB 2014/15 in CCM (ELENA_Btrain_acq.m and myCallback.m under /nfs/cs-ccr-nfs6/vol28/u1/elenaop/chgrech/ELENA_qualitative). The mat files consists of B-train signals from C1 and C2 systems (bdot included), the marker times, marker signals and possibly the current from the Sampler. The files are produced separately for each cycle, and this script takes care of skipped cycles in the numbering of files.

```
% The script outputs all performance measurements possible to be measured.  
% Make sure that cycle parameters are set accordingly. There are two  
% options for cycles measured.
```

Close and clear all previous data

```
clear all;  
close all;
```

Constants

```
numberOfCyclesStat = 1999; % Number of cycles recorded  
numberOfMarkers = 15; % Number of cycles recorded  
number_recorded = 0; % Iterations recorded  
skippedData = 2; % Iterations skipped due to 'incorrect' data  
n = 100; % Downsampling constant for plotting large number of iterations  
oper = 'OP_HMMD1';  
spare = 'SP_HMMD1';  
marker_times = 'MarkerTime_PBMD1';  
curr = 'current';  
marker = 'Marker';  
iter = [];
```

Define cycle parameters

```
Fs=1000; % FESA Sampling freq  
I_drift = 5.4e-4; % in A/s  
signal = 0; % 1 if HMMD1/PBMD1 cycle or 0 is PBMD2 cycle  
  
if signal == 1  
    cyclename = 'PBMD2'; % Antiproton cycle deceleration  
    injection = 2727; % time in ms  
    extraction = 31000;  
    fieldCheckAtD1 = 3350;  
    fieldCheckAtD2 = 3650;  
    fieldCheckAtD1e = 31000;  
    fieldCheckAtD2e = 34000;  
    dBdI_inj = 3598/275.946;  
    dBdI_ex = 493/37.132; % in G/A  
    Marker_fraction_plot = 10;  
    stable_time = 300;  
elseif signal == 0  
    cyclename = 'HMMD1'; % GBAR_100KeV_SHORT_CYCLE  
    injection = 1029;  
    extraction = 1229;  
    fieldCheckAtD1 = 1000;  
    fieldCheckAtD2 = 1090;  
    fieldCheckAtD1e = 1500;  
    fieldCheckAtD2e = 1700;  
    dBdI_inj = 454.765/34.79;  
    dBdI_ex = 493.2/37.752;  
    Marker_fraction_plot = 6;  
    stable_time = 90;  
elseif signal == -1  
    cyclename = 'PMD2'; % Proton test cycle  
    injection = 1029;  
    extraction = 1229;  
    fieldCheckAtD1 = 890;  
    fieldCheckAtD2 = 1090;  
    fieldCheckAtD1e = 1600;
```

```

fieldCheckAtD2e = 1800;
dBdI_inj = 454.765/34.79;
dBdI_ex = 493.2/37.752;
Marker_fraction_plot = 6;
stable_time = 200;
elseif signal == 2
    cyclename = 'PBMD1'; % Accelerating test cycle
    injection = 1029;
    extraction = 3138;
    fieldCheckAtD1 = 890;
    fieldCheckAtD2 = 1090;
    fieldCheckAtD1e = 3138;
    fieldCheckAtD2e = 5000;
    dBdI_inj = 454.765/34.79;
    dBdI_ex = 493.2/37.752;
    Marker_fraction_plot = 6;
    stable_time = 200;
end

id = 'MATLAB:colon:nonIntegerIndex';
warning('off',id)
k = 0;

```

Quality of B-train signals

```

for i = 1:numberOfCyclesStat
    filename2 = sprintf('%s_%d.mat',oper,i+skippedData);
    % Load files
    filename3 = sprintf('%s_%d.mat',spare,i+skippedData);
    filename6 = sprintf('%s_%d.mat',marker_times,i+skippedData+1);
    % MarkerTime_
    if signal ==0
        filename4 = sprintf('%s_%d.mat',curr,i+skippedData);
    end

    if exist(strcat(pwd,'\\data1\\'), 'file' ) == 2
        % Check if file exists, else skip
        number_recorded = number_recorded+1;
        load(strcat(pwd,'\\data1\\',filename2));
        load(strcat(pwd,'\\data1\\',filename3));
        load(strcat(pwd,'\\data1\\',filename6));
        if signal ==1
            load(strcat(pwd,'\\data1\\',filename4));
            curr_(:,1) = I_sam(1:10:end);
        end;

    timeNewBtrainOp = linspace(0, length(OP_signals(:,1))/1000, length(OP_signals(:,1)));
    newBtrainOp = double(OP_signals(:,1));
    timeNewBtrainsP = linspace(0, length(SP_signals(:,1))/1000, length(SP_signals(:,1)));
    newBtrainSp = double(SP_signals(:,1));

    % MarkerCTime values
    M1markerC1(i) = Check_signals(1);
    M1markerC2(i) = Check_signals(2);
    M2markerC1(i) = Check_signals(3);
    M2markerC2(i) = Check_signals(4);

    % Bdot signals
    newBdotOp = OP_signals(:,2);
    BdotC1(i) = max(abs(newBdotOp));
    newBdotSp = SP_signals(:,2);
    BdotC2(i) = max(abs(newBdotSp));

    % DeltaC1C1 in Gauss
    DeltaC1C2_inj(i) = newBtrainOp(injection)-newBtrainSp(injection);
    DeltaC1C2_ex(i) = newBtrainOp(extraction)-newBtrainSp(extraction);

    % Drift calculation
    DriftC1(i) = ((newBtrainOp(fieldCheckAtD2)- newBtrainOp(fieldCheckAtD1))/((fieldCheckAtD2/1000)-(fieldCheckAtD1/1000))-(dBdI_inj*I_drift);
    DriftC2(i) = ((newBtrainSp(fieldCheckAtD2)- newBtrainSp(fieldCheckAtD1))/((fieldCheckAtD2/1000)-(fieldCheckAtD1/1000))-(dBdI_inj*I_drift);
    DriftClex(i) = ((newBtrainOp(fieldCheckAtD2e)- newBtrainOp(fieldCheckAtD1e))/((fieldCheckAtD2e/1000)-(fieldCheckAtD1e/1000))-(dBdI_ex*I_drift);
    DriftC2ex(i) = ((newBtrainSp(fieldCheckAtD2e)- newBtrainSp(fieldCheckAtD1e))/((fieldCheckAtD2e/1000)-(fieldCheckAtD1e/1000))-(dBdI_ex*I_drift);
    DriftC1_new(i) = driftcalc(newBtrainOp, fieldCheckAtD2, fieldCheckAtD1, dBdI_inj, I_drift);
    DriftC2_new(i) = driftcalc(newBtrainSp, fieldCheckAtD2, fieldCheckAtD1, dBdI_inj, I_drift);
    DriftClex_new(i) = driftcalc(newBtrainOp, fieldCheckAtD2e, fieldCheckAtD1e, dBdI_ex, I_drift);
    DriftC2ex_new(i) = driftcalc(newBtrainSp, fieldCheckAtD2e, fieldCheckAtD1e, dBdI_ex, I_drift);

    % Stability
    for j = 1:stable_time
        stable_rmsC1(i,j) = (newBtrainOp((fieldCheckAtD1+j))-newBtrainOp(fieldCheckAtD1)-(DriftC1(i)*(j/1000)));
        stable_rmsC2(i,j) = (newBtrainSp((fieldCheckAtD1+j))-newBtrainSp(fieldCheckAtD1)-(DriftC2(i)*(j/1000)));
        stable_rmsClex(i,j) = (newBtrainOp((fieldCheckAtD1e+j))-newBtrainOp(fieldCheckAtD1e)-(DriftClex(i)*(j/1000)));
        stable_rmsC2ex(i,j) = (newBtrainSp((fieldCheckAtD1e+j))-newBtrainSp(fieldCheckAtD1e)-(DriftC2ex(i)*(j/1000)));
    end;

    stab_rmsC1(i) = std(stable_rmsC1(i,:));
    stab_rmsC2(i) = std(stable_rmsC2(i,:));
    stab_rmsClex(i) = std(stable_rmsClex(i,:));
    stab_rmsC2ex(i) = std(stable_rmsC2ex(i,:));
    pp_rmsC1(i) = max(stable_rmsC1(i,:))-min(stable_rmsC1(i,:));
    pp_rmsC2(i) = max(stable_rmsC2(i,:))-min(stable_rmsC2(i,:));
    pp_rmsClex(i) = max(stable_rmsClex(i,:))-min(stable_rmsClex(i,:));

```

```

pp_rmsC2ex(i) = max(stable_rmsC2ex(i,:))-min(stable_rmsC2ex(i,:));

% Repeatability at injection and Extraction
RepClinj(i) = newBtrainOp(injection);
RepC2inj(i) = newBtrainSp(injection);
RepClex(i) = newBtrainOp(extraction);
RepC2ex(i) = newBtrainSp(extraction);

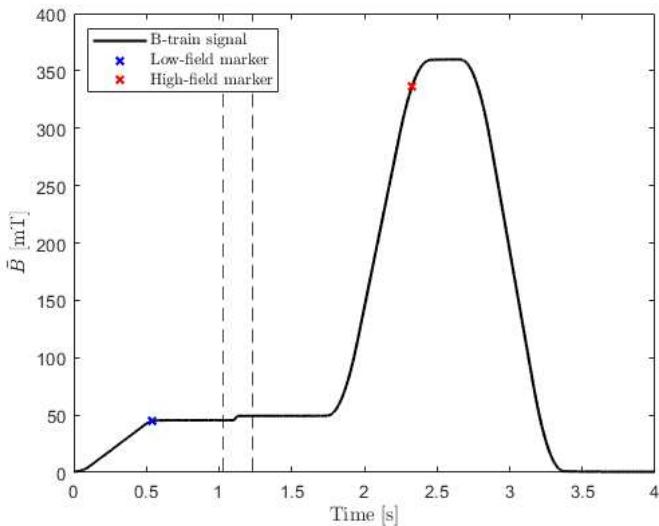
% FFT calculation
fft_C1(i) = fftnew(Fs, length(newBtrainOp), newBtrainOp, 200);
fft_C2(i) = fftnew(Fs, length(newBtrainSp), newBtrainSp, 200);
fft_C1_full(i,:) = fftsig(Fs, length(newBtrainOp), newBtrainOp, 200);
fft_C2_full(i,:) = fftsig(Fs, length(newBtrainSp), newBtrainSp, 200);
fft_C1_Bdot(i) = fftnew(Fs, length(newBdotOp), newBdotOp, 200);
fft_C2_Bdot(i) = fftnew(Fs, length(newBdotSp), newBdotSp, 200);
disp(strcat('Data calculation ongoing...',num2str(i/(numberOfCyclesStat)*100,'%6.2f'),'%')) %update the loading bar
else
    k=k+1;
    disp(strcat('No file for iteration...',num2str(i-1)));
    iter(k) = i-k+1;
end
end
disp(strcat('Number of B-train signals recorded...',num2str(number_recorded)))
figure; movegui(gcf,'northeast'); plot( timeNewBtrainOp, newBtrainOp/10, 'k', 'LineWidth', 1.5); hold on;
plot( injection/1000, newBtrainOp(injection)/10, 'r+', extraction/1000, newBtrainOp(extraction)/10, 'g+' );
plot(M1markerC1(9), newBtrainOp(round(M1markerC1(9)*1000))/10, 'bx', M2markerC1(9), newBtrainOp(round(M2markerC1(9)*1000))/10, 'rx','LineWidth', 1.5); hold on;
ylabel('$\bar{B} [mT]', 'interpreter', 'latex');
xlabel('Time [s]', 'interpreter', 'latex');
xlim([0 4]);
xline(injection/1000, 'k--');
xline(extraction/1000, 'k--');
legend('B-train signal', 'Low-field marker', 'High-field marker', 'Location','northwest', 'interpreter', 'latex');

numberofMarkers_recorded = 0;
j=0;
figure; movegui(gcf,'southeast');

```

Data calculation ongoing...0.05%
 No file for iteration...1
 Data calculation ongoing...0.15%
 No file for iteration...3
 Data calculation ongoing...0.25%
 No file for iteration...5
 Data calculation ongoing...0.35%
 No file for iteration...7
 Data calculation ongoing...0.45%
 No file for iteration...9
 Data calculation ongoing...0.55%
 No file for iteration...11
 Data calculation ongoing...0.65%
 No file for iteration...13
 Data calculation ongoing...0.75%
 No file for iteration...15
 Data calculation ongoing...0.85%
 No file for iteration...17
 Data calculation ongoing...0.95%
 No file for iteration...19
 Data calculation ongoing...1.05%
 No file for iteration...21
 Data calculation ongoing...1.15%
 No file for iteration...23
 Data calculation ongoing...1.25%
 No file for iteration...25
 Data calculation ongoing...1.35%
 No file for iteration...27
 Data calculation ongoing...1.45%
 No file for iteration...29
 Data calculation ongoing...1.55%
 No file for iteration...31
 Data calculation ongoing...1.65%
 No file for iteration...33
 Data calculation ongoing...1.75%
 No file for iteration...35
 Data calculation ongoing...1.85%
 No file for iteration...37
 Data calculation ongoing...1.95%
 No file for iteration...39
 Data calculation ongoing...2.05%
 No file for iteration...41
 Data calculation ongoing...2.15%
 No file for iteration...43
 Data calculation ongoing...2.25%
 No file for iteration...45
 Data calculation ongoing...2.35%
 No file for iteration...47
 Data calculation ongoing...2.45%
 No file for iteration...49
 Data calculation ongoing...2.55%
 No file for iteration...51
 Data calculation ongoing...2.65%
 No file for iteration...53
 Data calculation ongoing...2.75%
 No file for iteration...55
 Data calculation ongoing...2.85%
 No file for iteration...57

Data calculation ongoing...100.00%
Number of B-train signals recorded...1000



Marker signals

```

for i = 1:numberOfMarkers
    filename5 = sprintf('%s_%d.mat',marker,i+skippedData+10);

if exist(strcat(pwd,'\\data1\\',filename5), 'file') == 2
    load(strcat(pwd,'\\data1\\',filename5));
    M1C1 = markers(:,1)/25;
    M2C1 = markers(:,2)/5;
    d by 2 in the ELENA B-train acquisition
    M1C2 = markers(:,3)/25;
    M2C2 = markers(:,4)/5;
    d by 2 in the ELENA B-train acquisition

    samples = 1:length(M1C1);
    samples2 = 1:length(M1C1);
    numberofMarkers_recorded = numberofMarkers_recorded+1;
    fft_M1C1(i) = fftnew(50e3, length(M1C1), M1C1,10000);
    fft_M2C1(i) = fftnew(50e3, length(M2C1), M2C1, 10000);
    fft_M1C2(i) = fftnew(50e3, length(M1C2), M1C2, 10000);
    fft_M2C2(i) = fftnew(50e3, length(M2C2), M2C2, 10000);

    subplot(2,2,[1 2])
        h1=plot(0.4339+(samples/50000), M1C1, 'r', 0.4339+(samples/50000), M1C2, 'k'); hold on; grid on;
        title('Low field');
        xlabel('Time [s]');
        ylabel('NMR voltage [V]');
    subplot(2,2,3)
        plot(1.5627+(samples2/50000), M2C1, 'r'); hold on; grid on;
        title('High field C1');
        xlabel('Time [s]');
        ylabel('NMR voltage [V]');
    subplot(2,2,4)
        plot(1.5627+(samples2/50000), M2C2, 'k'); hold on; grid on;
        title('High field C2');
        xlabel('Time [s]');
        ylabel('NMR voltage [V]');
else

```

```

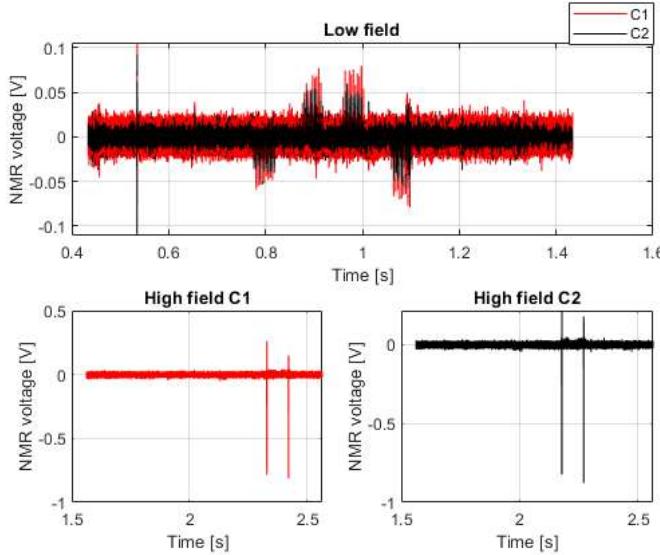
j=j+1;
disp(strcat('No file for iteration...',num2str(i-1)));
iter_marker(j) = i;

end
end;

lgd = legend(h1,'C1','C2');
set(lgd,'position',[0.82, 0.945, 0.06, 0.03]);
legend's position
disp(strcat('Number of Marker signals recorded...',num2str(numberofMarkers_recorded)))
% Adjusting

```

Number of Marker signals recorded...15



Remove zero elements due to skipped files, and calculate statistics

```

if length(iter) > 0
    for j = 1:length(iter);
        DriftC1(iter(j)) = [];
        DriftC2(iter(j))= [];
        DriftC1ex(iter(j)) = [];
        DriftC2ex(iter(j)) = [];
        DriftC1_new(iter(j)) = [];
        DriftC2_new(iter(j))= [];
        DriftC1ex_new(iter(j)) =[];
        DriftC2ex_new(iter(j)) = [];
        DeltaC1C2_inj(iter(j)) = [];
        DeltaC1C2_ex(iter(j)) = [];
        M1markerC1(iter(j)) = [];
        M1markerC2(iter(j)) = [];
        M2markerC1(iter(j)) = [];
        M2markerC2(iter(j)) = [];
        BdotC1(iter(j)) = [];
        BdotC2(iter(j)) = [];
        stab_rmsC1(iter(j)) = [];
        stab_rmsC2(iter(j)) = [];
        pp_rmsC1(iter(j)) = [];
        pp_rmsC2(iter(j)) = [];
        stab_rmsClex(iter(j)) =[];
        stab_rmsC2ex(iter(j)) = [];
        pp_rmsClex(iter(j)) = [];
        pp_rmsC2ex(iter(j)) = [];
        RepClinj(iter(j)) = [];
        RepC2inj(iter(j)) = [];
        RepClex(iter(j)) = [];
        RepC2ex(iter(j)) = [];
        fft_C1(iter(j)) = [];
        fft_C2(iter(j)) = [];
        fft_C1_Bdot(iter(j)) = [];
        fft_C2_Bdot(iter(j)) = [];
    end;
    dVM1C1 = nonzeros(dVM1C1)';
    dVM2C1 = nonzeros(dVM2C1)';
    dVM1C2 = nonzeros(dVM1C2)';
    dVM2C2 = nonzeros(dVM2C2)';
    fft_M1C1 = nonzeros(fft_M1C1)';
    fft_M1C2 = nonzeros(fft_M1C2)';
    fft_M2C1 = nonzeros(fft_M2C1)';
    fft_M2C2 = nonzeros(fft_M2C2)';
end;

dVM1C1 = dVM1C1(1:2:end);
dVM2C1 = dVM2C1(1:2:end-1);
dVM1C2 = dVM1C2(1:2:end);
dVM2C2 = dVM2C2(2:2:end);
fft_M1C1 = fft_M1C1(1:2:end);
fft_M2C1 = fft_M2C1(1:2:end-1);
fft_M1C2 = fft_M1C2(1:2:end);

```

```
fft_M2C2 = fft_M2C2(2:2:end);
```

Averages over N cycles

Bdot average

```
BdotC1_average = mean(BdotC1);
BdotC2_average = mean(BdotC2);

% Difference between measured chains
DeltaC1C2_average = mean(DeltaC1C2_inj);
DeltaC1C2e_average = mean(DeltaC1C2_ex);

% Drift
DriftC1_average = mean(DriftC1);
DriftC2_average = mean(DriftC2);
DriftC1e_average = mean(DriftClex);
DriftC2e_average = mean(DriftC2ex);

DriftC1new_average = mean(DriftC1_new);
DriftC2new_average = mean(DriftC2_new);
DriftC1e_new_average = mean(DriftClex_new);
DriftC2e_new_average = mean(DriftC2ex_new);

% Noise level
StabC1_average = mean(stab_rmsC1);
StabC2_average = mean(stab_rmsC2);
ppC1_average = mean(pp_rmsC1);
ppC2_average = mean(pp_rmsC2);
StabClex_average = mean(stab_rmsClex);
StabC2ex_average = mean(stab_rmsC2ex);
ppClex_average = mean(pp_rmsClex);
ppC2ex_average = mean(pp_rmsC2ex);

% Marker signal amplitudes
dVM1C1_average = mean(dVM1C1);
dVM2C1_average = mean(dVM2C1);
dVM1C2_average = mean(dVM1C2);
dVM2C2_average = mean(dVM2C2);
dVM1C1_std = std(dVM1C1);
dVM2C1_std = std(dVM2C1);
dVM1C2_std = std(dVM1C2);
dVM2C2_std = std(dVM2C2);

% Markers repeatability in sec
M1markerC1_rep = std(M1markerC1);
M1markerC2_rep = std(M1markerC2);
M2markerC1_rep = std(M2markerC1);
M2markerC2_rep = std(M2markerC2);

% Standard deviation time at injection and extraction time
Clinj_stddev = std(RepClinj);
C2inj_stddev = std(RepC2inj);
Clex_stddev = std(RepClex);
C2ex_stddev = std(RepC2ex);

% Mean field level at injection and extraction time
C1_mean = mean(RepClinj);
C2_mean = mean(RepC2inj);
Clex_mean = mean(RepClex);
C2ex_mean = mean(RepC2ex);

%FFT
FFT_C1_mean = mean(fft_C1);
FFT_C2_mean = mean(fft_C2);
FFT_C1_Bdot_mean = mean(fft_C1_Bdot);
FFT_C2_Bdot_mean = mean(fft_C2_Bdot);
fft_M1C1_mean = mean(fft_M1C1)';
fft_M1C2_mean = mean(fft_M1C2)';
fft_M2C1_mean = mean(fft_M2C1)';
fft_M2C2_mean = mean(fft_M2C2)';
FFT_C1_std = std(fft_C1);
FFT_C2_std = std(fft_C2);
FFT_C1_Bdot_std = std(fft_C1_Bdot);
FFT_C2_Bdot_std = std(fft_C2_Bdot);
fft_M1C1_std = std(fft_M1C1)';
fft_M1C2_std = std(fft_M1C2)';
fft_M2C1_std = std(fft_M2C1)';
fft_M2C2_std = std(fft_M2C2)';
```

Plots

```
figure; movegui(gcf,'northwest');
nbins = 10;
title('Noise distribution at injection')
subplot(1,2,1)
    histogram(stab_rmsC1, nbins, 'FaceColor','b','FaceAlpha',1);
    title('C1')
    xlabel('Noise level [G]');
    ylabel('Amount of samples')
subplot(1,2,2)
    histogram(stab_rmsC2, nbins, 'FaceColor','b','FaceAlpha',1);
    title('C2')
    xlabel('Noise level [G]');
    ylabel('Amount of samples')
```

```

figure; movegui(gcf,'southwest');
nbins = 10;
title('Noise distribution at extraction')
subplot(1,2,1)
    histogram(stab_rmsClex, nbins, 'FaceColor','r','FaceAlpha',1);
    title('C1')
    xlabel('Noise level [G]')
    ylabel('Amount of samples')
subplot(1,2,2)
    histogram(stab_rmsC2ex, nbins, 'FaceColor','r','FaceAlpha',1);
    title('C2')
    xlabel('Noise level [G]')
    ylabel('Amount of samples')

% Quality signals
figure; set(gcf, 'Position', [400, 150, 1100, 750])
subplot(3,3,8)
    boxplot([DeltaC1C2_inj; DeltaC1C2_ex],'Labels',{'Injection','Extraction'}); ylabel('Field [G]'); title('C1 - C2');
subplot(3,3,1)
    boxplot([DriftC1; DriftC2],'Labels',{'C1', 'C2'}); ylabel('Field per second [G/s]'); title('Drift at injection');
subplot(3,3,2)
    boxplot([DriftClex; DriftC2ex],'Labels',{'C1', 'C2'}); ylabel('Field per second [G/s]'); title('Drift at extraction');
subplot(3,3,3)
    boxplot([stab_rmsC1; stab_rmsC2],'Labels',{'C1', 'C2'}); ylabel('Stability [G]'); title('RMS Stability at injection');
subplot(3,3,4)
    boxplot([stab_rmsClex; stab_rmsC2ex],'Labels',{'C1', 'C2'}); ylabel('Stability [G]'); title('RMS Stability at extraction');
subplot(3,3,5)
    boxplot([pp_rmsC1; pp_rmsC2],'Labels',{'C1', 'C2'}); ylabel('Stability [G]'); title('p-p Stability at injection');
subplot(3,3,6)
    boxplot([pp_rmsClex; pp_rmsC2ex],'Labels',{'C1', 'C2'}); ylabel('Stability [G]'); title('p-p Stability at extraction');

% Other signals
figure; set(gcf, 'Position', [600, 0, 800, 550])
subplot(2,2,1)
    boxplot([dVM1C1; dVM1C2],'Labels',{'C1','C2'}); ylabel('Voltage [V]'); title('Low-field marker amplitude');
subplot(2,2,2)
    boxplot([dVM2C1; dVM2C2],'Labels',{'C1','C2'}); ylabel('Voltage [V]'); title('High-field marker amplitude');
subplot(2,2,3,5)
    boxplot([BdotC1; BdotC1],'Labels',{'C1','C2'}); ylabel('Bdot [G/s]'); title('Maximum Bdot value');

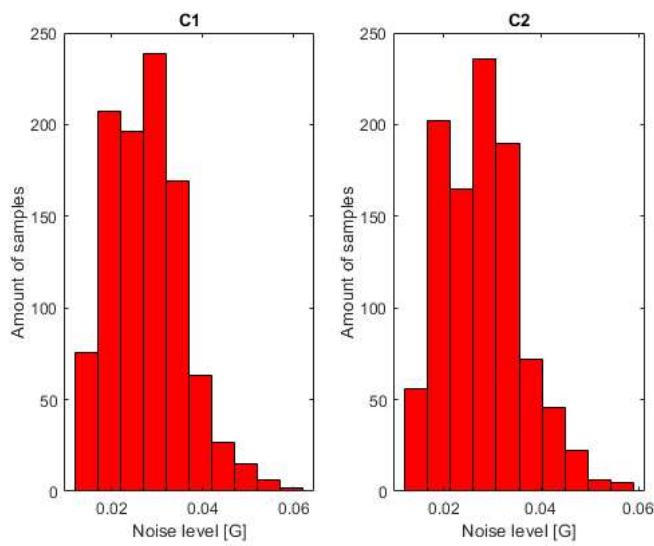
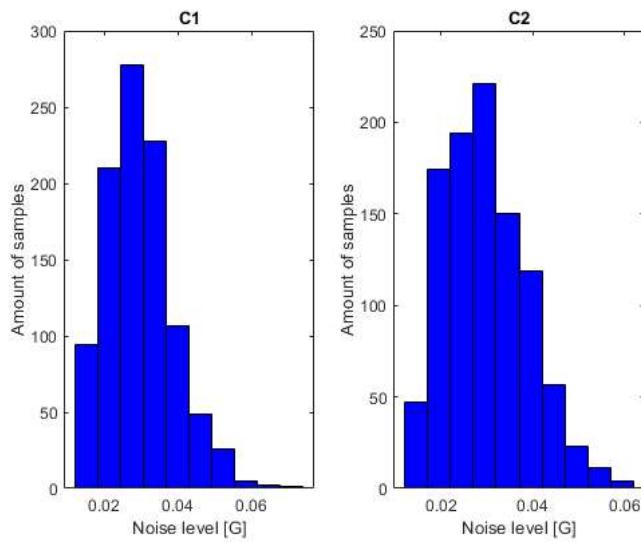
% FFTs
figure; set(gcf, 'Position', [600, 600, 800, 550])
subplot(2,2,1)
    boxplot([fft_C1; fft_C2],'Labels',{'C1','C2'}); ylabel('\Psi(f)'); title('\Psi(f) of B-train');
subplot(2,2,2)
    boxplot([fft_M1C1; fft_M1C2],'Labels',{'C1','C2'}); ylabel('\Psi(f)'); title('\Psi(f) of Marker 1');
subplot(2,2,3)
    boxplot([fft_M2C1; fft_M2C2],'Labels',{'C1','C2'}); ylabel('\Psi(f)'); title('\Psi(f) of Marker 2');
subplot(2,2,4)
    boxplot([fft_C1_Bdot; fft_C2_Bdot],'Labels',{'C1','C2'}); ylabel('\Psi(f)'); title('\Psi(f) of Bdot');

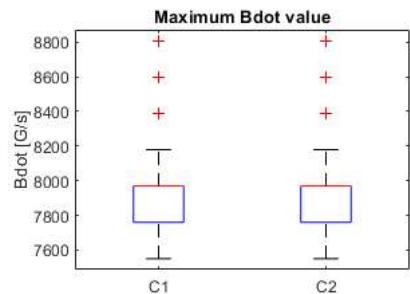
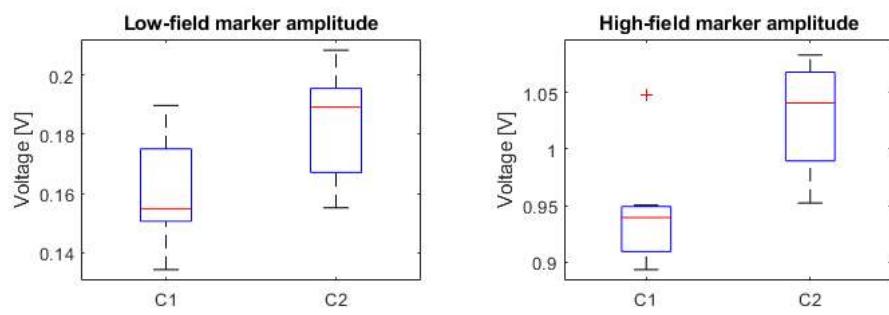
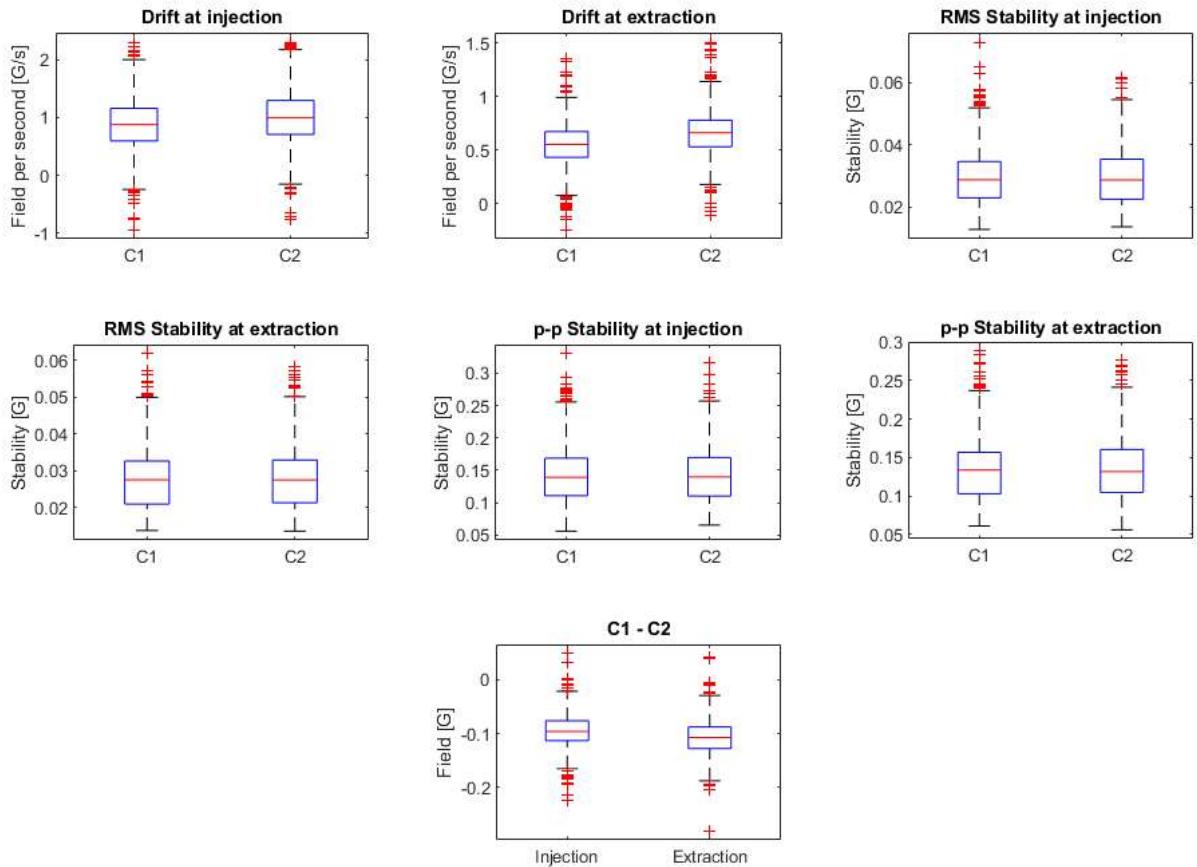
% Statistical measurements
figure; set(gcf, 'Position', [700, 150, 800, 550])
subplot(2,2,1)
    boxplot([M1markerC1; M1markerC2],'Labels',{'C1','C2'}); ylabel('Time [s]'); title('Low-field marker time');
subplot(2,2,2)
    boxplot([M2markerC1; M2markerC2],'Labels',{'C1','C2'}); ylabel('Time [s]'); title('High-field marker time');
subplot(2,2,3)
    boxplot([RepClinj; RepC2inj],'Labels',{'C1','C2'}); ylabel('Field [G]'); title('Field reproducibility at injection');
subplot(2,2,4)
    boxplot([RepClex; RepClex],'Labels',{'C1','C2'}); ylabel('Field [G]'); title('Field reproducibility at extraction');

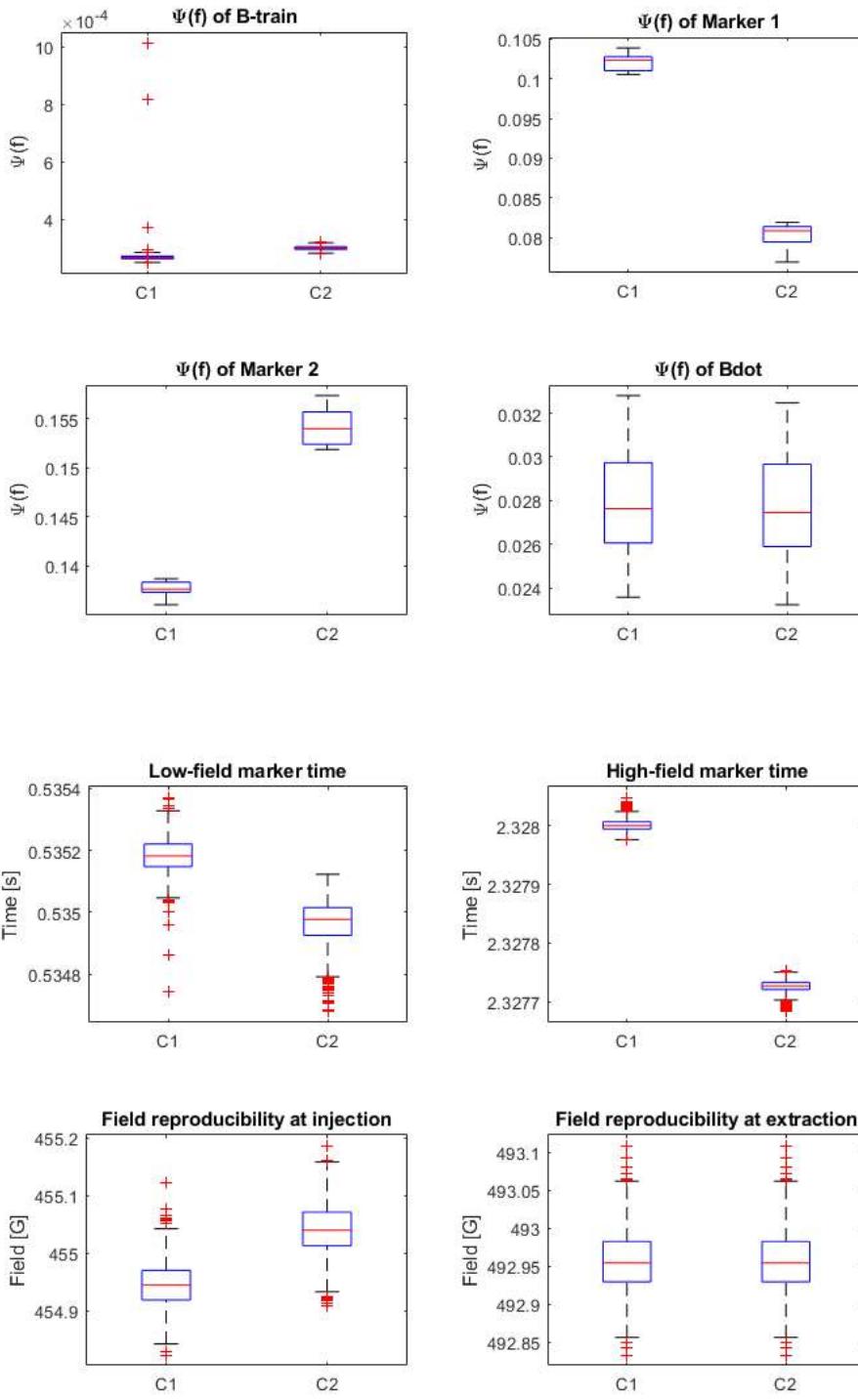
disp('Data plotting done') %update the loading bar

```

Data plotting done







Output indicators

```

disp('-----Per-cycle magnetic measurement indicators-----');
disp(strcat('B @ injection:.....', num2str(C1_mean), ' G'));
disp(strcat('B @ extraction:.....', num2str( Clex_mean), ' G'));
disp('-----Systematic errors-----');
disp(strcat('Measured chain difference @ injection:.....', num2str(DeltaC1C2_average), ' G'));
disp(strcat('Measured chain difference @ extraction:.....', num2str(DeltaC1C2e_average), ' G'));
disp('-----Drift-----');
disp(strcat('Measured drift C1 @ injection:.....', num2str(DriftC1_average), ' G/s'));
disp(strcat('Measured drift C2 @ injection:.....', num2str( DriftC2_average), ' G/s'));
disp(strcat('Measured drift C1 @ extraction:.....', num2str(DriftC1e_average), ' G/s'));
disp(strcat('Measured drift C2 @ extraction:.....', num2str( DriftC2e_average), ' G/s'));
disp('-----Drift NEW-----');
disp(strcat('Measured drift C1 @ injection:.....', num2str(DriftC1new_average), ' G/s'));
disp(strcat('Measured drift C2 @ injection:.....', num2str( DriftC2new_average), ' G/s'));
disp(strcat('Measured drift C1 @ extraction:.....', num2str(DriftC1enew_average), ' G/s'));
disp(strcat('Measured drift C2 @ extraction:.....', num2str( DriftC2enew_average), ' G/s'));
disp('-----Noise levels-----');
disp(strcat('Measured noise level C1 @ injection:.....', num2str(StabC1_average), ' G'));
disp(strcat('Measured noise level C2 @ injection:.....', num2str(StabC2_average), ' G'));
disp(strcat('Measured noise level C1 @ extraction:.....', num2str(StabClex_average), ' G'));
disp(strcat('Measured noise level C2 @ extraction:.....', num2str(StabC2ex_average), ' G'));
disp(strcat('Peak-Peak noise level C1 @ injection:.....', num2str(ppC1_average), ' G'));
disp(strcat('Peak-Peak noise level C2 @ injection:.....', num2str(ppC2_average), ' G'));

```

```

disp(strcat('Peak-Peak noise level C1 @ extraction:.....', num2str(ppClex_average), ' G'));
disp(strcat('Peak-Peak noise level C2 @ extraction:.....', num2str(ppC2ex_average), ' G'));
disp('-----Bdot signal-----');
disp(strcat('Peak Bdot level C1 :.....', num2str(BdotC1_average), ' G/s'));
disp(strcat('Peak Bdot level C2 :.....', num2str(BdotC2_average), ' G/s'));
disp('-----Marker Signals-----');
disp(strcat('Time harmonic B C1 :.....', num2str(FFT_C1_mean), ' ', '+/-', num2str(FFT_C1_std), ' '));
disp(strcat('Time harmonic B C2 :.....', num2str(FFT_C2_mean), ' ', '+/-', num2str(FFT_C2_std), ' '));
disp(strcat('Time harmonic Bdot C1 :.....', num2str(FFT_C1_Bdot_mean), ' ', '+/-', num2str(FFT_C1_Bdot_std), ' '));
disp(strcat('Time harmonic Bdot C2 :.....', num2str(FFT_C2_Bdot_mean), ' ', '+/-', num2str(FFT_C2_Bdot_std), ' '));
disp(strcat('M1 Voltage C1 :.....', num2str(dVM1C1_average), ' V', '+/-', num2str(dVM1C1_std), ' V'));
disp(strcat('M2 Voltage C1 :.....', num2str(dVM2C1_average), ' V', '+/-', num2str(dVM2C1_std), ' V'));
disp(strcat('M1 Voltage C2 :.....', num2str(dVM1C2_average), ' V', '+/-', num2str(dVM1C2_std), ' V'));
disp(strcat('M2 Voltage C2 :.....', num2str(dVM2C2_average), ' V', '+/-', num2str(dVM2C2_std), ' V'));
disp(strcat('Time harmonic M1 C1 :.....', num2str(fft_M1C1_mean), ' ', '+/-', num2str(fft_M1C1_std), ' '));
disp(strcat('Time harmonic M2 C1 :.....', num2str(fft_M2C1_mean), ' ', '+/-', num2str(fft_M2C1_std), ' '));
disp(strcat('Time harmonic M1 C2 :.....', num2str(fft_M1C2_mean), ' ', '+/-', num2str(fft_M1C2_std), ' '));
disp(strcat('Time harmonic M2 C2 :.....', num2str(fft_M2C2_mean), ' ', '+/-', num2str(fft_M2C2_std), ' '));
disp('-----Statistical magnetic measurement indicators-----');
disp(strcat('Measured field rep C1 @ injection:.....', num2str(C1inj_stdev), ' G'));
disp(strcat('Measured field rep C2 @ injection:.....', num2str(C2inj_stdev), ' G'));
disp(strcat('Measured field rep C1 @ extraction:.....', num2str(Clex_stdev), ' G'));
disp(strcat('Measured field rep C2 @ extraction:.....', num2str(C2ex_stdev), ' G'));
disp(strcat('Marker jitter C1 @ 450 G, 3400 G:.....', num2str(M1markerC1_rep*1e6), ' us,.....', num2str(M2markerC1_rep*1e6), ' us'));
disp(strcat('Marker jitter C2 @ 450 G, 3400 G:.....', num2str(M1markerC2_rep*1e6), ' us,.....', num2str(M2markerC2_rep*1e6), ' us'));
disp('-----END-----');

```

-----Per-cycle magnetic measurement indicators-----
B @ injection:.....454.9457 G
B @ extraction:.....492.9574 G
-----Systematic errors-----
Measured chain difference @ injection:.....-0.09507 G
Measured chain difference @ extraction:.....-0.10723 G
-----Drift-----
Measured drift C1 @ injection:.....0.8718 G/s
Measured drift C2 @ injection:.....0.9999 G/s
Measured drift C1 @ extraction:.....0.54858 G/s
Measured drift C2 @ extraction:.....0.65819 G/s
-----Drift NEW-----
Measured drift C1 @ injection:.....0.00089072 G/s
Measured drift C2 @ injection:.....0.0010013 G/s
Measured drift C1 @ extraction:.....0.00055572 G/s
Measured drift C2 @ extraction:.....0.00066226 G/s
-----Noise levels-----
Measured noise level C1 @ injection:.....0.029556 G
Measured noise level C2 @ injection:.....0.029677 G
Measured noise level C1 @ extraction:.....0.027707 G
Measured noise level C2 @ extraction:.....0.027889 G
Peak-Peak noise level C1 @ injection:.....0.1426 G
Peak-Peak noise level C2 @ injection:.....0.14296 G
Peak-Peak noise level C1 @ extraction:.....0.13441 G
Peak-Peak noise level C2 @ extraction:.....0.13488 G
-----Bdot signal-----
Peak Bdot level C1 :.....7948.208 G/s
Peak Bdot level C2 :.....7947.1606 G/s
-----Marker Signals-----
Time harmonic B C1 :.....0.00027025+/-3.0024e-05
Time harmonic B C2 :.....0.00030149+/-6.6725e-06
Time harmonic Bdot C1 :.....0.027868+/-0.0020958
Time harmonic Bdot C2 :.....0.027782+/-0.0020958
M1 Voltage C1 :.....0.16063 V+/-0.017929V
M2 Voltage C1 :.....0.94339 V+/-0.050715V
M1 Voltage C2 :.....0.18334 V+/-0.018222V
M2 Voltage C2 :.....1.0276 V+/-0.049095V
Time harmonic M1 C1 :.....0.10209+/-0.0011468
Time harmonic M2 C1 :.....0.1377+/-0.00088054
Time harmonic M1 C2 :.....0.080268+/-0.0016327
Time harmonic M2 C2 :.....0.1542+/-0.0020548
-----Statistical magnetic measurement indicators-----
Measured field rep C1 @ injection:.....0.039686 G
Measured field rep C2 @ injection:.....0.043457 G
Measured field rep C1 @ extraction:.....0.040156 G
Measured field rep C2 @ extraction:.....0.041897 G
Marker jitter C1 @ 450 G, 3400 G:.....58.508 us,.....9.8149 us
Marker jitter C2 @ 450 G, 3400 G:.....70.5241 us,.....10.2094 us
-----END-----