**COMMENTS: (with permission, I (Kasra) highlight the important parts for next references)**

**\* Tarje:**

Hi Kasra, on a quick look, this seems excellent, but from my experience it’s better to have **a very easy and quick first run of the tasks**, e.g. on **ONE page** with **all the important calls and major points we wish to achieve**... and delegate images and details to an **appendix**. So, let’s keep all this in but reshuffle it to have a concise early section. Tarje

I would structure it as I said in the email... pasting it here again:

Tutorial:

1) 10min intro(what is AXISEM, ObsPy; what we want to do in the tutorial)

2) 10min big run (maybe drop this, depending on progress next week)(2seconds, seismograms and not wavefields, 2 settings [different structures, sharpness of boundaries])

3) 60min virtual box: data and AxiSEM

4) 20min big run post processing

5) installation of AxiSEM, obspy locally

(**TASK÷**

50/100 s run (20min)

1) change AxiSEM input parameters to run this scenario, submit job (50s/100s for slow folks)

2) check mesh, background model, source-receiver geometry (google earth)

3) post processing on 50s run: filter, sum, rotate, movie snapshots

5s run with data (40min)

4) Load data from one of the 3 events with Obspy, plot cross sections

5) plot data vs. synthetics (PREM)

6) plot AxiSEM (PREM) vs SPECFEM vs data (20s)

7) load AxiSEM (IASP91), plot data vs. both axisem models (20s)

8) filter at 10s, then 5s, plot as in 7)

9) stick to “better” fitting AXISEM model: change source parameters (source CMT), plot data vs. both source parameters at 5,10,20s

The virtual box should therefore contain:

**(DONE)** - all 5 events (metadata and data), each in one directory

**(PENDING)** - AxiSEM source code structure (mesher, solver, manual, testing)

**(DONE\*)** - all axisem runs replicating the data for 2-3 background models and source choices

**(PENDING)** - plots of seismograms, movie

**\*: for each event there are: prem\_aniso and iasp91 for 5 ,10,50,100 seconds**

**AXISEM code practicing:**

The main goal of this tutorial is to use AXISEM for real scenarios, compare the results with real seismograms and explore the effects of source parameters and backgound model on the waveforms. Please follow the following lines. In case that you need more information, refer to Appendix-4 in which a complete example with all the commands and results is presented.

Start from within the ~/Desktop/EVENTS/SCRIPTS directory:

1. Plot one of the events (listed in EVENTS directory, we choose EVENT-1 in this example):

**$ python staev\_plot.py ~/Desktop/EVENTS/EVENT-1**

\* For more information on the folder structure of your Virtual-box, refer to Appendix-2.

2. To get an overview on both real data and pre-simulated seismograms: (e.g. PREM\_ANISO for 5 seconds dominant period)

**$ python epi\_plot.py ~/Desktop/EVENTS/EVENT-1/AXISEM\_PRE\_SIMULATED/PREM\_ANISO\_5sec/**

3. Cut the waveforms for Pdiff and PKiKP seismic phases and plot the seismograms: (compared with real data)

**$ python epi\_plot.py ~/Desktop/EVENTS/EVENT-1/AXISEM\_PRE\_SIMULATED/PREM\_ANISO\_5sec/ Pdiff**

**$ python epi\_plot.py ~/Desktop/EVENTS/EVENT-1/AXISEM\_PRE\_SIMULATED/PREM\_ANISO\_5sec/ PKiKP**

4. Change the filter in *epi\_plot.py* script and compare the results with *SPECFEM3D*: (for changing the filter, open *epi\_plot.py* and change hfreq and lfreq in INPUT section [line 34].)

**$ python epi\_plot.py ~/Desktop/EVENTS/EVENT-1/AXISEM\_PRE\_SIMULATED/PREM\_ANISO\_5sec/ Pdiff specfem3d**

**$ python epi\_plot.py ~/Desktop/EVENTS/EVENT-1/AXISEM\_PRE\_SIMULATED/PREM\_ANISO\_5sec/ PKiKP specfem3d**

5. Compare the results of two different background models (PREM\_ANISO\_5sec with IASP91\_5sec) for Pdiff phase:

**$ python epi\_plot.py ~/Desktop/EVENTS/EVENT-1/AXISEM\_PRE\_SIMULATED/PREM\_ANISO\_5sec/ Pdiff ~/Desktop/EVENTS/EVENT-1/AXISEM\_PRE\_SIMULATED/IASP91\_5sec**

6. Change the filter, as explained in step 4, and repeat step 5.

7. Compare the seismograms calculated for two different source parameters (PREM\_ANISO\_5sec and PREM\_ANSIO\_5sec\_GCMT) for Pdiff phase: (For more information about the source parameters, refer to Appendix-1)

**$ python epi\_plot.py ~/Desktop/EVENTS/EVENT-1/AXISEM\_PRE\_SIMULATED/PREM\_ANISO\_5sec/ Pdiff ~/Desktop/EVENTS/EVENT-1/AXISEM\_PRE\_SIMULATED/PREM\_ANISO\_5sec\_GCMT**

8. Change the filter, as explained in step 4, and repeat step 7.

9. Find the time shift between the synthetics and real data, shift the synthetics accordingly and plot the results:

**$ python epi\_plot.py ~/Desktop/EVENTS/EVENT-1/AXISEM\_PRE\_SIMULATED/PREM\_ANISO\_5sec/ Pdiff shift\_synthetics**

10. Map the calculated time shifts in step 9 on the location of stations:

**$ python tt\_plot.py ~/Desktop/EVENTS/EVENT-1**

**APPENDIX-1: Events**

Three events are selected for this tutorial (Figure-A1) with the following source characteristics:

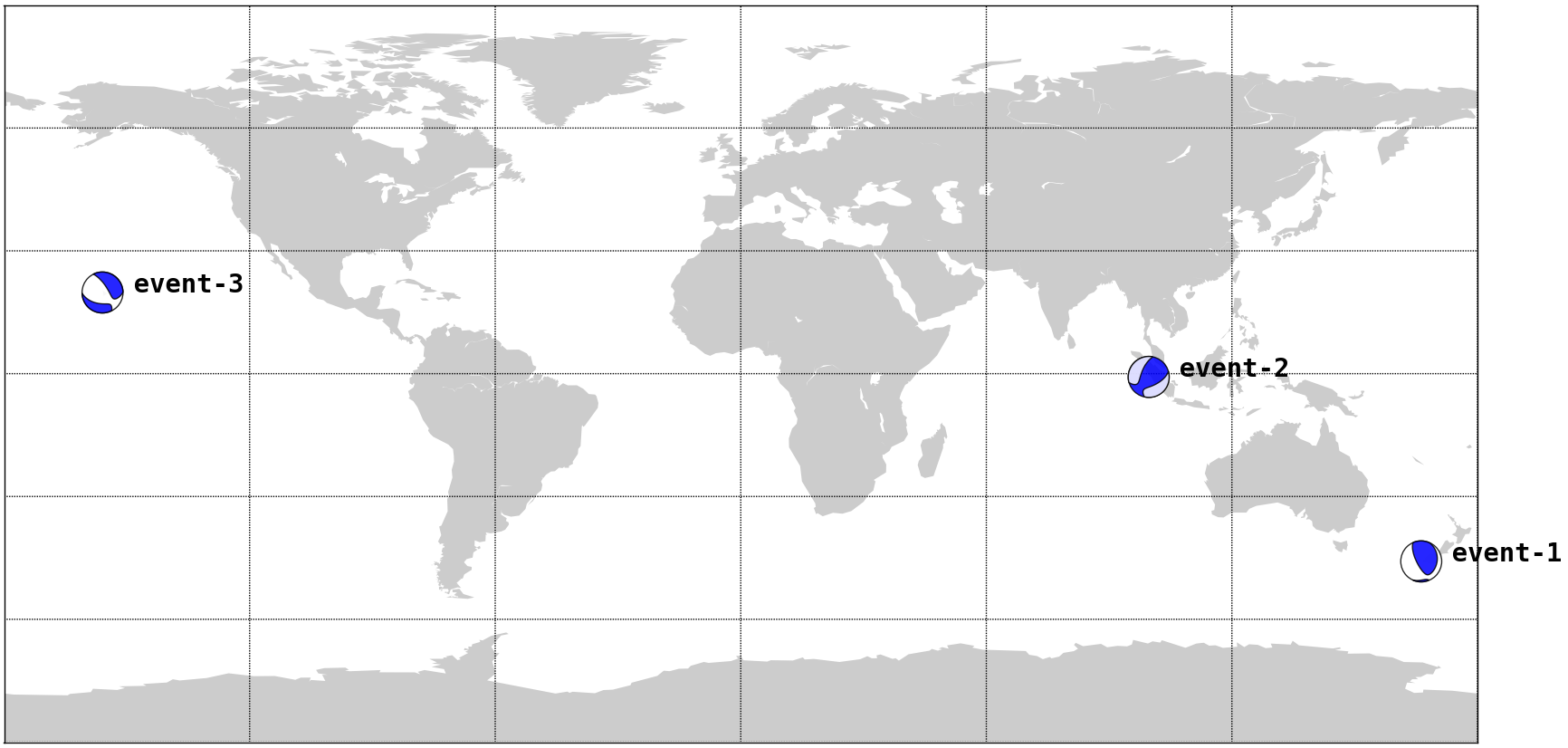
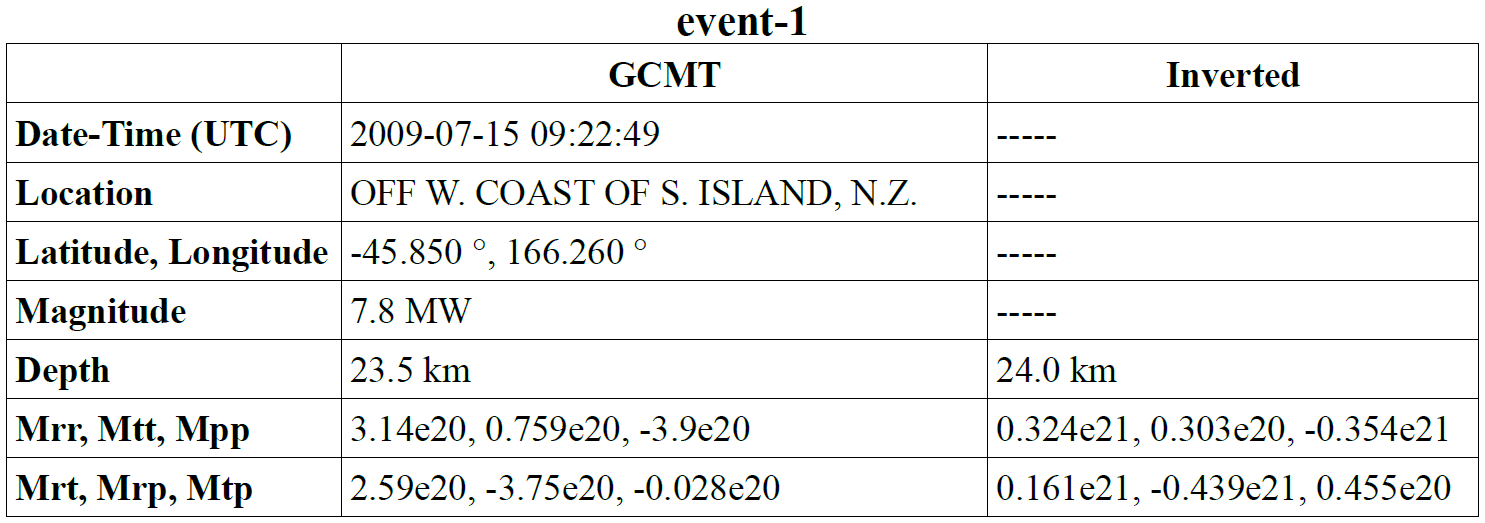
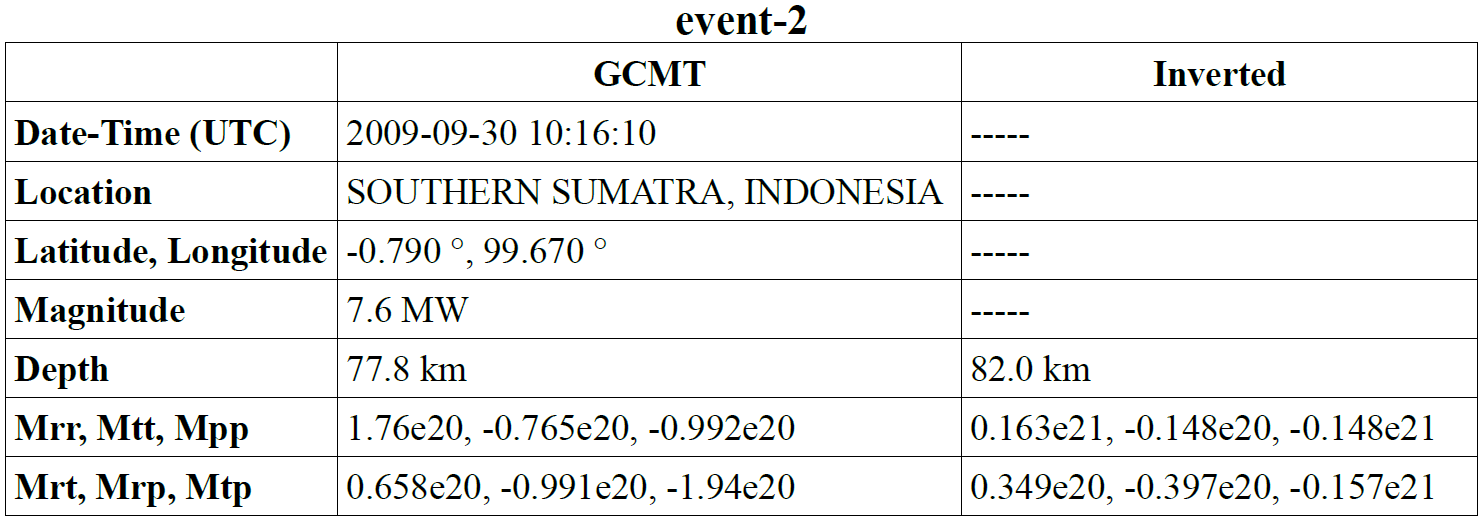
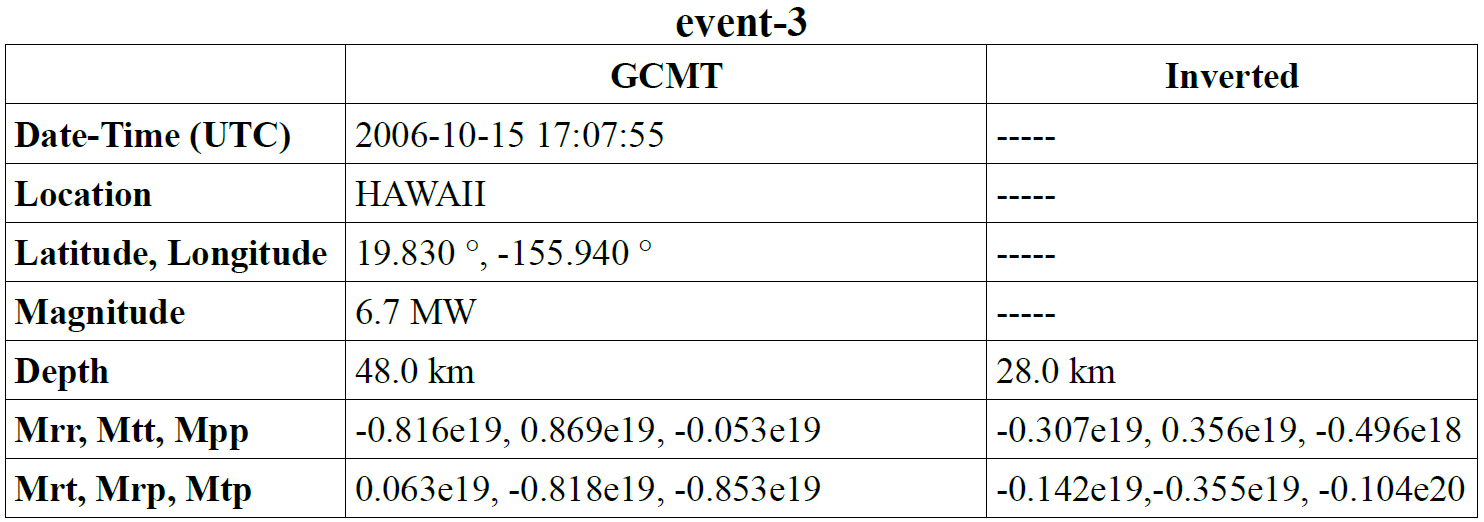


Figure A1: beach ball diagrams of event-1 to event-3 (based on GCMT catalog)







**APPENDIX-2: Folder structure**

Figure A2 shows how the events (and their meta-data), waveforms and scripts are organized in the Virtual-box:

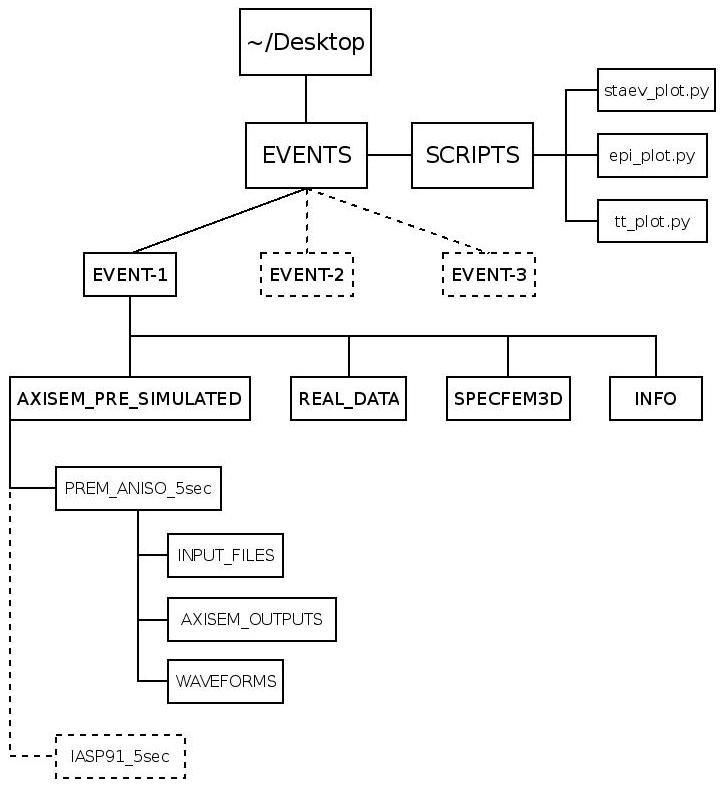


Figure A2: Folder structure

In *SCRIPTS* directory, there are three python scripts that we use here:

1. **staev\_plot.py**: maps event and stations of an event directory.

2. **epi\_plot.py**: plotting tool for comparison purposes.

3. **tt\_plot.py**: project the time shift derived by cross correlating the AXISEM waveforms and real data.

In *EVENTS* directory, there are three events, each with the following sub-directories:

1. **AXISEM\_PRE\_SIMULATED**: contains seismograms simulated by AXISEM with the required input files to re-produce them.

2. **REAL\_DATA**: seismograms retrieved from *IRIS.* (refer to APPENDIX-5)

3. **SPECFEM3D**: waveforms simulated by SPECFEM3D for comparison purposes (downloaded from *IRIS*, refer to APPENDIX-5)

4. **INFO**: information about the event and stations: event\_1.xml and STATIONS

**APPENDIX-3: A Quick Guide to PyAxi**

PyAxi is a Python script developed as an interface for AXISEM. All the options available in AXISEM are included in only one input file (inpython.cfg). By running the script, all the necessary steps (MESHER, SOLVER and Post-Processing) will be done automatically.

All you should do to run PyAxi for an input file (inpython.cfg) and station list (STATIONS) is:

**$ python PyAxi <inpython.cfg> <STATIONS>**

and the rest should be done automatically. inpython.cfg is a conguration file that contains all the AXISEM options. To change the input file, open inpython.cfg with an editor. However, you could find some already prepared inpython.cfg files for the events in Virtual-box. (refer to APPENDIX-2 for more information, Figure A2 *INPUT\_FILES*) Therefore, to run the AXISEM for the provided events (EVENT-1 and IASP91-5sec as an example), it is enough to replace:

**<inpython.cfg>: ~/Desktop/EVENTS/EVENT-1/AXISEM\_PRE\_SIMULATED/IASP91\_5sec/INPUT\_FILES/inpython.cfg**

**<STATIONS>:**

**~/Desktop/EVENTS/EVENT-1/AXISEM\_PRE\_SIMULATED/IASP91\_5sec/INPUT\_FILES/STATIONS**

**APPENDIX-4: Run AXISEM for EVENT-1**

In this appendix, we follow the steps in the main tutorial and show the commands and outcomes for each step. For this reason, we focus on one of the events, EVENT-1.

Start from within the EVENTS/SCRIPTS directory:

1. Plot one of the events (listed in EVENTS directory, EVENT-1 in this example):

**$ python staev\_plot.py ~/Desktop/EVENTS/EVENT-1**

\* For more information on the folder structure of your Virtual-Box, refer to Appendix-2.

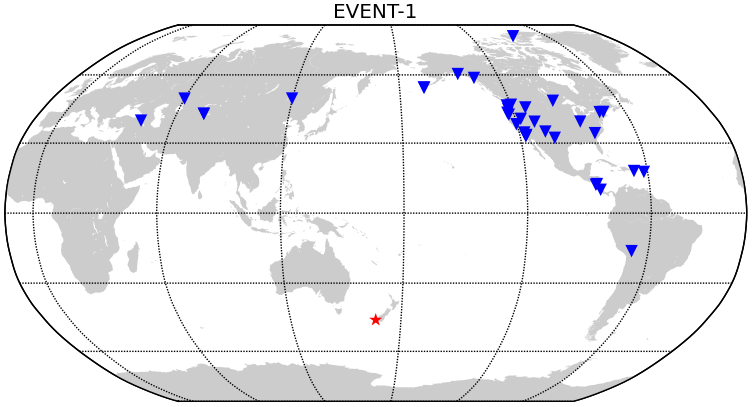


Figure A2: Event-station configuration for EVENT-1

2. To get an overview on both real data and pre-simulated seismograms: (e.g. PREM\_ANISO for 5 seconds dominant period) [Figure A3]

**$ python epi\_plot.py ~/Desktop/EVENTS/EVENT-1/AXISEM\_PRE\_SIMULATED/PREM\_ANISO\_5sec**

3.Cut the waveforms for Pdiff and PKiKP seismic phases and plot the seismograms: (compared with real data)

**$ python epi\_plot.py ~/Desktop/EVENTS/EVENT-1/AXISEM\_PRE\_SIMULATED/PREM\_ANISO\_5sec/ Pdiff**

**$ python epi\_plot.py ~/Desktop/EVENTS/EVENT-1/AXISEM\_PRE\_SIMULATED/PREM\_ANISO\_5sec/ PKiKP**

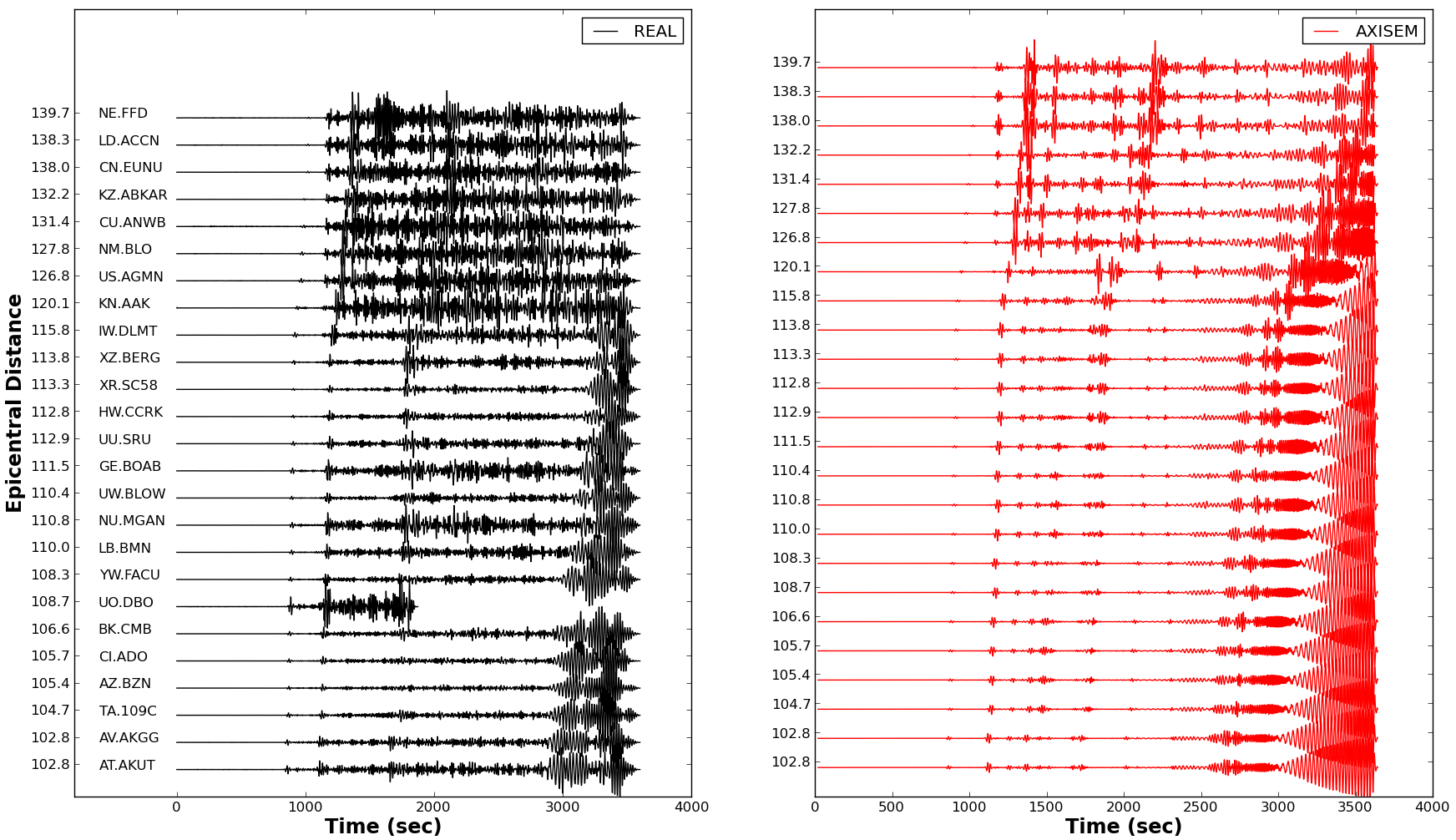


Figure A3:Real and AXISEM waveforms for EVENT-1

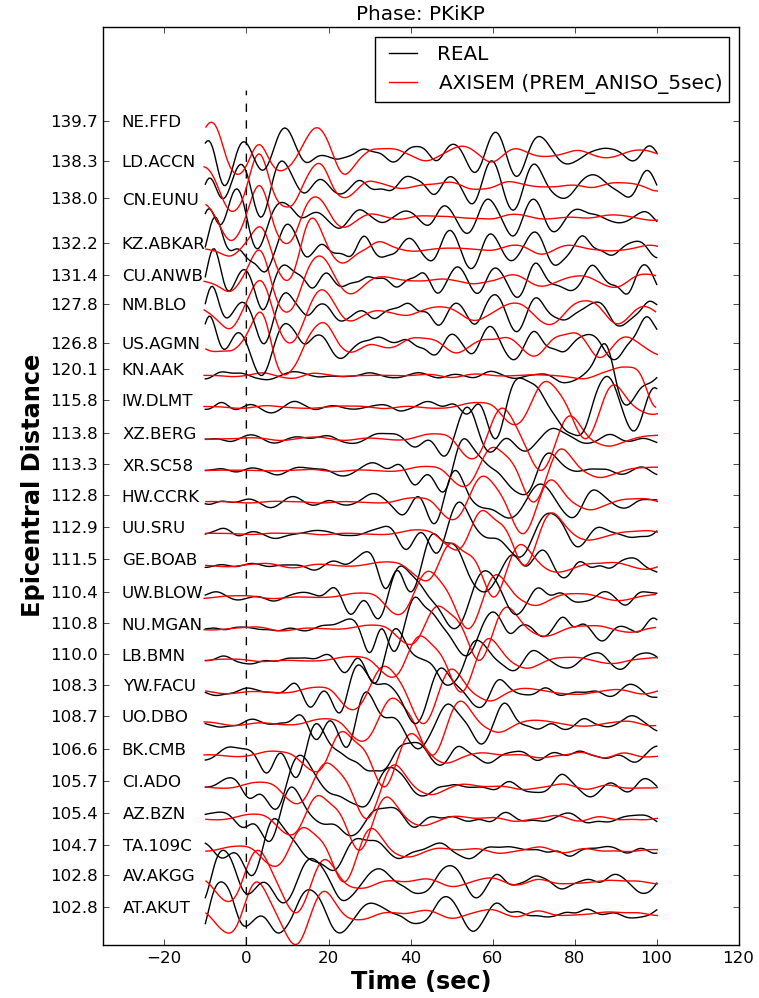
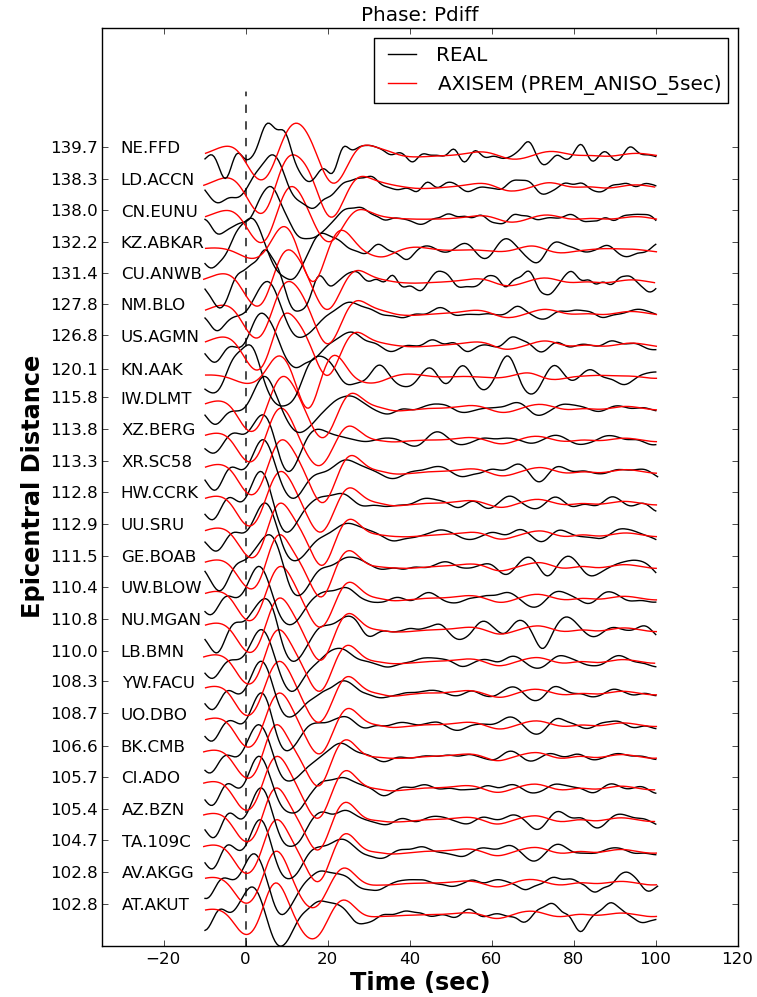


Figure A4: Comparison between real and synthetic waveforms for Pdiff and PKiKP phases.

4. Change the filter in *epi\_plot.py* script and compare the results with *SPECFEM3D*: (for changing the filter, open *epi\_plot.py* and change hfreq and lfreq in INPUT section [line 34]). For this example, we change hfreq (high frequency) in line 34 to 0.05Hz (20sec) and *lfreq* to 0.01Hz: (Figure A5)

**$ python epi\_plot.py ~/Desktop/EVENTS/EVENT-1/AXISEM\_PRE\_SIMULATED/PREM\_ANISO\_5sec/ Pdiff specfem3d**

**$ python epi\_plot.py ~/Desktop/EVENTS/EVENT-1/AXISEM\_PRE\_SIMULATED/PREM\_ANISO\_5sec/ PKiKP specfem3d**

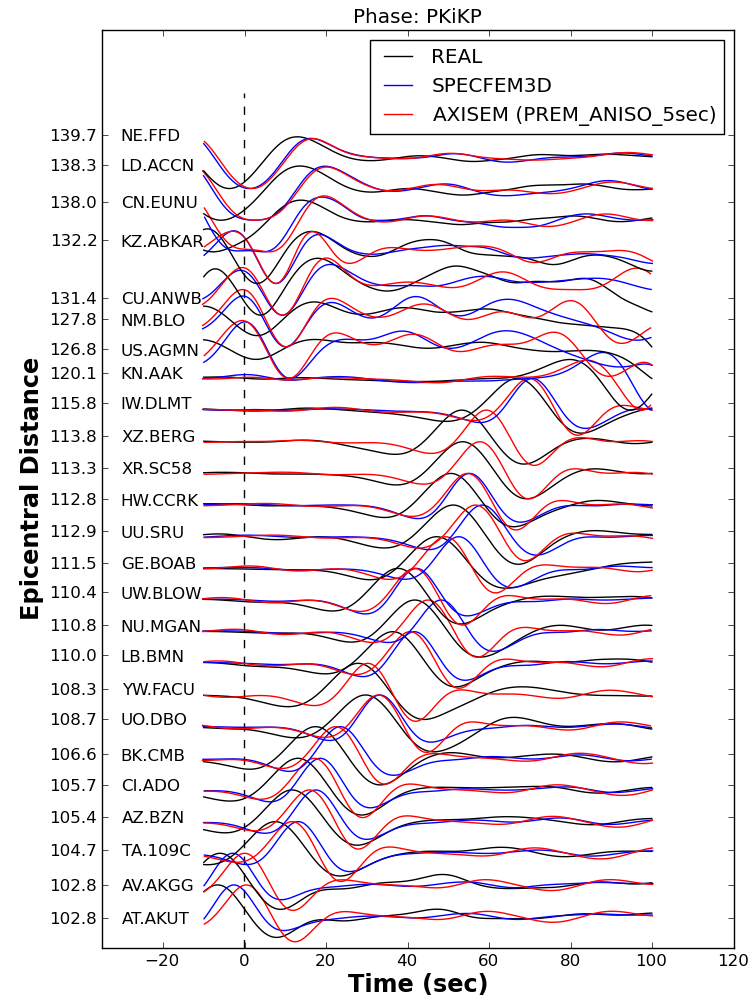
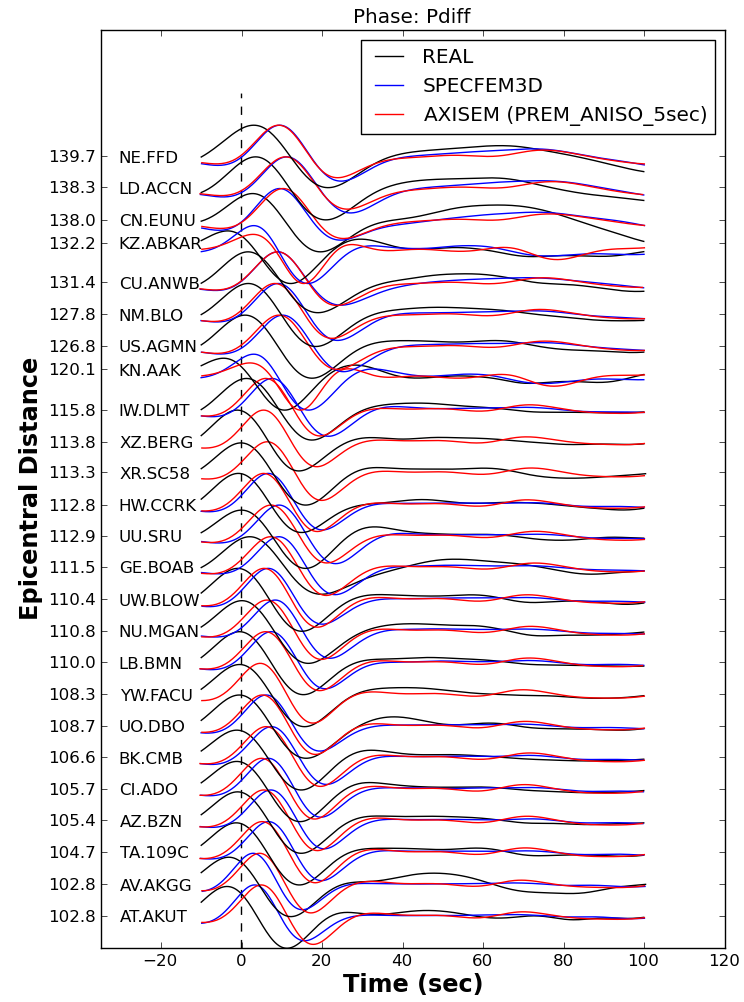


Figure A5: Comparison between real, AXISEM and SPECFEM3D waveforms for Pdiff and PKiKP phases.

5. Compare the results of two different background models (PREM\_ANISO\_5sec with IASP91\_5sec) for Pdiff phase: (Figure A6).

**$ python epi\_plot.py ~/Desktop/EVENTS/EVENT-1/AXISEM\_PRE\_SIMULATED/PREM\_ANISO\_5sec Pdiff ~/Desktop/EVENTS/EVENT-1/AXISEM\_PRE\_SIMULATED/IASP91\_5sec**

6. Change the filter, as explained in step 4, and repeat step 5. Here, we increase the *hfreq* to 0.2Hz and *lfreq* to 0.05Hz (Figure A7).

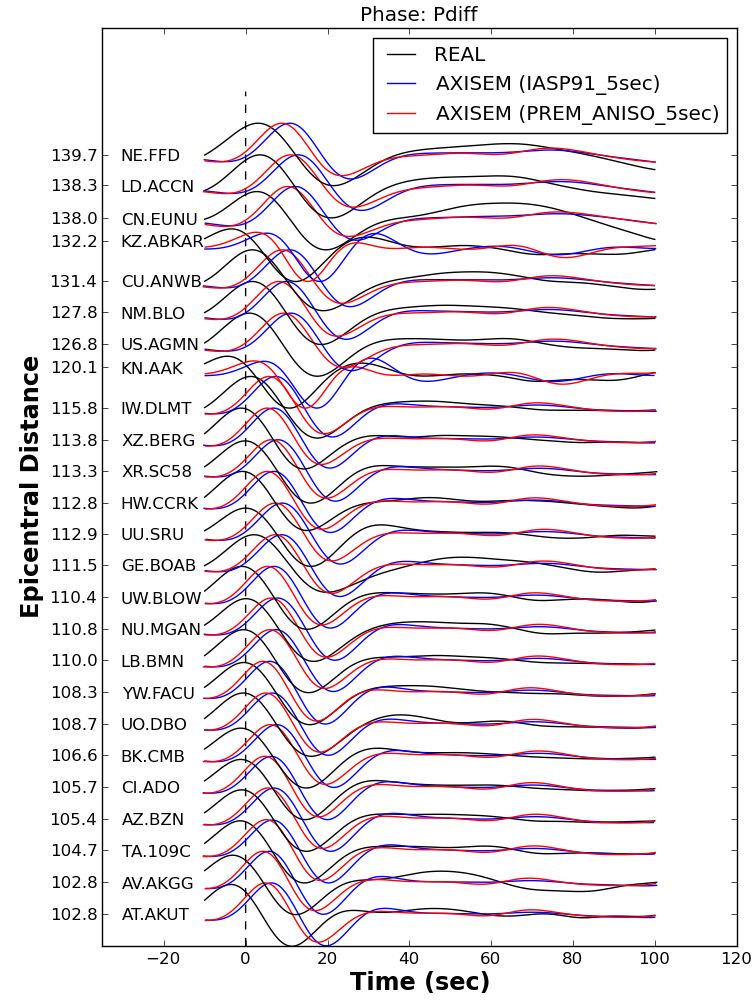


Figure A6: Comparison between real and AXISEM waveforms for two different background models (Pdiff).

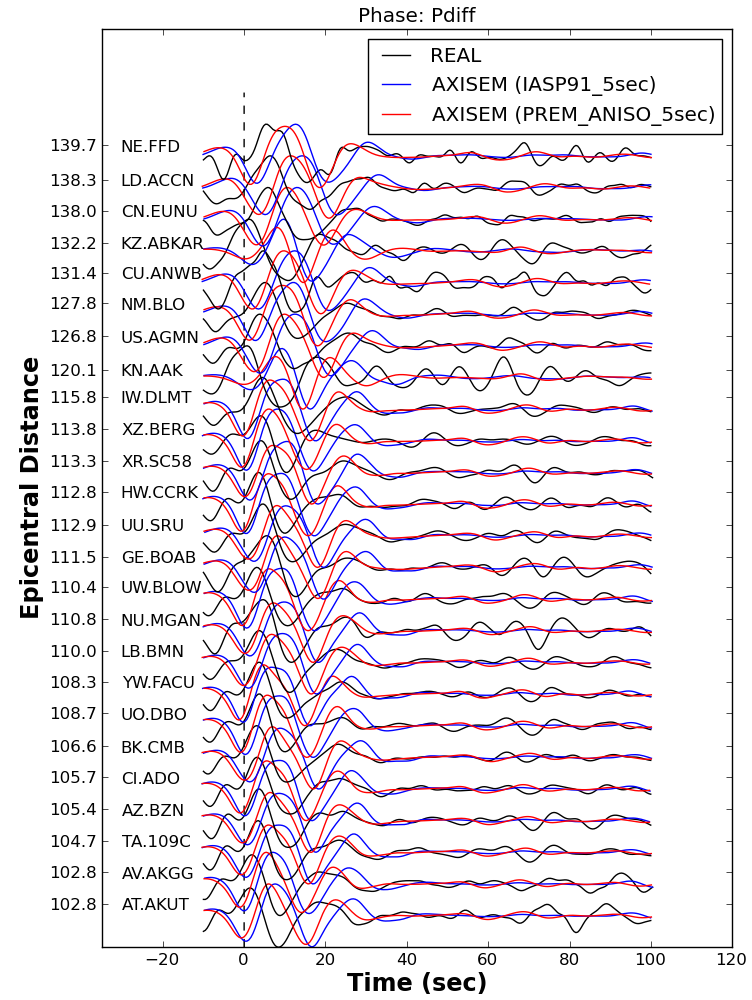


Figure A7: Comparison between real and AXISEM waveforms for two different background models (Pdiff).

7. Compare the seismograms calculated for two different source parameters (PREM\_ANISO\_5sec and PREM\_ANSIO\_5sec\_GCMT) for Pdiff phase: (For more information about the source parameters, refer to Appendix-1) [Figure A8]

**$ python epi\_plot.py ~/Desktop/EVENTS/EVENT-1/AXISEM\_PRE\_SIMULATED/PREM\_ANISO\_5sec/ Pdiff ~/Desktop/EVENTS/EVENT-1/AXISEM\_PRE\_SIMULATED/PREM\_ANISO\_5sec\_GCMT**

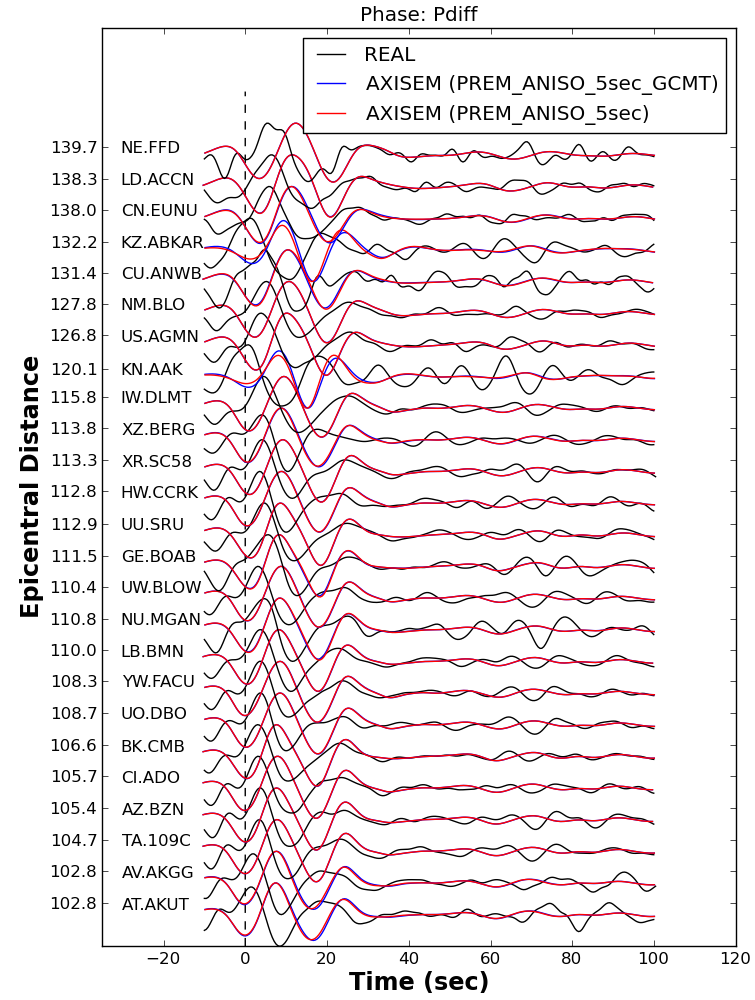


Figure A8: Comparison between real and AXISEM waveforms for two different source parameters (Pdiff).

8. Change the filter, as explained in step 4, and repeat step 7. Here, we decrease the *hfreq* to 0.05Hz and *lfreq* to 0.01Hz. (Figure A9)

9. Find the time shift between the synthetics and real data, shift the synthetics accordingly and plot the results: (Figure A10)

**$ python epi\_plot.py ~/Desktop/EVENTS/EVENT-1/AXISEM\_PRE\_SIMULATED/PREM\_ANISO\_5sec/ Pdiff shift\_synthetics**

10. Map the calculated time shifts in step 9 on the station locations: (note that it always plots the results of the latest comparison, i.e. Pdiff here)

**$ python tt\_plot.py ~/Desktop/EVENTS/EVENT-1**

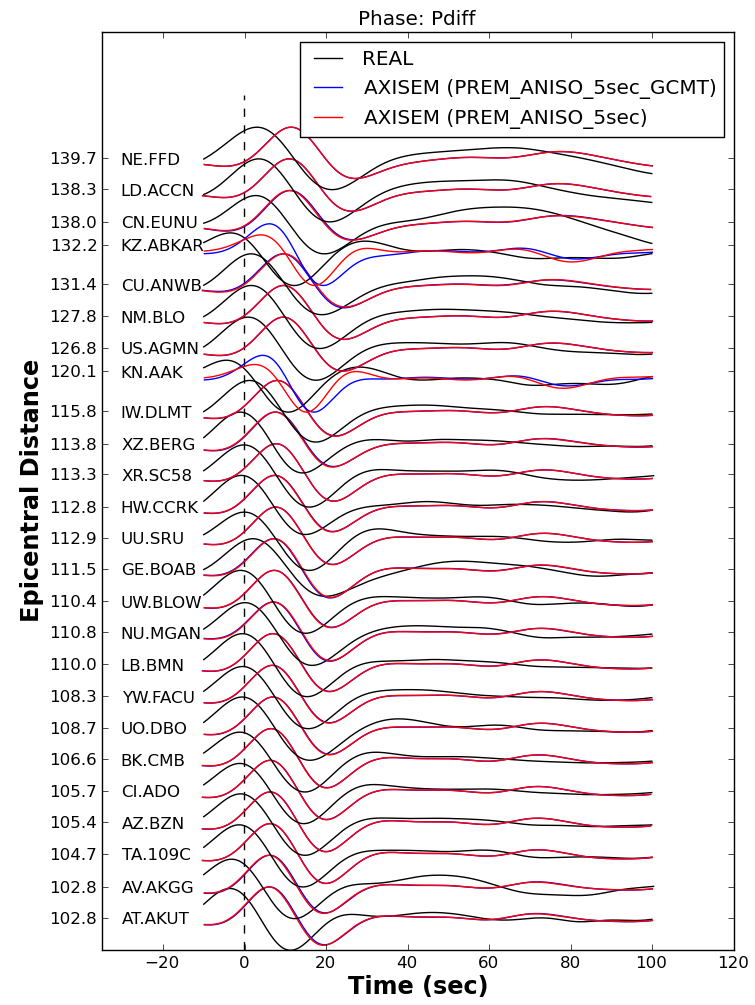


Figure A9: Comparison between real and AXISEM waveforms for two different source parameters (Pdiff).

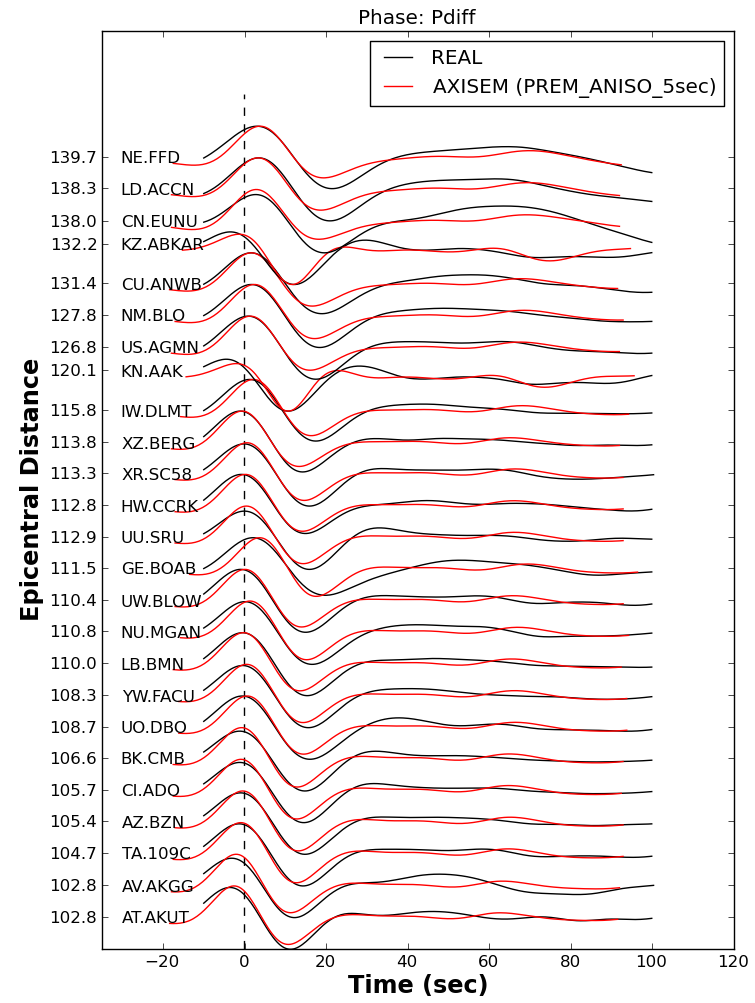


Figure A10: Comparison between real and AXISEM waveforms for Pdiff. AXISEM waveforms are shifted in order to gain the maximum cross correlation coefficient.

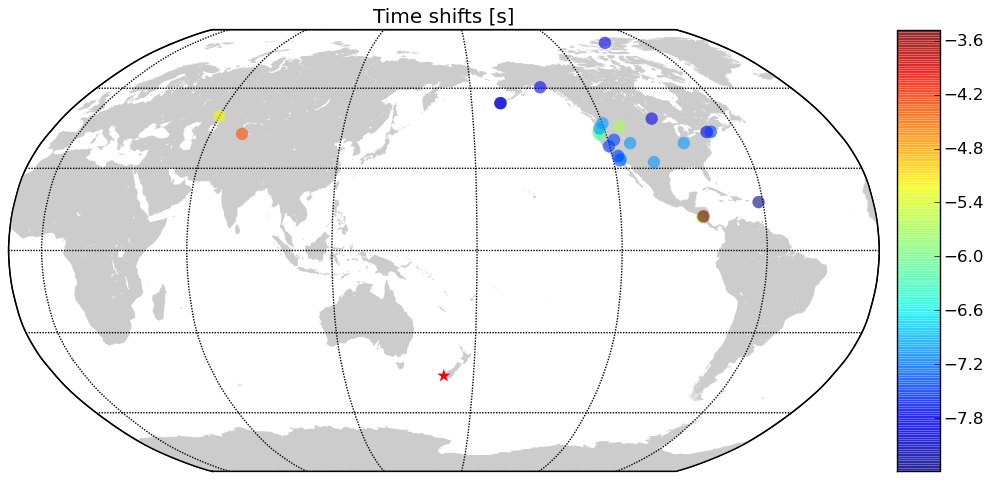


Figure A11: Time shift calculated by cross correlation between real and AXISEM waveforns. Blue color indicates that Pdiff in real data arrived sooner than that in the synthetic one.

**APPENDIX-5: Retrieving real data and SPECFEM3D seismograms automatically**

[obspyDMT](https://github.com/kasra-hosseini/obspyDMT) (ObsPy Data Management Tool) is a command line tool for retrieving, processing and management of massive seismological data in a fully automatic way which could be run in serial or in parallel.

This tool is developed to mainly address the following tasks automatically:

1. Retrieval of waveforms (MSEED or SAC), response files and metadata from [IRIS](http://www.iris.edu/ws/) and [ORFEUS](http://www.orfeus-eu.org/) (via [ArcLink](http://www.webdc.eu/arclink/)) archives. This could be done in *serial* or in *parallel* for single or large requests.
2. Supports event-based and continuous requests.
3. Extracting the information of all the events via user-defined options (time span, magnitude, depth and event location) from [IRIS](http://www.iris.edu/ws/) and [EMSC](http://www.emsc-csem.org/) (European Mediterranean Seismological Centre).
4. Updating the existing archives (waveforms, response files and metadata).
5. Processing the data in *serial* or in *parallel* (e.g. *Tapering, removing the trend of the time series, filtering and Instrument correction*).
6. Management of large seismic datasets.
7. Plotting tools (events and/or station locations, Ray coverage (event-station pair), epicentral-distance plots for all archived waveforms and seismicity maps).

Here, we use obspyDMT to retrieve both real data and SPECFEM3D seismograms. For more information about this tool please refer to the following webpage:

<https://github.com/kasra-hosseini/obspyDMT>

obspyDMT is installed on your virtual machine. By running the following commands, the real data used in this tutorial can be retrieved automatically:

**Event-1:**

./obspyDMT.py --datapath EVENT-1\_real --min\_date 2009-07-15 --max\_date 2009-07-16 --min\_depth 20 --list\_stas ~/Desktop/EVENTS/EVENT-1/INFO/STATIONS --offset 3600 --min\_mag 7.0 --req\_parallel --arc N

**Event-2:**

./obspyDMT.py --datapath EVENT-2\_real --min\_date 2009-09-30 --max\_date 2009-10-01 --min\_depth 70 --list\_stas ~/Desktop/EVENTS/EVENT-1/INFO/STATIONS --offset 3600 --min\_mag 7.0 --req\_parallel --arc N

**Event-3:**

./obspyDMT.py --datapath EVENT-3\_real --min\_date 2006-10-15 --max\_date 2006-10-16 --min\_depth 20 --list\_stas ~/Desktop/EVENTS/EVENT-1/INFO/STATIONS --offset 3600 --min\_mag 6.0 --req\_parallel --arc N

Moreover, the SPECFEM3D seismograms can be also retrieved in the same manner:

**Event-1:**

./obspyDMT.py --datapath EVENT-1 --min\_date 2009-07-15 --max\_date 2009-07-16 --min\_depth 20 --list\_stas ~/Desktop/EVENTS/EVENT-1/INFO/STATIONS --specfem3D --offset 3600 --min\_mag 7.0 --req\_parallel --arc N

**Event-2:**

./obspyDMT.py --datapath EVENT-2 --min\_date 2009-09-30 --max\_date 2009-10-01 --min\_depth 70 --list\_stas ~/Desktop/EVENTS/EVENT-1/INFO/STATIONS --specfem3D --offset 3600 --min\_mag 7.0 --req\_parallel --arc N

**Event-3:**

./obspyDMT.py --datapath EVENT-3 --min\_date 2006-10-15 --max\_date 2006-10-16 --min\_depth 20 --list\_stas ~/Desktop/EVENTS/EVENT-1/INFO/STATIONS --specfem3D --offset 3600 --min\_mag 6.0 --req\_parallel --arc N