OBSTOOL: Software for Processing UTIG OBS Data

Gail Christeson

University of Texas Institute for Geophysics
Technical Report No. 134

June 1995

Institute for Geophysics
University of Texas at Austin
8701 North MoPac Expressway
Austin, Texas 78759-8397

Table of Contents

Introduction

OBSTOOL is a graphical user interface that allows the user to process data collected by the University of Texas at Austin Institute for Geophysics (UTIG) ocean bottom seismographs (OBSs). ***OBSTOOL*** is designed to run a series of programs that will produce a SEG-Y file with filled headers from the original OBS field data, navigation, and start times.

Before the original OBS field data can be transformed into a final SEG-Y file, the OBS needs to be properly located and a clock correction needs to be calculated. The clock correction is necessary because there is significant drift on the OBS internal clock, which must be accounted for so that the shot times and recording times are in the same reference frame.

OBSTOOL is designed so that a user progressing from top to bottom of the main window (Figure 1) will follow all of the steps needed to properly produce a SEG-Y file. The suggested sequence is:

1. Enter OBS location, depth information for all OBS drops
2. Initial clock correction for all OBS drops
3. Initial processing for one OBS drop
4. Make water wave data subset for one OBS drop
5. Pick water wave arrivals for one OBS drop
6. Locate instrument, make secondary clock correction, create closest approach file for one OBS drop
7. Repeat 3-6 for each OBS drop in experiment
8. Final clock correction for each OBS drop
9. Create final SEG-Y files for each OBS drop

Step 7 is suggested before steps 8 and 9 for two reasons: 1) To determine if there is a bias in the secondary clock corrections, which is indicative of a shot delay (a shot delay can be calculated as part of the final clock correction). 2) To determine if the appropriate value for the variation in clock drift rate between sleeping-on-the-sea-floor (cold temperature) and data-acquisition (warmer temperature) phases was chosen (see Initial clock correction section).

The remainder of this document discusses each step in detail, but help is also available on line through the **HELP** buttons.

Information needed for processing

1. Shot file. The shot file merges shot time, navigation, and bathymetry into one file. ***OBSTOOL*** expects the shot file in the following format:
   shot_number year julian_day hour minute second latitude longitude depth
   Example: 6001 93 131 4 16 38.660 8.91521 -104.61215 3046
   Since the format of the navigation, bathymetry, and shot time files will differ from experiment to experiment, no attempt is made for ***OBSTOOL*** to generate the shot file. Generally, more than one shot file is used in the processing sequence - one shot file that includes all shots (for use in making the water wave subset), and several smaller shot files that include each seismic line (for use in making final SEG-Y files). Note: Shot times must increase (most plotting programs also prefer for the shot numbers to increase).

2. OBS start-up/clock-calibration capture files. Information from these files, together with the start and end times of data acquisition will be used to do the clock corrections.

3. OBS deployment location, water depth, water velocity. (Some OBS location programs can solve for water velocity if unknown).

4. Clock and BRG crystal calibration information for each OBS instrument from laboratory measurements. Information from these charts is used to properly choose the value for the variation in clock drift rate between sleeping-on-the-sea-floor (cold temperature) and data-acquisition (warmer temperature) phases.

5. Raw OBS data. Current programs assume that this data file is named obsdata.$obs, where $obs is the drop number e.g. obsdata.36.

Initial processing (convert to raw SEG-Y)

The initial processing converts input data from the raw OBS data format into a raw SEG-Y format file. During this process, it performs a sampling rate correction which accounts for the discrepancies between the Real Time Clock and the Baud Rate Generator. Details can be found in the appendix (see comments for program raw2segy). For most 8088 OBSs with scsi tape or disk there is a byte shift which occurs once per physical write, with each byte found in the next higher address than expected. This condition is checked for, and a byte shift correction is carried out if necessary.

Input information
OBS drop number
PATH0 - directory where input file in raw OBS data format is located
PATH1 - directory in which ouput raw SEG-Y data file will be located
PATH2 - directory in which output log and correction files will be located
Raw data file must have name $path0/obsdata.$obs, where $obs is the OBS drop number. This can be a symbolic link to a tape device if desired.
To run: Press **RUN** button. When program is finished, you may delete $path0/obsdata.$obs (make sure you have it backed up of course, as this is one file that cannot be reproduced).
Output files
Output raw SEG-Y file named $path1/obsdata.$obs.segy
Log file named $path2/obsdata.$obs.log
Correction file named $path2/obsdata.$obs.corr (includes sampling rate correction)
Header information files named $path2/hdrlist.$obs (raw header information) and $path2/hdrint.$obs (interpreted header information)
FORTRAN program: raw2segy.f

Initial clock correction

The initial clock correction will estimate clock drift rate during acquisition. This step can be done either before or after the Initial Processing step. Experience suggests that it is often easiest to enter clock corrections for all deployments at the same time.

Input information
OBS drop number
PATH2 - directory in which log and correction files are located

t1 c1 - Time of clock calibration about 1-2 hours before deployment, clock calibration (actual time - OBS clock time) at time t1
t2 c2 - Time of clock calibration close to deployment time, clock calibration at time t2
t3 - Deployment time plus 2 hours
t4 - Start of first data acquisition period
t5 - End of first data acquisition period
t6 c6 - Time of post-recovery clock calibration, clock calibration at time t6
Start and end of additional data acquisition periods (if applicable). Do not include data acquisition periods where the instrument is only on for a few minutes.
dcdw - Variation in clock drift rate between sleeping-on-the-sea-floor (cold temperature) and data-acquisition (warmer temperature) phases. This value is based on laboratory measurements on each OBS, and depends on recording medium (tape or disk) and estimated operating temperature, and to a lesser degree on number of recording channels.
t1-t6 are input as year julian_day hour minute
c1, c2, and c6 are input in seconds
To run: Fill in requested parameters. Press **RUN** button.
Output files
Log file named $path2/obsdata.$obs.log (new log information appended to this file)
Correction file named $path2/obsdata.$obs.corr (clock correction appended to this file)
FORTRAN program: clockcorr.f

Example:

Table 1. Clock calibration information for OBS drop 2, TICOSECT cruise

OBS Time format: T(time data), OBS drop #, yr, month, day, hr, min, sec, tenths of sec	**GPS Time** format: yr:hr:min:sec	
T029503291025002	088:10:25:01.754708100	
T029503291026001	088:10:26:01.655055000	(t1)
T029503291027001	088:10:27:01.655427200	
T029503291114001	088:11:14:01.672582700	
T029503291115001	088:11:15:01.672948000	
T029503291116001	088:11:16:01.673312400	
T029503291210001	088:12:10:01.692830100	(t2)
T029503291211001	088:12:11:01.693193000	
T029504022035001	092:20:35:11.711002400	(t6)
T029504022036001	092:20:36:11.712014900	
T029504022038001	092:20:38:11.713974500	

For OBS drop 2, deployment time was at 12:20, day 88. There were two data acquisition periods. Period 1 went from 20:58, day 89 to 20:30, day 90. Period 2 was from 19:00, day 91 to 11:00, day 92. The experiment took place in 1995. Example parameters:
t1: 95 88 10 26 c1: 1.555
t2: 95 88 12 10 c2: 1.593
t3: 95 88 14 20

t4: 95 89 20 58
t5: 95 90 20 30
t6: 95 92 20 35 c6: 11.611

Additional data acquisition periods:
tstart: 95 91 19 0
tend: 95 92 11 0
dcdw: 0.1

Figure 2 shows sample calibration measurements for two CPU-8088 real-time clocks. In general, the curves are steeper at low temperatures. The parameter dcdw (variation in clock drift rate between sleeping-on-the-sea-floor and data-acquisition phases) varies depending on the seafloor temperature and recording medium. Measurements indicate that during data-acquisition, the temperature increases by ~4° when writing to tape, and ~1° when writing to disk. To determine dcdw, measure the change in relative deviation for the appropriate temperature increase at the estimated bottom temperature, and multiply by 86400 sec/day. For our sample experiment, dcdw=0.1 for CPU 510 when writing to disk with a bottom temperature of 4°. Table 2 lists average values of dcdw for different conditions.

Table 2. Average dcdw values.

	dcdw - writing to disk	dcdw - writing to tape
deep water (4°C)	0.10	0.50
shallow water (15°C)	0.04	0.20

The procedure used by ***OBSTOOL*** to estimate a clock correction and clock drift rate for each data acquisition period is as follows:

1. Determine a pre-deployment clock drift rate from clock calibrations (c1 and c2) measured before deployment (at times t1 and t2). Extend this drift rate for two hours beyond deployment to allow for the thermal inertia of the instrument package after deployment. This will give you a an estimated clock calibration, c3, at time t3.
2. Determine post-deployment clock drift rate from time t3 to time t6, where t6 is the time of post-recovery clock calibration c6.
3. Vary the post-deployment clock drift rate based on the dcdw value. The clock drift rate will be less during data-acquisition (warmer temperature) phases than during sleeping-on-the-sea-floor (colder temperature) phases. Using the two drift rates determine a clock correction for each data acquisition period. There is a limit (currently four in version 1 of OBSTOOL) of the number of data acquisition periods.

Make water wave data subset (for OBS location)

This step will produce a subset of the OBS data file containing only data from shots fired near each OBS. The subset is used to pick water-wave arrivals (which are used to invert for OBS location and orientation), and to perform a quick quality check on the data. The output data file can be plotted with the **Plot SEG-Y File** option.

Input information
OBS drop number
PATH0 - directory in which raw SEG-Y data file is located
PATH1 - directory in which water wave data subset file will be located
PATH2 - directory in which log and correction files are located

Shot file
Maximum range - maximum shot to OBS range to consider in building water wave data subset (should be close to water depth for picking water-wave arrivals for OBS location inversion, may be larger for quality check on data)
Preliminary OBS location - If location for this OBS is not found in $path2/obsloc.dat, then a window will come up requesting OBS location information. Also, the OBS location file can be edited with the **Edit OBS location file** procedure.
Raw SEG-Y data file - created by **Initial processing (convert to raw SEG-Y)** procedure, named $path0/obsdata.$obs.segy

To run: Fill in requested parameters. Press **RUN** button.

Output files
Water wave data subset named $path1/obsdata.$obs.segy.ww
Log file named $path2/obsdata.$obs.log (new log information appended to this file)
Shot file subset containing only shots located near OBS, named $path2/$shot_file.ww

FORTRAN program: getwwdata.f

Auto pick water wave arrivals

This step produces a data file containing direct water-wave arrival travel times and azimuths. First, an automatic picking program searches for arrivals, based on a given set of input parameters. Next, a plotting window is brought up that allows the user to plot these arrivals. Pick travel times can be moved, deleted, or added. When satisfied, the user saves the picks and exits the plotting window. This output file is read, new azimuths are computed, and an output file is produced for use with the OBS location inversion routines.

Automatic picking

The automatic picking program will search the input data for a trigger, based on input threshold parameters. The current routine uses longterm and shortterm running averages of differences (i.e. amplitude of sample 2 - amplitude of sample 1). If the shortterm running average exceeds both the (longterm running average * relative threshold) and the absolute threshold, then a trigger condition occurs. Details in the code (appendix 1, program arrpick) explain how the correct first arrival travel time is determined from the trigger.

Input information
PATH1 - directory in which water wave data subset is located
PATH2 - directory in which log and correction files are located
OBS drop number
File extension - this should be .segy.ww
Relative threshold - difference between amplitude of pick sample and previous sample must exceed (relative threshold*longterm running average of differences) to trigger. This value will vary per experiment. A value too low will trigger on noise, and a value too high will miss arrivals. Typical values are 10-25.
Absolute threshold - difference between amplitude of pick sample and previous sample must exceed absolute threshold to trigger. This value will vary per experiment, and is used to pass over refractions and other arrivals which have relatively low amplitudes in comparison with the water wave arrivals. Suggestion: set this low, and increase only if changing the relative threshold does not seem to work. A typical value is 2.

To run: Fill in requested parameters. Press **RUN** button.

Output files
Log file named $path2/obsdata.$obs.log (new log information appended to this file)

Arrival file named $path2/obsdata.$obs.arr formatted for plotting purposes
Arrival file named $path2/wwtta.$obs formatted for input into OBS location inversion programs
FORTRAN program: arrpick.f

Water wave arrival time plotting

The water wave arrival time plotting runs the same program as the **Plot SEG-Y file** procedure, and detailed information on the parameters can be found in that section. In brief: Press the **Plot** button to plot the water wave traces and water wave arrival time picks. Type **q** in plot window to remove plot from screen. Move pick arrival times by typing **m** with cursor over present pick location, and a second **m** over new pick location. Type **s** in plot window to save picks to file. Press the **Done** button after saving picks, or if you do not want to change picks.

Input files
Water wave data subset named $path1/obsdata.$obs.segy.ww
Water wave pick file named $path2/obsdata.$obs.arr
Output file
Updated water wave pick file named $path2/obsdata.$obs.arr (Important: remember to press **s** in plot window to update pick file!)
C program: xpltsegy.c

Creating water wave travel time file formatted for input into OBS location programs

When the **Done** button is pushed in the above procedure, a program is run that loads in the water wave data subset and the updated pick file, and produces a water wave travel time file with correct azimuth information formatted for input into the OBS location programs.

Input files
Water wave data subset named $path1/obsdata.$obs.segy.ww
Water wave pick file named $path2/obsdata.$obs.arr
Output files
Log file named $path2/obsdata.$obs.log (new log information appended to this file)
Arrival file named $path2/obsdata.$obs.arr formatted for plotting purposes
Arrival files named $path2/wwtta_orig.$obs, $path2/wwtta.$obs, $path2/wwtta_az.$obs formatted for input into OBS location inversion programs
FORTRAN program: arr_final.f

OBS location

Direct water wave arrival travel times and horizontal polarizations can be used to invert for OBS location, orientation, velocity, and a secondary clock correction. There are several different inversion programs available - choose the correct one for your experiment type (for example, do not invert for orientation if you did not record horizontal components). Run the inversion procedure several times, changing input parameters, until you think you have the best possible results. You can look at the log file by pressing the **DISPLAY LOG FILE** button in the main ***OBSTOOL*** window to check results from each inversion. After you are satisfied with the inversion results, use **Edit OBS location file** to update the OBS location file. **NOTE**: The OBS location file is not automatically changed after an inversion (to safeguard against inversions that do not converge). You must use **Edit OBS location file** to update the OBS location file with inversion results.

Inversion Type

Choose which parameters you would like to invert for (see Table 4 for optimal conditions for each inversion type).

Plotting Increment

When inversion results are plotted, every nth arrival will be plotted.

Invert for OBS location

This program will solve for parameters specified under **Inversion Type**.

Input information for following inversion types: 1) *Latitude, Longitude, Clock Correction(s)* 2) *Latitude, Longitude, Velocity* 3) *Latitude, Longitude, Clock Correction(s), Velocity* 4) *Latitude, Longitude*

PATH2 - directory in which log file, OBS location file, arrival file, etc. are located
OBS drop number
Maximum number of iterations (depends on damping, 100 is good)
Initial clock correction (0.0 sec)
Damping parameter (>=1, depends on data, 10 should be sufficient)
Arrival file named $path2/wwtta.$obs
OBS location file - can edit initial location by pressing the **Edit OBS location file** button

Input information for following inversion type: 1) *Latitude, Longitude, Clock Correction(s), Orientation* 2) *Latitude, Longitude, Clock Correction(s), Orientation, Velocity*

PATH2 - directory in which log file, OBS location file, arrival file, etc. are located
OBS drop number
Maximum number of iterations (depends on damping, 100 is good)
Initial orientation (0 deg)
Initial clock correction (0.0 sec)
Damping parameter (>=1, depends on data, 10 should be sufficient)
Variance weight ratio (rad^2/sec^2) (>1 weights travel time fit higher, <1 weights horizontal polarization fit higher. Try somthing like 4 if the horizontal polarization fit looks good, 49 if there is a lot of variability in horizontal polarizations.)
Arrival file named $path2/wwtta_az.$obs
OBS location file - can edit initial location by pressing the **Edit OBS location file** button

Input information for inversion type: *Orientation*

PATH2 - directory in which log file, OBS location file, arrival file, etc. are located
OBS drop number
Initial azimuth (0 deg)

To run: Fill in requested parameters. Press **RUN** button.

Plot file: To exit plot file, hit return in plot window. To create a postscript file, enter 'p' and then a file name. To zoom, enter 'z'. A box will appear. Move the box with the 'i', 'j', 'k', and 'm' keys. To move at finer or coarser adjustments, type 'f' or 'c'. To switch between 'whole-moving' and 'side-moving' modes type 'w' or 's'. When box is in the correct location, enter 'z' again. To return to original plot, type 'r'.

Output files

Log file named $path2/obsdata.$obs.log (new log information appended to this file)
Inversion information file named $path2/obsloc.$obs.$ext, where $ext is .llc', '.llv', etc. depending on inversion type.

FORTRAN programs: obsloc_llc.f, obsloc_llv.f, obsloc_llcv.f, obsloc_llcaz.f, obsloc_ll.f, obsloc_llcvaz.f, obsloc_az.f

Note: If inversion does not converge, try a larger damping parameter.

Edit input

Used to delete water wave arrivals from input file for inversion.

To use: Double click with the left mouse button on any line you want to delete. Press the **SAVE** button to save. Reload original arrival file by pressing the **GET ORIGINAL FILE** button. Press **EXIT** when finished.

Plot again

Replot inversion results.

FORTRAN programs: plot_llc.f, plot_llv.f, plot_llcv.f, plot_llcaz.f, plot_ll.f, plot_llcvaz.f, plot_az.f

Plot clock correction

Plot secondary clock corrections for specified inversion type, along with curves representing different dcdw parameters (Figure 3). The dcdw values are plotted in increments of 0.05, from dcdw=0 to dcdw=0.5 (dashed purple lines, except dcdw=0, which is a solid blue line). The dcdw value used for the initial clock correction is shown by the solid green line. The secondary clock corrections are shown by the red squares. This plot is useful for determining the best input dcdw values if the seafloor temperature is not known.

FORTRAN programs: plot_clock.f

Plot inversion locations

Plot instrument locations from specified inversions, together with input shot locations and deployment and recovery positions if available (enter these with the **Enter OBS deployment and/or recovery positions** procedure located within the **Edit OBS location file and/or Create closest approach file** procedure).

To use: Set which inversion results you would like plotted (default is all inversion results for instrument). Press **Plot** button. Instrument locations for the various inversions will be plotted with green triangles, deployment and recovery positions will be plotted with red squares, and input shot locations will be plotted with blue plus signs. The coordinate (0,0) on the plot is the mean of the shot locations. Coordinates are in meters. Information on the instrument coordinates from each inversion will be written to the screen. To zoom, enter 'z'. A box will appear. Move the box with the 'i', 'j', 'k', and 'm' keys. To move at finer or coarser adjustments, type 'f' or 'c'. To switch between 'whole-moving' and 'side-moving' modes type 'w' or 's'. When box is in the correct location, enter 'z' again. To return to original plot, type 'r'.

FORTRAN programs: plot_obsloc.f

Edit OBS location file and/or Create closest approach file

Edit OBS location file - can either type in parameters, or replace existing parameters with inversion results. Create closest approach file - used for final clock correction.

To use: You can either manually change parameters, or replace parameters with results from the OBS location inversion programs. For example, if you press the **Replace with inversion result** button, and the **Inversion type** is set to **Latitude, Longitude, Clock Correction(s)**, the latitude, longitude, orientation, and secondary clock correction will be changed to the results from this inversion. Press the **SAVE** button to update the OBS location file. Press **Create closest approach file** button to extract clock corrections from results of given inversion type for use with the final clock correction. Press **Enter OBS deployment and/or recovery positions** to bring up a similar window that allows you to enter deployment and recovery positions (if entered, these positions will then be plotted together with the inversion locations with the **Plot inversion locations** procedure).

Input files:

OBS location file named $path2/obsloc.dat
Inversion information files $path2/obsloc.$obs.$ext, where $ext is '.llc', '.llv', 'etc. depending upon inversion type.

Output files:
OBS location file named $path2/obsloc.dat
Closest approach file named $path2/cl_app.$obs.dat. This file contains secondary clock corrections, as calculated by one of the OBS location inversion programs, at the closest approaches for use in the final clock correction procedure.
OBS deployment location file named $path2/obsloc_dep.dat
OBS recovery location file named $path2/obsloc_rec.dat

What are horizontal polarizations?

The horizontal polarization is a measurement of the partition of first arrival amplitude between the two horizontal components. Horizontal polarization values of 0°, 90°, 180°, and 270° are associated with the H1, H2, -H1, and -H2 directions, respectively. If an instrument is located close to a shooting line, the horizontal polarization values will change rapidly over a small range. Thus, horizontal polarization information can be used to accurately determine the distance of an instrument from the shooting line. In addition, the shape of the horizontal polarization curve will depend on the side of the shooting line the instrument is located on. Examples of theoretical horizontal polarizations for four different OBS locations are shown in Figure 4.

Example inversion:

Input travel times and horizontal polarizations can often be fit equally well with differing combinations of latititude, longitude, secondary clock correction(s), and water velocity. Example locations for one instrument from four different inversion types are shown in Figure 5. The locations (in the coordinate system plotted in Figure 5), secondary clock corrections, and water velocity from these inversions are listed in Table 3. It is often useful to use the final model from one inversion type as the starting model for a different inversion type. For example, the starting location and clock correction for the *Latitude, Longitude, Clock correction(s), Orientation* inversion plotted in Figure 6 is the final model from the *Latitude, Longitude, Clock correction(s)* inversion. In Figure 6, the error in the fit to the water wave arrival travel times is similar for both inversions, but the fit to the horizontal polarizations is improved by moving the instrument further from the shooting line (Figure 5). For this instrument, XBT results indicate a water velocity of 1500 m/s, which is greater than the water velocities from the *Latitude, Longitude, Clock correction(s), Orientation, Velocity* and *Latitude, Longitude, Clock correction(s), Velocity* inversions. Experience shows that the inversions for water velocity work best in deep water, with water wave arrival times out to 3-4 km range from the instrument - otherwise the travel times can be fit equally well with combinations of secondary clock corrections and water velocity values.

Table 3. Example inversion results.

inversion type	location	clock correction	water velocity
Latitude, Longitude, Clock correction(s)	(7,47)	-0.007	1500
Latitude, Longitude, Clock correction(s), Velocity	(-13,71)	-0.012	1485
Latitude, Longitude, Clock correction(s), Orientation	(-56,105)	-0.014	1500
Latitude, Longitude, Clock correction(s), Orientation, Velocity	(-56,105)	-0.019	1475

Which inversion to choose? Table 4 shows the best conditions for each inversion type. When in doubt, use the **Edit OBS location file and/or Create closest approach file** procedure to update the OBS location file with the final model from one inversion, and use this as the starting model in another inversion. This allows you to see exactly what is 'driving' the inversion in a certain direction. Occasionally, the removal of one or two outliers will reconcile the results from two inversions (do this with the **Edit input** procedure). Also, look at the errors for the fits to water wave arrival times and horizontal polarizations. Figure 7 illustrates an example where the error in the water wave arrival travel time fit increases from 8 to 58 ms, and the error in the horizontal polarization fit decreases from 21 to 15 degrees. In this situation, try increasing the variance weight ratio to weight the travel time fit higher than the horizontal polarization fit. If the resulting fit to the travel times is still unacceptable, first use the **Edit OBS location file and/or Create closest approach file** procedure with **Inversion Type** set to *Latitude, Longitude, Clock correction(s), Orientation* to update the orientation with the result from this inversion by pressing the **Replace with inversion result** button (this will also change latitude, longitude, etc.). Then change **Inversion Type** to *Latitude, Longitude, Clock correction(s)* and press the **Replace with inversion result** button - this will update thelatitude, longitude, and clock correction with the results from this inversion. Then press the **Save** and **Create closest approach file for final clock correction** buttons.

Table 4. Optimal conditions for different inversion types.

Inversion Type	When to use
Latitude, Longitude, Clock correction(s)	poor or no horizontals, large clock drift rate, known velocity
Latitude, Longitude, Velocity	very small clock drift rate, deep water, poor or no horizontals
Latitude, Longitude, Clock correction(s), Velocity	deep water, poor or no horizontals, large clock drift rate
Latitude, Longitude, Clock correction(s), Orientation	good horizontal polarizations, known velocity
Orientation	few arrivals (use another inversion to get location)
Latitude, Longitude	very small clock drift rate, known velocity, poor or no horizontals
Latitude, Longitude, Clock correction(s), Orientation, Velocity	deep water, good horizontal polarizations, large clock drift rate

Final clock correction

The final clock correction will estimate a shot delay and clock drift rate during data acquisition.

Input information
- OBS drop number
- PATH2 - directory in which log file, correction file, etc. are located
- Clock calibration information - input in **Initial clock correction** step

Secondary clock corrections at closest approaches - $path2/cl_app.dat - this file is created with the **Edit OBS location file and/or Create closest approach file** procedure.

To run: Choose whether or not to compute shot delay. (If set to yes, shot delay will be computed as the mean of all secondary clock corrections. Note that shot delay may change if you update any OBS locations - therefore if you plan on computing a shot delay you must locate all instruments before doing this step). Press **RUN** button

Output files:

OBS correction file named $path2/obsdata.$obs.corr, with shot delay and final clock correction appended

Log file named $path2/obsdata.$obs.log (new log information appended to this file)

FORTRAN program: clockcorr_final.f

The procedure used by ***OBSTOOL*** to calculate the final clock corrections and drift rates during each data acquisition period is as follows:

1. Using the clock correction and drift rates from the **initial clock correction** procedure, determine a clock correction at the start and end of each data acquisition period.
2. From the initial clock correction and drift rate, estimate the clock correction at each time in the closest approach file. Add to each clock correction the secondary clock correction found in this file. If applicable, subtract the shot delay from this correction.
3. For each data acquisition period, use a least squares procedure to fit a line through the clock corrections available for this time period. From this line, determine a final clock correction and drift rate during this data acquisition period. If no secondary clock corrections are available, then the final clock correction and drift rate will equal the initial clock correction and drift rate.

Create final SEG-Y file

This step will produce a final SEG-Y file suitable for use with other software packages.

Input information

PATH0 - directory where input file in raw SEG-Y data format is located (produced by the **Initial processing (convert to raw SEG-Y)** step

PATH1 - directory in which ouput final SEG-Y data file will be located

PATH2 - directory in which log file, correction file, shot file, etc. are located

OBS drop number

output file extension - output file will be named $PATH1/obsdata.$obs.segy.$ext

Shot file

Line number (for SEG-Y header)

Output number of channels

Advance window opening (sec) - time before shot time you want trace to start at zero offset. This is useful as a deep water delay time when a negative number is supplied e.g. -2 starts the trace at 2 seconds after the shot at zero offset.

Window shift rate (km/sec) - used to vary the trace start time based on shot distance - the reduction velocity you will use to plot the data is a good value

Length of time window (sec) - length of each output SEG-Y trace

Shot depth (m) (for SEG-Y header)

Anti-alias frequency (for SEG-Y header)

Azimuth for sign calculation - set to any negative number if you want to use the standard SEG-Y convention of positive ranges indicating shots approaching instrument, and negative ranges indicating shots moving away from instrument. Otherwise, shots between the given azimuth and the given azimuth + 180° will have a positive range. This is useful if you have several lines crossing the instrument shot in different directions. In general, input an azimuth that is 90° from the azimuth the line was shot.

Turn-on mute length (sec) - applied after a write to tape/disk has occurred (identified by gap in data) to mute out large exponential voltage decay transient signal or mechanical noise. Experience suggests a value of 0.15 sec, 3.8 sec, or 1.5 sec for writing to tape, 200 mB disk, or 500mB disk, respectively. The trace start and end times will be shifted forward by the mute length. No samples are zeroed.

Rotate Horizontals - click yes to rotate horizontal components to radial and transverse directions. This is usually done when instrument orientation has been well determined.

Change EBCDIC Header - If you click yes, a new EBCDIC header will be written that includes processing history for this OBS drop. If you are writing several SEG-Y files for this OBS drop, just click yes for the first SEG-Y file you produce.

To run: Input parameters. Press **OK**. If yes was clicked for the Edit EBCDIC header option, a window named **EBCDIC Header** will come up after a few seconds . Edit this information if needed (this will become the EBCDIC header of the SEG-Y output file), and press the **SAVE AND CREATE SEG-Y FILE**.

Output files:

Final SEG-Y file named $path2/obsdata.$obs.segy.final

Log file named $path2/obsdata.$obs.log (new log information appended to this file)

EBCDIC header file named $path2/ehdrfile

FORTRAN program: final_segy.f

Plot SEG-Y file

This procedure will plot SEG-Y files. Picks can be plotted, and interactively changed, deleted, and added. PostScript versions of the plots can be made.

Input information:

PATH1: directory in which SEG-Y file is located.

PATH2: directory in which pick file is located, and to which PostScript files will be sent.

Files:

SEG-Y file (put path name in PATH1 field)

Channel number: number of channel to plot (ignored if trace sequence number is chosen under X-axis option). Typically in four components are recorded, 1=vertical, 2,3=horizontals, and 4=hydrophone. This is experiment dependent.

Headers : Click off if SEG-Y file does not contain binary & ebcdic headers.

Picks File: Name of file that picks will be read from and written to (put path name in PATH2 field).

PostScript File: Name of output PostScript file (put path name in PATH2 field).

Axes Definition:

Y-axis: Plot with or without time grid. Flip option for refraction data. May specify start and end time to plot.

X-axis: Flip option. Plot by distance, shot number, or trace sequence number. May specify start and end distance/shot_number/sequence_number. Increment: For distance and shot number options, will increment every nth trace with given channel number. For sequence number option, will increment every nth trace ignoring channel number (i.e. if have 3 channels and SEG-Y file is written shot 1 channel1; shot1 channel 2; shot 1 channel 3, choose inc=3 to plot only channel 1, channel 2, etc.). Topographic correction/water wave flattening: If the file named shift.data is present in your SEG-Y file's folder, then each trace will be shifted by the given value. shift.data should contain a series of arrival times (previously picked ?) in seconds. Values are read sequentially and the ones corresponding to the extracted wiggles are applied.

Wiggle Plot Parameters:

Red. Vel. (Reduction velocity): For refraction data: plot y-axis as T - X/red_vel.

Gain/Dist: Processed as follows :

sample_value = sample_value * gain, where gain = 1 + (distance (m) /10000) power (gain_factor) and gain_factor is a user-defined value (1,2,...)

Horizontal gain: If value 'n' equals 2, then the maximum amplitude of a wiggle will not extend beyond the next 2 shots. It is independent from distance and constant over the entire profile.

Bias: If mean value of data is not zero, then the user may enter a bias value that will be considered the zero value for samples by the plotting program. This is especially useful for filled wiggle plots.

Threshold:

Nlin (non-linear): All sample values that lie between the begin and end time, and begin and end distance/shot number are sorted. If the Nlin threshold value is 90, then wiggles will be clipped when the amplitude exceeds the sorted sample value of the 90th percentile. The logic behind using the sorted sample value and not 90 percent of the maximum value is to disregard outliers. (Good value is about 80. A smaller value will increase the average amplitude of arrivals.)

Fixed: Clip wiggles when amplitude exceeds this value (useful when comparing different record sections).

Two information values are written to the screen:

Average of all positive amplitudes for extracted wiggles

Maximum amplitude found after threshold application.

Wiggle: Plot with or without wiggle.

DRT: Use/do not use delay time in SEG-Y header.

Fill Area: Positive and/or negative. Positive will always be plotted to right of trace independent of whether the flip X-axis option is on.

PostScript Parameters:

Plot title

Orientation: Landscape or Portrait

Default Size: Letter or Legal

x,y: coordinates in inches of upper left corner of plot

width: length of y-axis (inches)

height: length of x-axis (inches)

Merge PostScript Files

Plot Window:

Warning: Everytime you modify a parameter value, run a new plot to take this new value into account.

PostScript output: Press 'p' key to obtain a direct screen dump of your current window (including zoom windows). If no PostScript file name is given, plots are directly sent to the default printer.

Zoom: Press mouse at one corner of desired zoom area, hold down until reach opposite zoom corner. Press mouse in zoom plot to return to original plot.

Picking points procedure:

Points are loaded and saved in the file named in "Picks file" field.

'n': start a New profile - type profile number in the plot window. Arrival 0 is considered the water wave.
a': Add current point to picked arrival - get mouse X position and translate it to distance / time coordinates
'm': Move a point - select it first typing 'm', then move to new point place and retype 'm' or cancel with 'c'
'd': Delete a point - select it first typing 'd', then confirm removal with 'd' or cancel with 'c'
'i': Insert a point - first, select point to insert after with 'i', then move to new point place and create it with 'i' or cancel with 'c'
's': Save picked arrivals. The arrivals will be written to the PICKS File, with true arrival times (i.e. no reduction velocity, and including the delay time from the SEG-Y header).
'p': create a Postscript plot of current window
Quit plot window: Press 'q' key
C Program: xpltsegy.c

Output Files

General files:
ebc_head - top 15 lines of EBCDIC header
ehdrfile - latest version of EBCDIC header
info.* files - contain latest input parameters for the FORTRAN programs.
obsloc.dat - OBS location file
obsloc_dep.dat - OBS deployment location file
obsloc_rec.dat - OBS recovery location file

Specific files for each OBS (example for drop number 16)
cl_app.16.dat - secondary clock corrections at closest approaches
hdrint.16 - interpreted header file
hdrlist.16 - raw header file
obsclock.16 - input clock drift information
obsdata.16.arr - water wave arrivals in format for plotting program
obsdata.16.clock.dat - clock drift information in format for **Plot clock correction** procedure
obsdata.16.corr - correction file
obsdata.16.log - log file
obsdata.16.segy - raw SEG-Y data file
obsdata.16.segy.ww - water wave data subset in SEG-Y format
obsloc.16.az - inversion results for *Orientation* inversion (if applicable)
obsloc.16.ll - inversion results for *Latitude, Longitude* inversion (if applicable)
obsloc.16.llc - inversion results for *Latitude, Longitude, Clock correction(s)* inversion (if applicable)
obsloc.16.llcaz - inversion results for *Latitude, Longitude, Clock correction(s), Orientation* inversion (if applicable)
obsloc.16.llcv - inversion results for *Latitude, Longitude, Clock correction(s), Velocity* inversion (if applicable)
obsloc.16.llcvaz - inversion results for *Latitude, Longitude, Clock correction(s), Orientation, Velocity* inversion (if applicable)
obsloc.16.llv - inversion results for *Latitude, Longitude, Velocity* inversion (if applicable)
shotfile.16.ww - subset of shotfile containing only those shots close to OBS (actual name will depend on shotfile name)

wwtta.16 - water wave arrivals, used by inversion programs not needing horizontal polarizations
wwtta_az.16 - water wave arrivals, used by inversion programs which include horizontal polarization information
wwtta_orig.16 - original water wave arrival file

Authors

OBSTOOL was written by Gail Christeson using the TCL and TK toolkits. The underlying FORTRAN and C programs were written by Gail Christeson, Mark Wiederspahn, and Bertrand Toussaint. In many cases, these programs are based on originals written by Yosio Nakamura and Vik Sen. Questions on the SEG-Y plotting should be directed to Bertrand Toussaint (toussain@ccrv.obs-vlfr.fr). All other questions should be directed to Gail Christeson (gail@utig.ig.utexas.edu) or Mark Wiederspahn (markw@utig.ig.utexas.edu).

Appendix - FORTRAN Program Descriptions

FORTRAN programs are available on request from the authors. As of June 1995, code was located at UTIG in /disk/obsdata/markw (arr_final, arrpick, final_segy, getwwdata, raw2segy) and /disk/obsdata/gail/obstool/programs/obsproc (clockcorr, clockcorr_final, obsloc programs, plot_clock, plot_obsloc).

arr_final: Reads the pick file created by arrpick, which may have been edited by the user, and outputs updated travel times and horizontal polarizations formatted for OBS location inversion programs. The algorithm used to calculate horizontal polarizations is identical to that in **arrpick**.

arrpick: Searches continuously recorded, N-component data for water wave arrivals, using input threshold parameters. Travel times and horizontal polarizations are output. The input data can be a condensed data file output from **getwwdata**.

Algorithm:

1. Search for trigger. A trigger condition occurs if (*a1* > *anois*thr*) and (*a1* > *tha*), where *a1* is equal to the first difference between two continuous samples, *anois* is the short term running average of the absolute value of first differences, *thr* is the input relative threshold, and *tha* is the input absolute threshold. Note that *anois* uses the absolute value of first differences and *a1* does not - the assumption is made that water wave first motion is positive.
2. Search forward from trigger for zero crossing, and calculate travel time.
3. Add data near trigger (*MCORRLEN* samples) to a matrix *data* which will be used for stacking purposes. *MCORRLEN* is a parameter found in the file **arrpick.fin.**
4. Calculate horizontal polarization of arrival if horizontal components were recorded. For each horizontal component, let *spro*= sum over *MPOLAR* data samples of *diffh*diffv*, and *ssum*=sum over *MPOLAR* data samples of *diffh*diffh*, where *diffh* is the first difference of continuous samples on the horizontal channel, and *diffv* is the first difference of continous samples on the vertical channel. For each horizontal component *i*, let *aa(i)* = sign(sqrt(*ssum*), *spro*). The horizontal polarization is equal to atan2(*aa*(2), *aa*(1)). *MPOLAR* is a parameter found in the file **arrpick.fin**.
5. Repeat steps 1-4 until end of data is reached. Only use first *MPICKNUM* triggers in matrix *data*. *MPICKNUM* is a parameter found in the file **arrpick.fin.**
6. Create stack of water wave data. For each trace in matrix *data*, subtract mean and normalize by maximum amplitude. Sum these traces to create initial stack. Loop through all traces, subtract trace from stack, find correlation coefficient of trace and stack, and add trace plus lag associated with maximum correlation back into stack. Finally remove traces with poor correlation coefficient from stack.
7. Search for trigger in stack using procedure outlined in step 1. Search backwards from trigger for zero crossing - this is the onset of the water wave arrival. Now search forwards for next zero crossing. Calculate *twidth*, the time between zero crossings.
8. Subtract *twidth* from the travel times calculated in step 2.

clockcorr: Estimates clock correction and clock drift rate during data acquisition periods using clock calibration measurements. Procedure is detailed on page 4.

clockcorr_final: Estimates clock correction and clock drift rate during data acquisition periods using clock calibration measurement and secondary clock corrections from OBS location inversion procedures. Algorithm is detailed on pages 10-11.

final_segy: Creates final SEG-Y files. The output header information is described in Tables A1 and A2. Muting may be applied for two reasons. 1) Transient mute - length specified by user. The trace start and end times will be shifted forward by the mute length. No samples are zeroed. 2) Write to disk/tape mute. If the end of the wanted trace occurred while data was being written to the field media, then the missing samples will be set to zero.

Table A1. Binary Reel Header

byte	length	parameter
3201	4	job number = OBS drop number
3205	4	line number
3213	2	number of output channels
3217	2	sample interval
3221	2	number of output samples per trace
3225	2	output format = 1 = IBM float
3261	4	number of traces in file
3265	4	orientation of OBS channel H1 from north (degrees*100)

Table A2. Trace Header

byte	length	parameter
1	4	sequence number
5	4	sequence number
9	4	shot number
17	4	shot number
21	4	shot number
25	4	shot number
29	2	trace identification code = 1 (seismic data)
31	2	number of vertically summed traces = 1
33	2	number of horizontally stacked traces = 1
35	2	data use = 1 (production)
37	4	shot-receiver distance (sign depends on either shot-receiver azimuth or direction line is shot, according to input parameter)
41	4	receiver elevation (negative below sea level)
49	4	depth of airgun
61	4	water depth beneath shot

69	2	scalar to be applied to elevations and depths specified in bytes 41-68 = 1
71	2	scalar to be applied to coordinates specified in bytes 73-88 = -100 (divide by 100 - gives accuracy to ~0.3 m)
73	4	shot longitude
77	4	shot latitude
81	4	receiver longitude
85	4	receiver latitude
89	2	coordinate units = 2 (seconds of arc)
109	2	delay between shot time and first sample number (ms)
115	2	number of samples (some may be zeroes)
117	2	sample interval in microseconds
119	2	gain type of field instruments = 3 (floating point)
141	2	alias filter frequency
143	2	alias filter slope = 24
217	4	azimuth of shot (degrees*100)
235	2	component ID (negative if unrotated)
		1 - Horizontal component #1
		2 - Vertical component
		6 - Pressure
		7 - Horizontal component #2
237	2	array number (OBS drop number)
239	2	number of non-zero samples in trace

getwwdata: Produces a subset of the OBS data file containing only data from shots fired near each OBS. The input shot file must be time ordered.

obsloc programs:

*Algorithm for **obsloc_ll, obsloc_llc, obsloc_llcaz, obsloc_llcv, obsloc_llcvaz, obsloc_llv***

1. For each input water wave arrival travel time and/or horizontal polarization data point, calculate partial derivative information, and build normal equations. For **obsloc_llcaz** and **obsloc_llcvaz**, weight by variance weight ratio parameter.
2. Calculate rms error.
3. Solve normal equations.
4. Vector u holds model parameters, and vector b holds changes to model parameters. Apply damping (parameter *dmp*) and calculate new model parameters: $u(i)=u(i)+b(i)/dmp$.
5. Repeat steps 1-4 until maximum number of iterations is reached, rms error is below a set tolerance, or rms error between iterations increases.
6. Output final parameters.

*Algorithm for **obsloc_az***

1. For each input horizontal polarization data point, calculate model horizontal polarization value for given latitude, longitude, and input orientation.
2. Find residuals between input and model horizontal polarizations.
3. New orientation = input orientation - mean of residuals.

obsloc_az: For a given position, finds OBS orientation that best fits horizontal polarization data. Results can be plotted with **plot_az**.

obsloc_ll: Finds latitude and longitude that best fit water wave arrival travel time data. Results can be plotted with **plot_ll**.

obsloc_llc: Finds latitude, longitude, and secondary clock correction(s) that best fit water wave arrival travel time data. Results can be plotted with **plot_llc**.

obsloc_llcaz: Finds latitude, longitude, secondary clock correction(s), and OBS orientation that best fit water wave arrival travel time and horizontal polarization data. Results can be plotted with **plot_llcaz**.

obsloc_llcv: Finds latitude, longitude, secondary clock correction(s), and water velocity that best fit water wave arrival travel time data. Results can be plotted with **plot_llcv**.

obsloc_llcvaz: Finds latitude, longitude, secondary clock correction(s), water velocity, and OBS orientation that best fit water wave arrival travel time and horizontal polarization data. Results can be plotted with **plot_llcvaz**.

obsloc_llv: Finds latitude, longitude, and water wave velocity that best fit input water wave arrival travel time data. Results can be plotted with **plot_llv**.

plot_clock: Plots secondary clock corrections for the specified inversion (**obsloc_llc**, **obsloc_llcaz**, **obsloc_llcv**, or **obsloc_llcvaz**), together with curves representing different dcdw paramters.

plot_obsloc: Plots instrument locations from specified inversions (**obsloc_ll**, **obsloc_llc**, **obsloc_llcaz**, **obsloc_llcv**, **obsloc_llcvaz**, and/or **obsloc_llv**), together with deployment and recovery positions (if given) and shot locations. Coordinates are in meters, with (0,0) the mean of the shot locations.

raw2segy: Converts input data from the raw OBS data format into a raw SEG-Y format file. Also performs a sampling rate correction which accounts for the discrepancies between the Real Time Clock and the Baud Rate Generator. For most 8088 OBSs with scsi tape or disk there is a byte shift which occurs once per physical write, with each byte found in the next higher address than expected. This condition is checked for, and a byte shift correction is carried out if necessary.

Algorithm

1. Read first two records of raw OBS data file to determine if byte shift hardware bug exists in input data.
2. Determine optimal number of samples for each SEG-Y trace. Because SEG-Y traces can't be longer than 32768 samples, and existing plotting programs can't handle more than ~8192 samples, we break each OBS contiguous time series into multiple SEG-Y traces. We determine the number of samples that will break the OBS time series into approximately even pieces. Note: a big assumption is made in this step that the number of channels is constant for the entire raw OBS data file.

3. Read in data piece, perform byte shift, demultiplex, convert to output format if wanted (presently we keep the same format, because converting to IBM floating point would approximately double the size of the output data file), and write to output file. Continue until all data has been read.
4. Calculate correct sample interval. First, calculate mean residual count from the sampling residual counts (the time interval between the end of data acquisition for each record in the raw OBS data file and the next clock update, in units of 0.5013 ms - this number occupies byte 7 of the last record of each OBS raw data header). This count is used to calibrate the sample-timer against the real time clock. The deviation, d, in milliseconds, of the actual sampling interval from the theoretical interval is given by:

   $$d = \{100[1-frac(n*Tt)]-0.5013*(r+1)\}/n$$

   where frac(x) is the fractional part of x, n is the number of samples per record per channel (261120/(number of components)), Tt is the sampling interval in ms, and r is the residual count. Write out the sampling rate correction to the correction file.
5. Use correct sample interval to update times in SEG-Y headers to nearest millisecond. On Sun workstations, updating the headers causes significant paging problems in the system unless the disks are very fast or the memory is very large. Until a better way is found, the fix employed for this problem is to occasionally sleep during the header update process.
6. Output interpreted header information from the raw OBS data file.

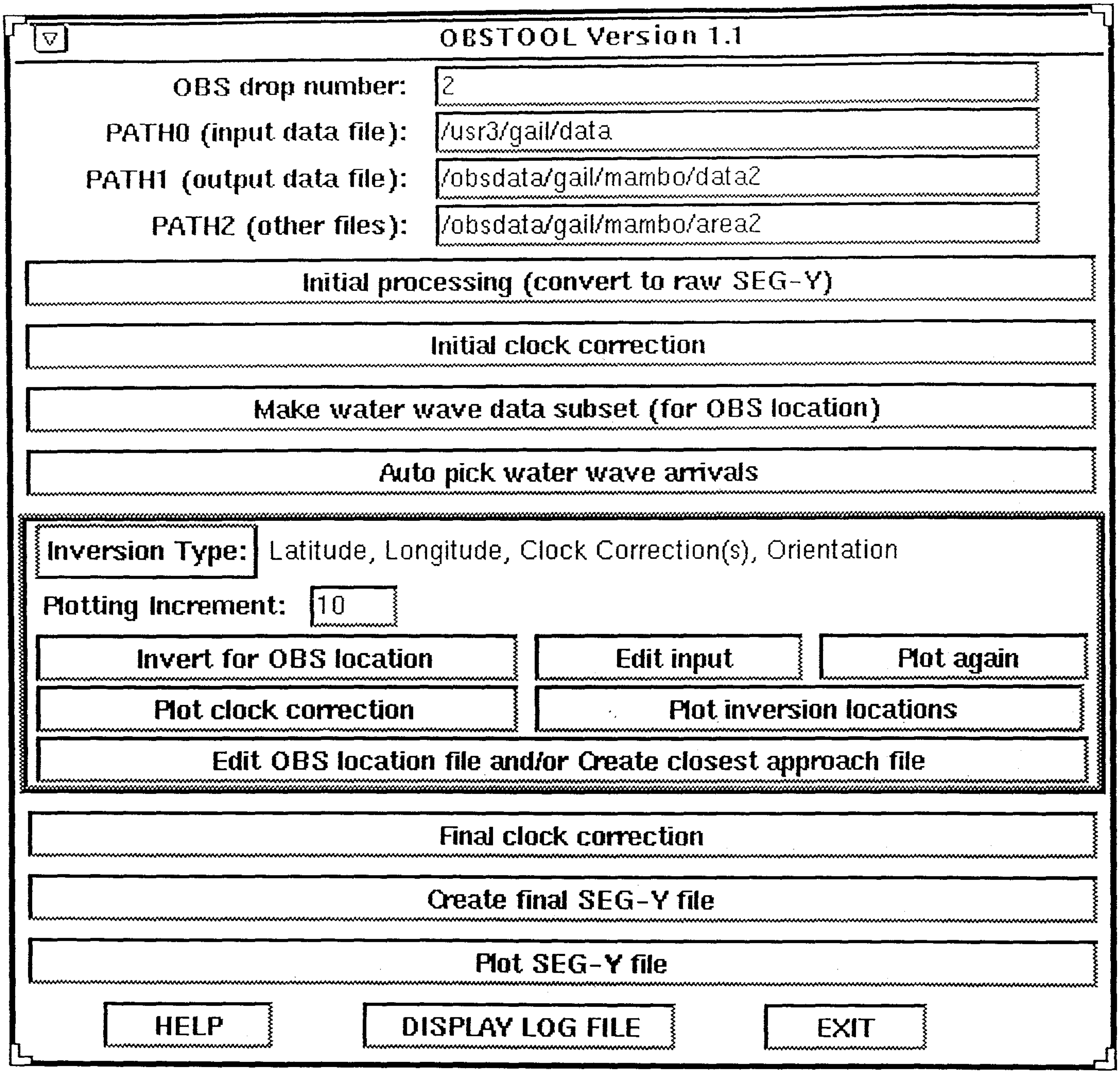

Figure 1. *OBSTOOL* main window.

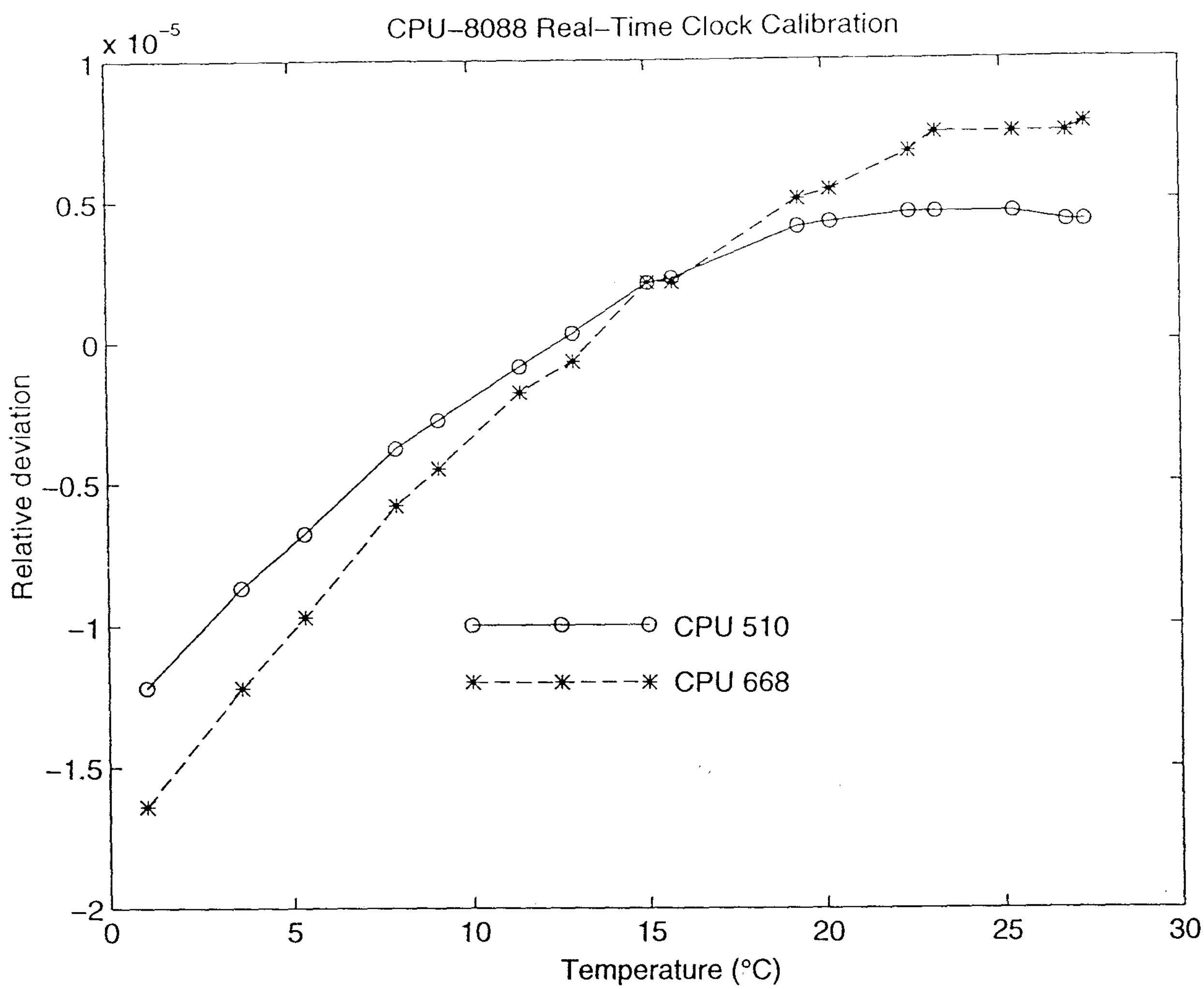

Figure 2. Sample crystal calibration curves. For a given temperature increase, the relative deviation is greater at low operating temperatures than at high operating temperatures. Thus, the variation in clock drift rate between sleeping-on-the-sea-floor (cold temperature) and data-acquisition (warmer temperature) phases is greater for low operating temperatures (large water depths) than for high operating temperatures (shallow water depths).

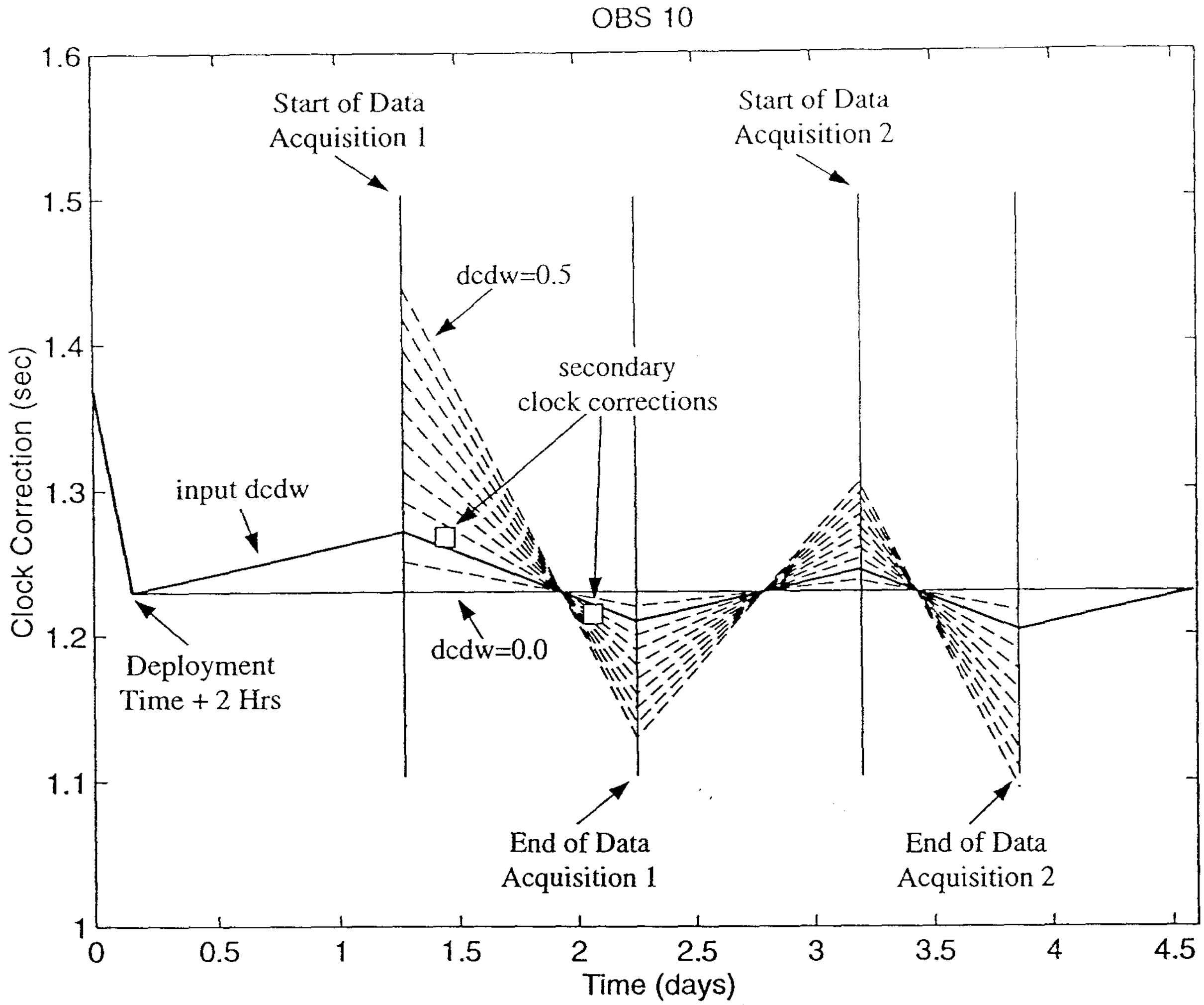

Figure 3. Example of plot from **Plot clock correction** procedure. This type of plot is useful for determining the best dcdw (variation in clock drift rate between sleeping-on-the-sea-floor and data-acquisition phases) value if seafloor temperature or crystal calibration curves are unknown. For this example, the input value of dcdw was 0.1; the secondary clock corrections (from an inversion for *Latitude, Longitude, Clock correction(s), Orientation*) suggest a dcdw value of ~0.13. There are no secondary clock corrections during the second data acquisition period because no shots were located over the instrument. The y-axis is scaled so that the clock correction for a dcdw value of 0.0 will plot as a horizontal line between time t3 (deployment time + 2 hours) and time t6 (time of post-recovery clock calibration).

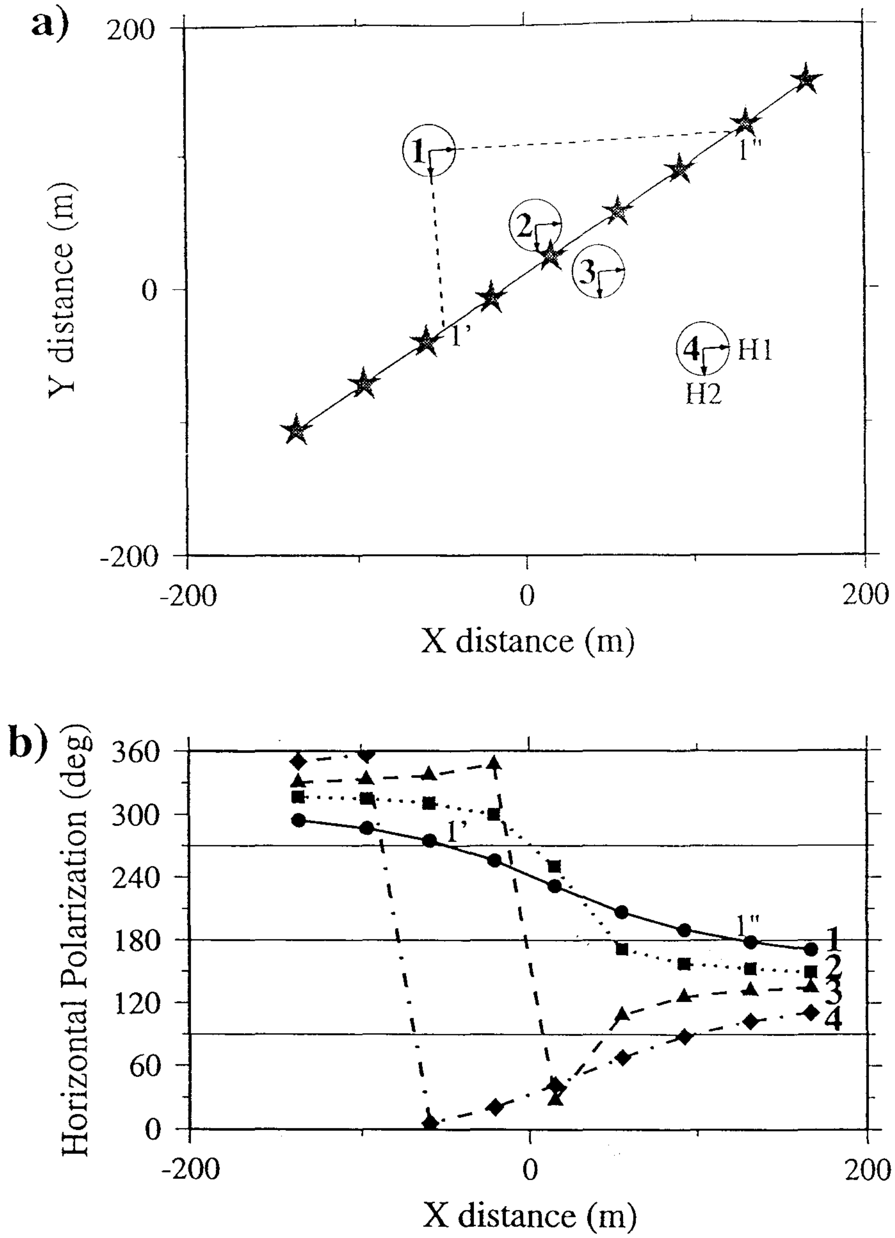

Figure 4. Example of horizontal polarizations for differing OBS locations. a) Plot of shooting line (solid line, with each shot denoted by star) and four OBS locations. The H1 and H2 directions are indicated by the vectors, with H1 at 86.3° and H2 at 176.3° clockwise from north. The dashed lines show the intersection of the H1 and H2 directions with the shooting line. At location 1', all energy from the shooting line is directed in the negative H2 direction for OBS location 1, and at location 1" all energy is directed in the negative H1 direction. b) Horizontal polarization plotted against X distance for shooting line and OBS locations plotted in Figure 4a. The horizontal polarization is a measurement of the partition in arrival amplitude between the H1 and H2 components, and will equal 0/360° for all energy in the H1 direction, 90° for the H2 direction, 180° for the -H1 direction, and 270° for the -H2 direction. For example, locations 1' and 1" are associated with horizontal polarizations of 270° and 180°, respectively. Note that the shape of the horizontal polarization curves is dependent both on distance from the shooting line and what side of the shooting line the OBS is located.

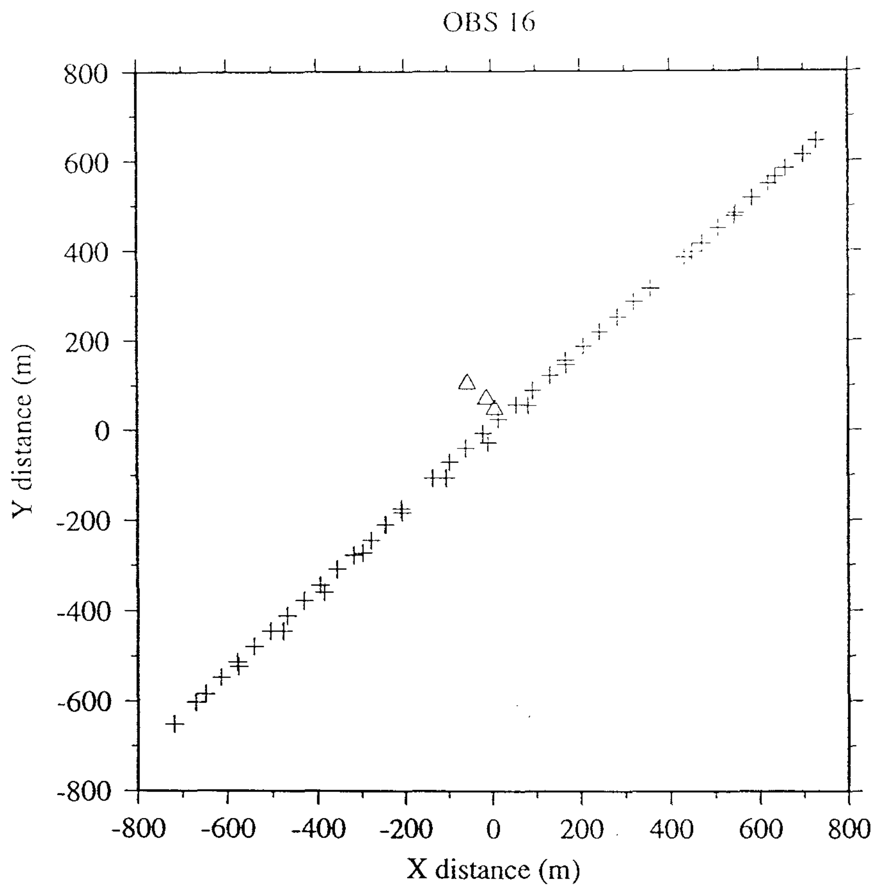

Figure 5. Example of plot from **Plot inversion locations** procedure. The triangles are the locations from the inversions for *Latitude, Longitude, Clock correction(s)* - coordinates (7,47), *Latitude, Longitude, Clock correction(s), Velocity* - coordinates (-13,71), *Latitude, Longitude, Clock correction(s), Orientation* - coordinates (-56,105), *Latitude, Longitude, Clock correction(s), Orientation, Velocity* - coordinates (-56,105). Shot locations are plotted with plus signs. (There is scatter in the shot locations because this line was actually shot twice, once at 50 m spacing and once at 125 m spacing).

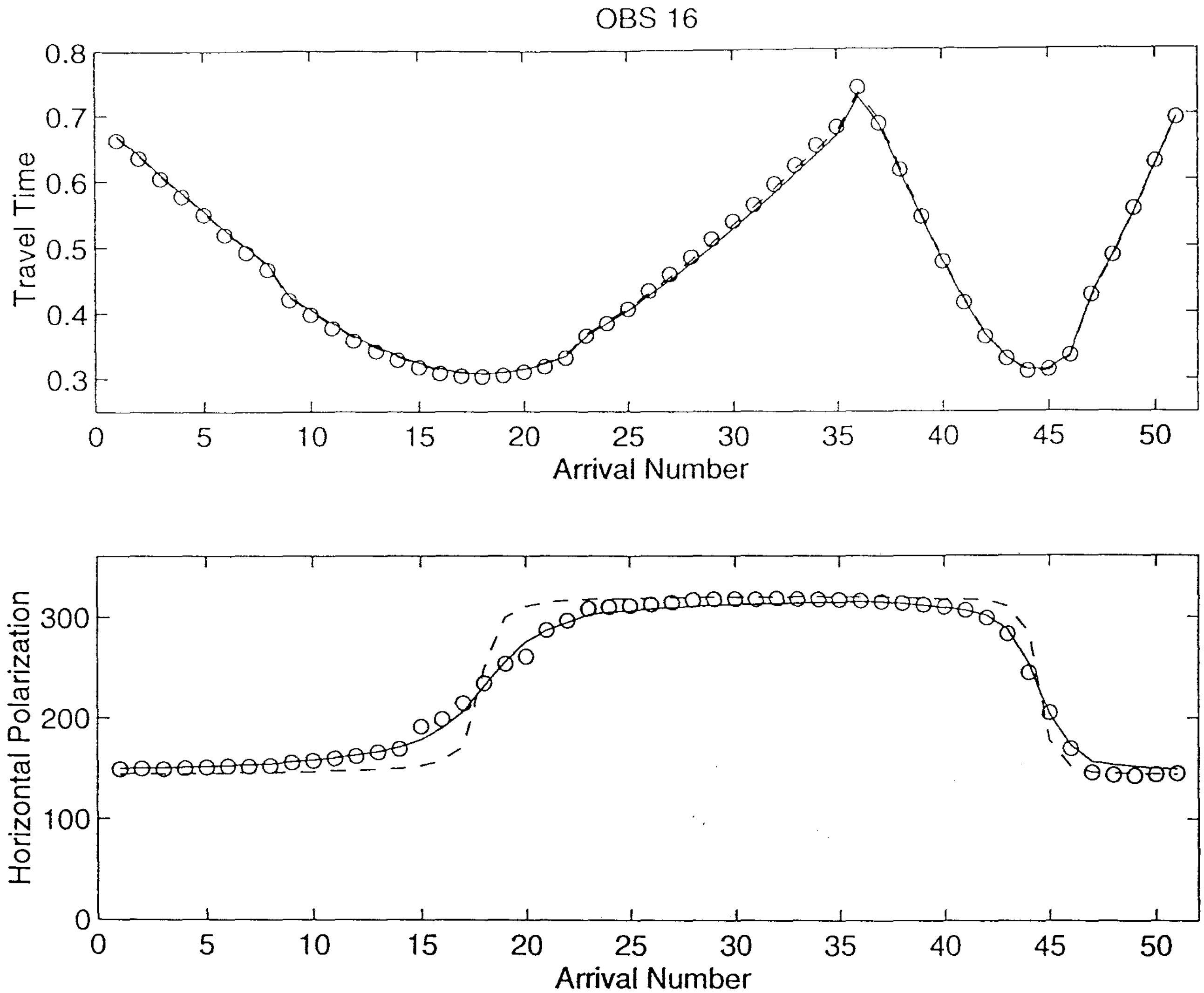

Figure 6. Example of plot from **Invert for OBS location** procedure: good fit. The **Inversion Type** was set to *Latitude, Longitude, Clock Correction(s), Orientation.* Upper panel: Plot of input water wave arrival times (circles), travel times for starting model (dashed line), and travel times for final model (solid line). Two lines were shot over the instrument; arrivals 1-35 are from the first line, and arrivals 36-51 are from the second line. Lower panel: Plot of input horizontal polarizations (circles), horizontal polarizations for starting model (dashed line), and horizontal polarizations for final model (solid line). For this inversion, the starting model was the location from the inversion for *Latitude, Longitude, Clock Correction(s),* with the starting orientation from a previous *Latitude, Longitude, Clock Correction(s), Orientation* inversion. The rms error in the travel time fit has increased from 5 to 6 ms, but the rms error in the horizontal polarization fit has decreased from 18 to 5 degrees. The final instrument location is 60 m from the initial location, and is further from the shooting line (Figure 5).

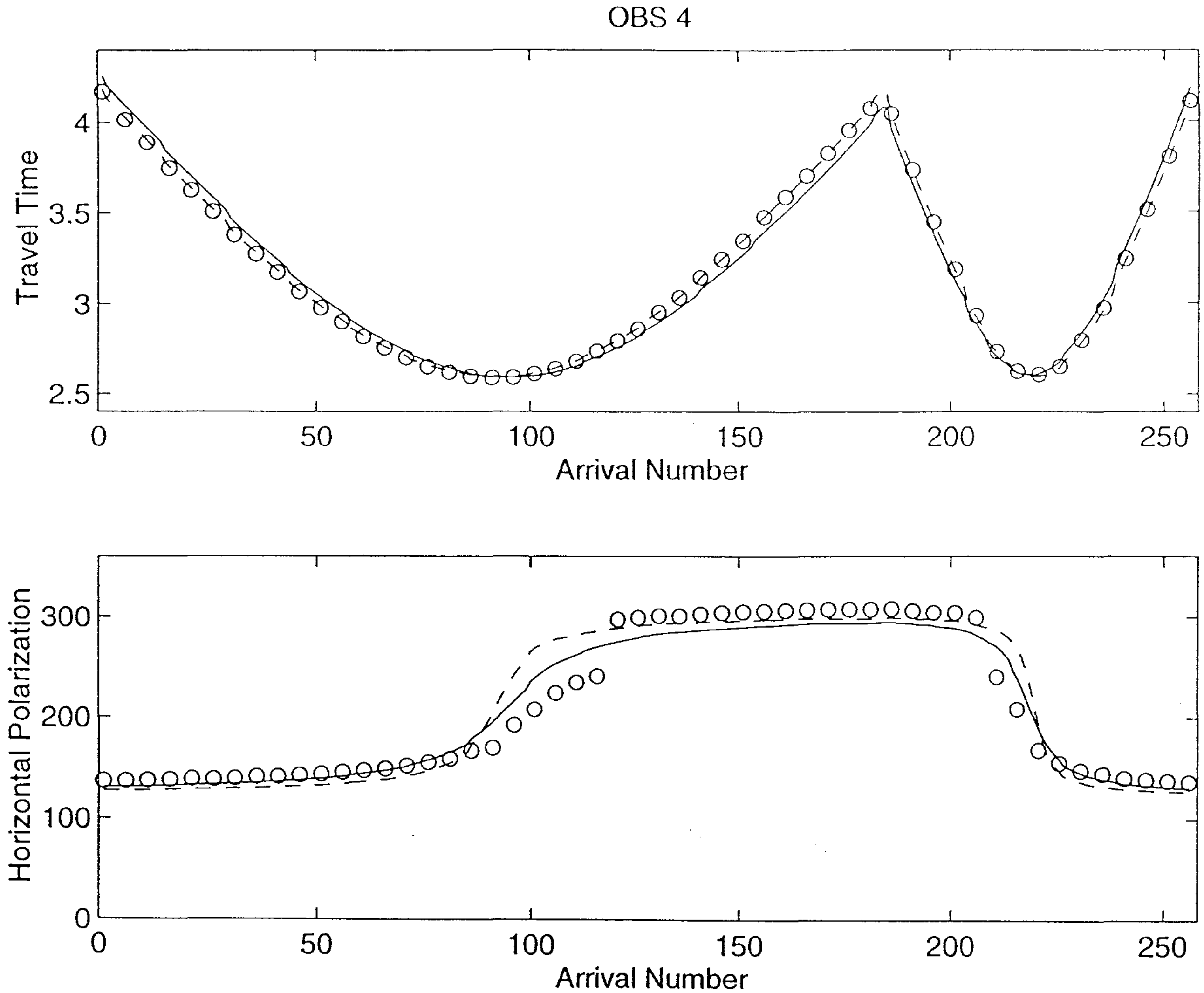

Figure 7. Example of plot from **Invert for OBS location** procedure: poor fit. The **Inversion Type** was set to *Latitude, Longitude, Clock Correction(s), Orientation.* Upper panel: Plot of input water wave arrival times (circles), travel times for starting model (dashed line), and travel times for final model (solid line). Two lines were shot over the instrument; arrivals 1-184 are from the first line, and arrivals 185-256 are from the second line. Lower panel: Plot of input horizontal polarizations (circles), horizontal polarizations for starting model (dashed line), and horizontal polarizations for final model (solid line). For this inversion, the starting model was the location from the inversion for *Latitude, Longitude, Clock Correction(s),* with the starting orientation from a previous *Latitude, Longitude, Clock Correction(s), Orientation* inversion. The error in the travel time fit has increased from 8 ms to 58 ms, and the error in the horizontal polarization fit has decreased from 21 to 15 degrees. The final instrument location is 354 m from the initial location, and is further from the shooting line.