Introduction to SAC,
Seismic Ananlysis Code

Monday, Aug 15, 2011
Short course on USArray data processing for the next generation of seismologists III
Suzan van der Lee

Preparatory:
Go to a work directory/folder (YourWork, for example) and unpack the set of files for this exercise.

```
> tar xvf /Volumes/usarray/tmp/Monday1.tar
```

Make and go to a data subdirectory, and grab a(ny) SEED volume, for example from /Users/usarray/data/SEED. This example uses station last year's earthquake in Ottawa.

```
> mkdir data
> cd data
```

Unpack the waveform data from the SEED volume in this new "data" directory with "rdseed". The "d" indicates that you'd like to extract the data (i.e. waveforms) from the SEED volume. Here's what it *sort of* looks like in my terminal window:

```
comboimac28:data usarray$ rdseed
<< IRIS SEED Reader, Release 5.0
Input  File (/dev/nrst0) or 'Quit' to Exit: /Volumes/usarray/data/SEED/ottawa.seed
Output File (stdout)    :
Volume #  [(1)-N]       :
Options [acCsSpRtde]    : d
Summary file (None)     :
Station List (ALL)      :
Channel List (ALL)      :
Network List (ALL)      :
Loc Ids (ALL ["--" for spaces]) :
Output Format [(1=SAC), 2=AH, 3=CSS, 4=mini seed, 5=seed, 6=sac ascii, 7=SEGY] :
Output file names include endtime? [Y/(N)]
Output poles & zeroes ? [Y/(N)]
Check Reversal [(0=No), 1=Dip.Azimuth, 2=Gain, 3=Both]:
Select Data Type [(E=Everything), D=Data of Undetermined State, M=Merged data,
R=Raw waveform Data, Q=QC'd data] :
Start Time(s) YYYY,DDD,HH:MM:SS.FFFF :
End Time(s)   YYYY,DDD,HH:MM:SS.FFFF :
Sample Buffer Length [20000000]:
Extract Responses [Y/(N)]     :
Input  File (/dev/nrst0) or 'Quit' to Exit: Quit
comboimac28:data usarray$
```

Or alternatively one can:

```
comboimac28[]$ rdseed –d –f  /Volumes/usarray/data/SEED/ottawa.seed
```

Running SAC:
Let's visually inspect the data or a subset of the data (the * and ? are unix wildcards):

```
> sac
SAC> r *.TA.?3?A..BHZ*.SAC
SAC> p1
```

Oops, what happened? Try looking at fewer seismograms at once and let’s zoom in on the earthquake wiggles:

```
SAC> p1 perplot 6     (press enter to continue to the next set of 6)
```

Are the traces ordered in some logical manner? How did that happen?
Try zooming in on the data, e.g. the first or the strongest arrival, using

```
SAC> xlim 200 400
SAC> p1        (repeating this command will use the settings from the last time the
command was issued, in this case “perplot 6”)
```

Maybe the time limits are not exactly 200 and 400 s for each set of traces and it may vary from one group of 6 traces to the next? This is because the time axes of seismograms are not relative to the same reference time. Assessing the seismograms (e.g. for research quality) is easier if we know what earthquake they’re associated with and if their time is relative to the earthquake’s origin time. Quit SAC (use the “quit” command) and gather the hypocenter parameteres for the earthquake associated with your SEED volume. For the Ottawa data the parameters are in file “event.list”. Edit its contents to reflect the parameters for your earthquake. In this file, accurate values are essential only for parameters lat, lon, depth, year, julian day, hr, mn, and sec. Then

```
> cp ../event.list .
>../codes/evinhdr *.SAC
```

Now try the reading (r), plotting (p1), and zooming (xlim) again now that the seismograms are all referenced to the same time. Do you see a difference? You know you’re a real seismologist if you enjoy the view when keeping the space bar pressed during “p1”.

Now choose any station, let’s call it STAT, with a decent looking three-component seismogram. Make copies of the three BH components and the LHZ component and move them one directory up.

```
> cp 2010.174.17.40.00.0000.TA.STAT..BHZ.M.SAC ../stat.bhz
> cp 2010.174.17.40.00.0000.TA.STAT..BHN.M.SAC ../stat.bhn
> cp 2010.174.17.40.00.0000.TA.STAT..BHE.M.SAC ../stat.bhe
> cp 2010.174.17.40.00.0000.TA.STAT..LHZ.M.SAC ../stat.lhz
> cd ..
```

Let’s take the bhz component (with 40 samples per second) and decimate it to 1 sample per second, the equivalent of the lhz component. The SAC command is “decimate” ☺ or “dec” in short. Decimation cannot be done with factor greater than 7 s we need do it three times to get a composite factor of 40.

```
> sac
SAC> r stat.bhz
SAC> dec 5 f off      (f is short for filter)
SAC> dec 4 f off
```

```
SAC> dec 2 f off
SAC> w stat.blz
SAC> r stat.lhz stat.blz stat.bhz
SAC> p1
```

Do the traces look good? Did you expect they would have these differences? Is anything odd, weird, or ugly? Try again with the filter on:

```
> sac
SAC> r stat.bhz
SAC> dec 5 f on        (f is short for filter)
SAC> dec 4 f on
SAC> dec 2 f on
SAC> w stat.blfz
SAC> r stat.lhz stat.blfz stat.bhz
SAC> p1
```

Do these traces follow your expectations? The reason why stat.blz and stat.blfz look different is the same as why instrument responses contain FIR filters. These filters are anti-alias filters and they're important for signal quality.

We can also do nice things, such as plot maps and cross sections of the wave's polarities, called particle motion plots.

```
SAC> r  stat.bhz stat.bhn
SAC> xlim 300 360
SAC> ppm               (short for plot of particle motion)
SAC> r stat.bhn stat.bhe
SAC> xlim 600 680
SAC> ppm
```

What seismic phases have you looked at in these particle motion plots. Do the mapped polarizations agree with your expectations based on the event-station geometry and seismic phase arrival windows?

And try having SAC calculate the Fourier Transform of a time series:

```
SAC> r stat.bhz stat.blz stat.blfz
SAC> fft
SAC> psp am linlin
SAC> color on
SAC> p2
SAC> q
```

Now we're going to rotate the horizontal components from the E-N coordinate frame to the R-T coordinate frame (radial and transverse). Re-open SAC and read both horizontal components

```
>sac
SAC> r stat.bhn stat.bhe
SAC> p1
SAC> rot               (short for rotate)
SAC> p1
SAC> w stat.bhr stat.bht  (w is short for write)
```

What's different? Do you get the same results if you read the bhe component into SAC before the bhn component? Why? The default for the rotate command is to rotate to the great-circle path and to keep normal polarity. The default thus puts the radial component first in memory and the transverse component second.

Now alter one of the horizontal traces

```
SAC> cut 200 1000
SAC> r stat.bhe
SAC> w over
SAC> q                    (short for quit)
```

Now start over with rotating the horizontal component, but reserve the current bhr and bht files:

```
>sac
SAC> r stat.bhn stat.bhe
SAC> p1
SAC> rot
```

Nice. SAC reports what's wrong and why you can't rotate these components as they are. So we'll "fix" them before attempting to rotate them again. But doing that manually for every pair of files with this problem takes too much time!

SAC Macros:
So we write a "sac macro" that represents a series of sac command steps in a single sac macro command. The sac commands for the macro go into a new file, call it mrot:

```
> cat mrot
setbb north $1$n              (4 lines of variale settings, filenames)
setbb east $1$e               (the settings require the file base name as input)
setbb rad $1$r
setbb tran $1$t
r %east %north                (read the east and north component)
lh npts cmpaz cmpinc          (list critical values for quality control purposes)
if &1,b gt &2,b               (5 lines to find latest begin time of the two traces)
  setbb cut1 &1,b
else
  setbb cut1 &2,b
endif
if &1,e lt &2,e               (5 lines to find earliest end time of the two traces)
  setbb cut2 &1,e
else
  setbb cut2 &2,e
endif
setbb dt &1,delta             (copy sampling interval from first trace)
evaluate to ddt 1.2 * %dt     (3 lines to have the traces differ by not even one sample)
evaluate to c1 %cut1 + %ddt
evaluate to c2 %cut2 - %ddt
cut %c1 %c2                   (2 lines to cut both traces to same begin time and same npts)
r
rot                           (rotate to gcp)
lh npts cmpaz cmpinc          (list critical values for quality control purposes)
```

w %rad %tran (2 lines to write rotated results to file and quit SAC)
q

This macro needs as input the base file name for this stations and it expects that the file names of the horizontal components end with “n” and “e”. As output it produces two files with the same base name and ending in “r” and “t”, for radial and transverse. It rusn like so:

SAC> m mrot stat.bh (m is short of macro)

and done. Much easier than having to type the whole sequence above. Any series of sac commands can go into a macro.

Another example of a macro is one that is generated by a shell script or piece of code that takes advantage of the sac library (sacio.a). Shell scripting and coding will be discussed tomorrow. For now you can run the “getorder” script in order to gain some control over the order in which we view the data:

SAC> quit
> ../codes/getorder *SAC

This produces a SAC macro called olist.m. In this case, the list is ordered by epicentral distance. If you uncomment line 12 (and comment line 11, using the “#” symbol) you can order the data by azimuth. Of course there are many different ways in which one can order data. Run the macro in SAC:

> sac
SAC> m olist.m
SAC> p1 perplot <n>
SAC> xlim 200 1200
SAC> p1

or you can use SAC’s “ppk” with which you page to the next or previous page of seismograms with the “n” and “b” buttons:

SAC> ppk p 6

You can resize the screen, or zoom between the X marker (move mouse to desired time and press the “x” key) and the mouse marker (move mouse to desired time and click). Press the “o” key to go back to the standard view and the “q” key to quit from “ppk”.

Important references:

SAC Users Guide:
http://www.iris.edu/software/sac/manual.html

SAC Command Reference:
http://www.iris.edu/software/sac/commands/alpha_commands.html

Prof. Mike Brudzinski’s SAC tutorial:
http://mymiami.muohio.edu/webapps/bbcms/portfolio/viewPortfolio.jsp?pid=_1414_1&tid=_458_1

Prof. George Helffrich’s list of SAC bug fixes and more:
http://www1.gly.bris.ac.uk/~george/sac-bugs.html

SAC mailing list:
http://www.iris.washington.edu/mailman/listinfo/sac-help