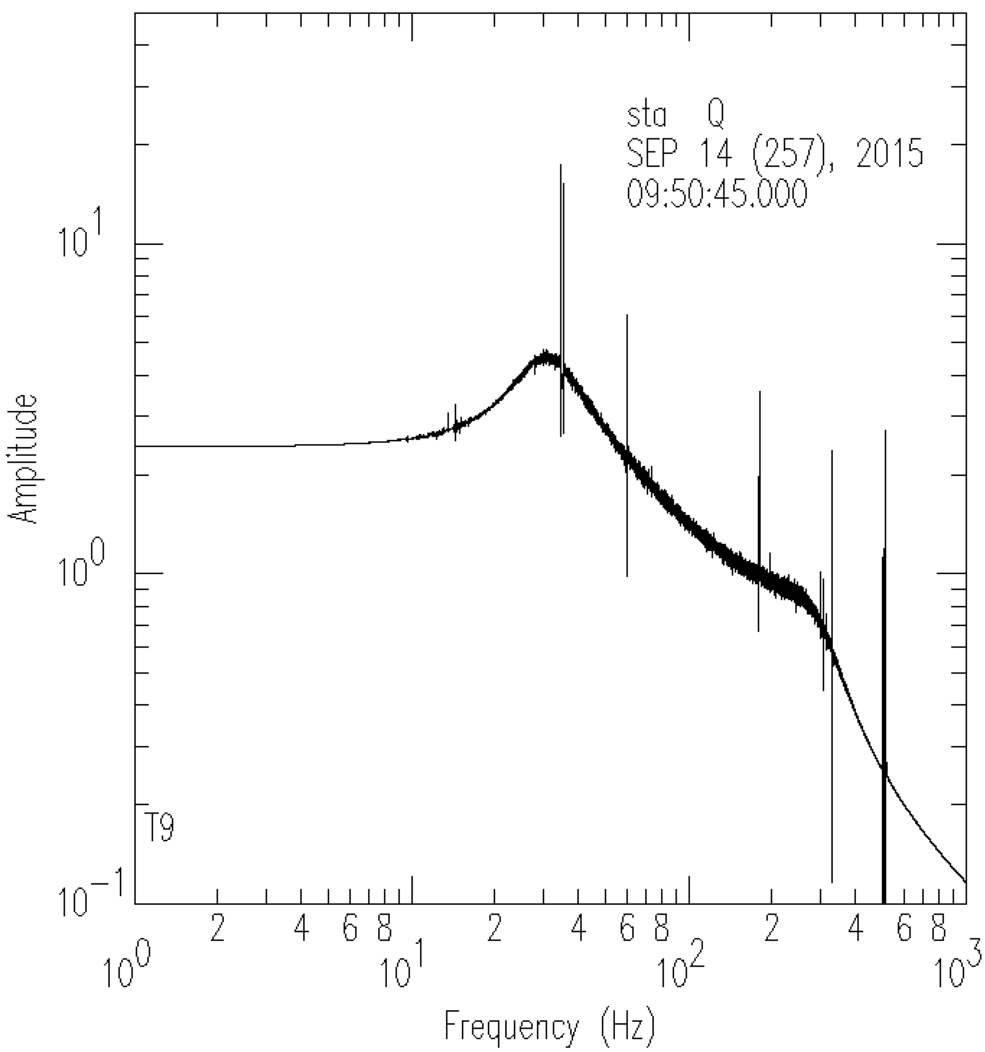
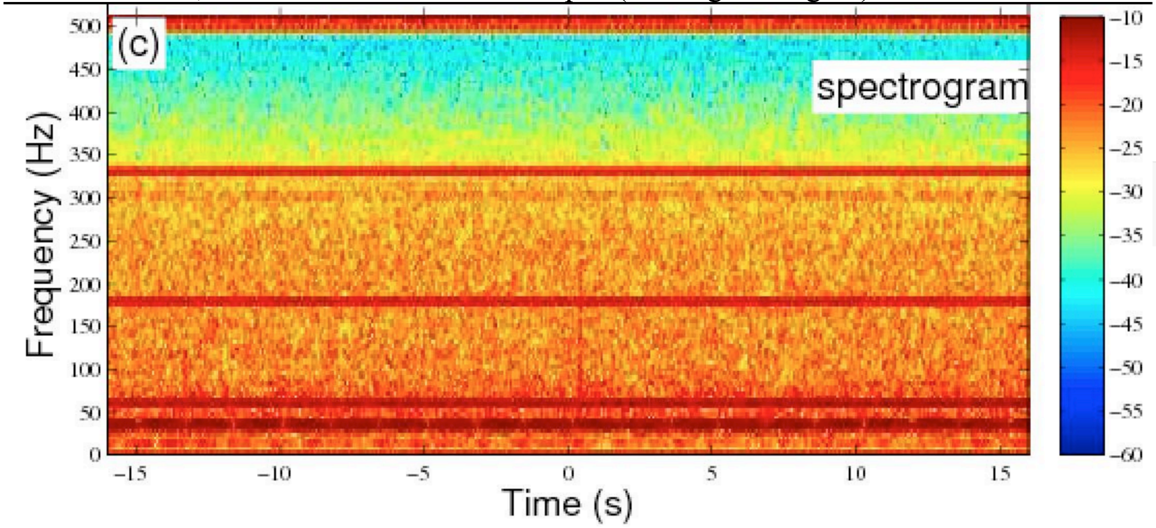
Homework 3 (EAS 8803: OBS SEISMOLOGY - SPRING SEMESTER 2019)

Total points: 100. Due 03/26/2019

1. The data associated with the gravitational waves detected by the LIGO instruments can be downloaded from the link below. Please try to reproduce the following tasks (25 pts):
   a. Please generate the spectrogram plot (i.e., frequency versus time) for both data. Identify the frequency bands that are associated with background noises. (You can use either your own codes from previous homework or built-in programs from Matlab or Python).
   b. Apply band-rejected filters to remove the frequencies associated with the background noises. Then plot them in the time domain, and use waveform cross-correlations to identify the time delays between these two data.
   c. Please take the synthetic data as templates, and compute running cross-correlations with both observed data. Identify the times where the synthetic match the best with the observed data.

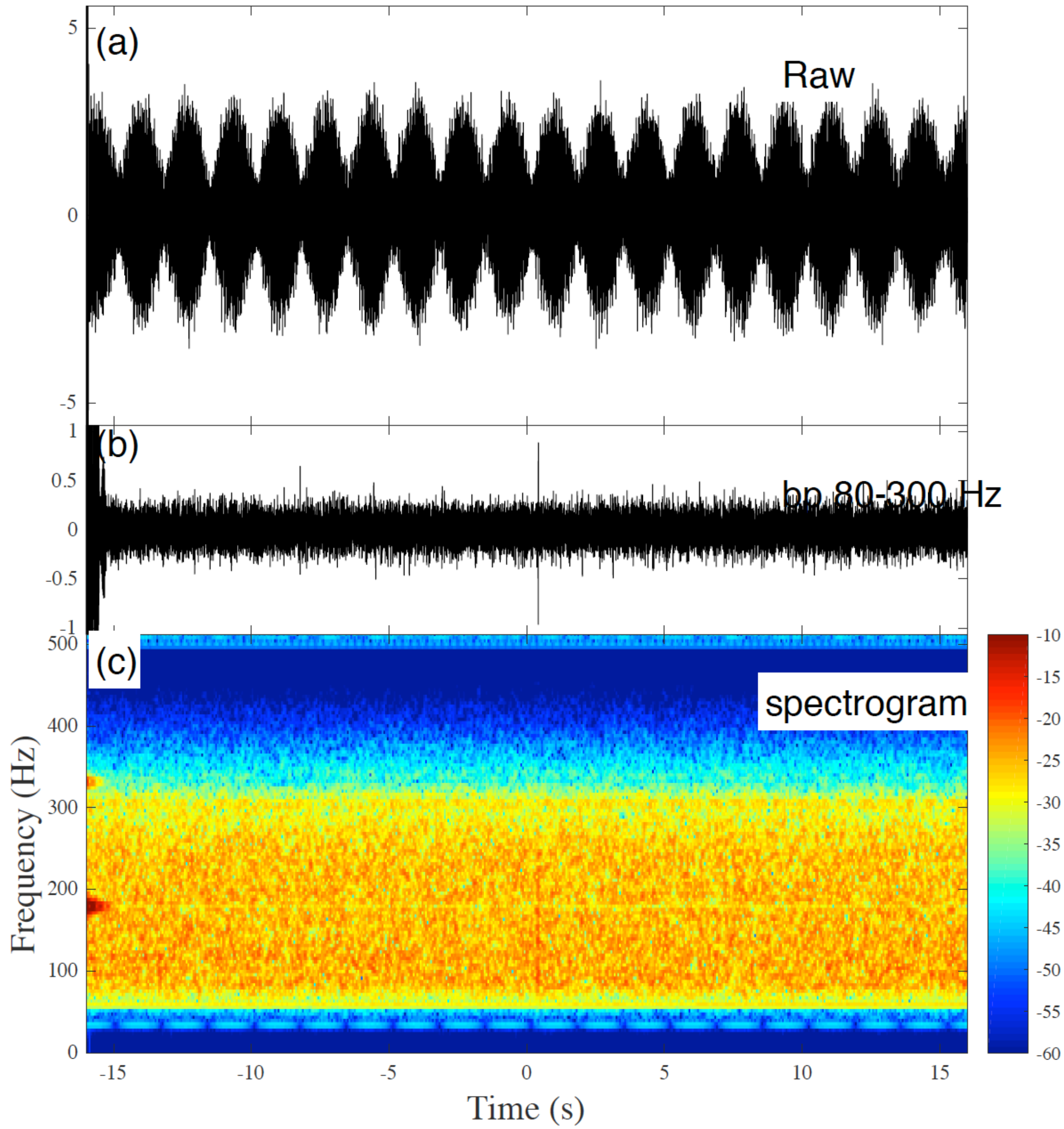


Based on the above FFT plot, and the spectrogram plot next page, we can see that there are peak noises in the following frequency ranges (20-80 Hz, including 35 Hz, 60 Hz), 180 Hz, and 330 Hz. In addition, there is a minor vertical stripes (the targeted signal) between 70-300 Hz.

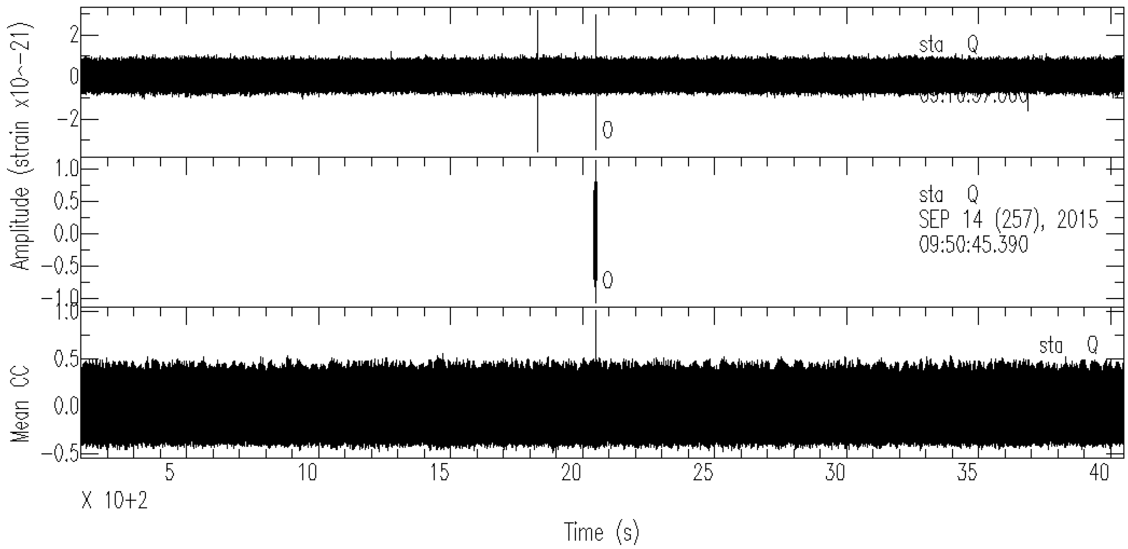


Hence, we apply a band-pass-filter of 80-300 Hz, and a notch filter (band-rejected filter) between 178-182 Hz and 329 to 333 Hz. See gen_spectrogram_Livingston.m for more details.

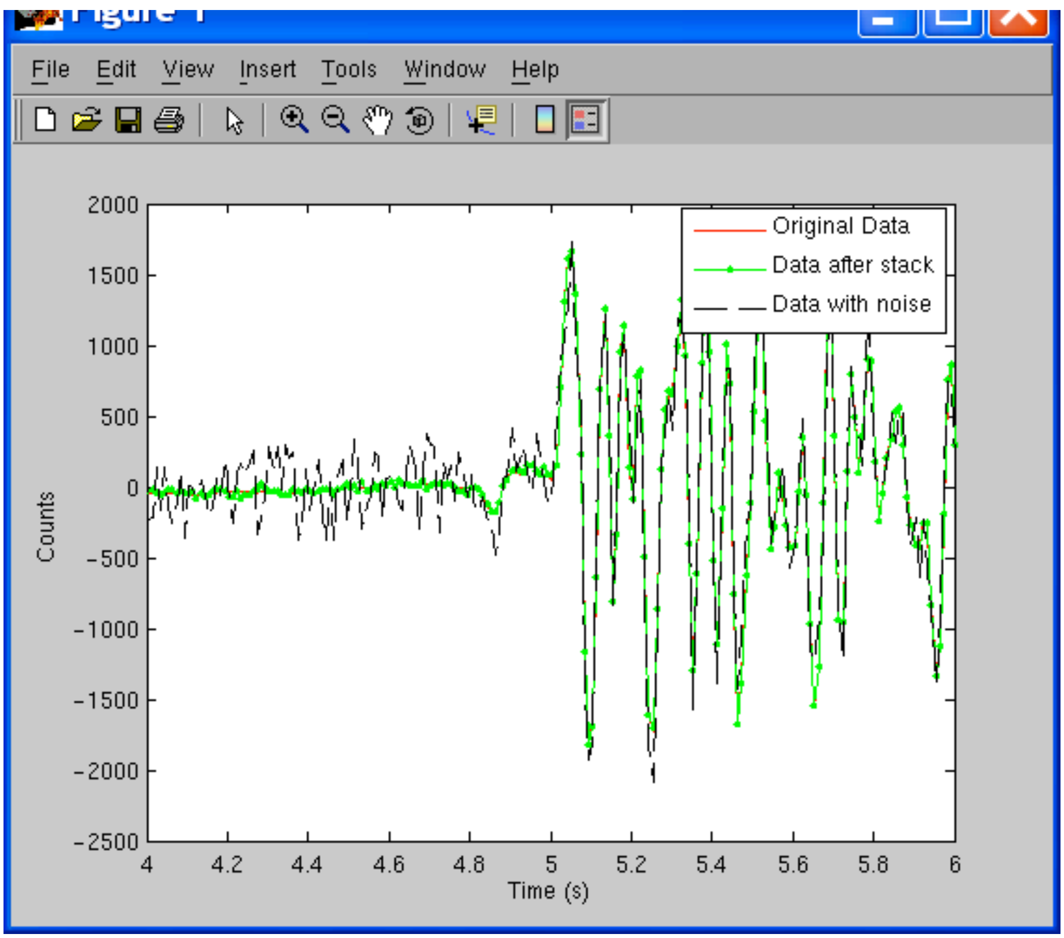
Event GW150914, stn Livingston



Matched filter Results (see next page)



2. Write your own code to add random noise to the following time series. Next, generate 100 new time series with random noises (e.g., 20% of the maximum amplitude of the time series). Then stack them together. Plot the 100 new time series together with the original and stacked trace. Comment on the improvement of signal to noise ratio before and after stacking around *P* wave (5 s) and head waves (4.8 s) (15 point).



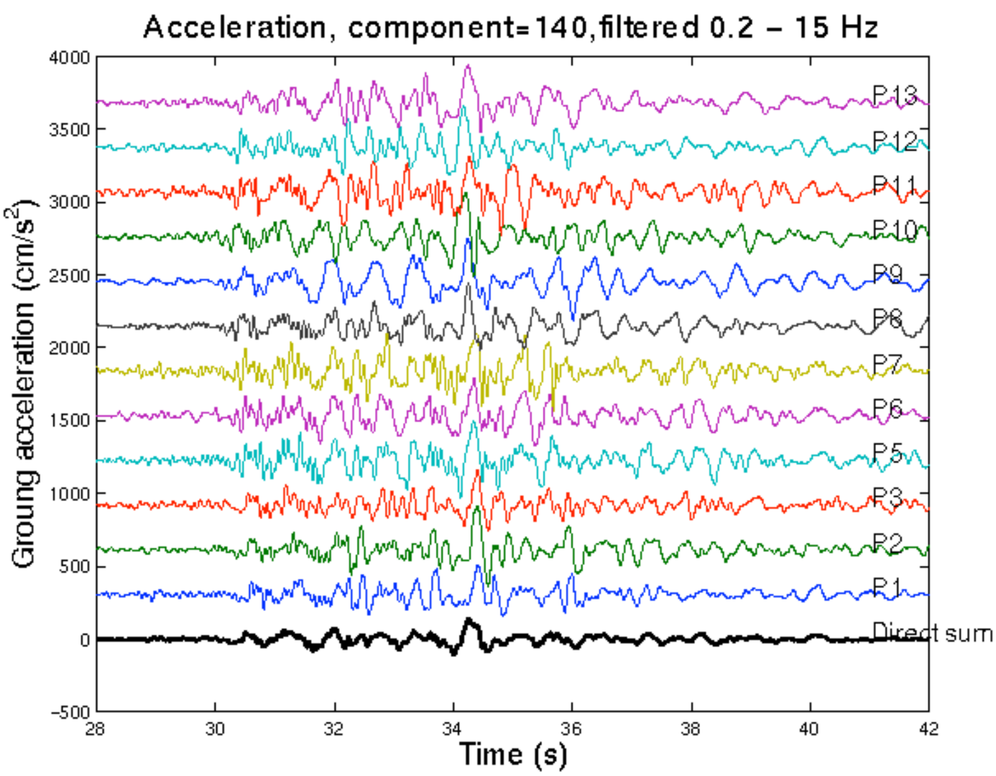
See attached code hw_32.m

3. Please read the Fletcher et al. (BSSA, 2006) paper first (included in the tar file below). Then follow the steps described in "Data and Method" section, and write your own array analysis code to perform the following analysis (60 points).

   0). Please download the tar file from the following website: http://geophysics.eas.gatech.edu/people/zpeng/Teaching/ObsSeis_2019/misc/UPSAR_PKD_ObsSeis_HW3.tar.gz

   The tar file contains the Fletcher et al. (BSSA, 2006) paper, the three component acceleration data in SAC format, station geometry, and site delays. See the readme.txt for details.

   1). Preprocessing the data: Take the two horizontal-component data for each station, cut the data between 28 and 42 s, rotate the data to fault parallel and normal component (the strike of the San Andreas fault is 140 deg from north), and apply a 0.2 to 15 Hz band-pass filter to the fault parallel data. Plot the fault-parallel component for each station together with the direct sum of the each component (10 points).



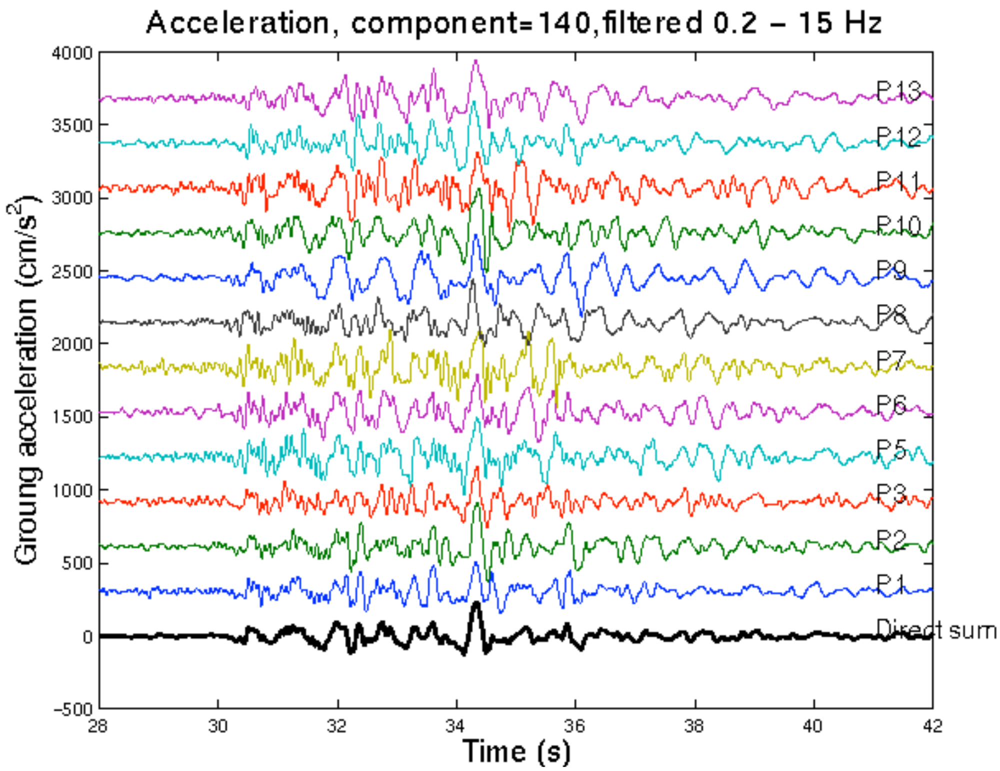
See attached code hw_331.m

   2). According to the Table 2 of the Fletcher et al. (BSSA, 2006) paper, the best-fitting apparent velocity C_{app} and back-azimuth θ for the following time window [34.1 – 34.6] s (the time window corresponds to data at the center station P06) is: C_{app} = 2.75 km/s, θ =

225 deg. Please use these numbers to compute the slowness component in the east (S_E) and north (S_N) directions (5 points).

Ans: S_N = -0.2571 s/km; S_E = -0.2571 s/km.

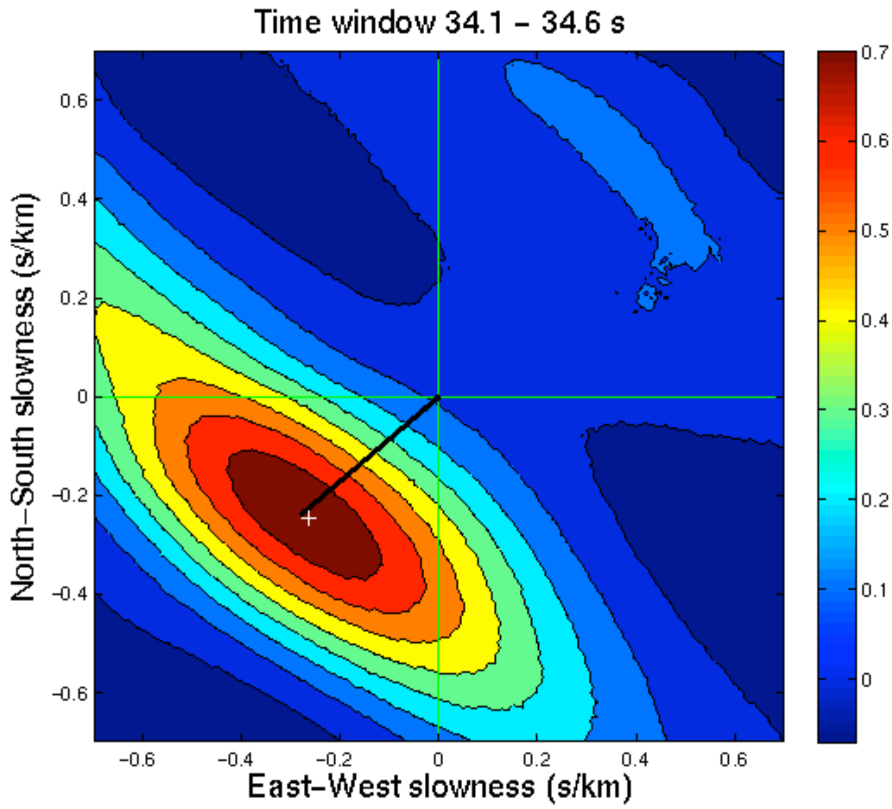
3). Using the obtained slowness component in 2), compute the time shift relative to the center of the array (station P06). Apply the time shift to the fault-parallel component data, and stack them together. You need to correct for the effect of elevation difference using $S_Z = \sqrt{1/c^2 - S_E^2 - S_N^2}$ as given in the paper. You should also correct for site delays using the table UPSAR_site_delays.dat. If a delay in (s) is positive, it means that the *S* wave arrives later than it should, because (for example) the shear wave speed under the station is lower than average. Plot the results and compare with those from step 1). Comment on the improvement of signal around the time window of [34.1 – 34.6] s (15 points).



See attached code hw_331.m

4). Perform a grid-search of slowness component in the east (S_E) and north (S_N) directions in the range of [–0.7 – 0.7] s/km, with a step of 0.01 s/km. Compute the cross-correlation coefficient (CCC) using equation 1 in the paper after correcting for the time lag, and average the CCC values for all possible pairs corresponding to each set of slowness. Plot the average CCC values as a function of slowness for time window [34.1 – 34.6] s. If everything works out well, your results should be similar to those shown in Figure 3 (bottom right panel). Compute your best-fitting apparent velocity and back-azimuth with those given in 2) (40 points).

See attached code hw_334.m



Note:

1. Your code can be written in any scientific languages (e.g., Fortran, C, Matlab). Please make sure that the code can be compiled under standard Linux machine. Please submit your code electronically to zpeng@gatech.edu, and submit a write-up that includes all the figures and answers to all questions.

2. The MatSAC package can be downloaded from the following link:
   http://geophysics.eas.gatech.edu/people/zpeng/Teaching/SAC_Tutorial/MatSAC.tar.gz
   Use fget_sac to read the SAC binary format data into Matlab.
   Alternatively, you can use sacdump program to extract data from SAC into two column ASCII output.