# Lab exercise #2
# Moment tensor inversion

PHYS3070

II semester 2018 Week 1-6

Weeks 4, 5 and 6, August 2018

**Due Wednesday of week 7 (in the lecture)**

Thanh-Son Phạm thanhson.pham@anu.edu.au

Lauren Waszek lauren.waszek@anu.edu.au

Hrvoje Tkalčić hrvoje.tkalcic@anu.edu.au

**Time-Domain Moment Tensor INVerse Code (TDMT_INVC)**
The code and original comments by D. Dreger (2002)
Lab description by H. Tkalčić for PHYS3070 (2009)

## Introduction

This seismic moment tensor inverse software package has been developed by Prof. D. Dreger and it has been in use at the University of California, Berkeley Seismological Laboratory (BSL) since 1993 and is employed to automatically investigate all $M_L$>3.5 events in northern California (e.g. www.seismo.berkeley.edu/~dreger/mtindex.html). The package has been successfully implemented at the Japan National Research Institute for Earth Science and Disaster Prevention (NIED; argent.geo.bosai.go.jp), and has been used by individual researchers in the United States, Europe, Australia and Asia. This exercise includes a set of programs for calculating a library of structural Green's functions, inverting broadband data for the seismic moment tensor, various data processing utilities and shell scripts, and code and scripts demonstrating how to automate the procedures. The frequency-wavenumber integration program (FKRPROG) written by Chandan Saikia of URS is included with his permission. The software may be freely distributed only by means of its gzipped tar file. There is no warranty either expressed or implied, and the author is not responsible for damages due to misinterpretation of results and misuse of the software.

**Finally, this software is freely distributed for non-commercial use. The reporting of seismic moment tensor solutions obtained using this software on the WWW and in the technical literature should include the following acknowledgement statement. "Moment tensors were computed using the mtpackagev1.1 package developed by Douglas Dreger of the Berkeley Seismological Laboratory, and Green's functions were computed using the FKRPROG software developed by Chandan Saikia of URS."**

## Basic Methodology and Program Assumptions

The general representation of seismic sources is simplified by considering both a spatial and temporal point-source.

$$U_n(x,t) = M_{ij} \cdot G_{ni,j}(x,z,t)$$

$U_n$, is the observed n$^{th}$ component of displacement, $G_{ni,j}$ is the n$^{th}$ component Green's function for specific force-couple orientations, and $M_{ij}$ is the scalar seismic moment tensor, which describes the strength of the force-couples. The general force-couples for a deviatoric moment tensor may be represented by three fundamental-faults, namely a vertical strike-slip, a vertical dip-slip, and a 45$^0$ dip-slip. The indices i and $j$ refer to geographical directions. The above equation is solved using linear least squares for a given source depth. In this distribution only the deviatoric seismic moment tensor is solved for, and the inversion yields the $M_{ij}$ which is decomposed into the scalar seismic moment, a double-couple moment tensor and a compensated linear vector dipole moment tensor. The decomposition is represented as percent double-couple (Pdc) and percent CLVD (PCLVD). Percent isotropic (PISO) is always zero for this deviatoric application. The double-couple is further represented in terms of the strike, rake and dip of the two nodal planes. The basic methodology and the decomposition of the seismic moment tensor is described in Jost and Herrmann (1989).

Source depth is found iteratively by finding the solution that yields the largest variance reduction,

$$VR = \left[1 - \sum_i \sqrt{(data_i - synth_i)^2} \Big/ \sqrt{data_i^2}\right] * 100 ,$$

where *data*, and *synth* are the data and Green's function time series, respectively, and the summation is performed for all stations and components.

Another measure that is useful for determining source depth in regions where explosive events are unlikely is the RES/Pdc, the variance divided by the percent double-couple where,

$$RES / Pdc = \sum_i \sqrt{(data_i - synth_i)^2} \; / \; Pdc$$

Dividing the variance by the percent double-couple tends to deepen the minimum.

It is assumed that the event location is well represented by the high frequency hypocentral location, and a low frequency centroid location is not determined. Second, the simplified representation above assumes that the source time history is synchronous for all of the moment tensor elements and that it may be approximated by a delta function. These assumptions are generally reasonable for $M_W$<7.5 events since long period waves (>10-20s) are used. It is noted however, that for larger events these point-source assumptions break down in the period range employed and alternative finite fault approaches (e.g. Dreger and Kaverina, 2000) or longer period waves and larger source-station distances (e.g. Fukuyama and Dreger, 2000) are required.

Finally, it is assumed that the crustal model is sufficiently well known to explain low frequency wave propagation. This software package will not work in a region if calibrated velocity models are not available. Calibrating velocity models to obtain a robust catalog of Green's functions is singly the most important step in successful seismic moment tensor applications.

In California, it was found that three 1D velocity models are adequate for the recovery of the seismic moment tensor. Different monitoring regions may require fewer or more crustal velocity models. Crustal velocity models that are sufficient for moment tensor analysis may be derived from models used to locate earthquakes, or by modeling of the broadband seismograms. There are numerous papers in the literature that describe how to model 3-component waveforms to constrain velocity structure and these (e.g. Dreger and Helmberger, 1990, 1993; Dreger and Romanowicz, 1994; Rodgers et al., 1999; Zhao and Helmberger, 1991; Song et al., 1996) are some examples for getting started. Two- and three-dimensional models may be used provided that the codes used to synthesize the Green's functions produces the full compliment of fundamental-fault responses.

The usage of the software is demonstrated in the following example.

## EXAMPLE

The objective of the example is to become familiar with computing Green's function files, performing post-processing filter operations, and using the seismic moment tensor inversion code by inverting synthetic waveform data. For today's exercise, students working on Linux machines should remotely login to Sun Solaris machines: *mars*, *venus*, *earth* and *uluru*, in such a way that 2 users occupy each machine. The files necessary for the example are found in /Tdata/public/PHYS3070/TDMT/. The example is located in the subdirectory EXAMPLE whilst the problem is located in the subdirectory called PROBLEM. You should copy the entire TDMT directory with its subdirectories to your disk space and use it as your working directory. Starting at your home directory, execute:
1) mkdir Lab2
2) cd ./Lab2
3) cp –r /Tdata/public/PHYS3070/TDMT .
In this directory you will find the following files; tdmt.config, MODEL.socal, run_parallel, run_fkrsort, run_filter, run_inversion, b2s.par, mt_inv.in, and testdata[1-3].

Before doing anything else set the system specific environment parameters. You will need to edit the tdmt.config file to set the environment variable MTPACKAGE to the full directory path of the software package. For example, if you are student00, change current setting for student17 to student00. After making this change, use the source program to invoke the environment variables (i.e. execure: source tdmt.config). This command sets a number of environment variables such as paths to executables and Green's function files, which will be clarified later on.

### *Step 1: Compute Green's functions*

The file, MODEL.socal, contains a sample 1D velocity structure appropriate for the Sierra Nevada and Southern California regions and the source at 8 km depth. The contents of this file are listed below.

MODEL.socal:

```
.F.
    0   64
GREEN.X
    6.0      8.00       X  512 1024    0.500      5    Y
    1    1    1    1    1    1    1    1    1    1    0
 0.5500E+01 0.5500E+01 0.3180E+01 0.2400E+01   600.00    300.00
 0.2500E+01 0.6300E+01 0.3640E+01 0.2670E+01   600.00    300.00
 0.8000E+01 0.6300E+01 0.3640E+01 0.2670E+01   600.00    300.00
 0.1900E+02 0.6700E+01 0.3870E+01 0.2800E+01   600.00    300.00
```

```
 0.4000E+03 0.7800E+01 0.4500E+01 0.3300E+01   600.00    300.00
    3
  0.4000000E+03  1.500000E+00         0
    1  10000.0     30.0      2.9       2.5
100.00      0.0 10.0
```

The first line of MODEL.socal is a debugging flag. Keep it set to ".F.". The second line specifies the frequency range for debugging, do not change. GREEN.X is the name of the output file. The fourth line specifies alpha (small complex number for integration stability), source depth (km), the starting frequency indice (X), the total number of frequencies (a power of two), the total number of time points (2 times the number of frequencies), the sample rate (seconds), the total number of layers, and the number of processors the code will be run on (Y). The values of X and Y are set by the preprocessing script, run_parallel (usage is described below).

Line 5 is a series of flags that turn on the various fundamental fault calculations. Do not change this line. Lines 6-10 list the model parameters such as layer thickness (km), p-velocity (km/s), s-velocity (km/s), density (g/cc), Q-alpha, and Q-beta. Note that the source must be located at an artificial boundary where the velocities above and below are the same (layers 2 and 3, lines 7 and 8, in the example). Line 11 gives the layer number below the source. Line 12 does not need to be changed. Line 13 specifies the number of stations to calculate (1 in this case), and the phase velocity window of the integration. There is no need to change the phase velocity values 10000.0 and 30.0 (km/s), but you will want to be sure that the other two are less than the Rayleigh wave velocity of the model. The lines that follow define the distance to the station, a delay time (not currently implemented), and a reduction velocity. It is recommended that you keep the reduction velocity set to 10 km/s. Green's functions produced by the code with these settings will begin at t=distance/(reduction velocity). The file as it is currently set up will generate Green's function responses for stations located at a distance of 100 km.

The first task of this example is to run the frequency-integration program (FKRPROG) through the run_parallel script to compute the Green’s functions used to invert the synthetic waveform data. **Very important note**; FKRPROG requires formatted input. Be careful not to change the column location of each field. Integer input is right justified, and floating point input should retain the position of the decimal point.

To generate the Green’s functions first run the run_parallel script. This script performs all of the steps required to generate the GREEN.* output files. The usage is:

run_parallel number_of_cpus name_of_inputfile

For example using the command “run_parallel 1 MODEL.socal” will first generate a new input file, MODEL1, which has the X and Y parameters each set to 1. Second, FKRPROG is launched generating an output file, GREEN.1  If the number of CPUs is larger than 1 (maximum value of 4) then multiple input files, MODEL[1-4], are created and FKRPROG is executed multiple times generating the corresponding GREEN.[1-4] files.

***Note to PHYS3070 students (by SP):***
*In the PROBLEM section, you will need to modify the velocity model (i.e., MODEL.socal) for different source depths. It is recommended to create a new copy of the velocity model having the depth information in the filename before making any modification, for example,*
*> cp MODEL.socal MODEL_d8.socal*
*The command to generate Green's functions will be:*
*> run_parallel 1 MODEL_d8.socal*
*where 8 is the source depth in this case.*
*You can find other velocity models for other source depths in the EXAMPLE directory.*

**Step 2: Post-processing to create Green's functions in usable ascii format**
Once the FKRPROG process(es) have finished the next step is to create ascii format, time domain Green's function files that will be used by the inversion code. Another script, run_fkrsort, is used for this purpose. This script first calls the wvint9 program, which reads the GREEN.* file(s) and performs an inverse FFT and writes all of the displacement (cm) Green's functions into a headerless binary file, vec. Note that options for wvint9 are hard-coded in the script. It is possible to output velocity (cm/s) Green's functions instead of displacement (cm) by changing the 'd' to 'v' in the standard input statement for wvint9. Subsequent to the wvint9 post-processing the specific time series are then extracted from the vec file and an eight component ascii file is constructed. The ordering of the components are as follows, and the same as in Jost and Herrmann (1989); transverse component vertical strike-slip (tss) and vertical dip-slip (tds) faults, radial component vertical strike-slip (xss), vertical dip-slip (xds) and 45-degree dip-slip (xdd) faults, and vertical component vertical strike-slip (zss), vertical dip-slip (zds), and 45-degree (zdd) faults.

The following is the usage for run_fkrsort.

run_fkrsort filename_prefix distance depth number_of_distances

In this example use 'run_fkrsort socal 100 8 1'

Where 100 stands for the distance of 100 km, and 8 stand for the depth of 8 km. This script will output the files with the naming convention, {filename_prefix}{distance}d{depth}.disp. If the number_of_distances is not equal to 1 then the script assumes that the distance given is the closest distance, and the following distances are for intervals of 5 km. Note if you chose to modify run_fkrsort so that it outputs velocity (see above) you should also change the file extension for the Green's function files to .vel.

**Step 3: Filter Green's functions**
Once the *.disp files have been created it is necessary to perform bandpass filtering using the run_filter script. The usage of run_filter is:

run_filter  infile_name outfile_name  high_pass low_pass

The infile_name is the output  filename from the run_fkrsort script. The outfile_name can be anything, but typically the infile_name with the '.disp' truncated is used. The high_pass and low_pass filter corners are given in hertz. An acausal (two pass), $4^{th}$ order butterworth filter is applied using SAC. The SAC commands are imbedded in the run_filter script.

In this example the Green's functions need to be bandpass filtered between 0.02 to 0.10 Hz. To do this use 'run_filter socal100d8.disp socal100d8 0.02 0.10'

***Note to PHYS3070 students (by SP):***
*In the PROBLEM section, you will need to repeat this filtering command for different epicentral distances. To avoid the repetition, you can use an alternative filtering command 'run_filter_for_many_files' whose usage is*
*>./run_filter_for_many_files prefix_name depth low_pass high_pass*
*In this example, use*
*>./run_filter_for_many_files socal 8 0.02 0.10*

**Step 4: Run Moment Tensor Inversion**
Once the filtered Green's function files have been computed you can use them to invert the test data for example 1. You will find three data files, testdata[1-3], in the working directory. These data files are listed in the provided ***tdmt_invc*** input file called mt_inv.in.

mt_inv.in has the following format

| | |
|---|---|
| 3  8 1 1 | <number of 3-component stations, source depth, distance weighting flag, plotting flag> |
| testdata1 100. 10. 0 120 | <data_filename, distance (km), azimuth (deg from north), sample-offset (Zcor), number_of_samples> |
| testdata2 100. 40. 0 120 | <ditto for station 2> |
| testdata3 100. 50. 0 120 | <ditto for station 3> |
| socal100d8 0 120 | <filtered GF_filename, zero-offset (always zero), number_of_samples (same as corresponding data> |
| socal100d8 0 120 | <ditto for station 2> |
| socal100d8 0 120 | <ditto for station 3> |

To run the moment tensor code execute the command "run_inversion". Apart from standard output that you will see on the screen, this will create an output file called "plot.ps". When tdmt_invc is executed, the following information is written on the screen, and to a log file (mt_inv_redi.out).

```
Depth=8
Station Information
Station(0): testdata1  R=100.0km  AZI=10.0  W=1.000  Zcor=5
Station(1): testdata2  R=100.0km  AZI=40.0  W=1.000  Zcor=5
Station(2): testdata3  R=100.0km  AZI=50.0  W=1.000  Zcor=6
Mo=9.13262e+19
```

Mw=2.6
Strike=20 ; 275
Rake=39 ; 155
Dip=70; 54
Pdc=92
Pclvd=8
Piso=0
Station(0)=99.418358  2.93047e-11
Station(1)=99.254257  1.7449e-11
Station(2)=83.429100  1.33789e-11
VAR=2.34412e-15
VR=95.81  (UNWEIGHTED)
VR=95.81  (WEIGHTED)
Var/Pdc=2.553e-17
Quality=4

Most of the output information is self-explanatory. W is the applied inverse distance weight. Since in this case the distance to the three test stations is the same the weight is the same. If the distances differ the more distant stations are given a larger weight. Zcor is the sample offset that the code obtains from cross-correlating the data with the fundamental fault Green's functions (tss, tds, etc.). Zcor is a very important parameter that is used to align the data with the Green's functions prior to inverting the data. Because the cross-correlation is against fundamental fault Green's functions its value should be checked. Typically testing values that are ±3 samples from the value obtained automatically is sufficient for finding the optimal value, but sometimes when the data is noisy or the velocity model used to construct the Green's functions is very approximate a larger search may be necessary. The optimal value is evaluated using the variance reduction for each station, where Station(0)=99.418358 is the variance reduction for the first station. Try changing the value of the offset by editing the sample_offset field (Zcor) in the mt_inv.in file and rerunning the tdmt_invc program. Figure 1 shows the best result that is obtainable.

***Note to PHYS3070 students (by SP):***
*The iteration over testing values of the sample offset (zcor) to find the maximum Variance Reduction has been employed in the command 'run_inversion'. Thus, the output file 'mt_inv_redi.out' is slightly different from the one shown above. The moment tensor plot (mt_plot_d8.ps) is identical to Figure 1 and you do not need to perform "a manual shift of one sample" to obtain that result as being said in the figure caption.*

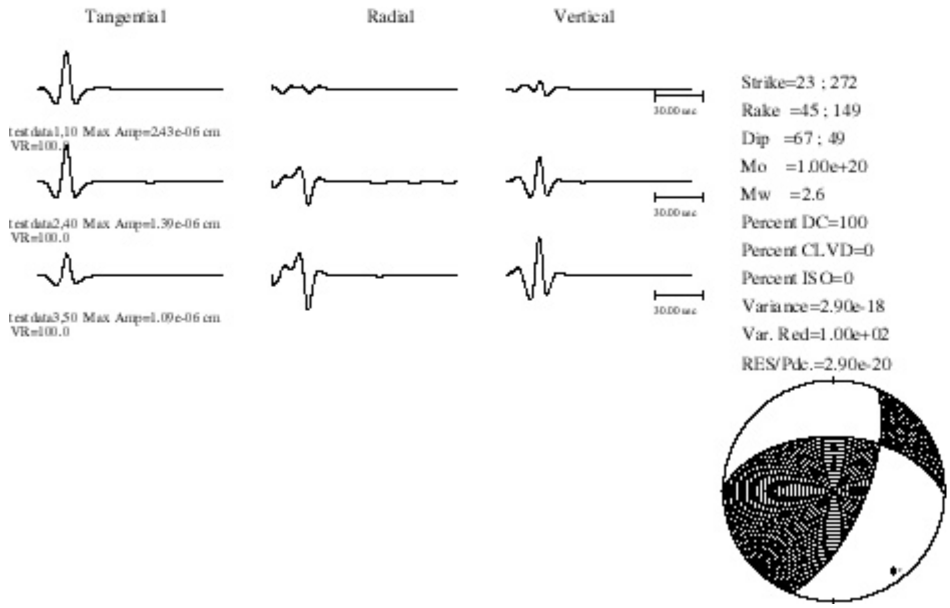


Figure 1. Example of ***tdmt_invc*** graphical output using Green's functions for a source depth of 8 km. The data are shown as solid lines, the synthetics are dashed. For each station three-component data and synthetics are compared, and the azimuth (10, 40, and 50-degrees), the maximum three-component trace amplitude, and variance reduction (VR) are provided. Solution information includes the strike, rake and dip for the two possible double-couple planes, the scalar seismic moment, and Mw. Information about the moment tensor decomposition in terms of percent double-couple (DC), CLVD, and isotropic (ISO) are also listed. Fitting parameters such as the variance, the variance reduction (Var. Red), and the variance modulated by the percent double-couple (RES/Pdc). To obtain this result it was necessary to manually shift the Green's functions for station 3 (testdata3) by one sample.

The output parameters VAR, VR, Var/Pdc and Quality are used to gauge the success of the inversion. VAR is the overall variance estimate, VR is the variance reduction (both unweighted and distance weighted estimates), Var/Pdc is the ratio of the variance to the percent double couple, quality is a subjective measure where 4 is the best and 1 is the worst. The higher the value of VR the better the solution. The Var/Pdc measure can also be very useful for areas where non-double-couple solutions are not expected.

To determine the source depth inversions are performed over a range of source depths taking the best solution as either the one with the greatest overall variance reduction or smallest Var/Pdc measure. Figure 2 illustrates the variance reduction plotted against source depth for this synthetic test example.

***Note to PHYS3070 students (by SP):***
*File 'VR_by_depth.dat' contains the maximum variance reduction as a function of source depths in this example. This data file is input for the command*

*>./plot_VR_by_depth*
*to generate a figure that looks similar to Figure 2. A similar data table is provided in the PROBLEM directory so that you can produce the figure for Task 1 in the PROBLEM.*

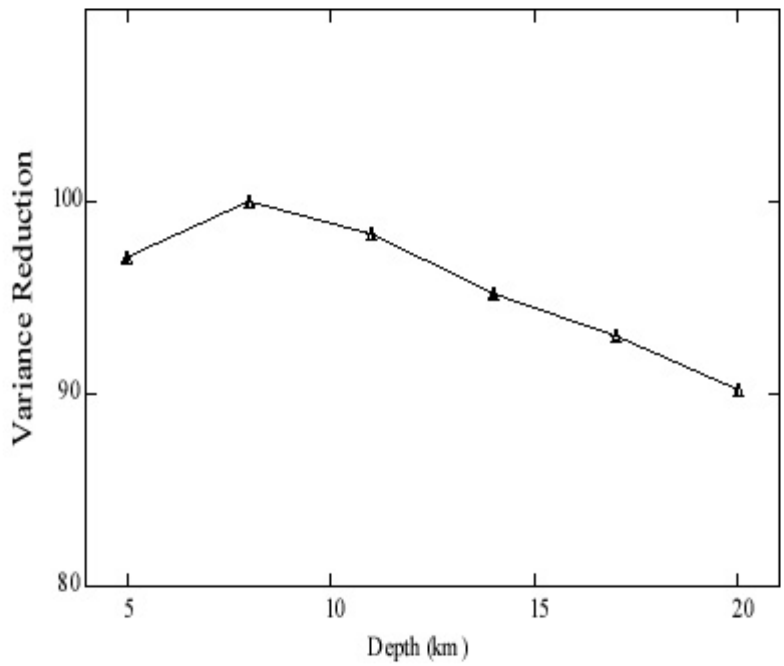


Figure 2. Variance reduction is plotted against source depth. For each inversion the Green's function alignment parameter has been optimized.

## PROBLEM

The objective of this exercise is to apply the data processing scripts to broadband data for a real earthquake, generate suitable Green's functions, and invert the data for the seismic moment tensor. This differs from the previous example in that now we fully prepare data files in a format suitable for the moment tensor inversion (this will be in the same format as testdata1, testdata2 and testdata3 from the example). In the subdirectory PROBLEM you will find the following files: tdmt.config, mt_inv.in, run_parallel, run_fkrsort, run_filter, run_inversion, MODEL.gil7, b2s.par. In addition, there is a subdirectory, DATA_SUN, which contains SAC binary files (raw waveforms) generated on SUN platforms. MODEL.gil7 is an input file for FKRPROG to compute Green's functions that are suitable for the Coast Ranges of central and northern California.

Before doing anything else, edit the tdmt.config file to set the environment variable MTPACKAGE to the full directory path of the software package, and then source the configuration file that is suitable for your operating environment.

To begin copy the binary data files for stations BKS, CMB, KCC and PKD from the appropriate data directory to your working directory. These stations will be used to illustrate the usage of the program. The following files should have been copied;

19980812141000.BKS.BHE 19980812141000.BKS.BHN 19980812141000.BKS.BHZ
19980812141000.CMB.BHE 19980812141000.CMB.BHN 19980812141000.CMB.BHZ
19980812141000.KCC.BHE 19980812141000.KCC.BHN 19980812141000.KCC.BHZ
19980812141000.PKD.BHE 19980812141000.PKD.BHN 19980812141000.PKD.BHZ
Use SAC to verify that the files are intact.

**Step 1: Instrument correct and filter waveform data and write data to ascii files**
The first step is to produce the ascii, three-component data files used by tdmt_invc. This step involves acquiring the pole-zero instrument response, using SAC to demean, deconvolve instrument response, integrate to displacement (cm), rotate to transverse and radial components, bandpass filter, resample to 1 sps, and finally write the ascii data files. All of this is done using a single script, tdmt_redi_prepdata, which is located in the MTPACKAGE/MTCODE/BIN directory. A detailed explanation of the processing script is not given, but users are encouraged to examine it to understand the SAC processing steps. Note that the instrument response database file, instr.resp, is read by the get_resp program which then writes the SAC instrument pole-zero files that are used to deconvolve the instrument response. The path to instr.resp is set using the REDI_MT_RESP environment variable in the tdmt.config configuration file. The program assumes that the pole-zero responses contained in instr.resp convert from digital units to velocity in m/s.

The usage of tdmt_redi_prepdata is shown below for BKS (the REDI_MT_BINDIR variable provides the path to the executable directory and is set by sourcing the configuration files). The execution will have to be repeated for each of the stations.

```
$REDI_MT_BINDIR/tdmt_redi_prepdata 19980812141000 BKS 1998 224 36.755 -121.464 0.02 0.05
```

The first command line argument is the filename prefix, which consists of the year (1998), month (08), day (12), hour (12), minute (14) and seconds (00) of the start of the record. This is followed by the station name (BKS), the year, the day of the year (224), the latitude (36.775) and longitude (-121.464) of the event, and the highpass (0.02) and lowpass (0.05) filter parameters. In this case we are using 0.02 to 0.05 Hz waves, which have been found to be effective in the regional distance range (e.g. Pasyanos et al., 1996; Fukuyama and Dreger, 2000). You can use any frequency passband you wish provided that both the data and Green's functions are processed using the same filter, and the velocity structure that is used to generate the Green's functions is a good representation of structure at the wave lengths that are being modeled. In practice we use a magnitude dependent frequency passband, where for M<4.0 the passband is 0.02 to 0.1 Hz, for 4.0 <= M < 5.0 the passband is 0.02 to 0.05 Hz, for M>= 5.0 the passband is 0.01 to 0.05 Hz. For very large events (M>7.5) a passband of 0.005 to 0.02 Hz is desirable (Fukuyama and Dreger, 2000).

After running tdmt_redi_prepdata for each station make a note of the azimuth and distance that is output to the screen. Alternatively, the program "dist_az" is available on our platforms to calculate the same parameters. You will need to log on www.berkeley.seismo.edu to obtain the BDSN stations coordinates (or extract this information from SAC headers, executing "lh" command in SAC). The values obtained for BKS are:

az = 3.314721e+02 <degrees from north>
dist = 1.420443e+02 <distance in km>

After running the script for the four stations you will have created the files, BKS_f0.05.data, CMB_f0.05.data, KCC_f0.05.data, PKD_f0.05.data.

**Step 2: Generate Green's functions for a suite of distances and source depths**
The next step is to generate a suite of Green's functions over a range of source depth for each source-station distance. The file MODEL.gil7 has been setup to generate Green's functions for 16 distances. Typically the distances are rounded to the nearest interval of 5 km. You can generate the gil7 synthetics using the following sequence of commands;

```
run_parallel 1 MODEL.gil7
run_fkrsort gil7_ 125 8 16
run_filter gil7_*d*.disp gil7_*d* 0.02 0.05
```

Note that you will need to explicitly enter the distance and depth in the spaces denoted by asterisks when filtering the Green's functions, and that nested foreach loops will be useful. This method for computing Green's functions may be used to compute a catalog of prefiltered Green's functions that may be used when automating these procedures as described in the next section.

Using the calculated Green's functions, the processed waveform data and the procedure for using tdmt_invc outlined in Example determine the best fitting moment tensor solution for this event. Figure 3 shows the automatic cross-correlation results using three stations (BKS, CMB and PKD), and Figure 4 shows the best result obtained when KCC is added and inversions were performed with Green's functions for a range of source depth.

***Note to PHYS3070 students (by SP):***
*Please see a note in the EXAMPLE section for the use of the 'run_filter_for_many_files' if you don't want to repeat 'run_filter' for multiple times.*

**Task 1**
By considering broadband data from 3 stations - BKS, CMB and PKD that we copied to the working directory, determine the best fitting moment tensor solution for this event. This means that you will have to investigate a range of depths (for this exercise, explore the following depths: 5, 8, 11, 14, 17 and 20 km). You should obtain the best result (the highest variance reduction) by varying Zcor parameter. One good way to do this would be to write a simple grid-search script that will loop over a range of Zcor values for each station (it should

be sufficient to consider ± 2 samples). *Show your best result* on a plot as in Figure 1. *Find the best fitting depth* and *show* the variance reduction as a function of depth (as in Figure 2). What can you say about the geometry of the fault that ruptured from the "beach ball" diagram that you obtained?

**Task 2**

Bring in the fourth station (KCC), and repeat the procedure. You can assume that the depth you obtained in Task 1 is the correct depth (so you do not need to investigate different depths any more). *Show your best result* on a plot as in Figure 1. Finally, *bring in* the fifth station of your choice. Consider the azimuth of each remaining station and select the station that has a good signal to noise ratio, and will at the same time fill in the azimuthal gap. Show your best solution for the case of including all 5 stations. What can you say about the final goodness of the fit? Does the increasing number of stations result in the increasing variance reduction? Explain why this might or might not be the case. (hint: think about the assumptions we made, starting from the consideration of the initial structural model).

**Task 3**

*Increase the crustal thickness* by changing the thickness of the sixth layer from 8 to 16 km. This means that we are now considering a perturbed earth model (which might be less accurate), which will result in different synthetic seismograms. By considering broadband data from only 3 stations - BKS, CMB and PKD (same as in Task 1), determine the best fitting moment tensor solution. *Show your best result* on a plot as in Figure 1. *Find the best fitting depth* and *show* the variance reduction as a function of depth (as in Figure 2). How do these results compare with those obtained in Task 1? What can you say about the variance reduction and the depth estimate? Are the waveform fits poorer or better now and why?

*Finally, if you are interested in applying the knowledge you acquired through this exercise to the Australian earthquakes, we can design together a research project at RSES, whose aim would be to study the Australian seismicity (although it doesn't have to be limited to the Australian earthquakes). You could benefit from expanding your knowledge on both forward and inverse modeling and learn how to use it in practice, improve your computer skills, etc.*

## References and Supplemental Reading

### Papers illustrating velocity model calibration:


Dreger, D. S., and D. V. Helmberger (1990). Broadband Modeling of Local Earthquakes, *Bull. Seism. Soc. Am.*, 80 1162-1179.

Dreger, D. S., and D. V. Helmberger (1993), Determination of Source Parameters at Regional Distances with Single Station or Sparse Network Data, *Journ. Geophys. Res.*, 98, 8107-8125.

Dreger, D. and B. Romanowicz (1994). Source Characteristics of Events in the San Francisco Bay Region, *USGS Open-file report*, 94-176, 301-309.

Saikia, C. K. (1994). Modified frequency-wavenumber algorithm for regional seismograms using Filon's quadrature; modeling of $L_g$ waves in eastern North America, *Geophys. Journ. Int.*, 118, 142-158.

Song, X., L. Zhao and D. V. Helmberger (1996). Broad-band modelling of regional seismograms; the Basin and Range crustal structure, *Geophys. Journ. Int.*, 125, 15-29.

Zhao, L., and D. V. Helmberger (1991). Broadband modelling along a regional shield path, *Geophys. Journ. Int.*, 105, 301-312.


### Papers describing various moment tensor applications:


Dreger, D., H. Tkalčić and M. Johnston (2000), Dilatational processes accompanying earthquakes in the Long Valley Caldera, *Science* 288, 122-125.

Fukuyama, E., and D. Dreger (2000). Performance test of an automated moment tensor determination system for the future "Tokai" earthquake, Earth Planets Space, 52, 383-392.

Gee, L. S., D. S. Neuhauser, D. S. Dreger, M. Pasyanos, R. A. Uhrhammer, and B. Romanowicz (1996), Real-Time Seismology at UC Berkeley: The Rapid Earthquake Data Integration Project, *Bull. Seism. Soc. Am.*, 86, 936-945.

Jost, M. L., R. Herrmann (1989). A student's guide to and review of moment tensors, *Seism. Res. Lett.*, 60, 37-57.

Pasyanos, M. E., D. S. Dreger, and B. Romanowicz (1996), Towards Real-Time Determination of Regional Moment Tensors, *Bull. Seism. Soc. Am.*, 86, 1255-1269.

Romanowicz, B. D. Dreger, M . Pasyanos, and R. Urhammer (1993). Monitoring of Strain Release in Central and Northern California Using Broadband Data, *Geophys. Res. Let.*, 20, 1643-1646.

Tkalčić H., D. S. Dreger, G. R. Foulger and B. R. Julian, The puzzle of the Bardarbunga, Iceland earthquake: No volumetric component in the source mechanism, Bull. Seismol. Soc. Am., 99: 3077-3085, 2009.