SEGY to ASCII Conversion and Plotting Program

by Mark R. Goldman[1]

Open-File Report 99-126

1999

This report is preliminary and has not been reviewed for conformity with U.S. Geological Survey editorial standards or with the North American Stratigraphic Code. Any use of trade, firm, or product names is for descriptive purposes only and does not imply endorsement by the U.S. Government.

U.S. DEPARTMENT OF THE INTERIOR
U.S. GEOLOGICAL SURVEY

[1]Menlo Park, California

Note for the Monthly List

"SEGY to ASCII Conversion and Plotting Program"

Open-File Report 99-126.

This report documents a computer program to convert standard 4 byte, IBM floating point SEGY files to ASCII xyz format. The program then optionally plots the seismic data using the GMT plotting package.

There is a single 3 1/2" floppy diskette, IBM compatible. The diskette contains this NOTE file, a README file, source code, a makefile for easy compilation, and an ASCII version of the documentation. The diskette also contains example test files, including a typical SEGY file along with the resulting ASCII xyz and postscript files.

Requirements for compiling the source code into an executable are a C++ compiler, including a mathematical library containing a function to convert from IBM to IEEE floating point format. The program has been successfully compiled under Sun Microsystem's Sun OS 5.5, using the CC compiler, version 4.1, and the sunmath.h mathematical library. Use of other compilers may require modifications to the existing source code.

Requirements for plotting the seismic data is the existence of the GMT plotting package. The GMT plotting package may be downloaded from the web site: http://www.soest.hawaii.edu/gmt/

README FILE

Date of Latest Revision: February 12, 1999

U.S. Department of the Interior
U.S. Geological Survey

Open-File Report 99-126

SEGY to ASCII Conversion and Plotting Program

Mark Goldman
U.S. Geological Survey
345 Middlefield Rd.
Menlo Park, CA   94025

This report is preliminary and has not been reviewed for conformity
with U.S. Geological Survey editorial standards or with the North
American Stratigraphic Code. Any use of trade, firm, or product
names is for descriptive purposes only and does not imply endorsement
by the U.S. Government.

Although this program has been used by the U.S. Geological Survey,
no warranty, expressed or implied, is made by the USGS as to the
accuracy and functioning of the program and related program material,
nor shall the fact of distribution constitute any such warranty, and
no responsibility is assumed by the USGS in connection therewith.

This disk contains all of the source code and shell scripts
necessary to run segyPlot, along with a makefile and a README file.

This disk also contains an example SEGY file, xyz ASCII file,
parameter file, and resulting postscript file. As a test,
one can run the parameter file to read in the SEGY file and
recreate the xyz and postscript files.

This disk also contains a text version of the open-file
report, which contains instructions and examples on
using segyPlot.

To start, compile the C++ source code using the makefile
provided. Just type "make all", sit back, and
watch the makefile run. If all goes well, the executable
"segyPlot" should be produced. If not, you may need to
modify the source code so it will run on your computer.
I have successfully compiled the source code using the
Sun CC Compiler version 4.1, and the sunmath.h library
to convert from IBM to IEEE floating point format.
I have not tried to compile this code on any other platform.

Note that it is possible to run segyPlot in "non-interactive"
mode, which does not require compiling most of the code.
One can copy, modify, and run the "test.par" shell script file
as opposed to using the more interactive segyPlot program.
Most of the source code is used to compile the interactive portion
of segyPlot. The programs that do the actual work of converting
from SEGY to ASCII, "segy2xyz", and "segy2xyz-flip", are standalone
C++ programs that do not make use of the mPointer, mArray, ... classes.
However, these two programs do use the sunmath.h library, and must be
compiled regardless of whether the interactive or non-interactive mode
is used.

```
List of Files:

PROGRAM NAME            DESCRIPTION
------------            -----------
README                  this readme file
segyPlot.example        example run of segyPlot
segyPlot.help           text version of open-file report
segyPlot.template       template shell script, do not change!

makefile                makefile

segyPlot.cc             interactive segyPlot executable source code

SegyPlot.cc             Primary SegyPlot class
SegyPlot.h

SegyIo.cc               SegyPlot interface class
SegyIo.h

SegyList.cc             SegyPlot parameter list class
SegyList.h

mParamFile.cc           parameter file class
mParamFile.h

mTextFile.cc            text file class
mTextFile.h

mAsciiFile.cc           abstract ascii file class
mAsciiFile.h

mFile.cc                abstract file class
mFile.h

mParam.cc               parameter class
mParam.h

mUtil.cc                utility class
mUtil.h

mString.cc              string class
mString.h

mArray.cc               array class
mArray.h

mError.cc               error class
mError.h

mPointer.cc             auto-pointer class
mPointer.h

mManip.cc               manipulator class
mManip.h

segy2xyz.cc             source code for standalone C++ program
                        to convert segy data to ascii
segy2xyz-flip.cc        like above but flips data laterally

test.segy               example SEGY file
test.cdpx               example cdp to x coordinate conversion file
test.par                example parameter file used to create test.xyz and test.250.ps
test.xyz                example ASCII xyz file
test.250.ps             example postscript file
```

Table of Contents

Introduction

SEGY has long been a standard format for storing seismic data and header information. Almost every seismic processing package can read and write seismic data in SEGY format. In the data processing world, however, ASCII format is the "universal" standard format. Very few general purpose plotting or computation programs will accept data in SEGY format. The software presented in this report, referred to as SEGY to ASCII (SAC), converts seismic data written in standard SEGY format (Barry et al., 1975) to an ASCII data file, and then creates a postscript file of the seismic data using a general plotting package (GMT, Wessel and Smith, 1995). The resulting postscript file may be plotted by any standard postscript plotting program. SAC allows complete user control over all plotting parameters including label size and font, tick mark intervals, trace scaling, and the inclusion of a title and descriptive text.

Input Data Format

SAC reads standard 4 byte IBM®-floating-point SEGY files, inputting the number of samples, sample rate, channel, and common depth point (cdp) values directly from the file. SAC assumes that the input SEGY file starts with a 3600-byte reel header, with bytes 3217-3218 containing the sample rate in microseconds and bytes 3221-3222 containing the number of samples. Each trace should be preceded by a 240-byte trace ID header, with bytes 13-16 containing the channel number and bytes 21-24 containing the cdp for the trace.

Output Data Format

SAC converts an input SEGY file to an ASCII file listing the seismic data in xyz format, where x is the channel number, cdp, or the x coordinate of the cdp for this trace, y is the time or depth of the trace sample, and z is the amplitude of the trace sample at the given time or depth. A line containing only the '#' character in column 1 precedes the listing for each trace. After this line the trace samples are listed, one sample per line, with the x, y, and z values of the sample separated by spaces. Samples are listed in order of ascending time or depth. All values are right justified within specific column ranges. The x value is in columns 1-8, the y value in columns 10-17, and the z value in columns 19-26. Shown in example 1 is an example for 2 traces (cdps 1 and 2), each with 2 samples per trace (sample times of 0 and 5 ms).

Example 1:

```
#
1        0        -1.6961
1        5        -1.4135
#
2        0        1.6445
2        5        1.2741
```

The postscript file contains the seismic data plotted in standard wiggle-trace, shaded-area mode. A title line and two subsequent lines of text are displayed above the plot. The plot's vertical axis (y axis) is in units of time or depth. The plot's horizontal axis (x axis) is in units of channel, cdp, or the x coordinate of the cdp, if a cdp-to-x-coordinate conversion file is present in the working directory. The cdp-to-x-coordinate conversion file has a free-form, xy-format, where the first field is the cdp number, and the second field is the x coordinate. Header and comment lines are not necessary, but may be included if they

start with a '#' character in column 1. Example 2 shows sample lines from a typical cdp-to-x-coordinate conversion file.

Example 2:

```
# optional comment line
1       97.2345
12      100.1256
```

Plotting Parameters

A large number of parameters must be specified using SAC. Parameters are used to determine the x axis unit of the ASCII file, set the plot boundaries, the plotting scale, title and text messages, and the size and offset of various plot components. Some parameters, such as text_title offset, may not need alteration. Other parameters, such as plot boundaries and text messages, may be changed quite frequently. Parameters are saved in a parameter file, with one or more parameter files present for each SEGY file.

Software Package Contents

SAC contains a sample SEGY file (test.segy) and the resulting ASCII and postscript files (test.xyz and test.250.ps), along with a sample cdp-to-x-coordinate file (test.cdpx) and a sample parameter file (test.par), which is a shell script that sets plotting parameters and calls GMT plotting commands. SAC also contains C++ source code, a makefile to compile the source code, and a template shell script used by SAC to create parameter files. SAC does not contain the GMT plotting package but assumes it to be already installed. SAC is currently designed to run under Sun Microsystem's operating system SunOS® 5.5. C++ programs were compiled using the Sun Microsystem CC compiler, version 4.1, and the sunmath.h mathematical library was used to convert from

IBM to IEEE floating point format. SAC may need modification to run under other operating systems or to be compiled by a different C++ compiler.

Software Package Program Descriptions

SAC may be run either interactively or non-interactively. In non-interactive mode the user creates, edits, and runs a shell script called a parameter file (more specifically a parameter "object", containing both data and commands to convert and plot SEGY data). In interactive mode the user runs the segyPlot program, which will prompt the user for parameter values and then create and run a parameter file automatically. First time users will find the interactive mode to be more intuitive. Experienced users will find the non-interactive mode to be faster and more flexible, as several parameter files can be run sequentially in batch mode.

Data Naming Conventions And Procedure

All data filenames are based on a "linename", a user-supplied name describing the seismic line and specific processing steps. Example linenames are:

Parkfield_agc150
Parkfield_agc300
Parkfield_agc300_decon

Various suffixes are appended to the linename to denote the type of data. Specifically:

1) the SEGY file must have a ".segy" suffix.
2) the corresponding parameter file will usually have a ".par" suffix

3) the parameter file will convert the SEGY file to an ASCII file having a ".xyz" suffix.

4) the parameter file will then use GMT to read in the ASCII file and create a postscript file
having a ".y_end.ps" suffix, where y_end is the ending time or depth of the plot.

Example for a linename of "test":

1) initial SEGY file should be named test.segy

2) create the corresponding parameter file test.par

3) test.par will convert test.segy to an ASCII file named test.xyz

4) if plotting down to 500 milliseconds (or 500 meters) test.par will then create a postscript
file named test.500.ps

5) a second plot down to 300 milliseconds (or 300 meters) will create a postscript file
named test.300.ps

Interactive Mode

The interactive program is a C++ executable named **segyPlot**. The program first
requests the linename, and verifies that the corresponding SEGY file linename.segy exists.
The program will then prompt the user for the name of a parameter file to open or create.
The user may either use parameter values from an existing parameter file or use default
values for each parameter. The program will then list the parameters and their values,
assigning a unique number to each parameter, for example:

List of All Parameters:

1) scale   =   50

2) x_start   =   5.00

3) x_end   =   0.00

After the parameter listing a set of possible commands is given. The user chooses a command by entering the first letter of the command name, along with any optional arguments (optional arguments are in brackets). segyPlot will prompt the user for needed arguments, if they are not given on the command line. For example, typing:

c 1 100

at the command line will change the scale parameter from 50 to 100.

Commands (type first letter of command, along with optional arguments):

Linename	[new linename]
Open parameter file	[parameter file]
Save parameter file	[parameter file]
Display parameters	[all] or [major] or [first-last] all displays all parameters major displays just major parameters first-last displays the selected range of parameters
Change parameters	[first-last] OR [number [new Value]] first-last prompts for the selected range of parameters
Execute parameter file	[convert] OR [plot] saves parameters, converts and/or plots seismic data no argument will both convert and plot seismic data

Quit segyPlot

For the Execute command, the user has the option to convert the SEGY file to ASCII, create a postscript file from an existing ASCII file, or both. If no argument is given then segyPlot will both convert and plot the SEGY file. If the "convert" argument is given then segyPlot will only convert the SEGY file to ASCII without plotting. If the "plot" argument is given then segyPlot will use the existing ASCII file to create a postscript file. One should not use the "plot" argument if:

1)   the SEGY file has been changed or updated

2) the x axis unit has been changed (unit is channel number, cdp, or meters)

3) the length of seismic data to be converted has been increased.

For example, if segyPlot first converted and plotted a SEGY file to 500 ms, then do not use the "plot" argument if plotting the same SEGY file down to 600 ms. Instead use no argument, so that the ASCII file is recreated down to 600 ms. If unsure, it is safest to use no argument, so that the ASCII file will always be recreated before plotting. Using the "plot" argument can save time, however, if only minor plotting changes are needed.

Interactive Mode Example

The following is a typical exchange using segyPlot, with user responses underlined.

Welcome to segyPlot   version 1.0   January 1999

Written by Mark Goldman
Earthquake Hazards Team, USGS Menlo Park

enter line name (hit return to select linename later): <u>test</u>
setting linename to: test

enter name of parameter file to open (hit return to open file test.par): <u><return></u>
opening parameter file: test.par

major parameters:
```
 1) scale          =    10       # plot scale: smaller number means larger traces
 2) x_start        =    90.00    # x axis start, in cdp, chan, or meter units
 3) x_end          =   350.00    # x axis end, in cdp, chan, or meter units
 4) x_tick         =    10.00    # x axis tick mark interval
 5) x_anot         =    50.00    # x axis tick mark annotation interval
 6) x_unit         = meter       # x axis unit, must be "cdp", "chan", or "meter"
 7) convert_file   = test.cdpx   # if x axis unit is meter then use this conversion file
 8) x_label        = default     # x axis label - use "default" for default x axis label
 9) y_start        =   -50.00    # y axis start (usually set to datum)
10) datum          =   -50.00    # datum to add to trace's y value (usually <= zero)
11) y_end          =   250.00    # y axis end, in time or depth units
12) y_tick         =    10.00    # y axis tick mark interval
13) y_anot         =    50.00    # y axis tick mark annotation interval
14) y_label        = Depth (m)   # y axis label
15) label_size     =    14       # axis label size of seismic plot
16) anot_size      =    12       # annotation label size of seismic plot
```

17) title	= TITLE	# title text
18) text_1	= TEXT 1	# first line of text
19) text_2	= TEXT 2	# second line of text

line: test   parameter filename: test.par
display mode: major parameters   execute mode: both convert and plot

Commands:

Line	[new_linename]
Open	[parameter_filename]
Save	[parameter_filename]
Display	["all"] or ["major"] or [first-last]
Change	[first-last] or [number [value]]
Execute	["convert"] or ["plot"]
Quit	

Enter Command Letter Plus Optional Arguments: <u>change 1 20</u>

Enter Command Letter Plus Optional Arguments: display

major parameters:

1) scale	=	10	# plot scale: smaller number means larger traces
2) x_start	=	90.00	# x axis start, in cdp, chan, or meter units
3) x_end	=	350.00	# x axis end, in cdp, chan, or meter units
4) x_tick	=	10.00	# x axis tick mark interval
5) x_anot	=	50.00	# x axis tick mark annotation interval
6) x_unit	= meter		# x axis unit, must be "cdp", "chan", or "meter"
7) convert_file	= test.cdpx		# if x axis unit is meter then use this conversion file
8) x_label	= default		# x axis label - use "default" for default x axis label
9) y_start	=	-50.00	# y axis start (usually set to datum)
10) datum	=	-50.00	# datum to add to trace's y value (usually <= zero)
11) y_end	=	250.00	# y axis end, in time or depth units
12) y_tick	=	10.00	# y axis tick mark interval
13) y_anot	=	50.00	# y axis tick mark annotation interval
14) y_label	= Depth (m)		# y axis label
15) label_size	=	14	# axis label size of seismic plot
16) anot_size	=	12	# annotation label size of seismic plot
17) title	= TITLE		# title text

9

18) text_1   = TEXT 1   # first line of text
19) text_2   = TEXT 2   # second line of text

line: test   parameter filename: test.par
display mode: major parameters   execute mode: both convert and plot

Commands:
Line   [new_linename]
Open   [parameter_filename]
Save   [parameter_filename]
Display   ["all"] or ["major"] or [first-last]
Change   [first-last] or [number [value]]
Execute   ["convert"] or ["plot"]
Quit

Enter Command Letter Plus Optional Arguments: execute plot
Error, parameters have been changed.
Must save parameters to a parameter file before executing.

Commands:
Line   [new_linename]
Open   [parameter_filename]
Save   [parameter_filename]
Display   ["all"] or ["major"] or [first-last]
Change   [first-last] or [number [value]]
Execute   ["convert"] or ["plot"]
Quit

Enter Command Letter Plus Optional Arguments: save test2.par
saving parameters to file: test2.par

Commands:
Line   [new_linename]
Open   [parameter_filename]
Save   [parameter_filename]
Display   ["all"] or ["major"] or [first-last]
Change   [first-last] or [number [value]]
Execute   ["convert"] or ["plot"]
Quit

Enter Command Letter Plus Optional Arguments: execute plot

running command: /bin/csh test2.par test.segy plot

creating postscript file test.500.ps

10

Commands:
Line   [new_linename]
Open   [parameter_filename]
Save   [parameter_filename]
Display   ["all"] or ["major"] or [first-last]
Change   [first-last] or [number [value]]
Execute   ["convert"] or ["plot"]
Quit

Enter Command Letter Plus Optional Arguments: quit
sure you wish to quit (y/n)?:_y

Non-Interactive Mode

The non-interactive program is a Unix C shell script called a parameter file. Usually there will be one or more parameter files for each SEGY file. The first part of the parameter file sets various parameters using the shell "set" command. Comments may be inserted in this part of the file by starting a line with the "#" character. Parameter values may be "toggled" by including several lines that set the same parameter, with all but one of the lines commented out. For example:

set title = "Parkfield"

#set text_1 = "agc set to 150"

set text_1 = "agc set to 300"

#set text_1 = "agc set to 500"

set text_2 = "bandpass set to 25-50-400-800"

The second part of the parameter file uses these parameters to convert a SEGY file (given as a command line argument) to an ASCII file and then create a postscript file. This part of the parameter file is normally not altered, except for one exception noted below.

11

The parameter file is run as a C shell script, and experienced Unix users may find it easiest to manually copy, edit, and run parameter files directly.

The parameter file requires as an argument the name of an input SEGY file. Optional arguments are to set the y axis depth, tick mark interval, and tick mark annotation interval, and the execute mode of "plot" or "convert". The normal operation is to both convert the SEGY file to ASCII and to plot the ASCII file. Y axis arguments allow the user to quickly generate plots of varying lengths. As noted above, it is best to create the longest length plot first, then use the plot option to create additional plots of shorter length, without recreating the ASCII file. Example arguments for parameter file test.par are:

test.par test.segy

test.par test.segy convert

test.par test.segy plot

test.par test.segy 500 50 100 plot

Segy2yxz Programs

The actual SEGY to ASCII conversion program is a C++ program called segy2xyz. This program is called by a parameter file, and is usually not invoked directly by the user. A similar C++ program called segy2xyz-flip also exists. This program is similar to segy2xyz, except that it flips the data laterally (i.e. the first seismic trace becomes the last seismic trace) before converting to ASCII format. To use this program instead of segy2xyz, substitute the name "segy2xyz-flip" for "segy2xyz" in the parameter file.

References:

Wessel, P., and W.H.F. Smith, 1995, The Generic Mapping Tools (GMT) version 3.0 Technical Reference & Cookbook, SOEST/NOAA.

Barry, K.M., Cavers, D.A., and Kneal, C.W., 1975, Recommended Standards for Digital Tape Format, Geophysics, vol 40, no 2, p. 344-352.