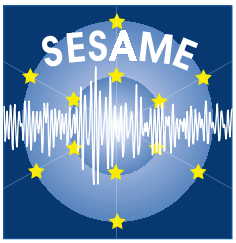


Site Effects Assessment Using Ambient Excitations

SESAME

European Commission – Research General Directorate
Project No. EVG1-CT-2000-00026 SESAME

Report on the

Quality control software

University of Potsdam, Germany

WP06

Array measurements:
Derivation of dispersion curves

Deliverable D05.06

December 2002

List of Contents

Summary 2
1 Waveform visualization tool Wavevis 3
2 GIANT / PITSA analysis system 4
3 H/V spectral ratio computation HVDB 5
4 Array analysis software “cap” 7
Acknowledgements 14
References: 14
List of Appendices: 14

List of Contributors
(in alphabetical order after the last name)

Gerd Härtel	IGUP, Potsdam, Germany
Matthias Ohrnberger	IGUP, Potsdam, Germany
Gudrun Richter	IGUP, Potsdam, Germany
Andreas Rietbrock	IGUP, Potsdam, Germany
Frank Scherbaum	IGUP, Potsdam, Germany
Daniel Vollmer	IGUP, Potsdam, Germany

Project coordinator:	Pierre-Yves Bard
Task B Leader, WP06 Leader:	Frank Scherbaum

Summary

In the following we report on the software used for quality in-situ checks during field experiments. The waveform visualization tool WAVEVIS as well as the programs HVDB (H/V spectral ratio processing) and CAP (array processing software) have been developed under the framework of the **SESAME Project** (Site Effects Assessment Using Ambient Excitations, EC-RGD, Project No. EVG1-CT-2000-00026 SESAME), Task B (Array measurement technique), Work Package 06 (WP06 – Array measurements: derivation of dispersion curves). During the field experiments carried out in summer 2002 (Central Italy and Greece), we have used this combination of software tools together with the GIANT/PITSA packages for quality checks on waveform data and to obtain preliminary array analysis results during field work.

1 Waveform visualization tool Wavevis

For fast visualization of waveform data in the field, Gerd Härtel (IGUP, Potsdam) has developed an easy to use software tool on basis of the Qt - library, version 2.3.0 (http://www.trolltech.com), called **wavevis**. The Qt-GUI toolkit has been chosen for this task, as it is an extensive C++ class library which is freely available for the most common platforms. The use of the Qt-library for developing is free for non-commercial applications on UNIX-platforms (Linux and Sun Solaris), but subject to licensing conditions under Windows and MacOS X.

Wavevis is a typically started from the command line (in Windows also by double clicking the executable program icon), either with one or no argument:

user@host:~/data> **wavevis [testfile.dat]**

The main feature of **wavevis** is the automatic file format identification, which includes at the time seven different seismic waveform formats (including sub-flavours):

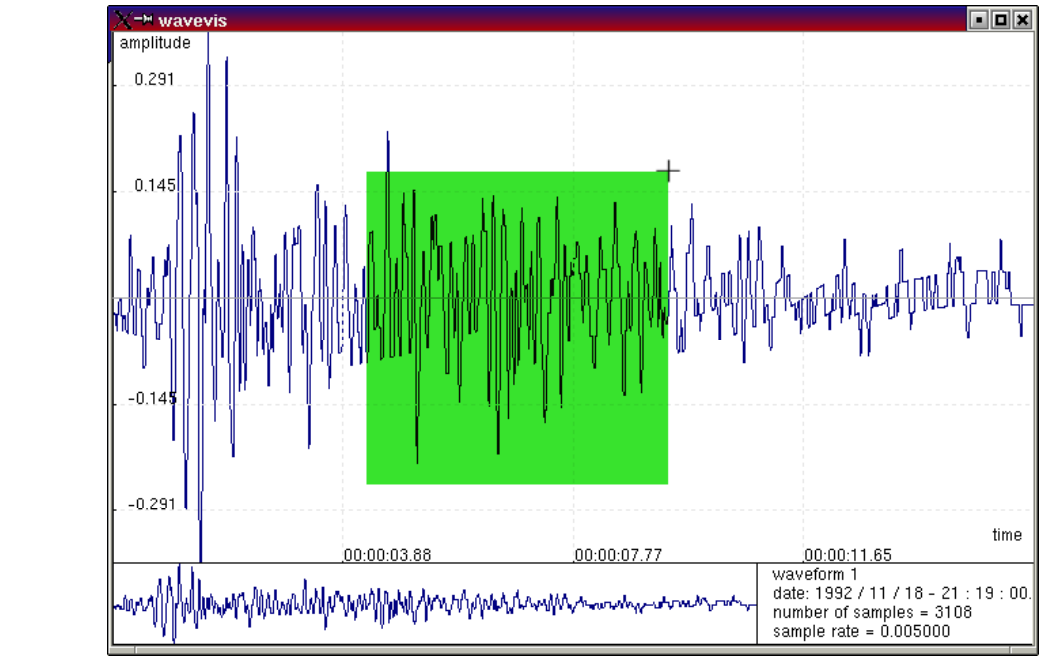
- GSE (INTV, FLTV for version1.0, CMP6/CM6 for versions 1. and 2.)
- SAC (binary and ASCII)
- MiniSEED
- CSS (2.8 and 3.0)
- SEGY
- ASCII for Pitsa
- AH

If the waveform data file given at the command line is recorded in one of the above given formats, a waveform display is shown, for unsupported file formats, an appropriate message is issued before exiting.

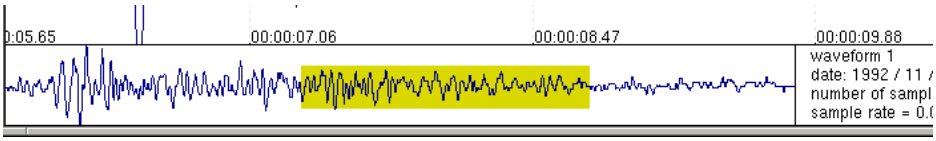
The display is splitted in three parts (compare Figure 1-1). The main window contains at startup the waveform display of the whole trace if a waveform file has been specified in the command line. If no argument has been given, a file can be opened from the main menu. The main menu is opened by clicking the right mouse button on top of the main display window. Choosing the option **"File open"** allows to choose a file from the file system. The main header information of the currently opened file is given in the lower right window.

Further options that can be chosen from the main menu are the "**Print**" option (either to file or printer), a **"Help"** function and, in case of loading a waveform format which allows storing multiple traces within a single file, to load the next or previous waveforms from the currently opened file. Information about the file format of the currently loaded file is obtained by pressing the entry **"File Format Info"** in the main menu.

By pressing the left mouse button and dragging over an area of interest, the trace can be zoomed. Releasing the mouse button, re-displays the zoomed area within the main window display. In the lower left window the complete waveform is visualized, and a yellow box indicates the zoomed region. Arrow buttons can then be used for scrolling forward or backward in time. The **"Return"** function in the main menu resets the waveform display to show the original trace in the main window display.



a)



b)

Figure 1-1 a) Appearance of wavevis after loading a trace file. The main display shows at startup the currently opened waveform. By pressing and dragging the left mouse over a specific area (green box), zooming is activated. b) In the lower left window, always the complete trace is shown, and in case of zooming indicates the position of the zoomed region in the trace as a yellow box. The lower right window displays header information of the currently loaded trace.

Additional information about **wavevis** is given in the html-files: wavevis_doku.html (in German) and wavevis_doku_en.html (in English).

2 GIANT / PITSA analysis system

The idea for the software package GIANT (Rietbrock, 1996, Rietbrock and Scherbaum, 1998) was born during the analysis of the aftershock sequence of the well known Loma Prieta earthquake which took place on October, 17th, 1989 in California. For this task a new processing system for large volumes of digital data sets was developed. The system was designed primarily for the analysis of large scale experiments in mobile seismology such as aftershock monitoring campaigns, but can also be used for the analysis of permanent networks or in our case for mobile array experiments. The main goal for the design of the system was trying to find a reasonable way to deal **quickly** with large volumes of heterogeneous mobile network data **in-depth**, without too many sacrifices in terms of modelling assumptions.

For organising the waveform dataset the dbVista Data Base Manager (Raima Corporation, 1991) is used inside GIANT. This Data Base Manager is based on the network model. Inside the network model the data is stored in *records* which are the basic units of data storage. The relationship between the different *records* is explicitly defined and directly maintained through *set*s. A *set* defines a *one-to-many* relationship between two *record* types. For example one seismic station is recording many seismograms and this dependency is described by the *station -> seismogram se*t. *Sets* are implemented with linked lists of pointers to the record

locations of the set members *(seismogram*s) and owners *(statio*n). The result is a *network* of interconnected *record*s. The great advantage of network based databases is a faster data access in comparison to relational database. Furthermore it is possible to transform every network oriented database into a relational database, which has the great advantage to use SQL (System Query Language) to retrieve data from the database.

Since the storage of the seismic waveform is always very space consuming the waveforms are not stored inside the database. They are stored in a separate directory (directory structure) which can be chosen appropriate for the system. This gives the possibility to work with the database even if the actual waveforms are not currently loaded into the computer system.

The major part of the functionality supplied in the GIANT system deals with the problem of providing fast and consistent solutions of source localization and source parameters of earthquake signals. For the analysis of ambient noise array measurements we make mainly use of the underlying data base concept of GIANT and the possibility to visualize and analyse the waveform data with PITSA (Scherbaum and Johnson, 1993).

An important issue for using the database structure of GIANT is the relatively easy import of information into the system with the GIANT utility program giant_setup. We need only the original waveform data and a simple ASCII file, which contains the important parameters of the field settings, as station names, coordinates, start and stop times of data acquisition to build a fully functional data base within a few minutes. The software tools HVDB and CAP, described in the following sections 3 and 4, are programmed with a direct interface to the GIANT database structure. This allows us to efficiently access the desired waveform data for processing.

A complete reference of GIANT and PITSA software are found in the software manuals attached in Appendix 1 of this report.

3 H/V spectral ratio computation HVDB

The software HVDB is a simple command line tool, allowing to compute H/V ratios for 3 component waveform data organized in a GIANT database. It is entirely written in C and makes use of the RAIMA data base manager library. As output, an ASCII file is created which contains H/V spectral ratios for the individual horizontal components (N/V and E/V), the geometric, and the arithmetic average of both horizontal components (sqrt(NE)/V, sqrt(N*N+E*E)/V).

The HVDB syntax is:

user@host:~> **HVDB –f YYYY[MM[DD[hh[mm[ss.sss]]]]] –l YYYY[MM[DD[hh[mm[ss.sss]]]]] –g GP01+GP02 –o file.out [-b] [-z] [-i 5] [-s]**

Necessary command line parameters are the –f, -l, -g and –o options, followed by an appropriate argument. The –b, -z, -i, and –s flags are optional.

-f	Start time of time interval to be processed in format: YYYY[MM[DD[hh[mm[ss.sss]]]]]
-l	End time of time interval to be processed in format: YYYY[MM[DD[hh[mm[ss.sss]]]]]
-g GP01+GP02	List of station names to be processed, separated by ‘+’ signs.
-o test.out	Output file name
-b	Apply Butterworth bandpass filter in a pre-processing step. Lower and upper corner frequency values, as well as the order of filter are

specified manually at command prompt.

-z In combination –b option specifies the use of a zero-phase filter. This options doubles the order of the filter (forward/backward processing)

-i 5 Integer decimation is performed in a pre-processing step, integer argument is required giving the decimation factor

-s Instruments are simulated to a common instrument response (Seidl, 1980). Calibration information must be present in GIANT database. Parameters (corner frequency and damping) of common instrument to be simulated are given manually at the command prompt.

After selecting an existing GIANT data base by setting the necessary environment variables (see GIANT manual, Appendix 1), the program HVDB can be run from the command line following the syntax given above.

The software accesses the database files and searches for the appropriate waveforms connected to the station(s), which have been selected with the -g option, and lie within the specified time bounds (specified with the -f and –l options). In a first step, the data is cross checked for the existence of three component data, eventual time gaps or changes in sampling rate for any of the selected data streams (station – channel combination). Then the waveform data is read from the referenced waveform data files and a continuous data stream for each channel is allocated in the memory.

In a first pre-processing step, the offset is removed over the entire window length for all channels to be processed. If the flag –b (with or without –z) is chosen, an interactive questioning is started to specify the lower and upper corner frequencies and the order of the Butterworth bandpass filter. The order is specified as number of sections (number of conjugate complex pole pairs fulfilling the Butterworth criteria).

If the optional –s flag has been set, the calibration files (GSE1.0 format – CAL section), which are connected to each of the selected channels are read into memory. The program asks then for the corner frequency and damping value of the common instrument which is to be simulated. After those values have been specified on the command prompt, the simulation is performed individually for each channel.

If the –i option has been selected, the data is subsequently lowpass filtered at 40% of the new sampling frequency and integer decimation is performed.

Then the processing is started. For each station and component, power spectra are estimated in two different ways, in a direct segment averaging manner (also known as Welch's periodogram estimate, Welch, 1967) or by calculating the FFT of the entire window and subsequent averaging of neighbouring Fourier coefficients to reduce the spectral variance (e.g. Press et al., 1994).

Therefore, the user is asked interactively to give a value for the segment length for the periodogram estimate of the power spectrum. This integer value must be a power of two. Additionally, the use has to specify, whether 50% overlapping windows shall be used or not. Furthermore, the user is asked to select the taper fraction to be used for the cosine taper window. A value between 0.5 and 1. should be typically used here.

For each data channel, the data is segmented according to the user's specification, and the periodogram estimate is computed. For the alternative power spectra calculation, the number of neighbouring Fourier coefficients to be averaged is determined from the ratio of the original number of points used for the entire length FFT and the given segment length. Although theoretically both methods should lead to the same power spectral estimate, in practice we see differences for the very low frequency part of the spectra. This is due to the additional offset removal applied to each segment in the periodogram estimate and the segmentation process itself.

For the periodogram estimate additional smoothing can be applied using the spectral smoothing technique proposed by Konno and Ohmachi (1998). This option can be toggled in the interactive dialogue before processing is started. (Remark: the smoothing is performed over the whole frequency band; hence, for larger segment lengths the smoothing can take a considerable amount of computing time).

As output a simple ASCII file is written. The file contains some header information regarding the selected processing parameters as comment lines (lines with leading # - UNIX convention). Below the header information follows the result of spectral ratio estimation, one line with 16 columns per discrete frequency. The columns contain the following results:

Column # 1	Frequency in [Hz]
Column # 2	Spectral ratio estimate N/V
Column # 3	Standard deviation of spectral ratio estimate N/V
Column # 4	Spectral ratio estimate E/V
Column # 5	Standard deviation of spectral ratio estimate E/V
Column # 6	Spectral ratio estimate HV1 = SQRT(N*N+E*E)/V
Column # 7	Standard deviation of spectral ratio estimate HV1
Column # 8	Spectral ratio estimate HV2 = SQRT(N*E)/V
Column # 9	Standard deviation of spectral ratio estimate HV2
Column #10	Spectral ratio estimate N/V (periodogram estimate)
Column #11	Spectral ratio estimate E/V (periodogram estimate)
Column #12	Spectral ratio estimate HV1 (as above, but periodogram estimate)
Column #13	Spectral ratio estimate HV2 (as above, but periodogram estimate)
Column #14	Amplitude spectral estimate for Z-component (from periodogram)
Column #15	Amplitude spectral estimate for N-component (from periodogram)
Column #16	Amplitude spectral estimate for E-component (from periodogram)

All H/V spectral ratios shown in the preliminary analysis (Appendix 3 of report WP05-D07.05) of array data sets have been computed by the use of HVDB. The software is included as binary executables for Linux Intel platforms (Kernel 2.x) as well as for SPARC SUN Solaris systems (2.5.x and higher) on the array data set CDROMs attached to deliverable WP05-D06.05. Due to the licensing conditions of the RAIMA library, we are not allowed to provide source code.

4 Array analysis software “cap”

The software tool “**cap**” (**c**ontinuous **a**rray **p**rocessing) allows performing several array processing methods on waveform data organized in a GIANT database structure. In its current version, the following methods are implemented: a semblance-based FK-algorithm (in frequency domain) which in its core of computation is similar to the code used in NORSAR (Kvaerna and Ringdahl, 1986), the high resolution frequency wavenumber (HRFK) method introduced by Capon (1969), and the modified spatial autocorrelation method (MSPAC) according to Bettig et al. (2001). In a previous version we had also implemented the slantstack method for linear array configurations (e.g. Louie, 2001). This method, however, is no longer included in cap, as from our experience with linear array configurations for the analysis of ambient vibrations, we could not positively confirm, that this procedure allows useful estimates of dispersion curves (Ohrnberger et al., 2001). “cap” is still under development and subject to be extended for additional processing methods. The basic processing scheme of “cap” has been presented in SESAME 2nd progress report (June, 2002).

The “cap” syntax is:

user@host:~> **cap –f YYYY[MM[DD[hh[mm[ss.sss]]]]] –l YYYY[MM[DD[hh[mm[ss.sss]]]]] –g GP01+GP02+GP08+GP10+GP12 –i parameters.cfg [-a mspac.in] [-h]**

Necessary command line parameters are the –f, -l, -g and –i options, followed by an appropriate argument. The –a flag is only to be used for repeated use of the MPSAC processing method.

-f	Start time of time interval to be processed in format: YYYY[MM[DD[hh[mm[ss.sss]]]]]
-l	End time of time interval to be processed in format: YYYY[MM[DD[hh[mm[ss.sss]]]]]
-g GP01+GP02+…	List of station names to be processed as an array configuration, names are separated by '+' signs.
-i parameters.cfg	File name of ASCII parameter file. (see below)
-a mspac.in	Only to be used with the MSPAC processing method. It allows reading previously calculated spatial autocorrelation curves from the specified file for inversion of SPAC curves to a dispersion curve estimate.

The processing of cap is controlled by parameters provided through a simple ASCII file. Only those parameters are specified at the command line, which often need to be changed. This approach has been chosen in order to ease the batch processing capabilities in combination with simple shell scripts in the UNIX environment.

A sample parameter file for the control of "cap" is provided in the box shown below. Parameters are specified by a certain keyword (as first text in the current line) followed by an appropriate value or string. Any other values or text strings are ignored by the program, which makes it possible to include comments of arbitrary form and length into this parameter file (as shown below).

After startup, cap is accessing the database records of a GIANT database searching for the appropriate waveforms of those stations (-g argument) for which the array analysis shall be applied (compare HVDB processing). The implemented pre-processing options are identical to the options in the HVDB software tool. The keywords SEIDL, BBP_FILTER and DECIMATE are used to toggle the instrument simulation, Butterworth band pass filtering or integer decimation options. The keywords HSIM, FSIM, BBP_LOW, BBP_HIGH, BBP_ORDER, ZERO_PHASE must be specified accordingly when these pre-processing options are desired. The WRITE_TRACES keyword specifies, whether the pre-processed waveforms shall be written to the disk for allowing a visual control of the input data (file format GSE2.0).

The processing method is selected with the METHOD keyword and an integer value as documented in the parameter file. Currently it is possible to select the FK, HRFK and MSPAC methods (in the sample parameter file given as BBFK, CAPON and MSPAC, values 0, 1, and 3). The SLANTSTACK (2) method is no longer implemented and the MUSIC (4) selection is currently used for the analysis of the eigenspectrum structure of the array covariance matrix (preliminary step for MUSIC method).

The FK-analysis is performed in a sliding window analysis, computing the frequency wavenumber spectra for each time window for a set of narrow frequency bands. A block diagram of the FK processing scheme is depicted in Figure 4-1.

The wavenumber domain is sampled in a user selectable grid layout. Window lengths and frequency band settings are separately given in the parameter file by the keywords WINLEN and the combination of keywords LOWEST_CFREQ, BANDWIDTH, BANDSTEP and NUM_BANDS.

The frequency bands used are spaced equidistantly on a logarithmic scale. The lowest centre frequency of interest is set with the keyword LOWEST_CFREQ. The bandwidth for all frequency bands is specified as a fraction of the centre frequency by the keyword BANDWIDTH. The higher frequency bands are always spaced equidistantly in a logarithmic frequency scale (linear in period). We specify therefore a multiplicative factor BANDSTEP to define the next higher centre frequency from the current frequency band (i.e. f_1 =

BANDSTEP*f_0, f_{n+1} = BANDSTEP*f_n). The total number of frequency bands is given by the value supplied with the keyword NUM_BANDS.

LOWEST_CFREQ ⇔ lowest centre frequency of interest for processing (e.g. 0.5)
BANDWIDTH ⇔fraction of centre frequency to be used as bandwidth (e.g. 0.1)
BANDSTEP ⇔next higher centre frequency is obtained by this multiplicative factor (e.g. 1.1)
NUM_BANDS ⇔ total number of frequency bands to be used. (e.g. 25)

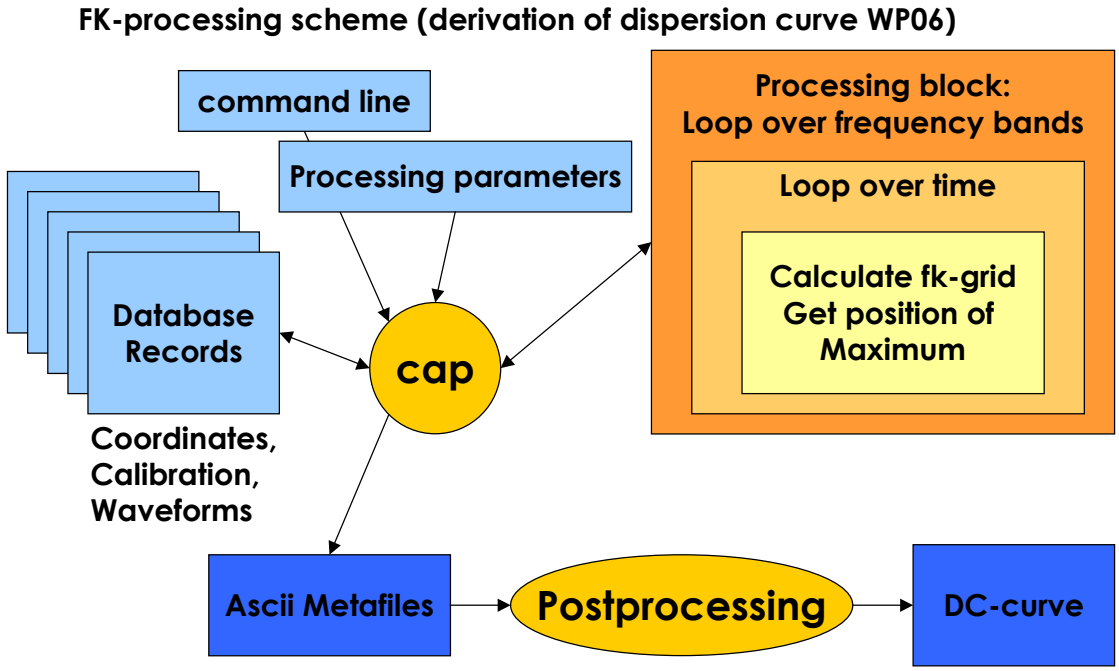


Figure 4-1 Block diagram of FK-processing scheme within cap software.

The WINLEN keyword specifies the use of a fixed window length (in seconds) in the sliding window analysis for all frequency bands. The keyword STEP gives then the value of time progression in seconds. As an alternative, the window length can be chosen automatically as a constant multiplicative factor (keyword WINFAC) of the centre period of the currently processed frequency band. If WINFAC is set to a positive value, it is interpreted as the multiplicative factor for the automatic determination of window lengths. The progression step is then also selected automatically as 50% of the window length for the highest frequency band processed. The WINFAC value then overrides the fixed window length settings of WINLEN and STEP. The automatic selection of window length allows the processing to be performed on constant time-bandwidth product tiles in the time-frequency plane (compare Figure 4-2).

WINLEN ⇔ fixed window length in seconds
STEP ⇔ value for time step between successive window computations in seconds
WINFAC ⇔ positive value: turns on automatic window length selection. Window length is the product of the centre period of the frequency band under consideration and the factor specified with WINFAC keyword.

The settings for the frequency bands and window length selections are used quivalently for the HRFK and the MSPAC methods. In contrast to the FK-technique, where a slowness vector is estimated for each individual time window and frequency band, the HRFK technique uses the time specified time windows as segments for determination of the averaged spectral array covariance matrix (Capon, 1969). A single slowness (and backazimuth) estimate is obtained for each frequency band.

In case of the MSPAC method, the correlation values between all station pairs are calculated for all time windows and each frequency band. Subsequently the corresponding station pairs with similar relative distances are automatically selected from the modified co-array geometry which is obtained for all relative

station pair difference vectors (compare Bettig et al., 2001). Then the averaged spatial autocorrelation curves for various distances can be computed. The result of the spatial autocorrelation curve computation is stored to an ASCII file, then the inversion of the spatial autocorrelation curves is started. The inversion can be reapeatedly started with different parameters for the starting model (CR@1HZ, CREXP), a priori distribution of the dispersion curve (OMEGA, APRIORI), and restrictions to the valid wavelength range (frequency band) for the global inversion scheme (LAMBDA_MIN, LAMBDA_MAX). In order to suppress the computation of spatial autocorrelation curves, the –a option is used at the command line. The argument of the –a flag is the output file created previously for an initial run of the MSPAC calculation.

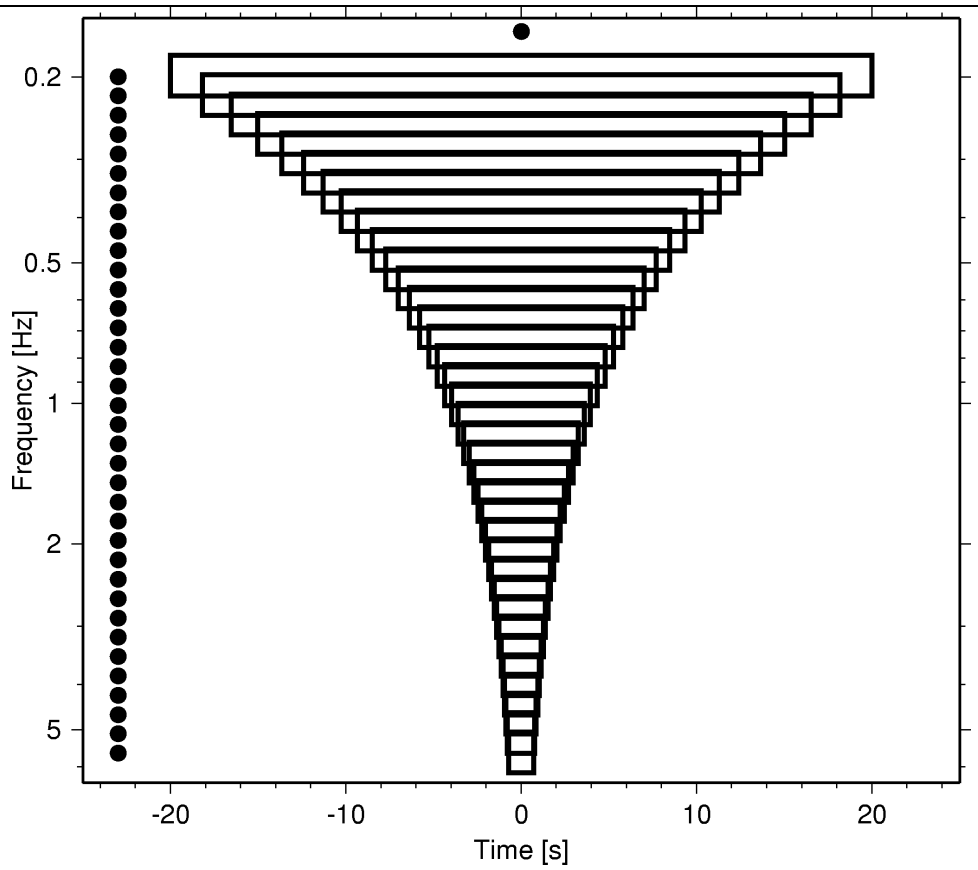


Figure 4-2 Setting the WINFAC value in cap's configuration file to a positive value selects the automatic window length selection for the different frequency bands. Constant time bandwidth tiles are then used for the FK-analysis processing at each time step. The processing windows for all frequency bands are centred on the common midpoint in time (indicated by black solid circle on top of the figure). In the example given, WINFAC has been set to 8, the value for LOWEST_CFREQ is set to 0.2, and BANDWIDTH and BANDSTEP are set to 0.1 and 1.1, respectively (note logarithmic scale for frequency axis). The centre frequencies of the 35 frequency bands (NUM_BANDS equals 35) are spaced equidistantly on a logarithmic scale (black solid circles on lefts side).

Both FK and HRFK (and in future also the MUSIC) methods require a wavenumber grid to be specified. The geometry and characteristics of the sampling in the wavenumber domain are set by the following keywords:

GRID_LAYOUT ⇔ toggles between polar or cartesian grid layout
GRID_TYPE ⇔toggles between grid spacing proportional to slowness or phase velocity.
GRID_RESOL ⇔specifies number of grid points in x,y (cartesian) or radial (polar) direction.
GRID_MAX ⇔maximal slowness (phase velocity) value of the map
NPHI ⇔ in case of polar grid layout, selects number of angular steps.

The raw results of the array analysis are stored in metafiles. The filename is built using the string supplied with the OUTPUT_FILE keyword as a base name and an automatic extension adding to distinguish various output files. If the OFILE_TYPE value is set to ASCII, then the metafiles occupy usually a large disk space.

A binary output file format can be used instead – however, working on these files will then become platform dependent. So far we are not applying a fixed post-processing scheme to the raw analysis result within the cap program itself. For scanning the raw results of the FK-analysis we provide a small utility called fk2disp, which performs easy statistics on the ".max" output files.

```
********** processing settings ************************
METHOD 0                    # select method of processing:
                            0: BBFK
                            1: CAPON
                            2: SLANTSTACK
                            3: MSPAC
                            4: MUSIC

********** applies just for BBFK method ****************
PREWHITEN    0    # toggle prewhitening on or off
                  0: toggles off
                  1: toggles on

DETECT_DOMINANT          0          # toggles detection of single dominant signal
                            # in current window by determination of
                            # eigenspectra characteristics - needs SINGVAL_THRESH

SINGVAL_RATIO               # ratio of first to second eigenvalue
                            # from eigenvalue decomposition of covariance matrix

****** applies to BBFK, CAPON (new), MUSIC and MSPAC ***

NUM_BANDS       10          # number of bands for FK or MSPAC

LOWEST_CFREQ    0.5         # centre frequency of lowest band

BANDWIDTH       0.1         # bandwidth of FK or MSPAC bands as fraction of
                            # centre frequency - filter (1-bw)*fc <-> (1+bw)*fc

BANDSTEP        1.1         # factor used to multiply centre frequency in order to
                            # get to next higher centre frequency

******** applies only for MSPAC method ****************

FILTER_TYPE     0           # select filter type for MSPAC to apply
                            # a narrowband correlation of station pairs
                            # not implemented! - cosine taper used instead
                            # similar implementation as B. Bettig
                            0: Butterworth bandpass
                            1: Gaussian bandpass

OMEGA           -1.         # smoothing for a priori gauss distribution
                            # of model parameters for MSPAC dispersion curve
                            # inversion - if set less than 0 - a priori
                            # information is set to unity matrix

APRIORI         1.          # standard deviation of a priori distribution
                            # of model parameters for MSPAC dispersion curve
                            # inversion - if Omega is set less than 0
                            # this parameter is not used...

CR@1HZ          0.6         # cR(2*PI) at f = 1 Hz, Rayleigh wave velocity at 1 Hz
                            # for initial dispersion curve model (MSPAC)

CREXP           0.1         # exponent for initial dispersion curve model
                            # c(2*PI*f) = c(1)*(2*PI*f)^-CREXP

LAMBDA_MIN      0.1         # use this to determine valid max freq. for
                            # mspac inversion scheme (fraction of wavelength)

LAMBDA_MAX      0.5         # use this to determine valid min freq. for
                            # mspac inversion scheme... (fraction of wavelength)

******* was formerly introduced for CAPON - now obsolete ********
```

```
FLOW            0.5     # lower frequency for CAPON

FHIGH           5.0     # upper frequency for CAPON

***** applies to all grid dependent computations *****************
    BBFK, CAPON, MUSIC

GRID_LAYOUT     1       # select grid layout
                        0: POLAR
                        1: CARTESIAN

GRID_TYPE       0       # select grid type
                        0: equal distance in SLOWNESS
                        1: equal distance in APPARENT VELOCITY
                        2: adaptive: using a priori information about the
                          range of apparent velocities to be observed
                          and the limits of array geometry - this is to
                          be implemented soon!

GRID_RESOL      101     # grid resolution in number of points
                        # for polar grid layout this is resolution in r

GRID_MAX        2.5     # maximum of grid either app. vel. or slowness

NPHI            36      # number of azimuthal steps for polar grid layout

********* applies to all methods ******************************

COMP            1       # select component to process
                        1: vertical component Z
                        2: north component N
                        3: east component E
                        22: radial component R
                        33: transverse component T
                        123: all three components

WINLEN          5.0     # window length in seconds

STEP            1.0     # forward step in seconds

WINFAC          5.0     # window length is adjusted to centre frequency
                        # of processed frequency band FCENT -
                        # window length is set to:
                        # WINLEN = WINFAC * 1./FCENT
                        # WINFAC set to positive value overrides
                        # settings for WINLEN and STEP!

TAPER_FRAC      0.5     # fraction for cosine taper

STACK_MAPS      0       # flag for specifying stacking of maps
                        # BBFK only
                        0: don't stack maps
                        1: stack maps

********** applies just for SLANTSTACK and is now obsolete ******

POWSPEC                 0       # flag whether power spectrum is calculated by
                        # stacking windows or smoothing in spectral domain
                        0: window stacking
                        1: smoothing stacked spectrum

STRIKE          -1.     # strike of line for slantstack analysis
                        # values < 0 indicates use of regression result
                        # from linear array configuration

******** I/O settings *********************************

STAT_LIST S01_+S02_+S03_+S04_   # station list to process
                                # is now read from command line
                                # => obsolete parameters, but
                                # may be used to keep track of
                                # processing parameters...
```

```
OUTPUT_FILE     test.out   # base name of output file -
                           # extensions are added for output files

OFILE_TYPE      0          # flag for output file type
                           0: write out ASCII file
                           1: write out BINARY file

MAP_WRITE       0          # flag for map writing
                           # applies only for CAPON in the moment
                           # use STACK_MAPS for BBFK stacked maps
                           0: don't write maps
                           1: write maps

WRITE_TRACES    0          # flag if pre-processed traces should be written out
                           0: don't write out pre-processed traces
                           1: write out pre-processed traces

******** pre-processing parameters **********************

DECIMATE        5          # integer decimation factor - 1 turns off

SEIDL           0          # flag for instrument simulation
                           0: don't simulate common instrument response
                           1: simulate common instrument response
FSIM            0.2        # corner frequency for common instrument simulation

HSIM            0.7        # fraction of crit. damping for instrument simulation

BBP_FILTER      0          # flag for Butterworth bandpass filtering
                           0: don't filter
                           1: filter

BBP_LOW                    0.1        # lower corner frequency for Butterworth bandpass

BBP_HIGH        5.0        # upper corner frequency for Butterworth bandpass

BBP_ORDER       2          # number of sections for Butterworth bandpass

ZERO_PHASE      0          # flag for zero phase filtering
                           0: just forward filtering
                           1: zero phase filter - forward/backward filtering

GAUSSNOISE      0.1        # if value larger 0 then Gaussian noise is
                           added to all traces - the float value specifies
                           the standard deviation of the Gaussian noise
                           as a fraction of the standard deviation of the
                           traces itself

******** more specialized parameters **********************

TIME_CORR       0          # flag if time correction have to be applied
                           0: don't need time correction
                           1: need time correction

3DCORRECT       0          # flag whether 3D array geometry is evaluated
                           0: option turned off
                           1: best plane fitted to 3D geometry of array
```

All preliminary results of the array analysis shown in Appendix 3 of report WP05-D07.05 for the test sites have been computed by using the FK-method in cap. The software is included as binary executables for Linux Intel platforms (Kernel 2.x) as well as for SPARC SUN Solaris systems (2.5.x and higher) on the array data set CDROMs attached to deliverable WP05-D06.05. Due to the licensing conditions of the RAIMA library, we are not allowed to provide source code.

Acknowledgements

This project (Project No. EVG1-CT-2000-00026 SESAME) is supported by the European Commission – Research General Directorate.

References:

Aki, K., *Space and time spectra of stationary stochastic waves, with special reference to microtremors*, Bull. Earthquake Res. Inst. Univ. Tokyo, Vol. 35, pp. 415-457, 1957.

Bettig, B., Bard, P.-Y., Scherbaum, F., Riepl, J., and Cotton, F., *Analysis of dense array noise measurements using the modified spatial auto-correlation method (SPAC). Application to the Grenoble area*, submitted, 2001.

Capon, J., *High-Resolution Frequency-Wavenumber Spectrum analysis*, Proceedings of the IEEE, Vol. 57, No. 8, pp.1408-1419, 1969.

Konno, K., and Ohmachi, T., *Ground-Motion characteristics Estimated from Spectral Ratio between Horizontal and Vertical Components of Microtremor*, Bulletin of the Seismological Society of America, Vol. 88, No. 1, pp. 228-241, 1998.

Kværna, T., and Ringdahl, F., *Stability of various f-k estimation techniques*, in: Semiannual Technical Summary, 1 October 1985 - 31 March 1986, NORSAR Scientific Report, 1-86/87, Kjeller, Norway, pp. 29-40, 1986.

Louie, J.N., *Faster, Better: shear-Wave Velocity to 100 Meters Depth From Refraction Microtremor Arrays*, Bull. Seism. Soc. Am., Vol. 91, No. 2, pp. 347-364, 2001.

Ohrnberger, M., Scherbaum, F., Hinzen, k.-G., Reamer, Sh.-K., and Weber, B., 2001, *Vibrations on the Roll - MANA, a Roll Along Experiment to map Local Site Effects Across a Fault System*, Eos. Trans. AGU, Vol. 82, No. 42, Fall. Meet. Suppl., Abstract S21D-0606, 2001.

Raima Corporation, *dbVista Data Base Manager*, 1991.

Rietbrock, A., *Entwicklung eines Programmsystems zur konsistenten Auswertung großer seismologischer Datensärtze mit Anwendung auf die Untersuchung der Absoorptionsstruktur der Loma-Prieta-Region, Kalifornien*, Ph.D. Dissertation, Fakultät für Geowissenschaften der Ludwig-Maximilians-Universität München, 1996.

Rietbrock, A. and Scherbaum, F., The GIANT analysis system, *Seismological Research Letters*, Vol. 69, No. 6, pp. 40-45, 1998.

Seidl, D., *The Simulation Problem for Broad-Band Seismograms*, Journal of Geophysics, Vol. 48, pp. 84-93, 1980.

Scherbaum, F. and Johnson, J., PITSA, Programmable Interactive Toolbox for Seismological Analysis, *IASPEI Software Library*, Vol. 5, 1992.

Tokimatsu, K., *Geotechnical site characterization using surface waves*, in: Earthquake Geotechnical Engineering, Ishihara (ed.), pp. 1333-1368, Balkema, Rotterdam, 1997.

Welch, P.D., *The use of Fast Fourier Transform for the estimation of power spectra: a method based on time averaging over short, modified periodograms*, IEEE Trans. Audio Electro – Acoust., AU – 15, pp. 70-73, 1967.

List of Appendices:

Appendix 1: Software documentation

GIANT manual
PITSA manual

These manuals are available on the site http://lbutler.geo.uni-postdam.de/service.htm and on the CD appended to the 18th month report.