ENSC 477/895

Biomedical Image Acquisition

Lab 1 – MEG

Lestley Gabo	**301170055**	**lgabo@sfu.ca**
Yousra Wakil	**301298273**	**ywakil@sfu.ca**
Karen Ly-Ma	**301148442**	**klyma@sfu.ca**

1 Introduction

The brain generates magnetic fields. These magnetic fields are measured in units of Tesla (T) and brain waves specifically are of the order 10-100 FemtoTesla (ft) or 10^{-15} Tesla, thus the magnetic fields generated by the brain are very small. The small magnetic fields are created by the neurons inside the brain when they transmit information to other nerve cells, muscles, and gland cells. As they transmit information, they create small magnetic fields that add up together which then become measurable and detectable signals. [1]

This lab focuses in a non-invasive way of measuring these signals. A magnetoencephalography (MEG) machine was used on a volunteer to directly image the neuronal activity. An MEG measures a 2D magnetic field (fT) data on a helmet surface over time (seconds) from a 3D brain response. So the MEG Systems measure the summation of the magnetic fields generated by the currents created by the neurons in the brain. [1]

1.1 Purpose

We want to explore the reconstruction of MEG images. To do this, we collect data using a 151 Channel CTF MEG in an unshielded open environment. Since the MEG machine is very sensitive to human subject has to prepare for this lab by being metal free when inside the machine.

Firstly, we will be collecting unshielded noise data at 300 Hz and at 1200 Hz to understand the effects of sample rate and length on our data. Secondly, we placed in the MEG machine a phantom object that created a magnetic dipole at 600 Hz with head localization before and after. The phantom should allow us to observe the signal morphology and amplitudes across the different sensor locations. Thirdly, the volunteer goes in the MEG machine relaxed and eyes open for 200 seconds, at 1200 Hz and then again with eyes closed. We will be observing the volunteer's brain neurons firing and see the differences of the brain signals with open and closed eyes. Lastly, the volunteer is tested with sound delivery for 200 seconds at 1200 Hz for us to get an auditory evoked data allowing us to observe the signals coming from the primary auditory cortex and to allowing us to observe the latencies between the sound delivery and brain response. The observations made from each experiment above should allow us to reach our goal which is to explore the reconstruction of the MEG images generated by our data.

2 Methods

2.1 Data Collection

The software allowed us to look at our collected data. We could see our data in frequency domain and could even see it when added with notch, low, and high filters. We processed our data by turning on 3rd gradient, noise correction, and 60 Hz notch. However, we were notified that the actual data we will be working on this lab will only be raw data. The raw data would have bad channels, noise, and it would not have any filters added into it. The necessary steps to reconstruct a MEG image would have to be done by ourselves using MATLAB.

2.1.1 Unshielded Noise Data

To gather the unshielded noise data, we set the software in the computer that controlled the MEG machine to start collecting data at 300 Hz for 10 seconds for 10 trials. Then, we changed the collection of data to 1200 Hz for 2048 points for 20 trials. We made sure that nothing metallic was near the MEG machine, we were all at least 15 meters away near the computer area.

2.1.2 Magnetic Phantom Data

A CTF Magnetic Dipole Phantom was placed inside the MEG machine. MRI Co-registration is needed so even though it was just a phantom it was still hooked up with the three landmarks. These are the nasion, left, and right preauricular landmarks. These MRI landmark points are digitally loaded to become landmarks in MEG space [1]. The phantom data is then gathered in 600 Hz frequency for 1 second for 1 trial with head localization before and after.

2.1.3 Resting MEG Eyes Open and Resting MEG Eyes Closed

This time a volunteer was put inside the MEG machine. The person's nasion, left, and right preauricular landmarks had to be located. There were three wires for each landmark. The nasion wire was taped right between the eyes of the volunteer. As for the left and right auricular wires, they were placed right at the junction between the edge of the jaw and right below the left and right ears. This created a local coordinate system. While setting up for the MRI Co-registration, the ear monitors which are also part of the MEG machine are worn by the volunteer. This part is used in the Auditory Evoked Data procedure.

When inside the MEG machine, the person had to be relaxed while staring straight at something. The volunteer must not move at all and must only breathe and blink for 200 seconds. After 200 seconds, the volunteer then does another 200 seconds but with eyes closed. For both eyes open and eyes closed the data will be gathered at 1200 Hz with head localization before and after.

2.1.4 Auditory Evoked Data

After the closed eyes procedure, still with closed eyes, the MEG machine will start delivering sounds to the volunteer at random time intervals. The evoked data will be collected in 1200 Hz for 200 seconds.

2.2 Data Processing

2.2.1 Empty Room Noise Data

We begin by reading in the empty room noise data at 300 Hz and 1200 Hz. We transformed the data of 1200 Hz in channel 120 to Fourier space and averaged the data over 20 trials. Similarly, we performed the same process on the data of 300 Hz except averaging over 10 trials for channel 120. The results can be seen in Section 3.1.

2.2.2 Phantom

For Phantom data we added a 6^{th} order Butterworth low pass, high pass and band-stop filter at 30 Hz, 4 Hz and 60 Hz respectively. This filtered out environmental, noise floor and powerline signals. The band-stop filter behaved similar to a notch filter. We plotted an overlay of the MEG sensors from channels 32 to 181. The results can be seen in Section 3.2.

We found the peak channel by locating the maximum data point which was from channel 120. The Fourier transform of the channels were plotted. To obtain the topography map we took x and y coordinates between channels 32 to 181. Channels 106, 109, and 110 were removed since they were very noisy compared to the other channels.

The maximum data point from channel 120 occurred at around 0.24 seconds, we used this value as the peak latency to plot the topography map. We performed the function '*meshgrid*' to obtain 2D grid coordinates of each x and y coordinates and then used '*griddata'* to interpolate the data at the grid points. The topography is viewed top down using *'view'* as seen in Section 3.2.

2.2.3 Rest Data

We imported in the rest data file and downsampled the large amount of data, 240,000 points, by a quarter to 60,000 points. This reduced our sampling frequency down to 300 Hz. Similar to Section 2.2.2, we filtered the data with a 6^{th} ordered Butterworth low pass, high pass and band-stop filter at 30 Hz, 3 Hz and 60 Hz respectively. This filtered out the environmental, noise floor and powerline signals. We ran the data for the first 2000 time points to run the data for 10 seconds. We removed channel 130 due to the wide fluctuation of the signal compared to other channels. We removed artifacts by epoching the artifacts out. We transformed the data into Fourier space and plotted the stacked fast Fourier transform of all the channels. The results can be seen in Section 3.3.

Then we selected a channel with high alpha activity, channel 52, and plotted the data in frequency domain for the case of when the eyes were open and when the eyes were closed.

2.2.4 Auditory Evoked Data

In this part of the lab, we observed the evoked MEG data collection of a human brain. For this purpose, we imported the CTF AEF data file into MATLAB. The AEF data has the sampling frequency of 1200 Hz, and 200 seconds sampling time with 1 trial. Similar to the Rest Data, we downsampled our data from 240,000 by a factor of 60,000 points. This reduced our sampling frequency down to 300 Hz.

We added the stimulus channel by adding channel 1 to our MEGlist. We also applied a 6th order Butterworth high pass filter at 3 Hz to filter out the lower frequencies, a 6th order Butterworth low pass filter at 30 Hz to remove the higher frequencies above 30 Hz, and a 6th order Butterworth band-stop filter at 60 Hz. These filters filtered out the environmental noise, noise floor and powerline signals.

After that, we took Fourier transform of our filtered data across the good channels, and plotted the stacked fft. We observed that our stimulus channel had a trigger around 1.8 seconds, thus we selected the -100 ms to begin at 1.8 seconds and run for a period of 700 ms, stopping at 2.6 seconds. We removed the artifacts in our data by epoching out the time segment the artifact occurred at. We epoched the time interval out and shifted concatenated data following the artifact as a replacement. We then plotted our filtered data with respect to the epoch time interval of 1.8 to 2.6 seconds. Lastly, we plotted a topography map of the good channels at the peak of the event which occurred at 1.8 seconds using the function '*meshgrid'*, *'contour*,' and '*view'* in a similar process to how we plotted the topography map of the phantom in Section 2.2.2.

3 Results

3.1 Empty Room Noise Data Results

The results for the data collections for 300 Hz and 1200 Hz each at 10 and 20 trials respectively can be seen below. Figure 2 shows the plot for a stacked fft of channels 32 to 181 at 300 Hz for 10 trials. Figure 2 and Figure 3 shows a close up of the stacked fft to have a clearer view of the peaks and the averaged data of the empty room noise data over 10 trials at 300 Hz respectively.

Figure 4 shows the plot for a stacked fft of channels 32 to 181 at 300 Hz for 20 trials. Figure 5 and Figure 6 shows a close up of the stacked fft to have a clearer view of the peaks and the averaged data over 20 trials of the empty room noise data over 20 trials at 1200 Hz respectively.

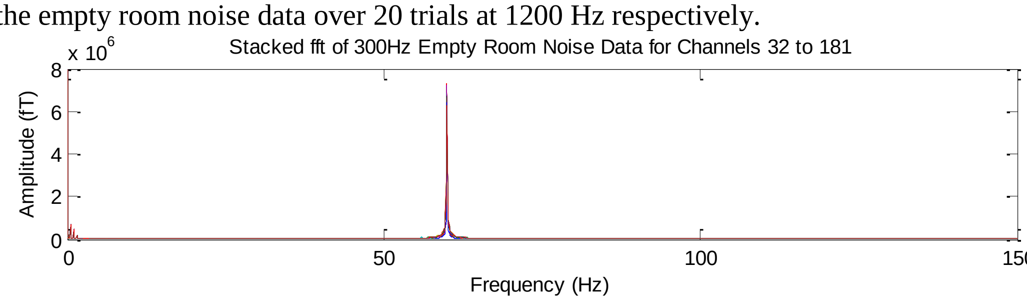

Figure 1: Stacked fft for empty room noise collection of Channels 32 to 181 at 300 Hz

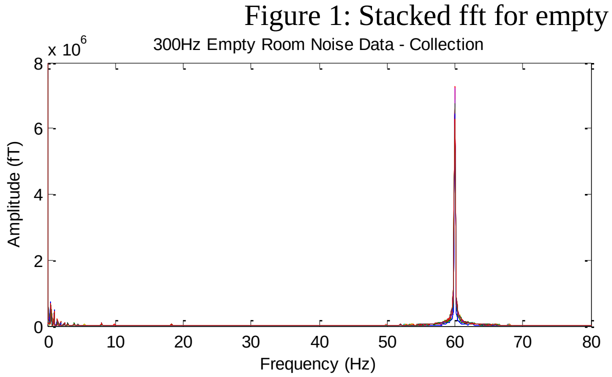

Figure 2 : Close up of Stacked fft for empty room noise collection of all channels at 300 Hz

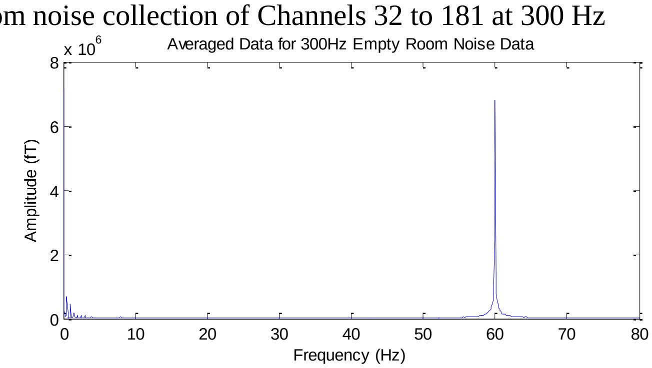

Figure 3 : Averaged data of empty room noise collection data at 300 Hz

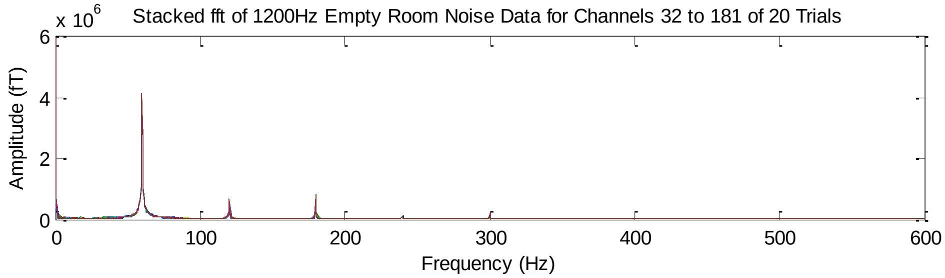

Figure 4: Stacked fft for empty room noise collection of Channels 32 to 181 at 1200 Hz

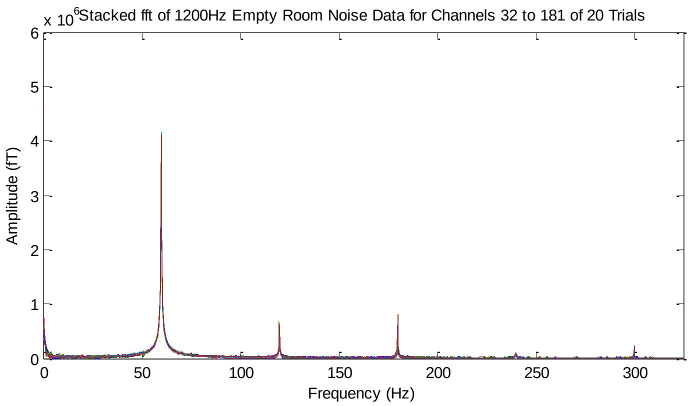

Figure 5 : Close up of Stacked fft for empty room noise collection of all channels at 1200 Hz

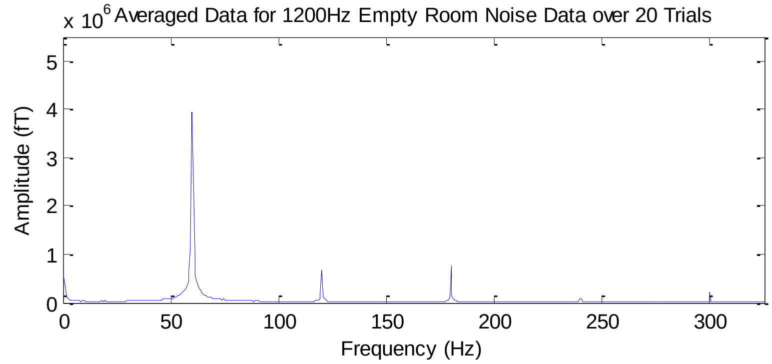

Figure 6 : Averaged data of empty room noise collection data at 1200 Hz

3.2 Magnetic Phantom Data Results

In Section 2.1.2, we read the file of CTF magnetic phantom data at 600Hz for 1 second with 1 trial. Figure 7 below shows the results of magnetic phantom data overlaid after the bad channels were removed and filtered in time domain.

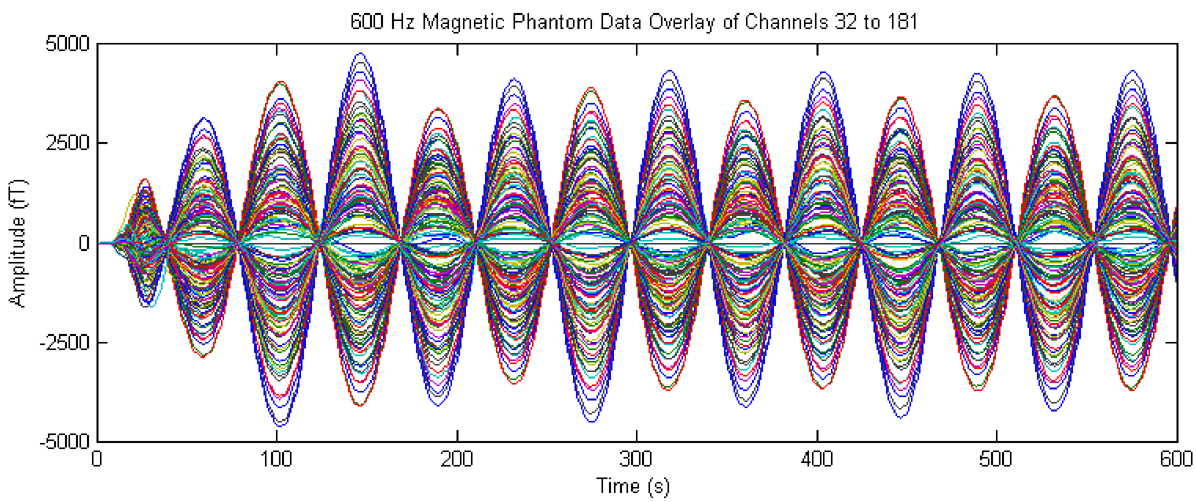

Figure 7: Magnetic phantom data overlay after bad channels removed

The result of the magnetic phantom data after it was filtered then Fourier transformed at a peak channel, channel 120, is shown in Figure 8.

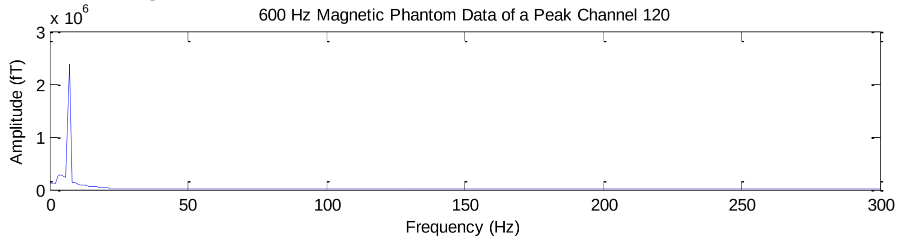

Figure 8: Magnetic phantom data at 600 Hz for peak channel 120

The topography map of the magnetic phantom data at the peak latency of 0.24 seconds is illustrated below in Figure 9.

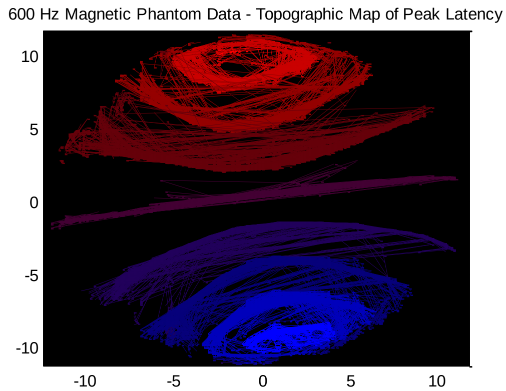

Figure 9 : Magnetic phantom data topographic map of the peak latency at 0.24 seconds

3.3 Rest Data Results

The result of the rest data for open and closed eye for channel 32 to 181 in frequency domain can be seen in Figure 10 and Figure 11 respectively

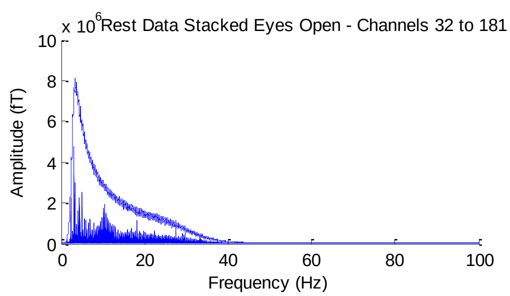

Figure 10: Rest Data Stacked Eyes Open for Channels 32 to 181

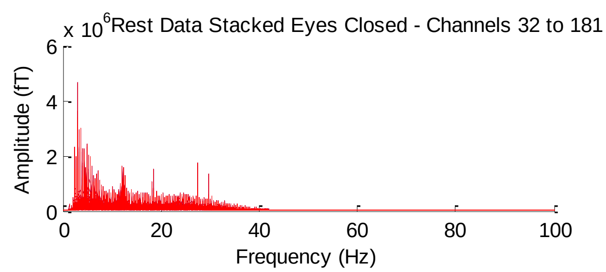

Figure 11:Rest Data Stacked Eyes Closed for Channels 32 to 181

The data for the channel with high alphas, channel 83, is plotted in time domain for open and close eyes as seen in Figure 12 and Figure 13 respectively. Figure 14 shows channel 83 data in frequency domain for both open and closed eyes. Figure 15 and Figure 16 shows a closer up of the frequency domain plot for channel 83 for open and closed eyes respectively.

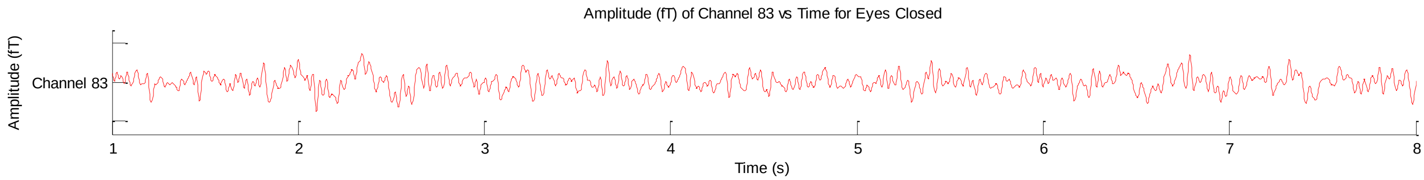

Figure 12: Amplitude of Channel 83 for Eyes Closed in Time Domain

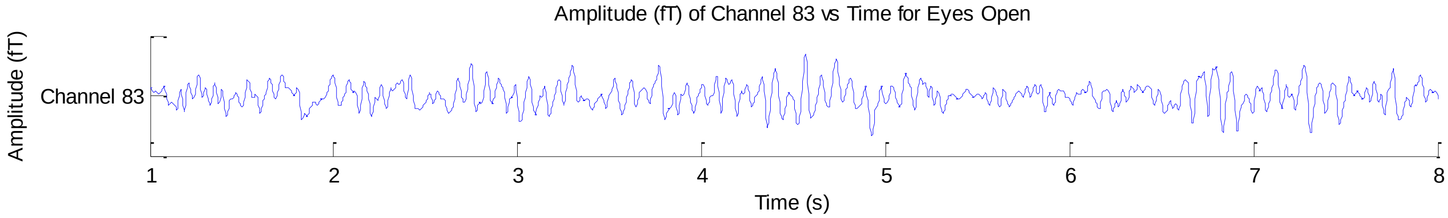

Figure 13:Amplitude of Channel 83 for Eyes Open in Time Domain

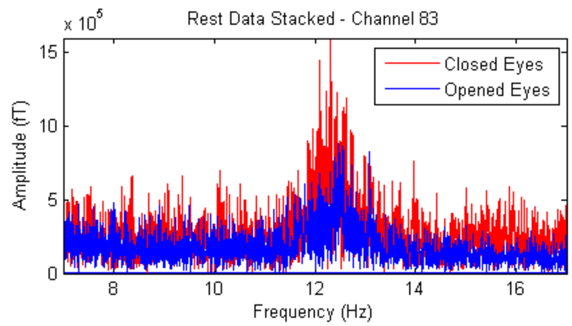

Figure 14: Rest Data Stacked for High Alpha channel Channel 83 in Frequency Domain

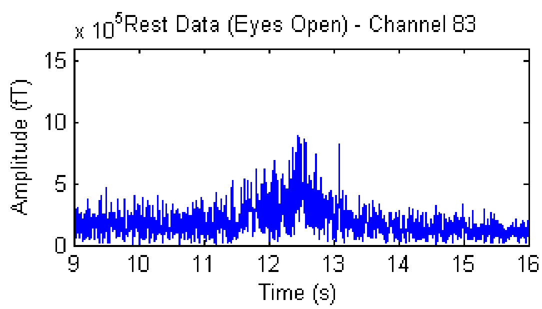

Figure 15: Rest Data for Eyes Open for Channel 83

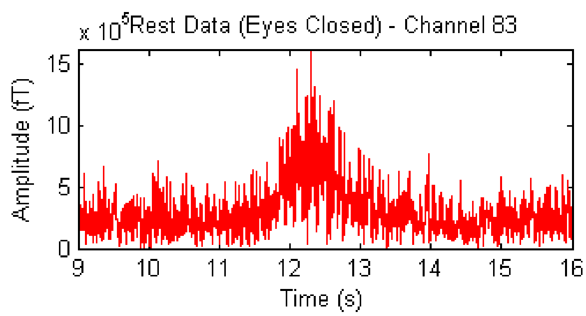

Figure 16: Rest Data for Eyes Closed for Channel 83

3.4 Auditory Evoked Data Results

The good channels were plotted together in time domain as seen in Figure 19 after filtering between time 1.8 to 2.6 seconds. The data was then averaged over time and the resulting plot is shown in Figure 20. The topography map for the good channels is illustrated in Figure 21.

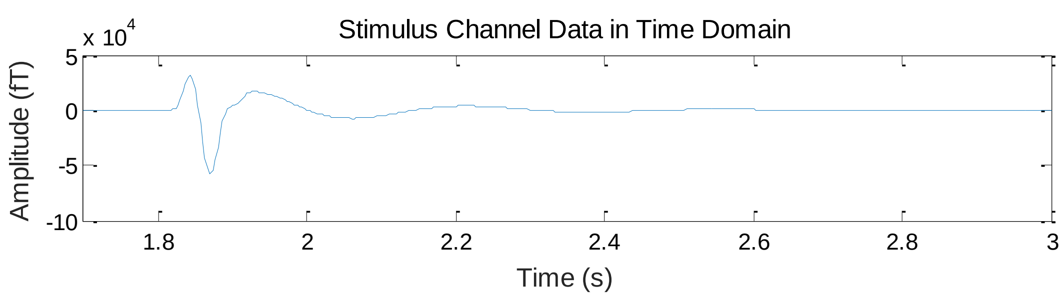

Figure 17: Stimulus Channel Data

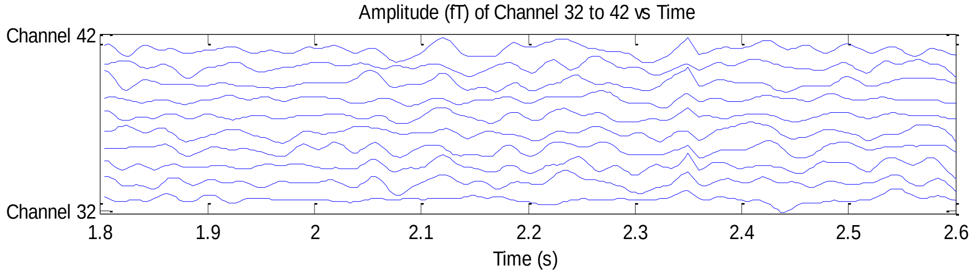

Figure 18: Data for Channel 32 to 42 vs Time

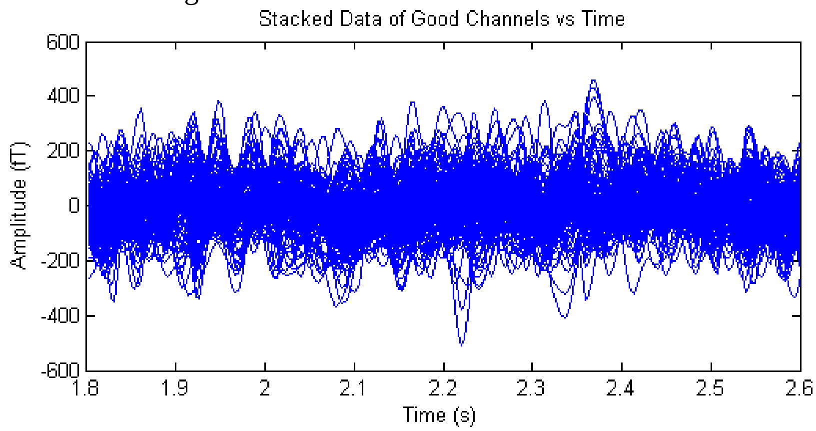

Figure 19: Overlaid Data of Good Channels in Time Domain

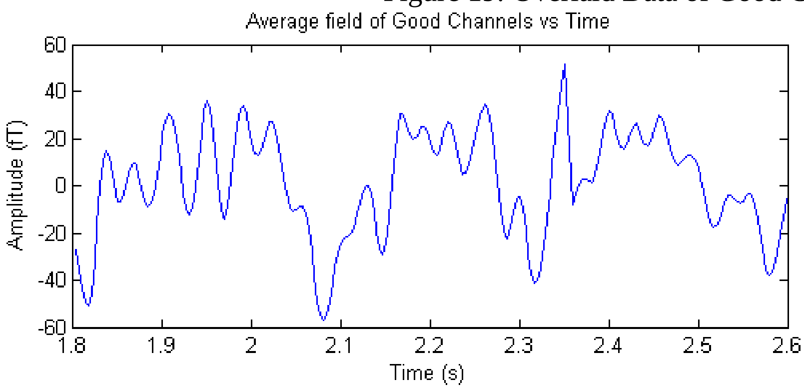

Figure 20: Averaged Signal for Good Channels in Time Domain

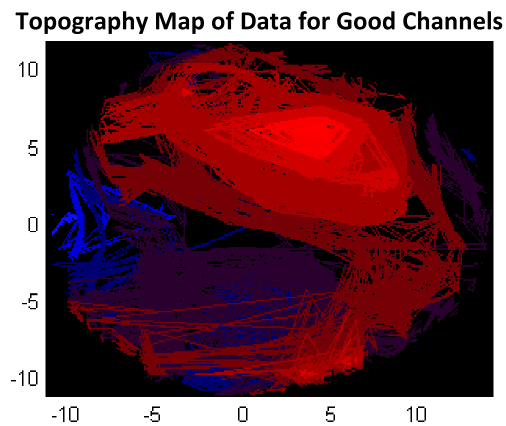

Figure 21: Topography of the Data for Good Channels

4 Discussion

4.1 Empty Room Noise Data

We observed that the MEG machine is very sensitive because it can measure very small magnetic fields. Even by bringing a small piece of metal close to the machine, we saw large changes in the data. By comparing Figure 1 and Figure 4 can see that 1200Hz. This tells us that the MEG machine is more sensitive at a higher frequency. The two figures also show us that even without putting anything inside the machine there are still noise data that are picked up. The power line frequencies that are at 60 Hz are also picked up by the machine.

The environmental noise and noise floor are detected in 1200 Hz data collection as the small peaks after 60 Hz in Figure 4, we can remove them by adding a low pass filter. The noise floor is the small peaks at around 4-7Hz as seen in Figure 1 and Figure 4. The noise floor can be removed by adding a high pass at 3 Hz. The filter cut off frequency occurs around 60Hz for 300 Hz but for 1200 Hz, the peaks start to get cut off at 300 Hz.

We can see from 1200 Hz data collection, we would obtain better resolution since we sample more often but we also pick up noise compared to 300 Hz this can be seen in Figure 1 and Figure 4. We see peaks at 125 Hz, 175 Hz and 300 Hz in Figure 4 whereas we do not see these in Figure 1. 1200 Hz allows you to capture signals up to 600 Hz from Nyquist Shannon Theorem whereas 300 you can only compare signals up to 150 Hz.

4.2 Magnetic Phantom Data

After removing the powerline noise at 60 Hz, we noticed the largest signal occurred at around 10 Hz when we plotted the fft of a peak channel, channel 120. This correspond to the frequency of the phantom dipole as seen in Figure 8. The magnetic phantom works as a validation for our MEG source reconstructions. The phantom enables us to simulate real brain data. Allowing us to measure magnetic fields even without a human subject.

4.2.1 Magnetic dipole and difference from a current dipole

To our understanding of the map at Figure 9 of a magnetic dipole, we can tell that the magnetic dipole is settled north to south. We know that the top dipole of the map are positive intensities while the bottom dipole are negative intensities. Therefore, we can conclude that when compared to the topographic maps shown in the lecture notes, the bottom dipole would be the blue coloured intensities while the top dipole would be the red coloured intensities.

If there was a current dipole instead then the topography map would show a perpendicular version of Figure 9. The current dipole going from north to south would show up as a red coloured intensity to the left of the map and blue coloured intensity to the right of the map. This is because the current dipole is perpendicular to the magnetic dipole.

4.3 Resting MEG Eyes Open and Closed

4.3.1 Eyes Open

When a human brain is placed near the sensors the difference in signals is observed which shows that the green signals in the data represent the left channels, while the blue signals represent the right channels. The brain activity is represented by alpha rhythm. The alpha rhythm is generated from the synchronous activity of a large number of neurons, and has a frequency of 8-12 Hz, which is the frequency at which the brain functions.

The neuron is a nerve cell that consists of a nucleus, dendrites, and axons. When there is a significant change in the voltage, an electrostatic potential is created called the action potential, which therefore can be called the firing of a neuron.

This can be seen in Figure 16 as it shows the high alpha activity meaning the neurons are active and firing. We see the firing of neuron in the form of brain waves. This firing of neurons generate oscillations called the alpha rhythms in the signal. The frequency of alpha rhythm is between 8 to 12 Hz in our case we can see them at 12 Hz. Figure 15 shows that there is less alpha activity when the eyes are opened compared when they are closed. This can be clearly seen when they are stacked together as illustrated in Figure 14. We observed the high alpha activity in the brain signals in channel 83 for both open and closed eyes as seen in Figure 12 and Figure 13.

4.3.2 Eyes Closed

From comparing Figure 10 and Figure 11 for eyes and open and closed, we can see that both plots show peaks at similar frequencies. There is a peak at around 5 Hz, 12 Hz, 19 Hz and around 30 Hz for both plots. The peak at 12 Hz for when the eyes are closed is slightly higher. The peak at around 4 Hz is higher when they eyes are closed up to 8×10^6 fT compared to 6×10^6 fT.

We observed a difference in the alpha rhythm between eyes open and eyes closed conditions. The alpha activity was strongest when eyes closed showing more brain activity, while a decrease in alpha activity is observed with eyes opened as seen in Figure 14. These alpha rhythms are used to represent the relaxation mode of the brain. Therefore, their activity is increased during eyes closed which is a more resting state than eyes opened when the brain is busy in processing information from outside.

4.4 Auditory Evoked Data

We used the stimulus channel which represents the tone to analyze the brain signals when it processes the tone. From Figure 17, we can see that the trigger starts at 1.8 seconds. We can see that the brain starts processing the tone after 1.9 seconds since the signals show more activity as seen in Figure 18 for channels 32 to 42. We observed the brain activity by creating plots similar to Figure 18 for all good channels. After a while we noticed the brain response dies until the next pulse of the trigger, which again activates the brain after 100 ms. So, in this way a trigger is provided to observe the brain activity to the tone. We notice the signals of the channels that had more brain activity or were more affected by the tone and corresponded them to the channel names starting with '*ML'* which is the left side of the brain.

5 Conclusion

When looking at the empty room noise data we observed that using a higher frequency like 1200 Hz we can get more details but we get more noise. On the other hand, using a lower frequency like 300 Hz generates less details and less noise. We can conclude that the MEG machine is more sensitive at higher frequencies.

The magnetic phantom data gave a peak signal at around 10 Hz and we were able to stimulate real brain data by creating the topographic map of that peak signal. It showed us two dipoles in the top and bottom of the map therefore we could conclude that the magnetic dipole is settled from north to south.

With the resting eyes open data, we saw neurons firing when there were higher alpha activities. We saw these alpha activities at 12 Hz. These alpha activities were even higher in the resting eyes closed data. Comparing the open and closed eyes data we can conclude that when our eyes are closed our brain's alpha activities are higher than when our eyes are open.

For the auditory evoked data, we observed the brain response after a stimulus was given. There are a total of 150 channels, but we observed that only some channels had brain activity on each stimulus. If only the channels corresponding to the left side of the brain responded to the stimulus, then we can conclude that that the stimulus was evoked only to the right ear.

6 Reference

[1] T. Cheung, “Medical Image Acquisition,” BME 477 - Lecture 4, 2016.

7 Appendix

MATLAB code for Empty Room Noise Data

```
%Ensc 477 code snippet for reading CTF data.
% I am using Matlab2011b

%%preliminary setup
%Execute the lines below to load the CTF toolbox

clear all

% addpath F:\Users\Daniel\Documents\ENSC477\Lab1\CTF-Matlab
%replace the path with the path you are using
%if you are running under windows, the path will contain backslashes \
%rather than forward slashes /

%%
%% Read data example
%replace file path with your data's file path
clear all

filePath='MEG_Noise_20161021_KL01.ds';
unit='fT';
prec='double';
trials=1:20;

ds1=readCTFds(filePath);
MEGlist=strmatch('M',ds1.res4.chanNames);
%MEGlist=2:180;
MEGdata1=getCTFdata(ds1,trials,MEGlist);
preTrigPt1 = ds1.res4.preTrigPts;
[MEGdata1,ds1]=setCTFDataBalance(MEGdata1,ds1,'G3BR'); %remove far field noise using 3rd gradient balance
nChannels=size(MEGlist,1);

%Calculates frequency axis for 300 Hz
% Fs=300;
% T = 1/Fs;             % Sampling period
% L = 3000;             % Length of signal
% t = (0:L-1)*T;        % Time vector
% f = Fs*(0:(L/2))/L;
%Calculates frequency axis for 1200 Hz
Fs=1200;
T = 1/Fs;             % Sampling period
L = 2048;             % Length of signal
t = (0:L-1)*T;        % Time vector
f = Fs*(0:(L/2))/L;

yaxis=[];
for trials = 1:20
Y=fft(MEGdata1(:,1,trials));
%Analyze only positive frequencies
```

```
P1 = abs(2*Y(1:L/2+1));
yaxis=[yaxis P1];
plot(f, yaxis)
title('Stacked fft of 1200Hz Empty Room Noise Data for Channels 32 to 181 of 20 Trials');
xlabel('Frequency (Hz)');
ylabel('Amplitude (fT)');
hold on
end
figure
avg_data=mean(yaxis,2);
plot(f, avg_data)
title('Averaged Data for 1200Hz Empty Room Noise Data over 20 Trials');
xlabel('Frequency (Hz)');
ylabel('Amplitude (fT)');
hold on

sample_rate=ds1.res4.sample_rate;
no_samples =ds1.res4.no_samples;
theTime=-preTrigPt1/sample_rate:1/sample_rate:(no_samples-preTrigPt1-1)/sample_rate;
tr =1; %plot trial 1
%time plot of first trial
figure
for(i=1:150)
plot(MEGdata1(:,i,tr));
hold on
end
```

MATLAB code for Phantom

```
%Ensc 477 code snippet for reading CTF data.
% I am using Matlab2011b

%%preliminary setup
%Execute the lines below to load the CTF toolbox
close all
clear all

addpath F:\Users\Daniel\Documents\ENSC477\Lab1\CTF-Matlab
%replace the path with the path you are using
%if you are running under windows, the path will contain backslashes \
%rather than forward slashes /

%%
%% Read data example
%replace file path with your data's file path
clear all
close all
filePath='F:\Users\Daniel\Documents\ENSC477\Lab1\MEG_MagneticPhantom_20161021_KL06.ds';
unit='fT';
prec='double';
```

```
trials=[1];

ds1=readCTFds(filePath);
MEGlist=strmatch('M',ds1.res4.chanNames);
%MEGlist=2:180;
%remove bad channels
goodChannels = MEGlist([1:105 107:108 111:141],:) ;
MEGdata1=getCTFdata(ds1,trials,goodChannels);
preTrigPt1 = ds1.res4.preTrigPts;
[MEGdata1,ds1]=setCTFDataBalance(MEGdata1,ds1,'G3BR'); %remove far field noise using 3rd gradient balance
nChannels=size(MEGlist,1);

sample_rate=ds1.res4.sample_rate;
no_samples =ds1.res4.no_samples;
theTime=-preTrigPt1/sample_rate:1/sample_rate:(no_samples-preTrigPt1-1)/sample_rate;

tr =1; %plot trial 1

%Calculates frequency axis
Fs=600;
T = 1/Fs;             % Sampling period
L = 600;             % Length of signal
t = (0:L-1)*T;        % Time vector
f = Fs*(0:(L/2))/L;

yaxis=[];
td_yaxis=[];

%Highpass Filter
[b_hpf,a_hpf]=butter(6,3/(Fs/2),'high');
%Lowpass Filter
[b_lpf,a_lpf]=butter(6,30/(Fs/2),'low');
%Notch filter
[b,a]=butter(6, [55/(Fs/2) 65/(Fs/2)],'stop');

%Frequency domain plot; 119 is peak channel
for sensors=120:120
    %Add band stop filter, low pass, high pass
    filter60Hz= filter(b,a, MEGdata1(:,sensors,1));
    LPF= filter(b_lpf,a_lpf, filter60Hz);
    HPF =filter(b_hpf,a_hpf, LPF);
    filtered_data(:,sensors) = HPF;
    %Fourier transform
    Y=fft(filtered_data(:,sensors));
    P1 = abs(2*Y(1:L/2+1)); % Regard only positive frequencies
    yaxis=[yaxis P1];
end
avg_data=mean(yaxis,2);
%Plot in freq domain with filter
%USe a peak channel: channel 120
```

```
plot(f,yaxis)
title('600 Hz Magnetic Phantom Data of a Peak Channel 120');
xlabel('Frequency (Hz)');
ylabel('Amplitude (fT)');
hold on

%time domain plot
figure('units','normalized','outerposition',[0 0 1 1])
for sensors=1:138
    %Add band stop filter, low pass, high pass
    filter60Hz= filter(b,a, MEGdata1(:,sensors,1));
    LPF= filter(b_lpf,a_lpf, filter60Hz);
    HPF =filter(b_hpf,a_hpf, LPF);
    filtered_data(:,sensors) = HPF;
    plot(filtered_data)
    title('600 Hz Magnetic Phantom Data Overlay of Channels 32 to 181');
    xlabel('Time (s)');
    ylabel('Amplitude (fT)');
    hold on
end
hold off
%Peak Channel
peak_val=max(max((filtered_data)))
%channel 120 is the peak channel, after bad channels removed
% 120 => peak_val = 4770.26363
% 4770.2636 is at 147/600 in the number of samples
% therefore use 147s to find the topographic data

%Creating topography map
X=[]; Y=[];
for i=[32:136 138:139 142:172]
    Xaxis= ds1.res4.senres(1, i).pos(1,1);
    Yaxis=ds1.res4.senres(1, i).pos(2,1);
    X=[X Xaxis];
    Y=[Y Yaxis];
end
figure
[X1, Y1]=meshgrid(X,Y);
Z1=griddata(X,Y, filtered_data(147,:,1), X1,Y1);
contour(X1,Y1,Z1)
view([0 90])
title('600 Hz Magnetic Phantom Data - Topographic Map of Peak Latency');
```

MATLAB code for Rest Data

```
%Ensc 477 code snippet for reading CTF data.
% I am using Matlab2011b

%%preliminary setup
%Execute the lines below to load the CTF toolbox
%close all
clear all

addpath F:\Users\Daniel\Documents\ENSC477\Lab1\CTF-Matlab
%replace the path with the path you are using
%if you are running under windows, the path will contain backslashes \
%rather than forward slashes /

%%
%% Read data example
%replace file path with your data's file path
clear all
%close all
filePath='F:\Users\Daniel\Documents\ENSC477\Lab1\MEG_Rest_20161021_KL06.ds';
unit='fT';
prec='double';
trials=[1];

ds1=readCTFds(filePath);
MEGlist=strmatch('M',ds1.res4.chanNames);
%MEGlist=2:180;
MEGdata1=getCTFdata(ds1,trials,MEGlist);
preTrigPt1 = ds1.res4.preTrigPts;
[MEGdata1,ds1]=setCTFDataBalance(MEGdata1,ds1,'G3BR'); %remove far field noise using 3rd gradient balance
nChannels=size(MEGlist,1);

sample_rate=ds1.res4.sample_rate;
no_samples =ds1.res4.no_samples;
theTime=-preTrigPt1/sample_rate:1/sample_rate:(no_samples-preTrigPt1-1)/sample_rate;

%Downsample the time and data by 1/4
ds_time = downsample(theTime,4);
ds_data = downsample(MEGdata1,4);

%Calculates frequency axis
Fs=1200/4;
T = 1/Fs;              % Sampling period
L = 240000/4;              % Length of signal
t = (0:L-1)*T;         % Time vector
f = Fs*(0:(L/2))/L;

%Highpass Filter
```

```
[b_hpf,a_hpf]=butter(6,3/(Fs/2),'high');
%Lowpass Filter
[b_lpf,a_lpf]=butter(6,30/(Fs/2),'low');
%Notch filter
[b,a]=butter(6, [55/(Fs/2) 65/(Fs/2)],'stop')

yaxis=[];
td_yaxis=[];
for sensors=1:150
    %Add band stop filter, low pass, high pass
    filter60Hz= filter(b,a, ds_data(:,sensors,1));
    LPF= filter(b_lpf,a_lpf, filter60Hz);
    HPF =filter(b_hpf,a_hpf, LPF);
    filtered_data(:,sensors) = HPF;
    %Fourier transform
    Y=fft(HPF);
    P1 = abs(2*Y(1:L/2+1)); % Regard only positive frequencies
    yaxis=[yaxis P1];
end
figure
for i=1:150
    hold on
    plot(f, yaxis(:,i),'b') %Plot frequency domain
end
ylim([0 1600000])
% end
title('Rest Data Stacked - Channels 32 to 181');
xlabel('Time (s)');
ylabel('Amplitude (fT)');

%time domain plot
for sensors=1:150
    %Filters all data
    filter60Hz= filter(b,a, ds_data(:,sensors,1));
    LPF= filter(b_lpf,a_lpf, filter60Hz);
    HPF =filter(b_hpf,a_hpf, LPF);
    plot(HPF);
end

%if you want to remove a pretrigger offset, you can uncomment this code
%offset1=sum(MEGdata1(1:preTrigPt1,:))/preTrigPt1;
%for i=1:nChannels
%    MEGdata1(:,i)=MEGdata1(:,i)-offset1(i);
%end



for (i=52:52)
    %196:2701 corresponds to 1 to 9 seconds
    hold on
    plot(ds_time,(filtered_data(:,i)),'r');
```

```
    hold on
end
%Creates manual ticks and labels
NumTicks = 3;
L = get(gca,'YLim');
set(gca,'YTick',linspace(L(1),L(2),NumTicks))
set(gca, 'YTickLabel',{'','Channel 83',''})
title('Amplitude (fT) of Channel 83 vs Time for Eyes Closed');
xlabel('Time (s)');
```

MATLAB code for Auditory Evoked Data

```
%Ensc 477 code snippet for reading CTF data.
% I am using Matlab2011b

%%preliminary setup
%Execute the lines below to load the CTF toolbox
close all
clear all

addpath F:\Users\Daniel\Documents\ENSC477\Lab1\CTF-Matlab
%replace the path with the path you are using
%if you are running under windows, the path will contain backslashes \
%rather than forward slashes /
%% Read data example
%replace file path with your data's file path
clear all
close all
filePath='F:\Users\Daniel\Documents\ENSC477\Lab1\MEG_AEF_20161021_KL06.ds';
unit='fT';
prec='double';
trials=[1];

ds1=readCTFds(filePath);
MEGlist=strmatch('M',ds1.res4.chanNames);
%Add stim channel
add_stim = [1 ; MEGlist];
%Removes bad channels
goodChannels = add_stim([1:39 42:61 63 65:66 68:145 147:148 150:151],:);
%MEGlist=2:180;

MEGdata1=getCTFdata(ds1,trials,goodChannels);
preTrigPt1 = ds1.res4.preTrigPts;
[MEGdata1,ds1]=setCTFDataBalance(MEGdata1,ds1,'G3BR'); %remove far field noise using 3rd gradient balance
nChannels=size(MEGlist,1);

sample_rate=ds1.res4.sample_rate;
no_samples =ds1.res4.no_samples;
theTime=-preTrigPt1/sample_rate:1/sample_rate:(no_samples-preTrigPt1-1)/sample_rate;
```

```

%Downsample the time and data by 1/4
ds_time = downsample(theTime,4);
ds_data = downsample(MEGdata1,4);

%Calculates frequency axis
Fs=1200/4;
T = 1/Fs;               % Sampling period
L = 240000/4;               % Length of signal
t = (0:L-1)*T;          % Time vector
f = Fs*(0:(L/2))/L;

%Highpass Filter
[b_hpf,a_hpf]=butter(6,3/(Fs/2),'high');
%Lowpass Filter
[b_lpf,a_lpf]=butter(6,30/(Fs/2),'low');
%Notch filter
[b,a]=butter(6, [55/(Fs/2) 65/(Fs/2)],'stop')

yaxis=[];
td_yaxis=[];
for sensors=2:length(goodChannels)
    %Add band stop filter, low pass, high pass
    filter60Hz= filter(b,a, ds_data(:,sensors,1));
    LPF= filter(b_lpf,a_lpf, filter60Hz);
    HPF =filter(b_hpf,a_hpf, LPF);
    filtered_data(:,sensors) = HPF;
    %Fourier transform
    Y=fft(filtered_data(:,sensors));
    P1 = abs(2*Y(1:L/2+1)); % Regard only positive frequencies
    yaxis=[yaxis P1];


end
avg_data=mean(yaxis,2);
plot(f, avg_data,'b'); %Plot frequency domain

%time domain plot
for sensors=2:length(goodChannels)
    %Filters all data
    filter60Hz= filter(b,a, ds_data(:,sensors,1));
    LPF= filter(b_lpf,a_lpf, filter60Hz);
    HPF =filter(b_hpf,a_hpf, LPF);

    td_yaxis=[td_yaxis  HPF];
end
avg=mean(td_yaxis,2);
plot(avg);
%if you want to remove a pretrigger offset, you can uncomment this code
%offset1=sum(MEGdata1(1:preTrigPt1,:))/preTrigPt1;
%for i=1:nChannels
%    MEGdata1(:,i)=MEGdata1(:,i)-offset1(i);
```

```
%end

newseg=ds_time(1231:1306)-1.74 %grabs new segment of data
epochtime=[ds_time([542:706]) newseg]; %creates a new epoch interval
epochdata=filtered_data([542:706 1231:1306],:); %concatenate data with new segment
figure
td_yaxis=[];
for i=2:144
    plot(epochtime,300*i+(epochdata(:,i))); %
    hold on
    xlim([1.8 2.6])
    td_yaxis=[td_yaxis epochdata(:,i)];
end
%creates manual ticks and labels
xlim([1.7 2.6])
ylim([400 3600])
NumTicks = 2;
L = get(gca,'YLim');
set(gca,'YTick',linspace(L(1),L(2),NumTicks))
set(gca, 'YTickLabel',{'Channel 32','Channel 42'})
title('Amplitude (fT) of Channel 32 to 42 vs Time');
xlabel('Time (s)');

avg=mean(td_yaxis,2);
figure
plot(epochtime,avg);

title('Average field of Good Channels vs Time');
xlabel('Time (s)');
ylabel('Amplitude (fT)');
X=[];
Y=[];
%To plot topography map grab good channels
for i=[32:69 72:91 93 95:96 98:175 177:178 180:181]
Xaxis= ds1.res4.senres(1, i).pos(1,1);
Yaxis=ds1.res4.senres(1, i).pos(2,1);
X=[X Xaxis];
Y=[Y Yaxis];
end
figure
[X1, Y1]=meshgrid(X,Y);
Z1=griddata(X,Y, epochdata(2,2:144), X1,Y1);
%mesh(X1,Y1,Z1)
contour(X1,Y1,Z1)
view([0 90])
```