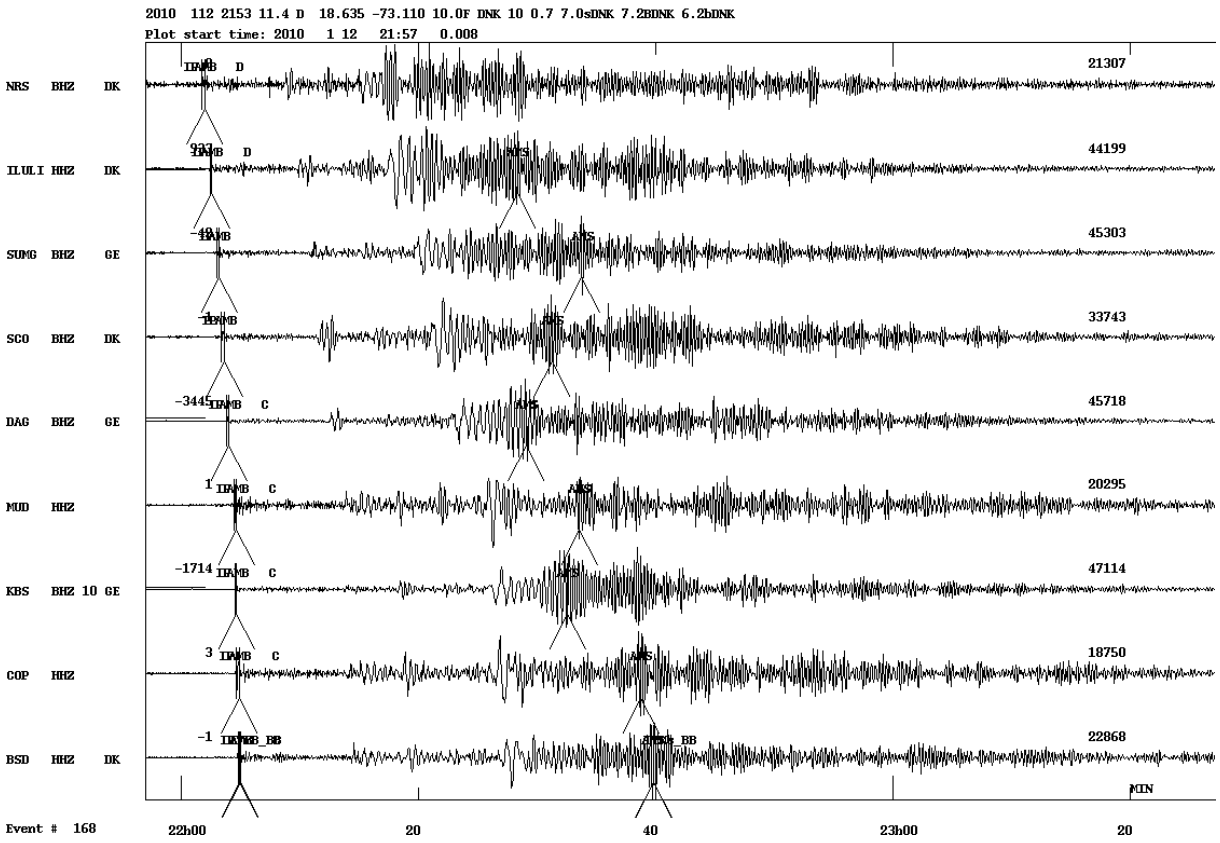
SEISAN
EARTHQUAKE ANALYSIS SOFTWARE

FOR WINDOWS, SOLARIS, LINUX and MACOSX

Version 9.0.1



Editors

Lars Ottemöller[(1)] lars.ottemoller@geo.uib.no
Peter Voss[(2)] pv@geus.dk
Jens Havskov[(1)] jens.havskov@geo.uib.no

(1) Department of Earth Science
University of Bergen
Allgaten 41
5007 Bergen
Norway

(2) Geological Survey of Denmark and Greenland
Øster Voldgade 10
1350 Copenhagen K
Denmark

May 2011

Cover
The seismograms on the cover show the devastating earthquake in Haiti, January 12, 2010. Note how the broad band bodywave magnitude of 7.2B gives a better estimate than the old Mb.

Citiation
If you need to cite this manual : Ottemöller, Voss and Havskov, SEISAN EARTHQUAKE ANALYSIS SOFTWARE FOR WINDOWS, SOLARIS, LINUX and MACOSX , 2011.

Copyright ©2011 Ottemöller, Voss and Havskov.

Contents

1 Introduction **1**

1.1 Latest changes . . . 2

1.2 Information about SEISAN online . . . 5

2 STRUCTURE OF SEISAN **7**

2.1 Directories . . . 7

2.2 The database . . . 7

2.2.1 Phase data and hypocenters . . . 9

2.2.2 Waveform data and formats . . . 10

2.2.3 Continuous waveform data . . . 12

2.3 File types used with SEISAN . . . 14

2.4 Upper and lower case . . . 15

2.5 Moving data between Sun, Linux, MacOSX and Windows . . . 15

3 INSTALLATION **17**

3.1 Unix (SOLARIS and Linux) . . . 18

3.2 MacOSX . . . 21

3.3 Cygwin . . . 22

3.4 Windows (95, 98, 2000, NT, XP and Vista) . . . 23

3.5 Database directories for your own system, MAKEREA . . . 24

3.6 Default parameter files for the main programs . . . 25

3.7 Color settings . . . 25

3.8 Compiling SEISAN programs . . . 27

3.9 Program Version . . . 28

3.10 Setting general SEISAN defaults, `SEISAN.DEF` . . . 28

3.11 Format description in parmater files . . . 32

3.12 Using SEISCONF to modify parameter files 32

4 USING SEISAN **37**

4.1 SEISAN TRAINING COURSE 37

4.2 Short user guide 38

4.3 Getting data into the database 41

4.3.1 System with digital data 41

4.3.2 System without digital data 42

4.3.3 Database security 42

4.3.4 Data base tools, content and checks 43

4.3.5 High accuracy in SEISAN 43

4.4 Interactive work with earthquake locations, EEV command 44

4.5 How EEV works 44

4.5.1 EEV driver program: JSEISAN 55

4.5.2 EEV Windows driver program: SEISAN 67

4.6 System response 69

4.7 Working with catalogs 71

4.7.1 Explosions in SEISAN 71

4.8 Printing 72

4.9 General Work with SEISAN 72

4.10 Graphics in SEISAN 73

5 SEISAN TRAINING COURSE **75**

6 DESCRIPTION OF PROGRAMS AND COMMANDS **77**

6.1 Hypocenter location programs: HYPOCENTER, HYPO71 and HYPOINVERSE 77

6.1.1 The hypocenter program, HYP 77

6.1.2 HYPO71 (Sun only) 91

6.1.3 The Hypoinverse program, HYPINV (SUN and PC) 95

6.1.4 HYP_ISC (Unix and Linux only) 96

6.2 Trace plotting, phase picking and spectral analysis, MULPLT 97

6.2.1 MULPLT main functions 97

6.2.2 Use of MULPLT from EEV 99

6.2.3 Continuous plotting of one channel 99

6.2.4 Commands in MULPLT, overview 100

6.2.5 Registering new events into SEISAN . . . 112
6.2.6 Adding BUD archive waveform data to nordic file, WAV2BUD . . . 113
6.2.7 Phase picking, amplitude, weight and polarity . . . 113
6.2.8 Theoretical arrival times for global and local phases and location . . . 114
6.2.9 Instrument correction and magnitudes Ml, mb and Ms . . . 115
6.2.10 Determine azimuth of arrival (3 comp or array) and component rotation . . . 117
6.2.11 Data manipulation commands . . . 118
6.2.12 Spectral analysis, s(Spec) . . . 123
6.2.13 Particle motion plots . . . 129
6.2.14 The `MULPLT.DEF` and `SEISAN.DEF` files . . . 129
6.2.15 Distance trace plot with GMT, TRACE_PLOT (Unix only) . . . 136
6.3 Plotting epicenters . . . 137
6.3.1 EPIMAP . . . 137
6.3.2 W_EMAP, Version Windows based map program . . . 143
6.3.3 GMAP, Plotting epicentres with Google maps or Google Earth . . . 147
6.3.4 SEIS2VIEWER, Plotting hypocentres in 3D . . . 153
6.4 Searching in the database, SELECT . . . 155
6.4.1 Searching for text string in nordic files, SELECTC . . . 161
6.5 Extracting events from the database, COLLECT . . . 161
6.6 Inserting events into the database, SPLIT . . . 162
6.7 Updating final locations in database, UPDATE and UPD . . . 163
6.8 Using `filenr.lis`, DIRF and DELF . . . 165
6.9 Making a bulletin, BUL . . . 165
6.10 Reports and statistics . . . 166
6.11 Waveform file management tools . . . 171
6.12 File conversion and modification programs . . . 183
6.13 Signal processing programs . . . 201
6.13.1 PITSA . . . 201
6.13.2 SAC2000 . . . 202
6.13.3 Degtra A4 . . . 202
6.14 Calculating b-value, BVALUE . . . 204
6.15 Automatic phase picking, AUTO, AUTOPIC, AUTOSIG, CONDET . . . 206
6.15.1 AUTO and AUTOPIC . . . 206
6.15.2 AUTOSIG . . . 210

6.15.3 Detection program for continuous data, CONDET 214
6.16 Fault plane solution 216
6.16.1 FOCMEC 216
6.16.2 FPFIT 221
6.16.3 HASH 223
6.16.4 Moment tensor inversion program, INVRAD 232
6.16.5 Moment tensor inversion program, PINV 232
6.17 Calculation of coda q, CODAQ 233
6.18 Merge events near in time ASSOCI 240
6.19 Making synthetic seismograms 240
6.20 Calculation and plotting of travel times 248
6.20.1 IASPEI travel time software, program TTIM 248
6.20.2 Calculation of travel times for layer and gradient model, TTLAYER 250
6.20.3 Plotting of travel times, TTPLOT 250
6.20.4 IASP, travel times for MULPLT 251
6.21 Inversion for Q_{Lg}, QLG 252
6.22 Wadati 253
6.23 Calculating spectra, the SPEC program 255
6.24 Seismic risk related programs 260
6.25 Magnitude relations, MAG 275
6.26 ML inversion, MAG2 280
6.26.1 MAGSTAT 282
6.27 Explosion filtering, EXFILTER 282
6.28 Inversion of travel time data and joint hypocenter determination 284
6.28.1 VELEST 284
6.28.2 NOR2DD 289
6.28.3 NOR2JHD_PUJOL 289
6.29 Analysis of volcanic earthquakes 290
6.30 FK Analysis 293
6.31 Surface wave analysis (SUN) 295
6.32 Instrument response 295
6.32.1 Create instrument response files, RESP 295
6.32.2 Examples of main response files from seismometers and accelerometer 305
6.32.3 SEED response 309

6.32.4 SEED response to GSE, SEEDRESP2GSE 310
6.32.5 GSE response to SEED, GSERESP2SEED 310
6.33 Macroseismic data in SEISAN 311
6.34 Correlation of waveform signals, CORR and detection of event clusters XCLUST 313

7 Programming in SEISAN and list of files in SEISAN distribution 319
7.1 Programmers guide and some test programs 319
7.2 CONTENTS OF PRO, LIB, INC, INF, COM, DAT, SUP, ISO and PIC DIRECTORIES 321

8 Acknowledgments 323

A The Nordic format 329

B The Seisan waveform file format 339

C Response file formats used in SEISAN 347

Chapter 1

Introduction

The SEISAN seismic analysis system is a complete set of programs and a simple database for analyzing earthquakes from analog and digital data. With SEISAN it is possible using local and global earthquakes to enter phase readings manually or pick them with a cursor, locate events, edit events, determine spectral parameters, seismic moment, azimuth of arrival from 3-component stations and plot epicenters. The system consists of a set of programs tied to the same database. Using the search programs it is possible to use different criteria to search the database for particular events and work with this subset without extracting the events. Most of the programs can operate both in a conventional way (using a single file with many events), or in a database manner. Additionally, SEISAN contains some integrated research type programs like coda Q, synthetic modeling and a complete system for seismic hazard calculation.

The data is organized in a database like structure using the file system. The smallest basic unit is a file containing original phase readings (arrival times, amplitude, period, azimuth, and apparent velocity) for one event. The name of that file is also the event ID, which is the key to all information about the event in the database. Although the database in reality only consists of a large number of sub-directories and files (all of which the user has access to), the intention is that by using the surrounding software, the user should rarely need to access the files directly, but rather do all work from the user's own directory. Test data and a tutorial (see chapter 5) are supplied with the system.

The programs are mostly written in Fortran, a few in C and almost all source codes is given, so the user should be able to fix bugs and make modifications. The programs have been compiled and linked with system compilers and linkers on SUN, GNU compiler on Linux Windows and MaxOSX. SEISAN runs under Sun Solaris, Linux, MacOSX, Windows95/98/NT/2000/XP/Vista/Windows7. For graphics, X is used on Unix systems and DISLIN (www.dislin.de) used under Windows. No format conversion is needed to move data files (binary and ASCII) between the systems if one of the standard formats (SEISAN, GSE, SEED, SAC ASCII) is used.

This manual resides in the directory INF (see below), when the system has been implemented on your computer. The file is called `seisan.pdf` (Adobe PDF).

The SEISAN system is built of programs made by many different individuals without whom it would never have been possible to make SEISAN. Acknowledgement is made throughout this manual where appropriate or in the acknowledgement section at the end. SEISAN now contains so many programs that when a new version is released, it is not possible to check all the options in all programs and we rely on the user to help finding the bugs, please report!

SEISAN is freely available for all non-commercial use.

In this manual names of computer programs are given with capital letters, names of files and command line options are given by typewriter font.

1.1 Latest changes

Changes in version 9.0.1

- Fixed some minor bugs in version 9.0.
- New options and changes in EEV: IFP: Input munually fault plane solution, INPUTFPS: input complete fault plane solution line, COML: Input geographical location line, COMF: Input felt event comments, U: Update event (was UPDATE, UP: Update event list (was U), M: Input model indicator on header line.
- program FOC now also makes rose diagrams.

Changes in version 9.0

- New compiler and graphics system for Windows: This is the largest change. Windows now uses Gfortran and gcc and the graphics library is DISLIN. This has stabilized the graphics on Windows end enabled to use the same compiler on most platforms. However, the gfortran on Windows has created new problem in some programs most of which probably has been solved.
- SEISAN can now extract and plot data from a BUD and a SEISCOMP archive, read more at page 12.
- EEV has several new options: CM: Copy many files, DD: Duplicate header, FH: HASH fault plane solution, FP: FPFIT fault plane solution, FO: Plot all fault plane solutions, IFP: Input munually fault plane solution, INPUTFPS: input complete fault plane solution line, COML: Input geographical location line, COMF: Input felt event comments.
- FOC: A program to plot and analyze fault plane solutions.
- FPFIT: FPFIT fault plane solution program with SEISAN driver program FPFIT_SEISAN.
- HASH_SEISAN: HASH fault plane solution program.
- GETPDE - A new program that grap the PDE from the USGS web page and add the events in a database, see page 184.
- SEIS2VIEWER - A new program for plotting earthquakes on a map, command smap, see page 153.
- EPIMAP: Can now plot fault plane solutions.
- MULPLT: Has a three component option facilitating working with three component data.
- AVQ: new program to average Q-relations.
- MAG2: new program to invert amplitudes for ML scale.
- Individual size of graphics windows for different programs can now be set in COLOR.DEF.
- The following programs do not work with Gfortran: HYPINV, HYP_ISC, NORHYP, ARCSEI

Changs in Version 8.3

- Manual now written in Latex, html version is available
- Unix setup files were remamed to SEISAN.csh and SEISAN.bash to make them visible
- MULPLT: Plot hour and minute on time axis, show shortcut keys on menu, select picked traces from trace selection, new keyboard shortcuts when reading amplitudes, also shortcut keys for the first 10 traces in trace selection (these changes were put in by Wayne Crawford)
- Two events are now included in the SEISAN software, so that one can plot data after unpacking SEISAN, both events are found by typing `eev 199606 TEST` (Note : in unix one must first source the `COM/.SEISAN` file)
- PINV, new program for estimation of fault plane solution using polarities, see page 232
- 64 bit, have only been tested on Linux
- SEISAN can now use SAC PAZ response as created with rdseed
- code now compiles with gfortran
- AUTOREG has new option for moving waveform file to WAV, see page 171.
- EEV accept SEED orientation codes : A,B,C,1,2,3,U,V,W,S and Z,N,E
- MULPLT orientation code 1 and 2 are read as N and E, respectively. 1 and 2 are used if orientation is different from N and E. (This still requires a more general solution)
- ASCSEI, A bug in in the reading of the input file was found and fixed, the first sample was lost if data was not PSN data.
- ISCNOR : A bug in the reading of surface wave amplitudes was fould and fixed.
- The problem with extracting time windows on the last page in MULPLT continous mode has been fixed, page 100.
- EEV copies the name and path of the current s-file to a file named `eev.cur.sfile`, when the system command `'o'` is used.
- Plot STATION?.HYP and SEISAN polygon files with Google Earth using GMAP, see page 147.
- The Herrmann modeling programs finally work under Windows.
- FOCMEC: Use of amplitudes has been improved and bugs fixed in amplitude section.
- New broadband body wave and surface wave magntudes have been implemented.
- New conversions programs: AHSEI, DIMASSEI, DATABASE2MSEED
- Noise spectra from continuouis data, CONNOI and EVANOI.
- Magnitude implementation has been adjusted to the new IASPEI standard, see table below.

Codes for reporting amplitude readings				
Magnitude	Old SEISAN	ISC	Old IASPEI	NEW IASPEI
Local	AMP AMPL	AML	IAML	IAML
mb		AMB	IAmb	IAmb
mB	AMP AMB AMpb AMb AMPB		IAmB	IVmB_BB
Ms	AMP	AMS	IAMS20	IAMs_20
MS			IAMSBB	IVMs_BB

Version 8.2.1

- Improved handling of continous data bases, particularly with SEED data
- New and expanded training document
- Several bug fixes reported after version 8.2
- Mistakes in manual reported after releases of version 8.2
- More examples of making response files
- Improved filtering in MULPLT by introducing tapering

Version 8.2

- Improved SEED reading and writeing (still not perfect, sometimes problem with Steim2)
- SEED channel naming convention now used
- SAC reading and writing also under windows
- New conversion programs
- Improved WAVETOOL
- SEISEI use all formats for input and SEISAN and MiniSEED for output
- Store waveform data in memory for faster plotting (see MULPLT section, page 97)
- Array processing of teleseismic P- arrival on regional network using plane wave approach, PFIT
- Plot of arrival times using EEV
- Epicenter plotting using GoogleMap or GoogleEarth (program GMAP)
- Spectral analysis also of teleseismic events
- Particle motion plot in MULPLT
- In MULPLT from continuous plot with one channel, it is possible to extract out time windows in a data file, page 100.
- SEISAN has been tested on Vista and there are problems with some graphic programs like LSQ.
- EDRNOR, conversion program for USGS parametric data
- GSERESP2SEED, prorgam to create dataless SEED volumes from GSE response files, using GSE2SEED
- QLg program has new features to test effect of noise and source perturbation
- Instructions on how to use SEISAN under Cygwin have been added

1.2 Information about SEISAN online

SEISAN homepage

The URL address where SEISAN and related software can be found is:
`https://www.uib.no/rg/geodyn/artikler/2010/02/software`

Here you can find information on the latest changes in SEISAN, access the online manual, download the software and much more.

SEISAN anonymous ftp server

Seisan is available from the following ftp server:

`ftp.geo.uib.no` (or `129.177.55.28`)

Login: ftp

Password: <your email address>

The files are stored in the directory `/pub/seismo/SOFTWARE/SEISAN`.

SEISAN mailing lists

There is a mailing list, which is set-up to improve the exchange of information and questions on SEISAN. We strongly recommend that all users subscribe to the SEISAN list. The list is:

`seisan@geo.uib.no`

The purpose of the list is:

To subscribe to the list, send an email to `seisan-join@uib.no` or to `seisan-request@uib.no` with subject/body subscribe.

Subscription can also be done online via `http://mailman.uib.no/listinfo/seisan`. As a member of the list, it is possible to look through the archive (since 2008) for questions and answers on SEISAN. Anyone is welcome to reply to questions and a response to a question should be send to the complete mailing list.

Chapter 2

STRUCTURE OF SEISAN

2.1 Directories

The whole SEISAN system is located in subdirectories residing under the main directory SEISMO. For more details, see chapter 3 on installation. The system contains the following main subdirectories:

REA:	Earthquake readings and full epicenter solutions in a database
WOR:	The users work directory, initially empty
TMP	Temporal storage of files, initially empty
PRO:	Programs, source code and executables
LIB:	Libraries and subroutines
INC:	Include files for programs and subroutines in PRO and LIB
COM:	Command procedures
DAT:	Default and parameter files, e.g. station coordinates
WAV:	Digital waveform data files
CAL:	System calibration files
INF:	Documentation and information
ISO:	Macroseismic information
SUP:	Supplementary files and programs

In the following, the above subdirectories will mostly be called directories to avoid always referring to SEISMO. All directories use capital letters, however this only makes a difference in the Unix versions. The directory structure is used as a tree like structure for quick access to individual files in the REA directory, which therefore will appear as a simple database to the user. The next section is a description of the database directories; the other directories are described in chapter 7. Figure 2.1 shows the tree structure of SEISAN.

2.2 The database

The database of SEISAN consists of the two directories REA and WAV. The REA directory and its subdirectories contain readings and source information while all waveform data is normally in the directory WAV (see 2.2.2) with no subdirectories. Optionally WAV can also be divided into a similar subdirectory structure, see 2.2.2, which is useful when storing continuous data in particular. The DELET database

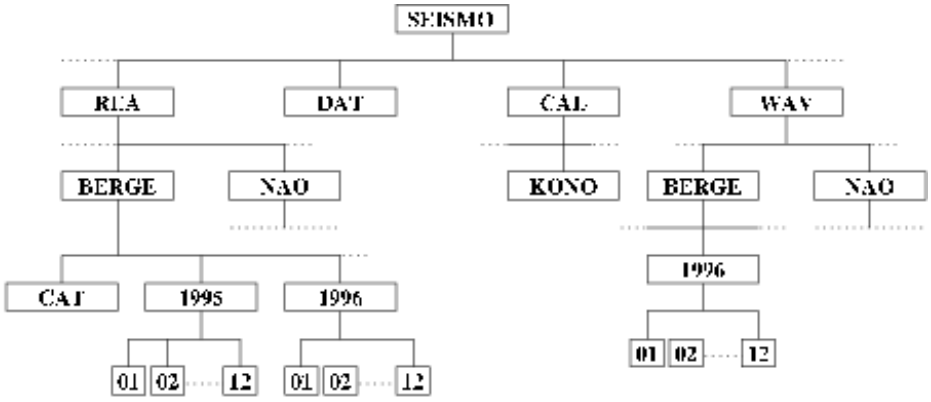


Figure 2.1: Structure of SEISAN. Note that BERGE under WAV is optional and DELET (not shown) under REA has a similar directory structure as e.g. NAO.

contains all events deleted from any of the databases (here BERGE/BER and NAO). Filenames are identical between all platforms.

2.2.1 Phase data and hypocenters

The REA directory contains phase readings and derived source information like hypocenters, fault plane solutions etc. The REA directory has one or several subdirectories corresponding to separate databases (see Figure 2.1 for an example with two databases). The database names can have between 3 and 5 characters. If less than 5 characters are used, the character '_' is added in the file system to make it 5. The user does not have to put the '_' when running a program, they will be added by the software. If a directory is made manually, the '_' must be put in. It is assumed that a database is always present in the system. The name of the default database is given by an environmental variable (see section 3.1), however if not set, it will default to AGA for agency. Here, BER will be used as an example throughout the manual. A database has a duplicate storage of the events. For quick reference and interactive work the events are stored in single files (S-files) in yearly directories and monthly subdirectories. When new data is entered into the database, it comes in as individual event files. However, once the interactive work has finished, the single event files are overwritten with the final location and additionally stored in monthly files, which are only changed when updating (UPDATE command, see section 6.7). The monthly files, called CAT-files for catalog, are stored separately in the CAT directory and primarily used for quick searching and backup for the single files. In addition to the event data, there is also a LOG directory in each database to keep a log of the data processing, see section 6.7.

S-file database structure

The structure for the single file storage is as follows (Windows example):

\REA\BER__\	Main readings directory, all data
\REA\BER__\1999\	Data for 1999
\REA\BER__\1999\01\	Data for January 1999, each event in one file

On Unix, the last line would have been /REA/BER__/1999/01

Each event contains original phase readings in the Nordic format (Appendix A.) which includes file names of all corresponding waveform files. One event is one file. Each event has an ID line. The ID line contains a unique ID, which will follow the event through all COLLECT and SPLIT operations (see section 6.5 and 6.6). The ID line also contains status information about the event like last action, when it was updated etc. The ID-number can be fixed, which is useful if data is taken out from the database, processed on another computer and later put back into the database, since otherwise the ID of an event might be changed and the existing file would not be overwritten. An example of an S-file name is :
`27-1112-11L.S199401`

The S-files are used as input for the location program and, when making a permanent update, also for output, see 6.1. The letter in front of the "." indicates the event type and can be L, R or D for local, regional or distant event respectively. It is the same indicator as given in the header line of the S-file, see the Nordic format page 329. The remaining numbers give (in order) day, hr, min, sec, year and month.

As mentioned above, the system can contain many other databases, which may function exactly like the BER directory. A data base can be used to store a subset of data or data from different networks. Data can be moved between databases or in and out of the databases, for details, see description on EEV (4.4 and 4.5).

Monthly location files, the CAT directory

Events located in monthly files are in a directory called /SEISMO/REA/BER__/CAT in addition to the individual S-files. Additional databases like e.g. NAO will have epicenters stored under /SEISMO/REA/NAO__/CAT. The monthly epicenter files are called texttt199901.CAT for e.g. January 1999. Although the files generated by SEISAN normally are monthly files, the CAT directory can also contain yearly files or any other time interval. The only rule is that the name of the file must give the year and month of the first event in the file. This is because the search program SELECT uses the file names to search requested time intervals. If a user has a historical catalog, this can be added as an individual file. If the historical catalog starts in 1820, the file name would be `182001.CAT`. The files in CAT do not need to be continuous in time, but they must not have overlaps in time and each file must have data in chronological order. The format of the CAT files is the same as for the S-files. Additionally, CAT files can also be compact files, meaning just the header lines of the S-files (see also section 2.3).

2.2.2 Waveform data and formats

SEISAN works with various waveform formats including SEISAN, GSE, SEED/MINISEED and SAC binary and SAC ASCII. The SEISAN format is described in Appendix B, while for a format description of GSE and SAC the user is referred to GSETT-3 [1997] and Goldstein [1999], respectively. The SEED format is described in IRIS Consortium [1993]. The GSE reading routines are based on the codeco routines written by Urs Kradolfer, Klaus Stammler and Karl Koch. The different formats can be used in parallel by several programs. With MULPLT for example it is possible to plot data in the four formats at the same time. Other formats can be added by adding reading routines and adding the respective calls to `LIB/wave.for` . Note that SAC binary files can also be used on Windows from SEISAN version 8.2. To use other formats, a conversion program must be used first, see section 6.12.

In general it is recommended to keep the waveform data in one format only, mainly for simplicity and maintenance reasons. There may be different arguments for or against one or the other format depending on the user's preferences and requirements. SAC and GSE are widely used formats and therefore may be attractive. SEISAN is a multi-trace binary format with direct read access to individual traces. The SEISAN format is probably your best choice if your main processing system is SEISAN and because it is easily used on all computer platforms. SAC is a single trace binary or ASCII format with a large number of header parameters. The SAC format is widely used in research-oriented programs. GSE is a multi-trace ASCII waveform format that includes various sub-formats. It is widely used for data exchange. Although the GSE format can keep any number of traces, it is recommended to include no more than 3 traces in a single file depending on the number of samples, since when reading a particular trace, the whole file may have to be read.

For the future, the SEED/MINISEED format might be the best option since most data centers use it. However, the SEISAN implementation should probably be tested a bit more. SEISAN cannot read SEED files using all options possible in SEED, but data from the largest data centers as well as many observatories have been used for testing. With respect to MINISEED, there are probably less problems since MINISEED is simpler than SEED. SEISAN can also write MININSEED (program WAVETOOL), but cannot write SEED (unless GSE2SEED is used). The WAV directory contains files with digital waveform data. The directory normally has no subdirectories or any other organization. However, in case of large databases, WAV can be subdivided, see below. In addition any directory can contain waveform data, it has to be specified in SEISAN.DEF (section 3.10). The amount of data that can be stored is only limited by the disk size. The analysis system will always look in WAV for particular files if they are not in the user's own directory. Waveform files will automatically be transferred to WAV on initial registration into the database (see MULPLT). Registration is the process of automatically creating an S-file in the database with the name of the waveform file and header information. Phase pickings are done later. See section 6.2.

There is normally no requirement for particular filenames for the waveform files in WAV or elsewhere, however many programs will make file names like:

`yyyy-mm-dd-hhmm-ssT.NETWO_nnn` e.g. `1995-01-23-1230-20T.BERGE_013`

With the abbreviations yyyy: year, mm: month, dd: day, hh: hour, mm: minute, ss: second, T: file type indicator (normally S), NETWO: maximum 5 letter network code and nnn: number of channels.

Recommended file type indicators are: S: Standard SEISAN, R: Resampled, A: Appended, M: Miniseed/SEED

WAV database: In case a large number of waveform data is stored, it might be an advantage to also split up the WAV directory in subdirectories. This is done in the same way as in the REA directory, e.g. waveform files for BER from July 1994 would be found in `WAV/BER__/1994/07`. Programs that use waveform files will automatically search, in order, the current directory, TMP, WAV and the monthly WAV directory. How it is a requirement for all programs running outside EEV that the waveform data is in the default data base since only that one is searced. **When storing in the WAV database, it is a requirement that the waveform names start with either yymm (like 9902) ,yyyymmdd (like 19990101) or yyyy-mm (like 1999-02).**

Waveform files created on Windows and Linux SEISAN version 7 or newer cannot be read on older SEISAN versions.

The SEISAN binary waveform format is explained in Appendix B. The files are written and read with the same Fortran statements on all platforms, however the internal structure and byte order are different. As of SEISAN version 5.1, files written on either machine can be read on the other and there is no need for any conversion when the binary waveform files are moved between Sun, Linux, MaxOSX and Windows.

Compression of waveform data

Waveform files can be stored in compressed format. The compression must be done by the user. Programs that access the compressed waveform files copy the file to the TMP directory, and uncompress there. The uncompressed file remains afterwards and will be found the next time one of the programs is looking for the same waveform file. The content of the TMP directory has to be deleted manually. On Unix, you may automatically delete the content of the TMP directory by a cronjob, see manual pages on crontab. On Unix the compression formats supported include gzip, compress, bzip2 and zip. So far, no automatic decompression is supported on Windows (will be put in). With the introduction of SEED format, there is less need for external compression since the SEED data usually is compressed and therefore decompressed on the fly when read.

Component codes

The SEISAN waveform format until version 8.2 has used 4 characters for the component code. The first character indicates the type of sensor, for example 'B' for broadband, 'S' for short-period or 'L' for longperiod. For acceleration data the first character has to be 'A' because SEISAN assumes that the corresponding response has been given as acceleration response. The fourth character has to give the channel orientation, 'Z' is used for vertical, 'E' for east-west and 'N' for north-south. Other orientation of the horizontal components is possible in GSE, SEED and SEISAN are not understood by SEISAN. If data are rotated, 'T' is used for transverse and 'R' for radial. The second and third characters can be chosen by the user. From SEISAN version 8.2, only 3 characters are used, the first 2 and the last. These 3 characters are then defined according to the SEED standard. SEED location codes and network codes are now also stored in the SEISAN format and are displayed when plotting the traces with MULPLT. No other programs, except some conversion programs, use network and location codes The component code is part of the response filename and is used to find the response corresponding to a given station and component. The network code is not part of the response files (except for SEED format) and not used,

so it is up to the user to put in the correct response which cannot be the same for two location codes at the same site. Program WAVFIX can be used to change station and/or component codes as written in SEISAN format files, but will not handle location or network codes.

The Nordic format only has space for two characters for the component code. The definition in SEISAN is that these are the first and fourth character of the waveform component code. This means that the relation between the component code in the Nordic file and the waveform data is non-unique.

The GSE and SEED waveform formats have three characters for the channel code, see GSETT-3 [1997] and IRIS Consortium [1993] for the detailed definition of the component codes. SEISAN, when reading waveform data in either GSE or SEED format internally keeps the first two characters and moves the third to fourth, so for example 'BHZ' becomes 'BH Z', however the user will only see the name as BH. Data files in SEED also have a location code, which allows to distinguish for example between two 'BHZ' components (for example a 30 second and 120 second sensor with the same sampling rate and high gain) at the same site. Z. When converting between SEISAN and SEED/MiniSEED, station, network and location codes are preserved while SAC and GSE only partly can store this information.. SAC has more than four characters for the component code and sacsei.def has to be used to define the conversion. However, normally SAC data will have three character component codes as well. Conversion of component codes from SEISAN to SAC is also defined in sacsei.def.

When converting between SEISAN and other waveform formats, component conversion is defined in the respective definition files, see section on conversion programs.

2.2.3 Continuous waveform data

In SEISAN one can plot or extract continous data from either a standard SEISAN database or from a BUD or a SeiscomP archive.

Continous data in a BUD or SeiscomP archive

SEISAN reading BUD and SeisComp archives
We are working on implementing archive reading in SEISAN and a first version is now being tested. The reading routines using Chad Trabant software have been implemented by Ruben Luis. Reading continuous data:
This works just like reading SEISAN continuous data, except there are no S-files, only the archive files. All the same functions are available:
Plotting, zooming and extracting segments and registering events.
Read archive data as an event from eev:
A reference to a segment is made in the s-file and it is treated as if it was a file. When a keyword for archive type (BUD or SCP) is found, the reading is directed to the archive instead of to a file. The archive reference is e.g.

```
ARC STAT  COM NT LO YYYY MMDD HHMM SS   DUR
BUD ROSA  BHZ PM    2010 1011 0100 00 14400                                  6
```

where ARC is archive type, STAT is station code, COM is component, LO is location code YYYY MMDD HHMM SS is start time and DUR is duration in secs.
Thus the segment in archive with given start time and duration is considered a file. If later plots require less data than the segment referenced, the whole segment is still read, like reading the whole trace in a file in archive with given start time and duration. A mixture of archive references and file names can be used.
The archive is defined in SEISAN.DEF as

```
ARC_CHAN                                PMOZ BHNPM
ARC_CHAN                                PMOZ BHEPM
ARC_ARCHIVE                             /uibsan/home/s2000/BUDARC
```

where each channel is defined as well as the location of the archive. The specification is the same for both BUD and SeisComp archives, and only when reading S-files or starting mulplt will the information about archive types be given. ONLY one archive type can be used at the same time when, maybe something to change.
Current it has only been tested on Linux but the plan is to also make it work on Windows.

Continous data in a SEISAN database

In SEISAN continuous data has no special format. Continuous data is simply ordinary waveform files that follow each other in time. In order to treat the data as continuous, the data can be put into a SEISAN continuous data base. Such a data base is made as follows:

- For each waveform file from a station or network, an S-file is created. The S-files only contain reference to the waveform file(s). Program AUTOREG can be used to create the S-files.
- The waveform data is optionally put into the corresponding waveform station directories, however they can also be in WAV or working directory. For large data sets it is strongly recommended to use the WAV database structure.
- The continuous databases are defined in SEISAN.DEF in DAT.

If e.g. data is to be stored from 3 different stations (three componet files), create 3 databases under WAV and REA with the name of the stations (program MAKEREA). If the continuous data consist of 20-minute files, this would mean about 2200 files pr month, which is a reasonable number. It is now possible for some programs (MULPLT, WAVETOOL) to get access to any or all of the traces in the continuous data base and plot and extract data. If the continuous data is archived from a real-time system it is best to have one database per station as it will at times be necessary to backfill gaps as data may not have arrived in real-time.

It is also possible to store the data without having a database for each station:

- Alternative 1: If the 3 stations have waveform files starting at about the same time and the same duration, they can be merged to 9 channel files and only one continuous data base is made. This may work well for data from a temporary deploymeny where all data is there when the data is put into database.
- Alternative 2: If the 3 stations have waveform files starting at about the same time and the same duration, the 3 waveform files can be listed in the S-file and only one data base is needed.
- Alternative 3: If the files are in individual channel files, 9 waveform files can be listed in the S-file and only one continuous database is needed.

The waveform files in a continuous data base can have different formats for different stations and one S-file can refer to more than one waveform file, provided they start at about the same time and have the same duration.

A simpler way to use smaller quantities of continuous data is to make a list of these files with DIRF and an application program can then use that list to work with the data. Currently two programs have

special options for this kind of continuous data. The MULPLT program will plot data from several files as if it was one file in one continues trace the RESAMP program will resample the data from several files and put it into one output file.

2.3 File types used with SEISAN

A description of the different file types is given below with typical names. Most names must be exactly as specified, others can be given names. However it is VERY important that no name including full path is more than 80 characters long. Until now this has not been a problem, however it has to be considered when SEISAN is installed.

The basic unit is a file in the Nordic format, (see Appendix A). For practical purposes 3 descriptive names are used for Nordic files:

S-file: Single event file with phase readings, with or without source parameters such as location and magnitude. In the database these files are named with the extension: .Syyymm This is the standard type of file in e.g. the `BER__/1998/08/`. An example is `11-1234-11L.S199808`.

CAT-file: A catalog file containing many S-files with location or just a catalog of hypocenters, a compact file, see below. This is the standard type of file in e.g. the `/REA/BER/CAT` directory. An example is `199801.CAT`. This file format is also output from several programs like SELECT and COLLECT. There is a blank line between events.

Compact file: This is a CAT-file with only the source information. One event is represented with one line, (the header line in the S-file). There is no blank line between events. A compact file can be generated by either COLLECT or NORHEAD (earlier called COMPACT).

In addition there are the following types of files:

SEISAN waveform file: Waveform data can be stored in SEISAN, GSE and SAC format, see section 2.2.2. An example of a name is `1992-01-11-2233-22S.BERGE_011`.

Response file: File giving the response of a given channel at a given station. They are typically generated with the RESP program, see description of CAL directory, section 6.32. This is the standard type of file in the CAL directory. An example of a name is `ODDA_S__Z. 1999-05-01-0000_SEI`. However, SEED and SAC response files extracted with rdseed can be used.

File listing: This is just a file with a list of numbered files. The file name is always `filenr.lis`, and it is generated with the DIRF program, see 6.8.

Index file: This file contains a listing of absolute paths to a series of S-files. The index file can be used as input instead of the CAT-files to several programs. Several programs generate index files as e.g. select and eev. The index file has the same format as the `filenr.lis` files described above and can be generated with the dirf command using S-files. The index file name must contain a '.'. An example is shown below:

1. \SEISMO\REA\TEST_\1993\09\29-2228-26D.S199309
2. \SEISMO\REA\TEST_\1994\06\16-1841-57D.S199406
3. \SEISMO\REA\TEST_\1996\06\03-1955-40D.S199606

2.4 Upper and lower case

Upper and lower case file names only makes a difference on SUN, Linux and MacOSX. The intention is that all permanent data file names used by SEISAN should be in upper case (e.g. S-files, crustal model file, directories (e.g. REA) while temporary files should be in lower case (e.g. print.out). Programs are also in lower case. It should then be a bit more difficult to delete the permanent files. NOTE THAT THROUGHOUT THIS MANUAL, PROGRAM NAMES ARE GIVEN IN UPPER CASE TO INDICATE THAT THEY ARE NAMES, HOWEVER WHEN USING THE PROGRAMS, LOWER CASE MUST BE USED ON SUN. In program MULPLT, commands are case dependent.

2.5 Moving data between Sun, Linux, MacOSX and Windows

All S-files and file names are identical on the three platforms. To move many events (S-files) from one system to another, make a COLLECT (section 6.5) on the original system and a SPLIT (section 6.6) on the receiving system. As mentioned in section 2.3, the SEISAN binary waveform files have different internal structure if written on Sun, Linux, MacOSX or Windows, but this is corrected for in the reading routine, so files can be copied directly. GSE files can be copied directly since they are ASCII files. SAC binary files are different between Linux and Solaris; SEISAN can only read files that were written on the same platform. SEED/MINISEED files can be used directly on all platforms.

The only other files that are different are the binary earth model files `IASP91_`*`platform`*`.HED` and `IASP91_`*`platform`*`.TBL` in the DAT directory (where platform is either sun, linux, macosx or windows). The platform is included in the filename so that possibly SEISAN can be used on different platforms with only one file system. For example the data may be kept on a Sun file system, but you also want to share the disks from a Windows system and process the data using the Windows version of Seisan. Otherwise, the files cannot be moved, but are easily regenerated with the IASP91 programs, see section 6.20 and 8.8 in the Hypocenter manual.

Chapter 3

INSTALLATION

SEISAN has been tested and compiled for Windows 2000/XP, Solaris, Redhat Linux and MacOSX.

Upgrade from version 7.0 or higher

Before you start, take a backup copy of your DAT directory. Note that when you upgrade, many parameter files will be overwritten so make sure old parameter files are copied before putting in a new version of SEISAN. The most important are in DAT: `STATION0.HYP, SEISAN.DEF, MULPLT.DEF`. Also the Unix setup file SEISAN.csh and SEISAN.bash is overwritten. You may also want to keep copies of PRO, LIB and INC to keep a copy of the old source code, especially if you have done any modifications to the code. You can keep almost all of your parameter files, only `SEISAN.DEF` has been changed. Check this file and change to your system. Some individual program parameter files like for SPEC have changed.

How to get SEISAN

SEISAN can be copied from `ftp.geo.uib.no` (`129.177.55.4`), login is ftp and password is your email address or from `https://www.uib.no/rg/geodyn/artikler/2010/02/software` On the AFTP server go to `/pub/seismo/SOFTWARE/SEISAN`. Use binary mode for the compressed files (tar and zip). Before copying, check the readme file for latest updates, changes and current content of the directory. The directory will at least contain the following files:

`seisan_X.Y_.unix.tar.gz`	a compressed tar file, whole distribution with executables and test data, X.Y stands for the latest distribution number and Unix for the respective Unix system (solaris or linux).
`seisan._X.Y.exe`	Windows distribution an install file
`seisan_X.Y.pdf`	The SEISAN manual, Adobe PDF
`seitrain_X_Y.pdf`	SEISAN training course
`testdata_X.Y.tar.gz`	SEISAN data for the training course

Alternatively SEISAN might be obtained on a CD with the same content as above (write to `jens@geo.uib.no`).

Section 3.8 gives additional information about modifications and recompilation.

3.1 Unix (SOLARIS and Linux)

Solaris: The SEISAN programs have been compiled on Solaris 7 using Sun Workshop 5, which means you have to recompile if you use an earlier version of the operating system or compiler. If you can recompile on Solaris, please do so! The programs on Solaris are compiled dynamically, which means not all system and compiler libraries are included in the executables. If you are running Solaris, the system libraries are normally installed, but the Sun system compilers might not be installed. If the compilers are not installed, you have the following options: (1) you install the Sun workshop compilers, license is not needed, since only the libraries are required; (2) you install the required libraries, which are part of the Solaris SEISAN distribution (instructions below).

Linux: The programs have been compiled under Redhat Linux7.2 using the GNU compilers gcc and g77. It is recommended to recompile the programs, since otherwise the programs might not run on your Linux distribution. In the Redhat distribution of Linux the Fortran compiler is not part of the standard distribution, it has to be installed (see your Linux manual for instructions). THE USER ACCOUNT MUST BE SET UP TO USE csh, tcsh (use SEISAN.csh) or bash (use SEISAN.bash), in order for the SEISAN scripts to work. Note that in the following SEISAN.csh stands for both SEISAN.csh and SEISAN.bash. Otherwise the scripts need to be adopted to the shell used.

Instructions

The first step is to install the distribution, the procedure is the same for all Unix platforms.

1. Get tar file

Copy the distribution file for your platform from CD or transfer it through FTP or from the web site to the SEISAN top directory, this could be a directory `seismo` under the home directory.

2. Decompress

`gunzip seisan_`*`version`*`_`*`system`*`.tar.gz`

There should now be the uncompressed file in your directory (without `.gz`).

3. Install SEISAN

```
tar xvf seisan.tar
```

Check that the SEISAN directories have been created.

If SEISAN has been installed without executable files, they can all be generated with the command 'make all' from the PRO directory. On Sun this requires that the Sun compilers be installed, on Linux/MacOSX it requires the GNU Fortran compilers (g77, only gcc before version 4.0; now gfortran). See also section on compilation (3.8).

Install Workshop libraries

In the SUP directory of the Solaris distribution the file `sun_ws_lib.tar.Z` includes the libraries that are needed to run SEISAN on Solaris in case the compilers are not installed. The file is a compressed tar file. The files can be extracted with `uncompress sun_ws_lib.tar.Z` and then `tar xvf sun_ws_lib.tar`. The library files can be stored in any directory in the system, but the environmental variable `LD_LIBRARY_PATH` has to be set accordingly. If you are using the C-shell, this can be done by adding to the `.cshrc` file the line `setenv LD_LIBRARY_PATH /path/:$LD_LIBRARY_PATH`. This would add `/path/` (which is the path to where the libraries are) to `LD_LIBRARY_PATH`, which normally is already defined.

4. Set system parameters

If you are doing an update, some of the following settings can be skipped.

Activate SEISAN:
csh/tcsh shell :
In your .cshrc file, the aliases and paths used by SEISAN are defined by adding the line
`source ../seismo/COM/SEISAN.csh`
where `../seismo` is the directory below which SEISAN has been installed. The `SEISAN.csh` script file assumes that you are running either csh or tcsh as your shell.
bash shell :
If you are using the bash shell add this line to your `.bashrc` file :
`. ../seismo/COM/SEISAN.bash`
bash might include a `select` program, if that is the case on your pc you also need to add this line in your texttt.bashrc file :
`alias select="/home/seismo/PRO/select"`
to use the SEISAN SELECT program.
If you are using another shell you need to modify the script accordingly or change the shell. It is assumed that X-windows is installed.

SEISAN path for programs:
In order for programs and subroutines to know the path to the SEISAN program directory, this must be defined in the file .SEISAN in COM. Edit that file and set the environmental variable SEISAN_TOP to the name of the top directory, meaning the directory structure below and including seismo e.g. /top/users/seismo. This variable is then used to set the path to SEISAN directories.

Search path for libraries:
To run the NANSEI conversion program under Solaris, the SEISAN LIB directory needs to be included in the environmental variable LD_LIBRARY_PATH. The LIB directory as default is already added to the library search path in the SEISAN.csh file.

SEISAN path for databases, parameter files etc:
The SEISAN database can be under the same top directory as programs, however it can also be different. This is practical if several users have their own databases, but use the same software. Set environmental variable SEISAN_TOP to top directory e.g. /top/users/seismo.

Java settings:
Aliases are defined to run the Java tools jseisan, seisconf and sformat. The setup in the SEISAN.csh file as default is:

```
alias jseisan 'java -DSEISAN\_TOP=\$SEISAN\_TOP -classpath \$SEISAN\_TOP/PRO/jseisan.jar SEISAN'
alias seisconf 'java -DSEISAN\_TOP=\$SEISAN\_TOP -classpath \$SEISAN\_TOP/PRO/jseisan.jar SEISCONF'
alias sformat 'java -DSEISAN\_TOP=\$SEISAN\_TOP -classpath \$SEISAN\_TOP/PRO/sformat.jar Sformat'
```

These settings may have to be changed in case you want to keep the jar files in some other directory. The CLASSPATH (which is used to search for classes) is also modified to include PRO and PRO/jseisan.jar, although this is not needed when using the aliases.

SEISAN agency:
In SEISAN.csh also set the environmental variable AGENCY (upper case) to your 3-letter agency code (upper case). This variable is only used by program MACROIN from EEV in connection with entering macroseismic data so for most users ignore this setting.

SEISAN default database:
To locate the default database directory (here BER) set environmental variable DEF_BASE in SEISAN.csh. If not set, the name AGA is used. The data bases are found under SEISAN_TOP.

SEISAN editor used in EEV:

The default editor is vi, any other editor can be set with the environmental variable SEISAN_EDITOR.

SEISAN calibration file directory:
By default, calibration files are in CAL, but they can be in a directory set with variable LOCAL_CAL. The directory name must be complete like /home/users/calibration/

SEISARCH
Gives the architecture, can be either solaris, linux or windows. Used in Makefile when compiling.

SACAUX
Path to SAC aux directory, required by the SAC routines for reading and writing, although not really used.

SACLIB
Specify path and filename to SAC libraries, only needed when you compile programs (Unix) and you have the libraries installed on your system.

Printer for Postscript plots:
The hard copy files from programs are sent to the printer from within the programs using the standard lpr command. In the SEISAN.csh file, define lpr using the standard environmental variable PRINTER. Remember that the printer must accept Postscript. PostScript files can also be viewed and printed on most printers outside SEISAN using GhostView, however in that cases files cannot be printed from within a program.

Scaling for Postscript plots:
By default, plots will be in A4 size. This can be changed by setting the environmental variables SEISAN_PSSCALE_X and SEISAN_PSSCALE_Y. The default for A4 size is 1.0 for both variables. For Letter size the Y-scaling can be set to 0.9.

Seisan Extension:
User specific code can be implemented by making use of the environmental variable SEISAN_EXTENSION. The idea is that programs read this variable, if set to the user specific string, the user's source code will be used instead of the default. An example could be the computation of error ellipses. Currently used codes are: BGS.

4. Testdata The testdata set can be extracted from the file `testdata_X.Y.tar.Z`. Use programs uncompress and tar to extract the data in the SEISAN top directory (keep subdirectory structure).

Dimensions

Most dimensions are set in file `seidim.inc` in the INC directory. In order to change dimensions, first change in the include file and then recompile the whole SEISAN distribution. The most important dimensions are:

Number of points in one trace	1 900 000 (1 200 000 on PC)
Number of points in memory buffer	30 000 000 (8 000 000 on PC)
Number of lines in NORDIC format file	4 000
Maximum number of traces in one plot	1 000
Maximum number of events in one month	90 000
Maximum number of calibration files	1 500
Maximum number of epicenters in epimap	90 000
Maximum number of lines in index file made with dirf:	9 999

The maximum number of points used by the SEED reading routine are set in `seed_internal.in.f`. Currently its is set to 1 900 000.

SEISAN has been tested with much larger dimensions, like 10 000 000 for number of points in one trace, however large dimensions might slow down the speed due to swapping (particularly if memory is not large) so a smaller dimension has been chosen. For continuous data, SEISAN works with many files so smaller dimensions can be used. For the PC version, dimensions may be different from above, check `seidim.inc`.

Note: In case programs don't work, you might have to recompile, see section 3.8.
Some Ubuntu users are missing the `libg2c.so.0` library file, it can be installed with the command (you might need to be online):
`sudo apt-get install gcc`
If this does not work, also try:
`sudo apt-get install libg2c0`

On at 64 bit computer the IASP files in DAT must be regenerated if you have the files from a 32 bit computer, with the programs `REMODL` and `SETBRN` otherwise `HYP` will crash.

Graphics problem: On Solaris, if no colors, make sure color setting is 8 bit. Can be set with command m64config -depth 8. See Solaris manual.

3.2 MacOSX

The MacOSX version does not come pre-compiled, and will have to be compiled by the user. The installation is basically the same as for Solaris/Linux, but compilation needs to be done, see section 3.8. in SEISAN.bash or .SEISAN.csh, set SEISARCH to 'macosx' (Intel-based=newer Macs) or 'macosxppc' (PowerPC based).

You may also need to change the line $(fc) seed.for to $(fc) -fno-range-check seed.for in LIB/Makefile

If you have gcc/gfortran 4+ installed (see below) and your Mac is Intel-based, you should now be able to compile. You also need X-windows, which should be preinstalled or on the installation disk for OSX 10.5 and higher (for earlier versions, they can be downloaded and installed).

Additional hints on MacOSX 10

In order to compile and link Seisan off the source distribution, you need to have gcc/gfortran installed. The simplest way to do this is to install the Apple Developer Tools. These come on an extra CD together with the OS. If you dont have access to that extra CD, DevTools can be downloaded from Apple (its a fairly major download, around 600 MB), but you need to go through a registration process before. The steps are:

1. Go to `http://developer.apple.com`

2. Click the Join Now link in the Partner with Apple section and follow the instructions to join Apple Developer Connection.

3. Once your Apple Developer Membership is confirmed, log in and click the Download Software link.The Mac OS X 10.0 Developer Tools are available for download there.

3.3 Cygwin

As alternative to running SEISAN directly under Windows, it can also be compiled and used under Cygwin. The cygwin website `http://www.cygwin.com` explains:

"Cygwin is a Linux-like environment for Windows. It consists of two parts:

- A DLL (cygwin1.dll) which acts as a Linux API emulation layer providing substantial Linux API functionality.
- A collection of tools which provide Linux look and feel. The Cygwin DLL currently works with all recent, commercially released x86 32 bit and 64 bit versions of Windows, with the exception of Windows CE."

It is attractive to use SEISAN with Cygwin as it looks like the Unix version. Under Cygwin SEISAN uses X graphics, which requires the Cygwin X server (part of Cygwin). This is likely to work better than the native Windows graphics, which has seen some problems with recent versions of Windows. The compilation under Cygwin is like Solaris and Linux, which means on the software side there are fewer differences to take care of.

Cygwin can be downloaded and installed from the website. However, to be able to compile SEISAN a number of non-default packages have to be included:

- Devel: gcc-g++ (and associated packages)
- Devel: g77
- Devel: make
- Libs: libncurses-devel
- Shells: tcsh
- X11: xorg-x11-base
- X11: xorg-x11-deve

We also recommend to install:

- Editors: vim
- Utils: diffutils

The choice of packages is done through the Cygwin installation tool. With some of the packages, additional required packages will be selected by default and must be installed.

To install SEISAN under Cygwin, use the Linux distribution and set SEISARCH to 'linux'. You have to compile as described for Linux below.

Under Cygwin, you can use csh or tcsh, which will allow to use the SEISAN.csh. If your default is bash, '(t)csh' can be started from the command line. The X server is started using command 'startx' from the Cygwin prompt. See the Cygwin documentation for more details.

3.4 Windows (95, 98, 2000, NT, XP and Vista)

It is assumed that you transferred the Windows distribution, copied from CD or decompressing directly from CD. In the following, it is assumed that you install on disk drive C. The windows graphics work with any resolution, small fonts should be used, however, SEISAN is designed to work with 1024x768. This version of SEISAN has not been tested on 95, 98, 2000 or NT. Vista has problems with some graphics programs.

Using install script

Click on seisan_8.2.1.exe and follow the instructions, use all defaults. If you later want to change some environmental variables, see instructions below. If SEISAN already is installed, a window comes up for a possible upgrade or removal of SEISAN. A new version cannot be installed witout removing the old one. SEISAN is now completely installed. Optionally, some parmeters can be set, se below.

OPTIONAL: Change environmental variables SEISAN_EDITOR, SEISAN_PSSCALE_X, SEISAN_PSSCALE_Y and DEF_BASE (see UNIX section for definition) using control panel - system - advanced and select environmental variables. The defaults are respectively
SEISMO, notepad, A4 and AGA. SEISAN_TOP would be set as `set SEISAN_TOP=\SEISMO`, but could also be e.g. `test\best\analysis` or `d:\seisan`. Note there is one blank character at the end and the first character MUST be "\" or the second ":". If no seismo directory, it could e.g. be just E:. The Unix variable AGENCY is not used on the PC.

Optional: If you want to recompile, you need the Fortran compilers installed.

Printer:
It is assumed that a Postscript printer is connected to the PC, either directly or by a network. When a program sends a plot to the printer, it issues the command SEISANPR. In the COM directory, there is a file called SEISANPR.BAT containing the print command. Several suggestions are made in that file, the default is to use `COPY filename PRN`. If you do not have a PostScript printer, it is possible to view and print PostScript files with GhostViev, a public domain software available from many sites, e.g.

`http://www.seas.ucla.edu/~ee5cta/ghostView/`
`http://www.cs.wisc.edu/~ghost/gsview/get47.htm`
`http://www.geo.uib.no/Seismologi/SOFTWARE/`

Testdata:
The testdata set can be extracted from the file seisan_test_data.tar.gz with for example the WinZIP program. Extract the data to the SEISAN top directory. To use the test data base as the default DEF_BASE must be set to TEST (done by default during installation).

Installing Windows SEISAN interface

The program is called SEISAN.EXE and located in the PRO directory. To install it on the desktop, do the following steps:

1. Right mouse click with cursor on the desktop
2. Select New/Shortcut
3. On the command line box, type `c:\seismo\pro\seisan.exe`, press Next
4. Type a name for the shortcut, e.g. SEISAN
5. Press finish button

In this installation example, SEISAN is installed under c:\seismo.

During installation, a file seisan.ini is created in the [c:\] directory (if, it does not exist) and thereby the configuration variables are set from this file. If you want to work with different configuration, the file has to be changed. In particular if you want to work with a data base on a different disk drive.

Example of seisan.ini file :

```
[Seisan]
SeisanDirectory=c:\seismo\
WorkDirectory=c:\seismo\work\
ProgDirectory=c:\SEISMO\PRO\
DatabaseDirectory=c:\seismo\REA\

[Database]
LastDatabase=CUB2\_
LastYear=1998
LastMonth=04
IndexFile=[None]
```

Running SEISAN on a PC with data on a Unix system, or the other way around.

With SAMBA, a disk can be mounted between different platforms, for example a Unix file system under Windows. The SEISAN_TOP must then be set to the name of the Unix disk a seen on the PC. All files can be read directly from the Unix disk, however the files IASP91_platform.HED and IASP91_platform.TBL must be present. SEISAN works out which of these files to use. With SAMBA, PC users can then access a Unix SEISAN data base directly using the programs on the PC.

Running SEISAN on networked PC's

SEISAN on one PC can be accessed from another PC. This can be an advantage if several users share the same data base. On the client PC, name the server PC disk where seisan is installed e.g. Z:. Then `set seisan_top=z:\.`

Potential problems

Program takes a long time to load: If the program is large, it might use disk swap files, which can take a lot of time. Reduce array size (seidim.inc in INC) and recompile.

Commands like P or L do not work in EEV. If swapping takes place, some damaged swap files or other files might remain which prevents the program from starting. Clean disk with command SCANDISK.

Digital Fortran seems to be very sensitive to viruses. If you get memory problems or programs will not run, particularly under EEV, check for virus.

Graphics problem: Graphics programs write text to a console window. If there is a program crash, the error message goes to the console window, which then disappears.

3.5 Database directories for your own system, MAKEREA

When you want to use SEISAN with your own data, the necessary directory structure must first be created and your own station coordinates and crustal model must be defined. Both REA and WAV directory structures are created with the program MAKEREA. The program asks for database name

(must be 1-5 letters and UPPER CASE), start time (year and month), end time (year and month) and the structure to create (REA or WAV). An output will then show the names of directories created. If a directory already exists, a message will be given. It is possible to create both structures at a time, since the program after finishing asks if the respective other structure should be created.

3.6 Default parameter files for the main programs

Once you start locating events, the location program HYP will look for an input file with station coordinates, location parameters and crustal model. This file is located in the DAT directory and called `STATION0.HYP`. To edit the file, move to the DAT directory by using command DA and edit the file. For more information about this file, see section 6.1. If you want to keep the original file for test purposes, first copy it to a file with another name.

When plotting epicenters (command EPIMAP) input files with map-contours for the EPIMAP program are used (type *.MAP). These files are also located in the DAT directory. If you want to use more detailed map contours, you must get hold of your own data and put them into file called e.g. MYMAP.MAP. In the DAT directory there are two sets of contours: WORLD.MAP and EUROPE.MAP, a more detailed European map than found in WORLD.MAP. Detailed map files in SEISAN format for the whole world is found at the SEISAN ftp site and the SEISAN CD.

The plotting program MULPLT can use a default file for those stations, which are to be routinely plotted, as well as other default information. This is defined in `MULPLT.DEF` in DAT, see example file. `MULPLT.DEF` also defines which keys are assigned to which phases and what character is used for the mouse. An example is given in DAT, see also example in 6.2.

Both the `MULPLT.DEF` and `STATION0.HYP` can also be in the working directory. Programs always look there first and on a multi-user system, this enables different users to have their own setup. It also makes it possible to work with different setups by just changing directory.

The AUTOPIC PROGRAM requires AUTOPIC.INP in the DAT directory.

Most SEISAN program use the `SEISAN.DEF` file (in DAT, see section 3.10) where many general parameters are set.

The bulletin program requires a front page with whatever you like and some set for fonts can be made. The text of this page is located in the BUL.INP file in the DAT directory.

3.7 Color settings

All programs using color can use a color definition file called COLOR.DEF. The file can be located in the current working directory or in DAT. Programs will first look in the working directory, then in DAT. If no COLOR.DEF file is given, default is used. Be careful with color setting, unexpected results might occur, like getting a blank screen when plotting white traces on a white background. Several color settings are universal like setting colors for titles, however several color settings are specific for particular programs, see example file below. Since colors can also be used for the Postscript file, there is an option for using color on the screen but not for Postscript, since it is more likely that the user will have a color screen than a color Postscript plotter. Sending a color plot file to a black and white laser printer results in a plot with gray tones.

The COLOR.DEF file:

This file is for defining SEISAN color objects and for setting the size(of full screen) of the graphics window for individual programs. If no file available, default colors, as given at end of line below, will be used. The entries can come in any order, however the keyword MUST appear as shown below. If an object color is not defined, it will retain its default value. If a black or white background and no colors are chosen, all colors will be reset accordingly.

```
This file is for defining SEISAN color objects. If no file available,
default colors will be used. The entries can come in any order, however the
keword MUST appear as shown below. If a object color is not defined,
it will retain its default value. If a black or white background and no
colors are chosen, all colors will be reset accordingly.
Since verison 9.0, the size of the window (% of full screen) for most graphics
programs is also set in this file.

The color codes are:

1: blue   2: green   3: red   4: yellow   5: white   6: black

color_screen            1   ! 0: no colors, 1: colors
color_hard_copy         0   ! ------------------------
color_back              5   ! background color
color_trace             6   ! seismic traces, map contours
color_pic               3   ! phase picking
color_zoom              2   ! zoo lines in mulplt
color_def               6   ! default color
color_frame             1   ! frames like epimap map frames, mulplt
color_title             6   ! titles on top of plots
color_spec              1   ! spectras
color_axis_not          3   ! axis notations
color_epi               3   ! epicenters
color_station           3   ! seismic stations
color_map_contour       1   ! epimap contours
color_map_grid          6   ! Lat-long/(x,y) grid
color_label_grid        6   ! Grid labels for map
color_symbol_key        6   ! Diagram key.
color_prompt            1   ! Prompt text.
color_section           3   ! section outline in epimap
color_bval_np           2   ! bvalue, number of events
color_bval_ac           1   ! ------, accumulated ---
color_bval_line         1   ! ------, lsq line
color_box               5   ! box for interactive input
color_box_letter        6   ! letters in --------------
color_foc_dilat         3   ! focmec dilatation
color_foc_comp          1   ! focmec compression
color_foc_p             2   ! focmec P-axis
color_foc_t             3   ! focmec T-axis
color_foc_plane         1   ! focmec fault planes
color_syn               1   ! synthetic picks, blue
size_bvalue            50
size_sample_graphics   80
```

```
size_catstat          60
size_codaq            70
size_corr              0   ! not implemeted
size_epimap           70
size_fk               60
size_focmec           70
size_lsq              60
size_mag              60
size of mulplt         0   ! set in MULPLT.DEF
size_presp            65
size_rmsdep           50
size_spec             80
size_ttplot           60
size_wad_plot         60
```

3.8 Compiling SEISAN programs

The SEISAN distribution for all platforms includes the executables. Therefore in general it is not necessary to recompile. However, you may have the source distribution or you might want to modify some of the programs for your own needs or remove bugs and will have to compile programs.

The SEISAN programs on all platforms can be compiled using the make utility On all platforms there is a 'Makefile' in both the PRO and LIB directories and the make file is the same for all operating systems supported. The file might not need any modification, however the following parameters must be set correctly:

SEISARCH (environmental variable): This variable is used as keyword for the compilation, and can be solaris, g77, gfortran, macosx, macosxppc or windows. While the gfortran option should work on all platforms, the other keywords allow to have specific compile options. The keywords are also used to define which programs are compiled and installed in addition to the default list of programs. See chapter 7 for differences between the platforms. (**Note:** Without setting SEISARCH, the compilation will not work since make will not know what SEISARCH is). On Linux/Unix system SEISARCH is set in the seisan.csh or seisan.bash in COM while on Windows it is set in the Makefile itself or it can be set manually as an environmental variable. Most of the programs are the same on all platforms, but not all.

After you have changed the Makefile according to your system, additional libraries/software must be installed.

(a) On all platforms the Chad Trabant MiniSeed library is used (new from version 9.0). In the distribution they are located in file libmseed-*.tar.gz in SUP. The files must be uncompressed and placed in ../LIB/libmeseed, e.g. gunzip libmseed*gz and tar xvf libmseed.tar. This must be done in LIB directory since libmseed directory is part of tar file.

(b) For the Windows platform, a graphics library and an include file is needed for the DISLIN software (new from version 9.0). The two files dislin.h and dismg.a are located in SUP and must be copied to ../LIB/dislin. The files are for 32 bit Windows so if running on a 64 bit system, different files must be used, see `http://www.dislin.de/`.

The compilation can now be started from the PRO directory (for windows, see compiler installation below) by starting 'make all'. From the Makefile in the PRO directory, the Makefile in the LIB directory is started to create the object libraries. A SEISAN archive in LIB for SEISAN routines is created,

'seisan.a' and in libmseed, an archive libmeseed.a is made. The archives contains all library subroutines, and you can easily link to the archives if you want to use SEISAN subroutines in your own programs. Finally all programs are compiled.

Single programs can be recompiled by starting 'make program' . If you do changes in the LIB directory you need to compile using 'make all', which will also create the archive file. Then you can recompile individual programs in PRO as explained above.

When compiling SEISAN on SUSE Linux it has been suggested to use the following compiler options '-malign-double -finit-local-zero' in addition to the ones already used. Testing this on Redhat Linux produced very large executables, but it may be worth trying on SUSE.

If graphics programs do no link on Linux/Unix systems, check that you have X11 libraries in
`$LD_LIBRARY_PATH`
You can check what is there by command
`echo $LD_LIBRARY_PATH`

Compilers used for SEISAN version 9.0

Linux 64 bit: Gfortran 4.1.2

Linux 32bit: Gfortran xx

Windows: Gfortran 4.5.0 under MinGW

Compiler installation:

For Linux/Unix, compilers are usually installed when the operating system is installed.

For Windows, the gcc/gfortran compiler is found at `http://sourceforge.net/projects/mingw/`. The Fortran compiler and the MinGW development toolkit must be installed. SEISAN can be compiled using the GW shell where the path to compilers is known. If you want to compile outside the GW shell (in the DOS window) in the same way as under Linux/Unix, the compilers must be defined outside the shell by adding c:\mingw\bin and c:\mingw\msys\1.0\bin to the path (assuming MinGW installation under c:\mingw).

3.9 Program Version

The version number of this SEISAN distribution is 8.2. New from version 7.0 is that all individual programs have a version number and a date of the last changes. This will help the users and developers to find out which version is in use and from which date. This is particular useful when bugs are reported. In case major bugs have been fixed, the new versions (same version number, different date) of the individual program will be available on the anonymous ftp site and a message will be sent to the seisan mailing list.

In all programs you can obtain the version number by starting a program with '`−version`' as argument. For example `collect −version` will show you the version and date of your COLLECT program.

3.10 Setting general SEISAN defaults, `SEISAN.DEF`

There are several parameters which are common for several programs which are set in the file `SEISAN.DEF`. This file can be in your local directory or the DAT directory. The system will first look in the local

directory, but normally the file is in the DAT directory. The file does not have to be there since there are defaults for all parameters, but the parameters might need to be set.

The format of the file is as shown below:

```
This file is for defaults for SEISAN and called SEISAN.DEF. The name must
be in upper case on Sun. The following shows the parameters which can be set. The file can
contain any lines in any order, only the lines with
recognized keywords and a non blank field under Par 1 will be read. The
comments have no importance. Numbers are real numbers.

KEYWORD.............Comments..............Par 1.....Par 2
CONFIRMATION        level 0 or 1          0.
CONT_BEFORE         start min before      20.
CONT_AFTER          start min after       1.
CONT_BASE           REA continuous bas    CBER      1.0
CONT_BASE           REA continuous bas    CBER
COPY_WAV_DIR        data base copy reg.   BER
CURSOR              0,1,2=poi,cr,crhair   2
EPIMAP_STATIONS     pl. st. a for all
EPIMAP_MAP_FILE     name of map           EUROPE
EPIMAP_PROJECTION   real number             3.
FOCMEC MAXSOL                             100.
FONT                graphics font
HERKIJ_DISTANCE                           100.0
HIGH_ACCURACY       high accuracy (0/1)     1.
HYPO71_OFFSET       offset in degree      0.0
INIT_IMGMAP_FILE    path to fi            c:\seismo\dat\IMGWORLD.GIF
MAP_LAT_BORDER      dist from center        3.0
MAP_LON_BORDER           "                  6.0
MAP_SERVER                                2
MERGE_WAVEFORM      Code for merging wa   NSN
SPECTRAL GEO_DEPTHS                       10.0       14.0
REG_KEEP_AUTO       keep phases when reg.  1.0
TEXT_PRINT          printer command EEV   nenscript -Psps
WAVEFORM_BASE       Waveform base name    AGA
WAVEFORM_BASE       Waveform base name    LOF
WAVEFORM_BASE       Waveform base name    BER
WAVEFORM_DIRS       Waveform directory     /net/seismo/seismo/WOR/seisnet
OUTPUT_DIR                                ./
INIT_IMGMAP_FILE                          /net/seismo/DAT/IMGWORLD.gif
MAP_SERVER                                0
IMGMAP_PATH                               /net/seismo/DAT/MAP
INIT_MAP_LOWER_LATITUDE                   -90.0
INIT_MAP_UPPER_LATITUDE                   90.0
INIT_MAP_LEFT_LONGITUDE                   -180.0
INIT_MAP_RIGHT_LONGITUDE                  180.0
INTERNET_BROWSER                          /prog/netscape
HELP_DIR                                  /net/seismo/INF
```

The parameters are:

AUTO_PROCESS: Set to 1. to run REG_AUTO_PROCESS.

CONFIRMATION: Level of confirmation required for example when deleting files, 0.=no confirmation and 1.=always confirm. (at the moment only used in eev)

Cont_base: Waveform database to be searched (there can be several). The base is selected as a default if flag is 1.0. Example:

```
CONT_BASE selected by default          ESK            1.
CONT_BASE not sel by default           EDI            0.
```

ARC_ARCHIVE: Path to BUD or SeiscomP archive

ARC_CHAN: BUD or SeiscomP data that MULPLT will plot, given by station code, channel name, network code and location id

CONT_BEFORE: Minutes to read into the memory before the required start time (must be at least the length of a waveform file)

CONT_AFTER: Minutes to read into the memory after the data that is plotted

COPY_WAV_DIR: Normally, waveform files are copied to WAV when registering an event. If this parameter is set, waveform files can be copied to the WAV data base specified. Max 5 characters.

CURSOR: Select cursor: 0. for pointer, 1 for cross and 2 for crosshair (Unix only, on PC only pointer is available)).

EPIMAP_MAP_FILE: The map coordinates file to use with MAP option in EEV

Epimap projection. The projection number used by EPIMAP, see EPIMAP program for choices.

EPIMAP_STATIONS: One letter indicator for plotting stations, in first column. See EPIMAP program for codes. A is all.

FOCMEC MAXSOL: Maximum number of solutions in FOCMEC grid search, default is 100, however, it may be required to allow for more solutions.

FONT: Select font available on X system. To check available fonts use command 'xlsfonts' or 'xfontsel' and 'xfd -fn *font*' to display a font. (Unix only)

HIGH_ACCURACY: Setting it to 1.0 enables high accuracy operation. This parameter affects programs MULPLT, HYP and UPDATE

HYPO71_OFFSET: Apply offset in degree to station and epicenter locations, required for example when not all stations are either east or west of 0 longitude.

MAP_LAT_BORDER, map_lon_border: These parameters are used with command MAP and GMTMAP in EEV, which plot a map centered on current epicenter. The two parameters give the distance in degrees from the epicenter that the map should be plotted. If both set to '0.' EEV will ask for the parameters.

MERGE_WAVEFORM: The network code given to waveform files merged with MULPLT when running from EEV. See MULPLT and EEV. Also used in WAVETOOL and SEISEI. Max 5 characters.

SPECTRAL_GEO_DEPTHS and HERKIJ_DISTANCE. See MULPLT spectral section for explanation. Parameters used to calculate geometrical spreading.

REG_AUTO_PROCESS: Name of program to run when registering event.

REG_KEEP_AUTO: If flag set to 1.0, keep automatic pics when registering event from EEV.

TEXT_PRINT: Printer command used to print an S-file from EEV.

WAVEFORM BASE: Name of waveform data base to be searched. Normally this is a 1-5 letter data base name. The name must be written as shown above under Par1. Not needed for the default data base.

WAVEFORM_DIRS: The complete path to directories where the system should look for waveform files.

OUTPUT_DIR: Output Directory for SEISAN commands results. Default "./"

INIT_IMGMAP_file: File name of the initial map represented as image. Default `/seismo/DAT/IMGWORLD.gif`, for JSEISAN.

MAP_SERVER: Type of map retrieved from Internet or locally; 0: Static local image (country boundaries), 1: Static remote image (country boundaries), 2,3,4,5: Dynamic Remote image (2:country boundaries, 3:Relief from GTOPO30 only land, 4:Two minute shaded relief, 5: Combine 3 and 4). Default 0, for JSEISAN.

IMGMAP_PATH: PATH for the static local maps (images) stored in the local hard disk used for zooming. Default ¨/seismo/DAT/IMGMAP¨; for JSEISAN.

INIT_MAP_LOWER_LATITUDE: Lower latitude of the initial map. Default -90.0, for JSEISAN.

INIT_MAP_UPPER_LATITUDE: Upper latitude of the initial map. Default 90.0, for JSEISAN.

INIT_MAP_LEFT_LONGITUDE: Lower longitude of the initial map. Default -180.0, for JSEISAN.

INIT_MAP_RIGHT_LONGITUDE: Upper longitude of the initial map. Default 180.0, for JSEISAN.

ACROBAT_READER: Path for the Adobe Acrobat Reader (needed for the help files). Default ¨/prog/acroread¨; for JSEISAN in Unix.

INTERNET_BROWSER: Location and place of browser

HELP_DIR: Directory of help files, usually INF

GMAP_DIR: The directory on your computer system where `gmap.html` is copied to when gmap is called in eev , see also section 6.3.3.

GMAP_TYPE: The type of background map used by Google Maps when gmap.html is opened in a browser options are :MAP, SATELLITE or HYBRID, TERRAIN.

GMAP_ICON_QUAKE: The gmap program used this parameter to defines the icon uses to illustrate an earthquake in Google Earth.

GMAP_ICON_EXPLOSION: The gmap program used this parameter to defines the icon uses to illustrate an explosion in Google Earth.

GMAP_ICON_PROB_EXPL: The gmap program used this parameter to defines the icon uses to illustrate a probable explosion in Google Earth.

GMAP_ICON_OTHER_EVENTS: The gmap program used this parameter to defines the icon uses to illustrate all other events in Google Earth.

GMAP_ICON_MSIZE: The gmap program will plot all events with a magnitude smaller than this with the size/scale of this value.

GMAP_ICON_XSIZE: The gmap program scale the epicenter icons with the formula :
scale=GMAP_ICON_XSIZE * Magnitude ** GMAP_ICON_YSIZE

GMAP_ICON_YSIZE: See GMAP_ICON_XSIZE

GMAP_APPEND_KML: With this parameter you can append yor own Google Earth KML code the the output file of gmap. Note, there can only be 100 GMAP_APPEND_KML lines and the KML code must be given between character no. 41 and no. 120.

3.11 Format description in parmater files

Since most SEISAN programs are written in Fortran, the format descriptions follow the Fortran convention. The following are the main format descriptors:

In: Integer format: E.g. I5 means an integer with 5 digits normally right justified. A completely blank field will be read as zero. Examples:

```
123456..    (Position)
  1100      read as 1100
    11      read as 11
            read as 0
```

Fn.k: Real number format: E.g. f7.3 is a real number occupying 7 places like 111.111 and the decimal point is 3 places from the right. Any real number can occupy the 7 places like

```
1234567     (Position)
  1.1
1.1
   1.1
```

All of these will be read as 1.1. If there is no decimal place given, it is automatically put k places from the right. Like the number 1234 read with f4.2 will be read as 12.34. nX . Spaces. Like 5x means 5 spaces.
An: Character format: Like A5 means reading 5 characters
Combining format specifications, example

```
1234567890123456789 position 2f5.1,1x,a4,2x,i2
11.1 12.1 text 12
```

Do not use tabulator instead of blanks

3.12 Using SEISCONF to modify parameter files

By **Bladimir Moreno**

Introduction

The SEISCONF program is a JAVA application for editing some of the configuration files defined in the SEISAN earthquake analysis software. The editing is based on a simple Graphical User Interfaces composed of dialog-boxes where the input data is validated and an on-line help about the parameter being

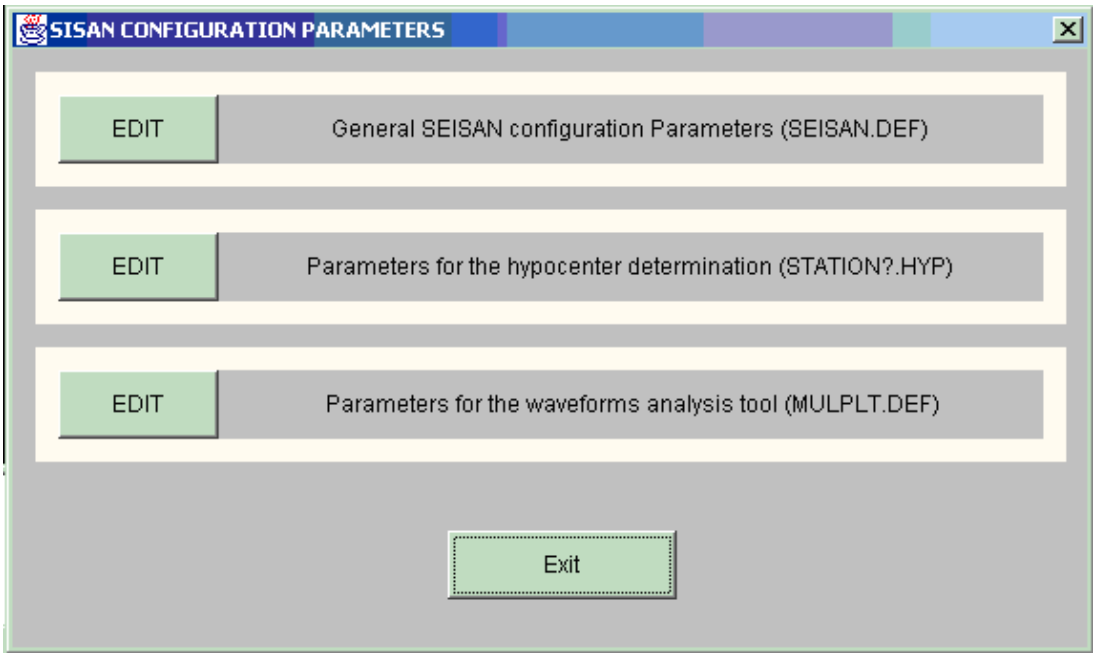


Figure 3.1: Main Window of the program.

configured is given. The user-friendly interface allows to minimize the possible error in the configuration file format as well as to avoid unrealistic values for a particular parameter. SEISCONF was developed with Visual Cafe 4 standard edition. The software operates on Unix and Windows.

When all settings are done as described under Installation, you can start the program from the prompt line as with the command 'seisconf'.

Program options

The program starts with a main window (Fig. 3.1) with three options: (1) Edit the general SEISAN configuration parameters (`SEISAN.DEF`), (2) Edit the parameters for the hypocenter determination (`STATION?.HYP`) and (3) Edit the parameters for the waveforms analysis tool (`MULPLT.DEF`). An option can be selected by clicking the associated button. Normally these files are expected to be in the DAT directory. If the selected configuration file is found in the current directory, the user is prompted for choosing which file, either the local or the global.

Changing SEISAN.DEF

The editing process is based on a list made up with the configuration parameters. The parameters which are present in the configuration file are marked with "*" (Figure 3.1). Parameters that are not found in the file, are set to defaults values. These default values are taken from the file `SEISANDEF.INP`. The user navigates through the list and can edit the associated values by using the edit-boxes labelled as VALUE1 and VALUE2. The entered data is validated according to the type and allowed range of values. The check-box labelled as "Use" permits to select/unselect the selected parameter to be saved in the configuration file. The button <Add New> allows add a new parameter, same type of the one selected in the list, if possible. A brief description about the meaning of the selected parameter is given at the bottom of the window.

Changing STATION?.HYP

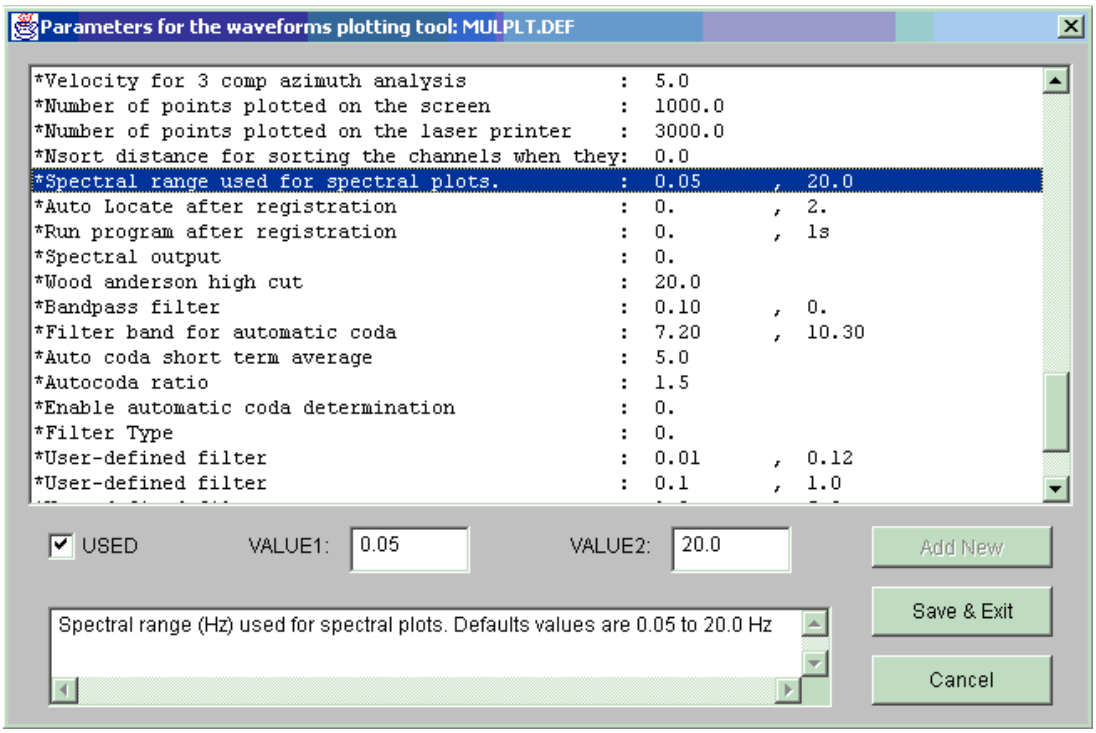


Figure 3.2: Edit Window of the configuration file: `MULPLT.DEF`

The editing window (Figure 3.2) is composed of four blocks: (1) A list with the general parameters (RESET TEST) of the HYPOCENTER program, (2) The seismic stations list, (3) The velocity model and (4) The control line. In addition there is a edit-box for the reporting agency and a combo-box labelled as Model for selecting which `STATION?.HYP` file is being edited. The question mark (?) in the file name takes the value selected in the combo-box. By default, `STATION0.HYP` is selected. The combo-box is made up from the set of station files (`STATION?.HYP`) found in the DAT directory or in the current local directory. There is two buttons <add> and <remove> in the station list block and velocity model block. Their function is for adding or removing items from the list. If you want to add a new item and there is one already selected (highlighted), you must click the button <add> and then change the value with the new one. Do not try to enter the data before clicking the button <add> because that will modify the current value of the selected item. The editing process is similar as it was explained in the previous section, except that the data for validation and on-line help is taken from the file `STATIONDEF.INP`.

Changing `MULPLT.DEF`

The changing is the similar to `SEISAN.DEF`. The data for validation and on-line help is taken from `MULPLTDEF.INP`.

Input definition file format

The input files `SEISANDEF.INP`, `MULPLTDEF.INP` and `STATIONDEF.INP` are used for data validation purpose and user's help support. Normally the user will not edit these files. The first two lines of the files are comments followed by three or more lines for each parameter: (1) The data numerical description, (2) The name and a short description of the parameter shown on the list and (3) A longer description (help) about the meaning of the parameter(s). This can be several lines. The data numerical description has

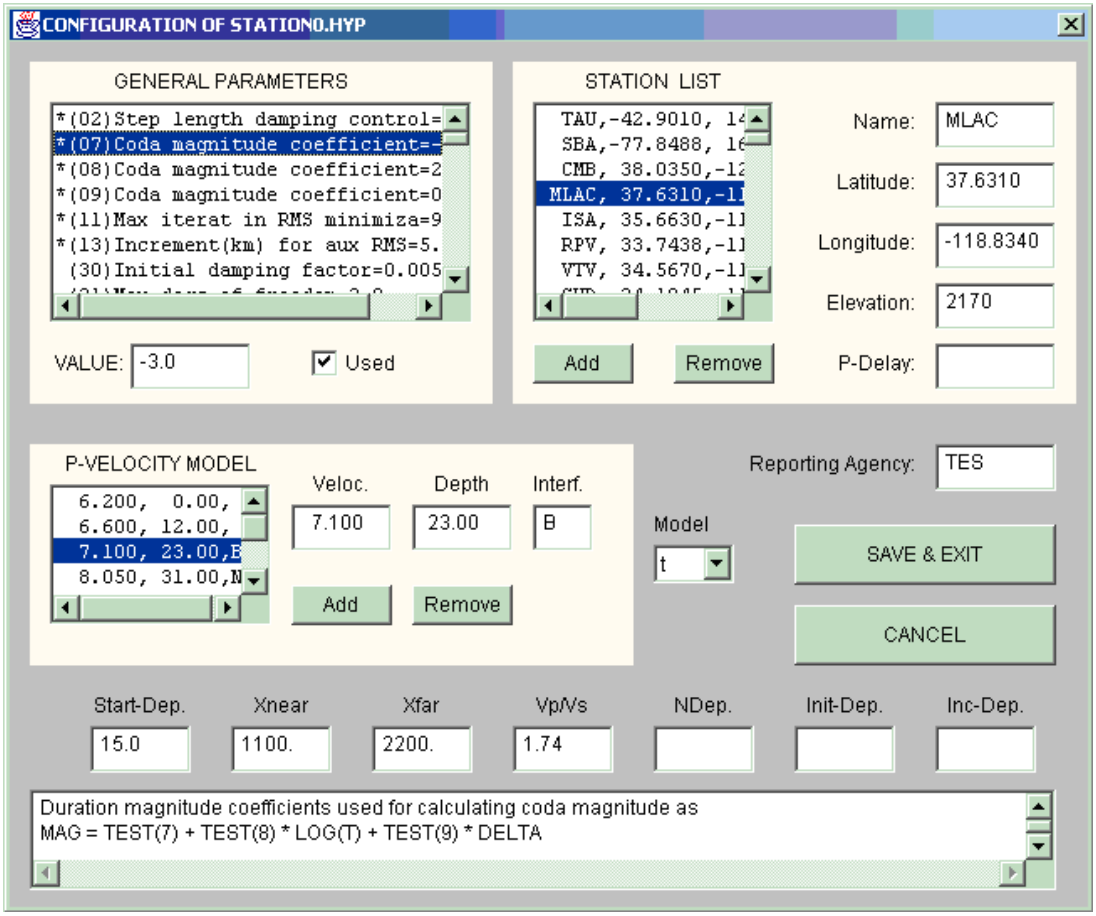


Figure 3.3: Edit Window of the configuration file: `STATION0.HYP`

the following format:

<Keyword>:<default>,<range>,<value1>,<value2>,...;<type>,<column>,<width>,<dec>—...

<keyword>: Identify the name of the parameter used in the configuration file.
<default>: Default value used when the parameter is not found in the configuration file.
<range>: A character for specifying when the parameter is enclosed by a range of values or is allowed to take a set of values. It is set with 'R' for range or 'U' for a set (see example below).
<value1>,<value2>: Represent the lower and upper limit of the range of values which are allow to be taken. It is make sense when 'R' is used in the <range> option. When using 'U', then newline <value1>,<value2>,..,<valueN> are the set of permitted values.
<type>: A character for identifying the type of data: 'F' float,'I' integer,'S' string of characters.
<column>: The position (column) of the parameter in the configuration file.
<width>: The number of characters occupied in the configuration file.
<dec>: The number of digits after the point for float values.

When more than one value is taken for a particular parameter, then the character '|' is used for dividing the two formats.

Examples:

SPECTRAL F-BAND:0.05,R,0,20 ; F,40,5,2 — 20.0,R,0,20 ; F,50,5,2

The keyword S̈PECTRL F-BANDḧas two values. The first one is set by default to 0.05 Hz and can take values between 0 and 20 Hz. It is a float value found in the column 40 of the configuration file and occupies 5 characters with 2 decimal digits. The second value is set by default to 20 Hz and has the same numerical format as the first one, except that it is found in column 50 of the configuration file.

MAP_SERVER:0,U,0,1,2,3,4,5 ; I,40,1

The keyword M̈AP_SERVERïs set by default to 0 and can only take the values 0,1,2,3,4 and 5. It is an integer data found in column 40 and occupies a 1 character position.

WAVEFORM_BASE:BER ; S,40,5

The keyword ẄAVEFORM_BASEïs set by default to B̈ER:. It is a string with a maximum of 5 characters and it is found in the column 40 of the configuration file.

Below is shown part of the content of `MULPLTDEF.INP`

```
# KEYWORD:D,R,V1,V2,..;T,C,W,X|D,R,V1,V2,..;T,C,W,X
# D:default,R:Range(R,U),V1:value;T:type(F,I,S),C:column,W:width,X:decimals
X_SCREEN_SIZE:90.0,R,10,100;F,40,4,0
Size of initial X-window in % of total screen
Size of initial X-window in % of total screen

SPECTRAL Q0:440.0,R,0,1000;F,40,6,1
Q0 for spectral analysis
Q is defined as q0*f**qalpha, default 0 meaning no Q-correction

SPECTRAL KAPPA:0.0,R,0,10;F,40,5,2
Spectral kappa - near surface attenuation
Spectral kappa - near surface attenuation, default 0 meaning no attenuation
.............................
```

Chapter 4

USING SEISAN

Once the system has been installed, it is ready to use. Usually all work should be done in the WOR directory or on a multi user system from your own directory. To move to WOR, type WO. Unless you have to do system work, it will not be necessary to move to any other directories. However to do so, just type the first two letters of the directory name like DA to move to the DAT directory. On a PC the Edit editor is default (invoked with command edit), and on SUN the vi editor.

The system has two basic modes of operation. The first is to work interactively with the database. That means jumping around from event to event, plotting, interactive phase picking, locating, deleting, typing, editing or appending events (S-files). This mode is invoked with the command EEV, which uses several programs, controlled by a driver program and is intended for testing and editing of single events. Once the input data seems OK, the second mode of operation can be used.

On Windows95/98/2000/NT, program SEISAN is equivalent to EEV and whenever EEV is mentioned, this is meant to also include W95/98/NT, SEISAN, see section 4.5.2.

The second mode is more like traditional data analysis where single programs are made to work on the whole or part of the database. In this mode the updated S-files and CAT-files are created. Examples are also plotting of epicenters, waveform data or searching for data fulfilling certain criteria.

The system comes with a test data set from different networks, mainly the Norwegian National Network for the time periods 199309 to 200002. The data has waveform data in different formats. The data set includes events from both local and teleseismic distances. The installation of test data is separate from installation of SEISAN.

If you want to try the system, go directly to section 4.4 to get a feeling for how the system works.

SEISAN problems: Some of the most common problems have been collected in the index under the header "Problem".

4.1 SEISAN TRAINING COURSE

The document 'Computer exercises in processing earthquake data using SEISAN and introduction to SEISAN' which is a tutorial for new users as well as experienced users, is included in the distribution. The testdata used in the exercises need to be installed, see chapter 3. Going through the exercises of the tutorial might be the best way to learn SEISAN. The document is given as PDF file (`seitrain.pdf`) in

the INF directory.

The main goal of the introductory training course is to become familiar with the database program EEV, the plotting program MULPLT and the location program HYP. Of course additional reading of relevant sections in this manual is required. The basic exercises can be completed within one or two days, while the advanced exercises take more time.

4.2 Short user guide

The SEISAN manual has been divided into sections describing the individual programs. However, many tasks require the use of many programs and it is not always easy to find what can be done and which programs to use. The following section intends to give an overview of some general problems that SEISAN can work with and a list of programs to use. The following tasks have been identified:

- Routine processing: Phase picking, hypocenter location and magnitudes
- Determination of source parameters: Fault plane solution, stress drop, etc
- Determination of back azimuth and apprent velocity using arrays and networks
- Crustal structure: Velocities, layer thickness and attenuation
- Seismic catalogs: ISC data, database management, completeness, statistics, etc
- Seismic hazard: Attenuation, catalogs, and soil response

Routine processing

The main work of a seismic observatory is to quickly process and organize incoming data from different sources. SEISAN has a simple time ordered database (see later section) and a set of programs for these tasks. The most important programs are:

EEV: The EEV program is the interactive program for working with single events in the database. The program is used to navigate in the database to find a given event as well as for housekeeping (splitting, merging and deleting events). Once an event has been selected, a large number of options are available like phase picking, earthquake location, fault plane solution, macroseismic information etc. All results of the interactive processing are stored in the database (S-files).

MULPLT: This is the general plotting and signal analysis program and can be used to pick phases and amplitudes, correct for instrument response, produce Wood-Anderson seismograms for determining Ml, simulate WWSSN SP and LP records, determine azimuth of arrival for 3 component stations, rotate seismograms, display theoretical arrival times for IASP91 phases to help identifying global phases and do spectral analysis. MULPLT can be used from EEV or as a stand-alone program.

FK and PFIT Determining apparent velocity and back azimuth using an array of a local /regional network.

HYP: This is the general program for hypocenter location and is based on HYPOCENTER [Lienert et al., 1986; Lienert and Havskov, 1995]. The program can use nearly all common crustal and global phases (8 character ISC codes), locate teleseismic events using the IASP91 model and use observed azimuth and apparent velocity. The program can therefore be used with all types of input data whether from single stations or arrays. HYP can be used from EEV or as a stand-alone program. Apparent velocity is currently only used for starting location.

EPIMAP: This is the general hypocenter plotting program for making epicenter maps and hypocenter profiles. The hypocenters can be plotted with elliptical error ellipses and EPIMAP can also be used for interactive selection of events in polygon areas. For plotting hypocenters, there is also an interface to GMT.

BUL: The function of this program is to produce a bulletin. The user can tailor the appearance to local needs and the program can produce bulletins of hypocenters only or both hypocenters and phase readings.

In addition to the above programs, several programs are available for database creation, input and output of large data sets and conversion and manipulation of waveform data.

In order to get an idea of how routine processing works, some examples of routine processing will be given below.

Case A: Telemetry network with 32 channel central recording

The network generates waveform event files, which are transferred to SEISAN. The tasks are:

1: Convert waveform files to SEISAN format or any of the other formats used by SEISAN. It is likely that the format is MiniSEED in which case no conversion is needed. (many events can be converted in one operation). Inspect events with MULPLT. From MULPLT, false triggers are deleted and real events are put into the database. Events are at this stage identified as local, regional or distant. Phase picks can be done at this stage, but is usually done later.

2: Interactive phase picking, earthquake location, magnitude etc done with EEV. Automatic phase picking is also possible at this stage.

3: Database is updated (UPDATE) once a suitable interval has been processed interactively, usually a month. Updating means permanently storing the hypocenters etc in the database.

4: Make hypocenter maps with EPIMAP.

5: Produce a bulletin with BUL.

Case B: 3 telemetry networks and one broad band station

The routine is the same as above except for one additional step between 1 and 2. Since several data sets are available, some of the detections from different networks or the broad band station might correspond to the same event. There are now two options. The first is to merge the waveform files for corresponding events and then put the events into the database. The second option is to put all real events into the database and then do the merging from EEV.

Case C: A mix of stations and networks and additional phase readings

The steps are as in case B except that before step 2, the additional phase data is put into the database. In this case the merging of events must be done with EEV

Case D: A network recording all data in continuous mode into a SEISAN continuous data base. In addition, there is likely to be network wide triggering put into SEISAN. In this case it is a question of inspecting the triggers with EEV/MULPLT as above and possibly extract additional data out of the continuous data base and put it into the event data base.

It should be noted that data collection and step 1 to 3 is fully automated using SEISNET [Ottemöller and Havskov, 1999].

Example of using EEV for interactive processing:

Find event in default database nearest the given date and time: EEV 1999020303
Once EEV is started, an EEV prompt is given and different EEV options are available. Examples are: E: Edit event, P: Plot event, L: Locate event, F: Make fault plane solution, d2201: Find event nearest day 22 at 01 hour, MAP: Start EPIMAP to show earthquake location and SAC: Start SAC processing of event using all parameter and waveform data from SEISAN database.

The above examples have mostly described the interactive processing of single events. However, once the data is in the database, operations can be done on the whole database, for any time interval or for events fulfilling certain criteria (like magnitude, area etc). Examples are relocating events, extracting data and determining coda Q.

Source parameters

The routine processing normally produces magnitudes and hypocenters. The fault plane solution can be determined using polarities and one event (Snoke et. al., 1984). Composite fault plane solutions can also be made. A second way of determining fault plane solution is to synthetically model the waveforms using the modeling programs. In addition, seismic moment, stress drop and seismic source radius can be determined by doing spectral analysis or spectral modeling. This can also be done automatically with AUTOSIG. The moment tensor of local earthquakes can be determined by inverting the amplitudes of the Pg and Sg waves [Ebel and Bonjer, 1990]

The full wave modeling programs integrated with SEISAN, are written by Bouchon [1981] and Herrmann (Herrmann,1996). The ray-tracing program is based on WKBJ and written by Chapman et al. [1988] and integrated with SEISAN by Valerie Maupin. All the above programs are executed from EEV in order to use known source parameters.

Crustal structure and Q

A large database can be a good source of information for determining structural parameters and SEISAN provides several programs to determine the crustal structure and Q. Using seismic arrival times, it is possible to invert for the crustal structure using the VELEST program [Kissling et al., 1994]. It is also possible to do forward modeling using the location program for a large number earthquakes, since it at the end of a run, a summary of average station travel time residuals and event RMS is given. A special option of HYP is to locate a data set with all permutation of a given range of models in order to find the model giving the lowest RMS.

Deep earthquakes under a local network produce clear phase conversion at crustal interfaces [Chiu et al., 1986]. They can be modeled with one of the full wave modeling programs both with respect to amplitude and arrival time.

SEISAN can, when displaying surface waves, make spectral files ready to be processed for surface wave dispersion with Herrmann's programs (Herrmann, 1996).

Attenuation can be determined using the coda Q method for local earthquakes (CODAQ). The coda Q program will calculate q for a series of events and stations at given frequencies. Average values are calculated and a q vs f curve is fitted to the calculated values. The principle for calculation is the standard coda q method, whereby a coda window is bandpass filtered, an envelope fitted and the coda q at the corresponding frequency calculated [Havskov et al., 1989]. The SPEC program will determine Q by calculating spectral ratios or the near surface attenuation using the spectral decay method. An alternative is to use spectral modeling where Q, stress drop and seismic moment are modeled simultaneously.

Catalog and database work

Once a large database has been created, several programs are used to manipulate and analyze the data. The catalog can be searched for a large number of parameters. Selection criteria are: Magnitude range,

magnitude types, event types (e.g. local, distant, volcanic, explosion), latitude, longitude and depth range, RMS of travel time residuals, number of stations used in the location, felt events, number of polarities, presence of certain stations etc. Events can also be selected in an area with the program used for hypocentral plots.
A very useful source of data is the ISC. Data from ISC CD ROM's can be read and converted to SEISAN format (hypocenters and phase data) and put into a database. The data can then be used for e.g. seismic hazard, fault plane solution or it can be relocated. A general task with catalogs is to homogenize magnitudes. Magnitude relations between e.g. Mb and Ms or Ms from one agency to Ms from another agency can be done with the program MAG. The program will also convert one magnitude to another once the linear regression has been determined. Event statistics can be made with STATIS and b-values calculated with BVALUE. The number of events as a function of time is plotted with CATSTAT.

Seismic hazard

Probabilistic earthquake hazard computations is done, using the EQRISK program [McGuire, 1976] or the CRISIS99 program [Ordaz, 1991, 1999]. EQRISK computes seismic hazard in terms of probabilities of exceedence vs earthquake intensity measures such as peak ground acceleration (PGA), for a given site or a grid of sites for up to eight different return periods. The site amplification is calculated with the SPEC program. This is used for making spectra of many seismic signals in a semiautomatic manner. The program is intended for two purposes: (1) making relative spectra for a series of pairs of stations terminated by the average spectra, (2) Making a series of spectra for a number of stations and events. The spectra can be corrected for distance, q, and instrument response.
This section involves a large number of programs and a more detailed description is given in section 6.24.

4.3 Getting data into the database

The first requirement for interactive work with the event editor EEV is to get the data into the database.

There are two ways to get data into the database, as described in section 4.3.1 and 4.3.2. It is of course possible to make the individual S-files directly in the REA directories with the editor. This would be rather slow, and be against the philosophy of the system. However, it is mentioned in order to point out how simple the database structure is.

The SEISAN system can be used with or without digital data, the only difference to the directory structure is that the WAV and CAL directories are present when using digital waveform data. However the way of getting data into the database differs in the two cases and will be described separately.

4.3.1 System with digital data

This means that the original data is individual digital event waveform files generated by some data acquisition system. The waveform data can be stored in SEISAN, GSE and SAC format as single or multi trace files. The files that are used in conjunction with the database are normally stored in WAV but can also be in the user's directory, e.g. WOR. The normal scenario would be that multiplexed files would be transferred from a digital field station, demultiplexed and converted to SEISAN waveform format. Programs are provided to convert from most of the popular waveform formats like MINISEED, GSE, PCSUDS and from commercial recorders. It is most practical to initially put the files in WOR, check the events for false triggers, save the true events in WAV, make the corresponding S-file and a hardcopy of the digital data.
All of this can be done with the program MULPLT. The program plots channels from a single waveform

file. The user can then interactively decide if this is an event to keep, in which case an S-file is created in the database and the event is moved to WAV.

Alternatively, all new waveform files can be auto-registered into the database (AUTOREG) and all checking takes place from EEV.

When digital data is the input to the analysis system, MULPLT is the program to use to get data into the database. From there on further analysis can be done with EEV (picking phases, locating and editing). MULPLT is also the program used with EEV. For more details on MULPLT, see detailed description in section 6.2.

4.3.2 System without digital data

In this case the user would get phase data from other sources, e.g. analog seismograms or files with readings from other stations and agencies. These files are assumed to be written in Nordic Format. Conversion can be done from other formats like ISC, NEIC and HYPO71.

If a user already has a file with one or several events in Nordic Format, this file can be split up into single files which are copied (from any directory) into the database by using the command SPLIT. Creating a new file in Nordic Format can also be done with the program NEWEVE (use command NEWEVE).

The SPLIT program then reads the NEWEVE output file and writes out single S-files with correct names either in the current directory (default) or in the database specified (BER or another). The reason that the database specifically must be given is that the user should not accidentally put data into the database (see section 6.6).

4.3.3 Database security

Duplicate ID:

Since the database consists of single files with names corresponding to time down to the second as well as the event type (L, R or D) it will sometimes happen that two events will get the same name. Thus copying in a new event with the same name could overwrite the existing event, and the user would never know. In SEISAN, from version 5.0, some security has been put in. New data can enter the database with 4 programs: SPLIT, EEV, MULPLT and AUTOREG. With all programs, the user will be prompted if a new event is about to overwrite an existing event. Both SPLIT and EEV have the possibility to create alternative ID's if the user wants both the new and old event, while MULPLT and AUTOREG just offers the possibility to skip a double event. If a new ID is created, an attempt will be made to use a time one second later. If that also corresponds to an existing event, the next second is attempted etc. This allows for 60 events to be registered in the database with the same minute and event type. If an event has got the ID changed, the header line in the file is NOT changed, however the ID line is of course changed. This will be indicated on the ID line with a 'd' at the end of the ID number.

Deleting events:

Event here means S-file in the database. Events are only deleted when using EEV, either with the EEV delete command D or the EEV append command A. In both cases, the deleted event is stored in the DELET database before being deleted from whatever database. Even if the system contains many databases, there is only one DELET database. This means that deleted events from different databases are mixed in DELET. In order to restore an event, enter DELET database with EEV and copy the deleted event back with the C command. It is up to the user to manually clean up the DELET database.
There is one more final security. If an event has been deleted from a database, but an UPDATE has not

yet been made, the event might be in the CAT part of the database and can be extracted by SELECT or the editor.

4.3.4 Data base tools, content and checks

Content of data bases, program BASE:

In the REA directory, a binary file called REA.LOG contains information about number of events in all data base. Initially the file has no information, but each time programs EEV, HYP, UPD, CHECK_BASE or COLLECT are executed, the information is updated for the months accessed. The information can be displayed with program BASE, which first shows available data bases and the user, can then select one to get info for particular months. Make sure to use right case for data base names, always in upper case on Unix systems. The program is still a bit experimental !!

Check content of S-files for magnitudes and residuals etc, program CHECKRE:

The program can read data bases or CAT files and check events for large residuals, abnormal depths etch. The program is intended for quality control, the parameters hardwired in the program might not suit all. Check program source listing.

Check for data base related errors, program CHECK_BASE

The data base depends on error free S-files and that there is a correspondence between the S-file name and the event ID. This should normally be ok, however errors can occur during editing or there can be program crashed producing errors. The program reads the data base and checks for:

Missing ID lines: If ID line is missing, it can be put in manually or doing an UPDATE.

No correspondence between ID line and S-file name: A serious error has occurred. try to find out what is correct, the ID or the file name. An UPDATE cures the problem, however data might be lost.

Error in S-file: All parameters are checked and files with non standard parameters are indicated. The error can be a number in a wrong position. The errors should be corrected.

For all the above 3 cases, an index file is generated with bad S-files and EEV can the be used directly with the index file to access the bad S-files. THIS ONLY WORKS WITH ONE DATA BASE AT A TIME.

It is recommended to run check_base in case of system crash or as a security, just before an UPDATE.

4.3.5 High accuracy in SEISAN

SEISAN can use higher accuracy than the default. The goal is to have an accuracy of 1 ms in time and 1 m in location.

In order to write out the high accuracy numbers, a new parameter has been added to `SEISAN.DEF`. The parameter is HIGH_ACCURACY. Setting it to 1.0 enables high accuracy operation. This parameter affects the programs MULPLT, FK, HYP and UPDATE.

Station locations: The station file looks like before except that in order to get higher accuracy of station locations, the minutes of latitude and longitude are specified without the point. E.g. the minutes 22.122 can now be written as 22122 in the same columns as before while if the point is given, only 2 decimals can be used as 22.12. This changes do not affect any old station coordinates. Programs reading station coordinates, will use high accuracy input if available.

EPIMAP will always read in high accuracy mode, if any high accuracy data is present, whether station locations or hypocenters.

FK will always read high accuracy station coordinates, if available and FK can therefore now be used with very small arrays.

Programs with output affected by high accuracy mode:

MULPLT will write the phase readings as f6.3 instead of f5.2 like e.g. 11.234 instead of 11.23. For normal use, this is not needed and the files look better if high accuracy mode is not used.

HYP and UPDATE writes an extra high accuracy hypocenter line which has been given type H. An example is

4.4 Interactive work with earthquake locations, EEV command

The idea of SEISAN for interactive work is that the user should be able to easily jump from event to event and run several different programs with one event without restarting every time. This is done with the command EEV (see below). In this interactive mode, events are picked, edited, located, moved, deleted etc. until a satisfactory solution is found. In the interactive mode, NO UPDATING of the location in the S-file or the permanent output CAT directory is done since it is too easy in interactive mode to accidentally change something. The permanent updating of S-files and CAT directories can only be done for one or several months at a time (see UPDATE command) in order to ensure that nothing is forgotten within a month.

Once the events have been updated, further work can be done (like searching for specific events or making a bulletin) using single programs which read directly from the database. Most of the analysis programs will also work without using the database structure that is e.g. searching in single file with many events. For more details of the analysis programs, see chapter 6.

4.5 How EEV works

It is now assumed that data has been entered into the database. The fundamental tool for the database is then the EEV program, which mostly works within the limits of one month in the standard database or with whatever the user has of S-files in his own directory. Optionally, EEV can also work with several months. A special option is to use a list of files in an INDEX file, see end of this section and SELECT program. Some of the commands available within EEV are also available within programs. See below for more details on EEV.

The EEV program reads the file names of all S-files in the database monthly directory (or local directory or index file), positions the pointer at the first event and asks for a command to be performed for the current event or to find another event. If the command is to use a program, control is handed over to that program, which on completion hands control back to EEV. In this way, many different independent programs can be used from within EEV, e.g. several different location programs can be installed.

EEV can be started in several ways:

EEV with one month in default database: EEV yyyymm.
E.g. EEV 199201 would work on January 1992 on the standard BER database. It is here also possible

to give a more precise start time like EEV 1992011520 to start with the first event at or after January 15 at 20 hrs.

EEV with one month in alternative database: EEV yyyymm BASE.
BASE is the database. To work on the NAO base, the command would be EEV 199201 NAO.

EEV with several months in default database: EEV yyyymm YYYYMM
yyyymm is start year and month and YYYYMM is end year and month.

EEV with several months in alternative database: EEV yyyymm YYYYMM BASE
yyyymm is start year and month and YYYYMM is end year and month.

EEV to work with events is local directory: EEV
Only the S-files in local directory will be used.

EEV to work with an index file: `EEV index.out`
EEV can work with an index file and the command would be `EEV index.out`, where `index.out` is the index file name (can have any name as long as it contains a '.' except when used with HYP). For information on index files, see 6.4.

Databases can have 1-5 letter names and the user specify 1-5 letters. The real names in the directory structure are always 5 letters so if the user specifies e.g. a base name of BA, the real name will be BA___ . The full 5-letter name can also be used.

The commands in EEV mainly use only one letter unless a date or a number has to be given. To get a short explanation, type ? and you will get:

```
------------------------------------------------------------------------
 Help on EEV
 ------------------------------------------------------------------------

 ?         Print this help file
 #xx:      Go to event # xx, also works without the #
 Axx:      Append event # xx to current event, original event
           remains
 AA:       Append current event to next event, original event remains
 AUTOSIG:  Automatic processing
 B:        Back one event
 BOUCH:    Run Bouchon's modeling program
 BOUSEI:   Make SEISAN file from Bouchon synthetic file
 C:        Copy event to another data base or to current directory
 CM:       Copy out several events to eev.out
 COMMENT:  Write comment lines in S-file
 COML:     Add one geographical location line
 COMF:     Add one or several felt information lines
 D:        Delete current event, you will be prompted to confirm
 DD:       Duplicate header line in S_file
 DUP:      Duplicate current event in data base, different id
 Dxxxxxx:  Go to first event on date xx, hour xx, min xx
 E:        Edit
 EXP:      Enter explosion information
 Eyyyymm:  Let EEV session end with year yyyy and month mm
 F:        Make a fault plane solution with FOCMEC
```

```
FH:       Make a fault plane solution with HASH
FI:       Make a fault plane solution with PINV
FP:       Make a fault plane solution with FPFIT
FO:       Plot all fault plane solutions, no questions
GRID:     Locate by grid search
GMAP:     Make epicenter maps with Google Map or Google Earth
GMTMAP:   Makes a GMT map  like the MAP commnad
HERRMANN: Run Herrmann's modelling program (not PC)
HERSEI:   Make SEISAN file from Herrmann synthetic file (not PC)
H:        Locate with Hypoinverse
HYPO71:   Locate with Hypo71
IASP:     Generate arrival times using IASPEI91 tables
IL:       ISC location program (unix only)
INPUTEPI: Input hypocenter and origin time in S-file from st. input
INPUTONE: Input an additional type one line (hypocenter line)
INPUTFPS: Input completer fault plane solution line
IFP:      Input fault plane solution strike, dip, rake, Ag., Prog, Q.
INPUTX:   Input of xnear  and xfar
INVRAD:   Make moment ternsor inversion with Invrad
Jyyyymm BAS:Jump to year yy and month mm in base BAS
L:        Locate event, will also calculate magnitude if not
          locatable but distance is present, Hypocenter
LL:       Locate current and next event together
Lxx:      Locate current and event xx together
M:        Input model indcator on header line
MAC:      Input macroseismic information
MACROMAP: Make a GMT based map of felt information, Unix only
MAP:      Make a map of current location
MODELS:   List MODEL.DEF file in DAT
NEW:      Add a new event to data base
WKBJ:     Make synthetic seismograms with WKBJ program
O:        Operating system command, e.g. ols is ls, ocd test is cd test
          do not currently work on command with prompt input like epimap and
          collect
P:        Plot event, also make hard copies and pic phases
PO:       Plot event with defaults
PF(IT):   Get back azimuth and app. Velocity for network P-arrival
PITSA:    Start Pitsa program (not on PC)
PRINT:    Print S-file on printer
PMAC:     Macroseismic Windows program
PUT:      Register event
Q:        Quit EEV
R:        Rename event type, must be L, R or D
REG:      Register event
RMSDEP:   Calculates and plots RMS as a function of depth
SAC:      Run SAC
Sxxxxxx:  Search for next two events which are within xxxxxx seconds. If time
          blank, a default of 180 secs is used
SS:       Find next unprocessed event in base
```

```
SYNT:      Make parameters for synthetic modelling
T:         Type event
TT:        Type only header line of event
TTPLOT:    Make travel time plot
U(PDATE):  Updates S-file with hypocenter etc.
USERCOM:   Start user defined command as 'usercom -sfile <sfile-name> '
WAD:       Make a wadati diagram
UP:        Update list of S-files
W:         Show location of waveform files
Z:         Auto pic current event, if readings available, new pics
           will be added with a flag
```

Note: Command letters can be upper or lower case.

Comments to commands:

#XXX : Go to event by number. When giving a number, only give the number of digits needed, no formatting. Thus e.g. to find event 7 or 777, write 7 or 777 respectively. If there is not an event corresponding to the parameter specified, EEV will go back to event #1. In the number command, # can be omitted.

Axxx: Append another event to current event. The event specified is appended to current event. All header and lines in both files are saved and put in order in the current event. The main first header is from the current event. The ID line for the appended event is saved as a comment line. The user will be questioned if the appended event is to be deleted.

AA: Same as above using next event.

AUTOSIG: Automatic processing with autosig program.

B: Back one event

BOUCH: Run Bouchon's modeling program

BOUSEI: Make SEISAN file from Bouchon synthetic file

C: Copy events
There are two options, copy the event to another database given by a 1-5 letter name (upper case) or to a file EEV.OUT in your working directory. Several files can be extracted within one EEV session to the same EEV.OUT file. A new EEV session deletes the previous eev.out file. The C option can be used to recover files from the DELET database of deleted events. In addition to making the EEV.OUT file, an index file is also made called indexeev.out. THIS FILE IS NOT DELETED WHEN EEV STARTS UP since the intention is to be able to use EEV to make an index file of interesting events from several months. You can then start eev with the selected events with command EEV eevindex.out. Note: The other data base can also be a local data base ",," in which case EEV should not operate on the same local data base. CM: Copy many files to eev.out. The copying starts at current file and the user is asked for the number of files to copy. COMMENT: Comment are written into S-file, terminated by a blank line.

DXXXXX; The D-command is used to jump to another event at a given date and time, normally only day is used: The hour can optionally be specified. E.g. d2205 will find the event nearest in time after day 22 at 05 hours. If both day and hour is used 4 digits MUST be given e.g. 0708. Highest accuracy is the nearest minute.

D: Delete event You are asked for confirmation. After the event has been deleted, all S-file names are

read in again and all event numbers after the deleted event are therefore changed. **The deleted event is automatically saved in the DELET database**. If the event is present in the CAT file, it remains there until the next update is done, see UPDATE command in 6.7.

DD: Duplicates the header line

DUP: Duplicates an event in the database. The duplicated event has an ID, which is one second different from the original event. The command can be used to split an event in two and then manually deleting phase lines in each.

E: Edit the event. As default on SUN vi is used and on PC edit is used. The editor can be changed, see section 3. When control goes back to EEV, the file is checked for possible typing errors or other format problems. If a problem is encountered, the line with the problem is displayed with an indication of where the mistake might be, and the user is returned to the editor. Alternatively the error can be ignored. The file is also checked for missing iD and consistency between file name and ID. Problem: Some editors will keep a backup copy of the original file so 2 files might be present with one e.g. with the additional extension .BAK. EEV (from version 7.2) will only use the original file, but there is no check on what backup files might accumulate.

Eyyyymm: Giving this command will make the current EEV session end with year yyyy and month mm within the same data base. When EEV gets to the end of the month, pressing return will move EEV to the first event of the following month instead of to the first event of the same month.

EXP: Input of explosion information. This command creates 3 new lines (see format description in Appendix A) and changes the main header line event type to explosion (E). The user is asked for location, time, charge and comments. The explosion agency is used to classify types of sites and can be used by SELECT for searching. If no event is available, a new event must be created with EEV command NEW.-

F: Make a fault plane solution. The program uses polarities. See section 6.16.1 for more details.

FI: Fault plane solution using PINV

FH: Fault plane solution using HASH.

FP: Fault plane solution using FPFIT.

GMTMAP: Start `gmtmap.exp` program (not included in SEISAN) to plot GMT map. GMTMAP automatically creates a map using GMT. (UNIX only)

GMAP: Make an epicenter map of current event using Google Earth or Google Map. It is also possible to make maps with many epicenters using GMAP outside EEV, see section 6.3.3 for more details.

GRID:. Hypocenter is started up and will ask for the grid: Latitude and longitude range and grid spacing. A maximum of 71 points can be used in each direction. The point with the lowest RMS is found and the corresponding location and residual is printed on the screen. It is now optionally possible to plot the contours on the screen. The map coordinates used are as defined in `SEISAN.DEF`. Note that the grid search is using exactly the same parameters as Hypocenter. This includes all weights and phase types. The depth is fixed to the depth given in the S-file header line. For more details and an example, see application note `epi.pdf` in INF.

H: Locate with Hypoinverse, no database update is made, no Nordic output format file.

HERRMAN: Herrmann's modeling programs, only on Sun, might work on Linux, not tested.

HERSEI: Make a SEISAN waveform file from output of Herrmann modeling, only tested on Sun.

HYPO71: Locate with HYPO71. The database is not updated (not well tested on PC).

IASP: Generate a file with theoretical arrival times for the current event. The command will only work if the event has an epicenter and origin time in header line or a subsequent type 1 line, see also INPUTEPI and INPUTONE. These theoretical times will then be displayed with mulplt, the next time command P is used in EEV. The theoretical times are listed in file iasp.out. See section 6.20.3 for more information. The command can also be used directly from MULPLT.

IL: Makes a location with the ISC location program. For more info, see section 6.1.4

INPUTONE: Makes an additional type one line (hypocenter line) in the file. Enter the data exactly under the columns indicated. The line will be entered exactly as written, so it is possible to enter any part of the information.

INPUTEPI: Works like INPUTONE, except that it overwrites information on the first header line if non-blank information is given. Use INPUTEPI to add information to the first header line like e.g. the depth. If existing nonblank characters on the line are to be replaced by blanks (e.g. remove a magnitude), use underscore "_".

INPUTX: Makes a comment line with xnear, xfar and start depth values.

INVRAD: Runs the moment tensor inversion program, see section 6.16.4.

Jyyyymm BAS: This command makes it possible to change month and database during an EEV session by giving a new year yyyy and month mm and optionally a new database BAS. If no database is given, the same database is assumed.

L: Locate event with HYPOCENTER (same as HYP). The location does not update the S-file.

Lxx: Locate current event with event number xx. This is used to check if two events belong together.

LL: Locate current and following event together.

MAC: Enter macroseismic information, you will be prompted for all information. For details of the type of information, see definition of Nordic format, Appendix A. See also command PMAC.

MACROMAP: Felt information is read from a file with macroseismic information and plotted with GMT. The file name of the file with macroseimic observations is given in the S-file.

MAP: Start EPIMAP program to produce a map of current location. If a location is given in the S-file, this location is plotted, otherwise the event is located if possible and the resulting location used for plotting. The parameters for generating the map are set in the `SEISAN.DEF` file (see 3.10).

MODELS: Lists `MODEL.DEF` file in DAT that assigns names to single characters in `STATIONx.HYP` file.

NEW: Creates a new event in the database. The user is asked to give date and time and the event is created in the current monthly database.

O: Give a command to the operating system. This is a very useful command, since it is possible to do almost anything without leaving EEV, including starting a new session of EEV !! E.g. the command ols on Sun and odir on PC would make a directory listing. The name and path of the current s-file is copied to a file named `eev.cur.sfile`, this makes it easy to write your own programs to handel seisan data.

PF or PFIT: Calculate the apparent velocity and back azimuth using the P-arrival times stored in the S-file. The calculation is done by a free standing program PFIT, which also can be called outside eev. It is assumed that the arriving wave can be approximated with a plane wave so this option is intended to be used with events which are far away relative to the size of the network which then can be considered a seismic array. The station coordinates are taken from the default station file and there is no correction for station elevation. When starting the pfit option, the user will be given a choice of reference station and maximum distance from the reference station. Linear distances will then be calculated from the reference

station and possible results will be associated with the reference station. All P-phases given as P, Pn, PN, Pg, PG, PKP, PB and Pb will be used and it is up to the user to ensure that the event file only contains the phases to be used. The output is displayed on the screen and the linear fit can be shown on a plot, which also can be used to interactively check individual station values, see example below.

Example run of PFIT

```
 Give reference station, enter for using first station in list
HYA
 Maximum distance from reference station, default is 1000 km

 2007  1 5 1652 23.3 D  55.991-159.065 13.5  BER 19 1.2 5.3SBER 5.5BBER 5.4BPDE

 Stations available: 18  Stations used: 14
 Stat      Delta t  Latitude Longitude         x         y
 HYA         0.000    61.165     6.182     0.000     0.000
 LOF       -44.970    68.132    13.541   394.668   774.733
 MOR8      -34.220    66.285    14.732   458.521   569.318
 NSS       -23.720    64.531    11.967   310.275   374.227
 MOL        -9.540    62.570     7.547    73.220   156.192
 DOMB       -6.840    62.073     9.112   157.158   100.965
 ASK         0.810    60.472     5.201   -52.618   -77.095
 RUND        4.060    60.414     5.367   -43.680   -83.581
 BER         4.820    60.384     5.335   -45.414   -86.880
 EGD         3.270    60.270     5.223   -51.403   -99.501
 ODD1        5.600    59.911     6.627    23.891  -139.457
 BLS5        9.920    59.423     6.456    14.712  -193.739
 KONO        9.630    59.649     9.598   183.221  -168.571
 SNART      18.280    58.339     7.210    55.130  -314.293

 Reference station is: HYA

 Back azimuth =358.8  Apparent velocity =19.81  corr =-0.99  rms =   0.03
```

Relative to the reference station, the above output gives relative P-times and relative x and y-coordinates (km). It is also seen that only 14 station were available within 1000 km from the reference station HYA. These results are also available in an output file `pfit.out`. See also array processing section 6.29 on FK-analysis.

PUT: Register event. This option is mainly meant to be used with the SEISNET data collection system. The command cleans up the S-file for all SEISNET operations. It removes commented out ID-lines and copies the waveform files given for the event from the current directory to WAV. The command is equivalent to the register command in MULPLT. If events are auto registered with AUTOREG, the command can be used to clean up and inspect incoming data without using MULPLT directly.

PMAC: Windows only program PROMAC for processing macroseismic information to calculate intensities from felt information and model the macroseismic intensities. The program can also plot associated pictures (in directory PIC). All information is stored in the S-file. The program was written by Bladimir Moreno, and has a separate manual, see INF directory. Program must be installed separately, zip file in SUP.

P: Plot event with MULPLT

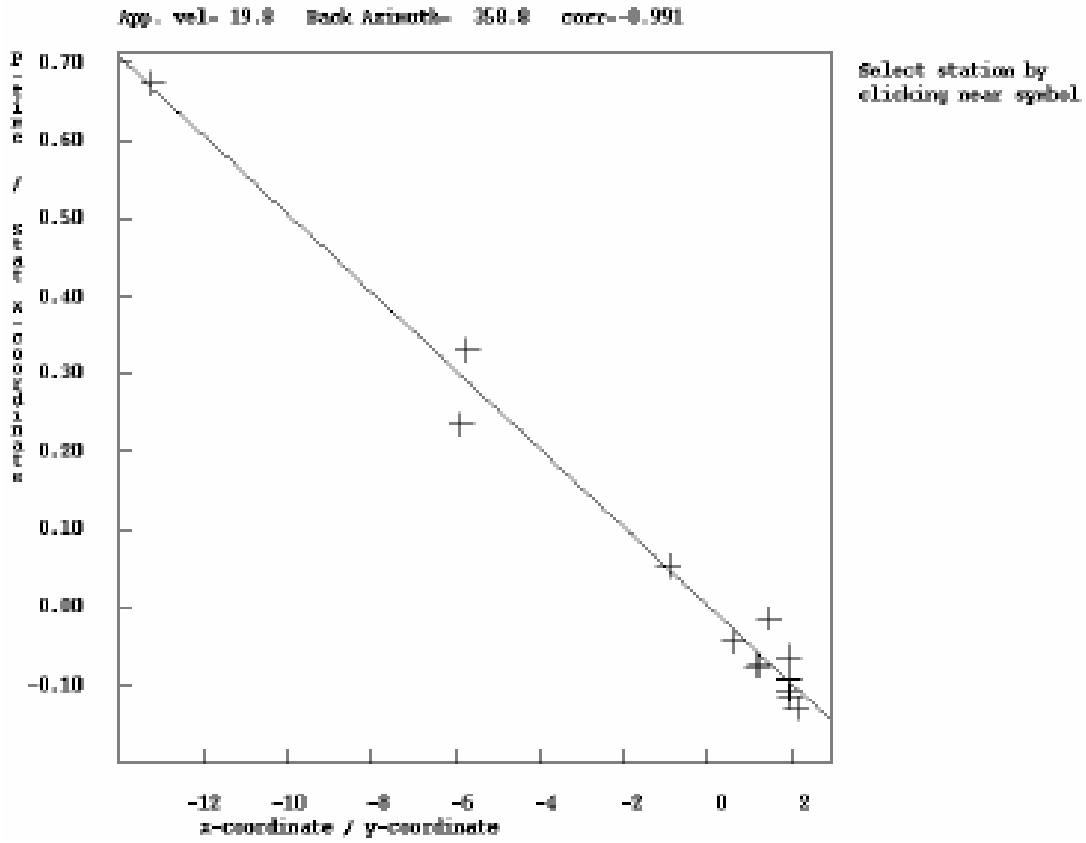


Figure 4.1: The linear fit of P-arrival times to a plane wave. For more details, see Havskov and Ottemöller [2010].

PO: Use MULPLT with defaults. This means that no questions will be asked and the plot appears in multi trace mode with default channels and default filters as given in the `MULPLT.DEF` file in DAT. Useful option for routine inspection of raw data.

PITSA: Run the Pitsa program, see section 6.13.1 (Not on PC).

PRINT: The current S-file is printed on the default printer, to set up printer command, see `SEISAN.DEF` (section 3.10).

Q: Quit EEV

REG: Same as PUT.

R: Rename event type Giving an event a new type requires changing the header in the S-file and the S-file name. All this is done with R-command. You are prompted for a new type (can be the same in which case nothing is done). A new S-file is made and the old deleted. The CAT-file is NOT changed so if no UPDATE is done, the event there will remain with wrong type. Event types are L: Local event, R: Regional event and D: Distant event.

RMSDEP: Calculates and plots RMS as a function of depth for current event. Note: Program starts by reading STATION0.HYP so if current events uses e.g. STATION1.HYP, STATION0.HYP must be there also.

SAC: Convert all data to SAC format and starts the SAC processing system (not distributed with SEISAN, must be obtained separately), not on PC.

Sxxxxxx: Search for next pairs of events separated in time by xxxxxx secs (max 999999). If no value is given, 180 secs is used. The command is intended for finding events to be merged after putting together two different data sets with SPLIT. If a new time instead of the default 180 is entered, it will remain in effect for the whole

EEV session. NOTE, that the search starts with the current event, so after using S, one return to go to the next event must be given to start a new search.

SS: Find next unprocessed event in database. Events, which have status in ID line as follows: SPL: split with SPLIT program, HYP: auto-located with HYP, NEW: new event from EEV or ARG: registered by AUTOREG. The idea is that when new unprocessed data have entered the database by one of these programs, it should be easy for the operator to find the event. In EEV, an N near the end of the prompt line indicates an event with this status.

T: Type event.

TT: Type only header of event.

TTPLOT: The program reads P and S-arrival times from S-file and makes a travel time plots. The program is useful for checking readings, see section 6.20.3. The lines connect the computed first arrivals for P and S, respectively.

UPDATE: Updates (overwrite) S-file with hypocenter, magnitudes, residuals etc. Note that the CAT file IS NOT UPDATED . This can only be done with stand-alone command UPDATE, see section 6.7.

U: Update EEV event list. All S-file names are read in again. Is useful if data arrives during an EEV session, like when using Copy command from another data base.

USERCOM: Starts user defined program with command `usercom -sfile <sfile-name>`, where usercom is the command name. This command is useful for example if you want to start your program to create a report based on the S-file, from EEV. Note: the usercom is not a SEISAN program.

W: Check if event has waveform files. If so, check in which directory they are if present on the system. The search will start in current directory, then WAV followed by all directories defined with keyword WAVEFORM_BASE in `SEISAN.DEF` in DAT.

WAD: The program reads the data for the event and then asks if all phases are going to be used or only phases of the same type like Pg and Sg. Ideally, only phases of the same type should be used, however in practice it might be interesting so see all data, it might give an idea about phase identification. The Wadati parameters will now be calculated and shown on the screen. Optionally a plot can now be made. The plot shows the Wadati diagram. On the left is shown all stations with corresponding S-P times. Any station on the plot can be identified with the cursor. Point the cursor near a symbol and click and the station data will be shown in the upper right hand corner. This facility is used to identify bad picks. The plot output file is called `wad_plot.eps`.

Z: Automatic phase picking. A waveform file must be present. See also the AUTO program section 6.15.

Below is shown a session with EEV on PC.

Example of using EEV for November 1993

```
eev 199311

1993 11 Reading events for base AGA          18
#    1  2 Nov 1993 17:06 48 L   60.443    4.512  2.0  1.5 N 1.8CBER    6  ?
#    2  5 Nov 1993 22:37 21 D                                          1  ?
#    3  5 Nov 1993 22:37 23 D                                          1  ?
#    4  5 Nov   93 22:39  2 L                                             ?
#    5  5 Nov   93 22:40 58 L                                             ?
#    6  7 Nov 1993 23:40 43 L   67.837   20.059 15.0  0.7  2.5CBER     7  ?
#    7  7 Nov 1993 23:43 17 L   66.307    6.919 31.0  1.4  3.1CBER     8  ? 17
#   17 19 Nov 1993 01:45 29 D   70.069  139.780   .1  0.1              7  ? t

File name: \seismo\REA\AGA__\1993\11\19 0145-29D.S199302
1993 1119 0145 29.0 D  70.069 139.780   .1  BER  7  .1                        1
                .19     999.9   821.9999.9   .3206E+06   .2536E+07   .2639E+08E
ACTION:UPD 97 03 25 21:28 OP:jh   STATUS:               ID:19931119014529     I
93111901.K41                                                                 6
  93 1119  153  6.5 D                                                        1
9311 19 0153 06S.NSN_09                                                      6
STAT SP IPHASW D HRMM SECON CODA AMPLIT PERI AZIMU VELO SNR AR TRES W  DIS CAZ7
KBS  SZ EP        151 54.8                                      13.4 0 3365 161
TRO  SZ EP        153 03.0                                        .010 4420 169
MOL  SZ EP        153 50.51                                       .010 5070 165
ASK  SZ EP        154 04.0                                        .010 5262 164
BER  SZ EP        154 05.0                                        .110 5274 165
EGD  SZ EP        154 05.5                                        .110 5285 165
KONO BZ EP   9    153 49.21                                     25.5 0 5413 167


#   17 19 Nov 1993 01:45 29 D   70.069  139.780   .1                     7  ?
#   18 21 Nov 1993 01:53 56 L   60.184    4.965 15.0  N 0.5  2.6CBER    11  ?
 1993 11 Reading events for base AGA          18
    1  2 Nov 1993 17:06 48 L   60.443    4.512  2.0     2.2  1.8CBER     6  ? q
```

In the above example (PC), the month has 18 events. For each event, vital information is displayed: Date, type, hypocenter, RMS, first magnitude and number of stations (number in S-file which might be larger than number used for location as given in S-file header line after a location). In this way the user can quickly search for events wanted and get important information without looking at all the details. The first event in the list is newly entered into the database as indicated with the N near the end of the line. In the above example, a return was made to go to next event until event #7 after which a jump was made to event 17. For this event, all parameter data was displayed with the 't' command. A return was made to event 18, another return and the event list was read in again and event #1 again became the current event. Note that not all events had a location.

Below are shown examples of the commands (C)opy, (D)ate, a(S)sociate and (A)ppend. Comment are preceded by '!' and written in bold. The database is EAF.

```
EEV 199405 EAF

1994  5 Reading events for base EAF         613   ! the month has 613 events
#    1  1 May 1994  1:18  8 D                                                ?
#    2  1 May 1994 11:37  6 L                                                ?
#    3  1 May 1994 12:00 33 D   36.607   68.449 15.0  2.4    ! go to day 20 ? d20
#  366 20 May 1994  5: 2  8 R                                                ? c
                                          ! copy an event to working dir.
 Copy event: Other database, give 1-5 letter name
             Working directory in file eev.out: return

#  366 20 May 1994  5: 2  8 R                                               ?
#  367 20 May 1994 10:59 32 D      ! jump to 530                            ? 530
#  530 26 May 1994  8:55 11 D      ! look for time association              ? s

  549 27 May 1994  9:27 41 L                                          Associated
  548 27 May 1994  9:27  1 L       ! append to next event             ? aa
 Event #  549 appended to event #  548   Appended event still present
 Do you want to delete appended event(y/n=return)y   ! delete appended event

Backup copy saved as: \seismo\REA\DELET\1994\05\27 0927-41L.S199405 ! del. ev. save
Deleted file          \seismo\REA\EAF__\1994\05\27 0927-41L.S199405 ! app.ev. del.
1994 05 Reading events for base EAF         612    ! event list updated
#  548 27 May 1994  9:27  1 L                    ! jump to 222            ? 222
#  222 12 May 1994 23:28 10 L                    ! change event type      ? r

 Change event type to L,R or D ?r

New file       \seismo\REA\EAF__\1994\05\12 2328-10R.S199405
Deleted file:  \seismo\REA\EAF__\1994\05\12 2328-10L.S199405
Reading events for base EAF         612
#  222 12 May 1994 23:28 10 R                                              ?
#  223 13 May 1994  1: 1 37 L                                              ?
#  224 13 May 1994  1:16 44 L                                              ? q

Stop   Program terminated.
```

****

When the interactive location is finished, the database should be updated, see section 6.1.
****

Using EEV on a subset of events or using alternative databases:

Since the EEV procedure or the HYP program will work on an index file, the user can create a subset of his own interesting events to work with by creating his own index file with just these events. The index file can be created by searching through the database using SELECT or it can be created manually with the C-command in EEV.

Local database:

If data is extracted by using the COLLECT or SELECT and then split up again using SPLIT, it is possible to keep all files in a working directory by not specifying database when splitting up. Another simple way is to use the Copy function in EEV and copy directly from a named data base to the local data base. Programs will then look for S-files in the current directory instead of in the database.

In addition to working with index files, there is also the possibility of storing data in different databases. By default, the data is always stored in BER. However, the user can also create another database structure (file structure) with another name and programs and procedures will work on that database too. There are some restrictions: The new database, which is a subdirectory under SEISMO/REA, just like BER, MUST have a 1-5 letter name. Currently, the alternative database is used in our Institute to store data from other agencies like NAO, which in some cases are copied to our own database (C-command under EEV). The name DELET is reserved for the DELET database, which is always present.

4.5.1 EEV driver program: JSEISAN

Software and manual by Bladimir Moreno

JSEISAN is a JAVA application for providing a user-friendly graphical interface of some of the functions of the SEISAN earthquake analysis system. The program uses the functions implemented in SEISAN by executing external commands, which mean that most of the data processing is performed with the SEISAN software. JSEISAN is mainly a tool for formatting the input data and display the results. The program was developed with Visual Cafe 4 Standard Edition. The software operates in Unix and Windows environments.

Using JSEISAN

The program is started by giving command jseisan and the window (Figure 4.2) below will appear.

The user interaction with JSEISAN can be classified as: (1) Searching for earthquakes and (2) working with a particular earthquake. It is assume that a standard SEISAN data base is available with S-files and CAT files in the CAT directories (see SEISAN manual). The interface has a main window where the user can search earthquakes (<Search> button) set up by a range of values (Figure 7). As a result of the searching process, an earthquake list is shown on the screen and a set of functions is offered. In addition, the epicenters of the earthquake list are plotted on the map. Because there is a direct link between the map and the list, the selected earthquake is highlight on both the map and the list when the user is navigating either through the list or through the map (**see Interactive mapping tool**).

Configuring JSEISAN

JSEISAN has several configuration variables to be set up by the user. In order to configure the variables, you have to click the button <Config> (Figure 6). The configuration parameters are stored in the file

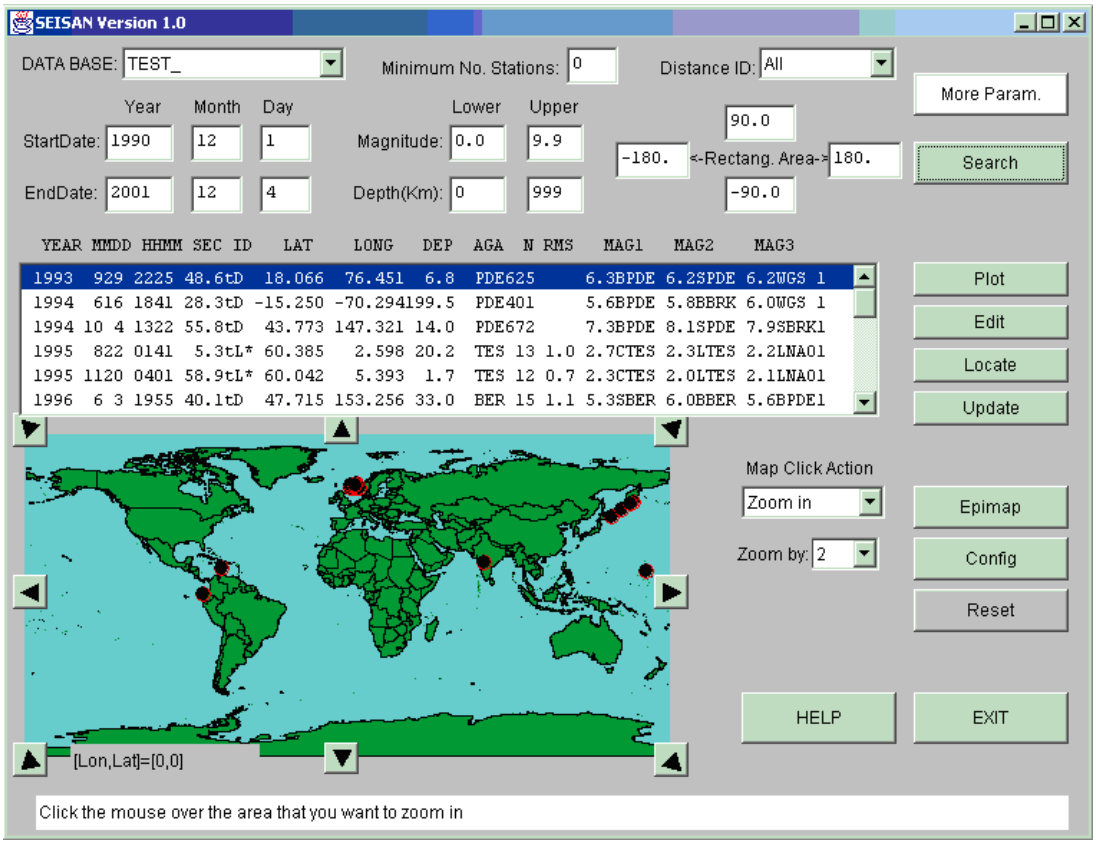


Figure 4.2: Main window of JSEISAN.

`SEISAN.DEF`. The editing process of this file is performed through the JAVA program `SEISCONF.class` (see SEISCONF). A complete description of the configuration parameters is given in at the end of this section.

Selecting the database to work with

The desired SEISAN database is selected from the combo-box labeled as "DATA BASE" located in the left upper part of the main window (Figure 6). The last item in the combo-box corresponds to a SEISAN CAT file labeled as "USER FILE". When this item is selected, the user can load a file, for example `collect.out`, to work with.

Searching

There are two windows for setting the parameters used in the searching process: (1) A main window, which contains the most common parameters (Figure 6) and (2) A second window (Figure 7) which contain the rest. This second window is opened by clicking the button <More Param.>. In order to perform the search you have to:

1. Set up the parameters (period of time, magnitude range, geographic area, etc.) needed in the searching process.
2. Click the button <More Param.> for refining the searching process (if needed). A dialog-box (Figure 7) for setting the additional parameters is shown.
3. Press the <Search> button.

Interactive Mapping Tool

Figure 6 shows a map with the locations of the earthquakes obtained during the searching process. The map has an equidistant cylindrical projection, which means that one unit of longitude has the same distance to one unit of latitude. This map can be used as interactive mapping tool. For this purpose, 4 options are included in the combo-box labeled as "Map Click Action" (Figure 6). The options are "Select", "Zoom in", "Zoom out" and "Polygon". The function of each radio-button is as follows:

"Select": When it is active, the user can click over the circle identifying the location of the earthquake. The associated earthquake in the list will be highlighted.

"Zoom in": When it is active, the user can zoom in the map by a factor of 2,4,8,16 or 32. The zoom factor is selected from the combo-box labeled as "Zoom by". The clicked point on the map will be the center of the new map.

"Zoom out": Similar to "Zoom in" but with reverse effect.

"Polygon": When it is active, the user can define the points of a polygon by clicking on the map. The polygon is closed when a click over the starting point is made. Once the polygon is defined, the user can edit its points by using the refined search window, which is reach by clicking the <More Param.> button. Epicenters inside the polygon can now be selected with the search function.

The zoomed map is taken either from Internet or locally. In the case of being retrieved locally, the images are taken from the directory specified by the configuration variable **IMGMAP_PATH**. The images are saved in sub-directories according with the level of zoom. For example in case of 6 levels of zoom you will find the directories ZOOM0, ZOOM1, ZOOM2, ZOOM3, ZOOM4, ZOOM5 and ZOOM6. ZOOM0 only has one image, the whole world. ZOOM1 contain 9 images, the world is divided into 9 equal areas. ZOOM2 contain 81 images, the world is divided into 81 areas and the last 6 level of zoom (ZOOM6) contains 9^6 images. The image files are numbered as IMG#.gif from left-up to right-down, like a matrix

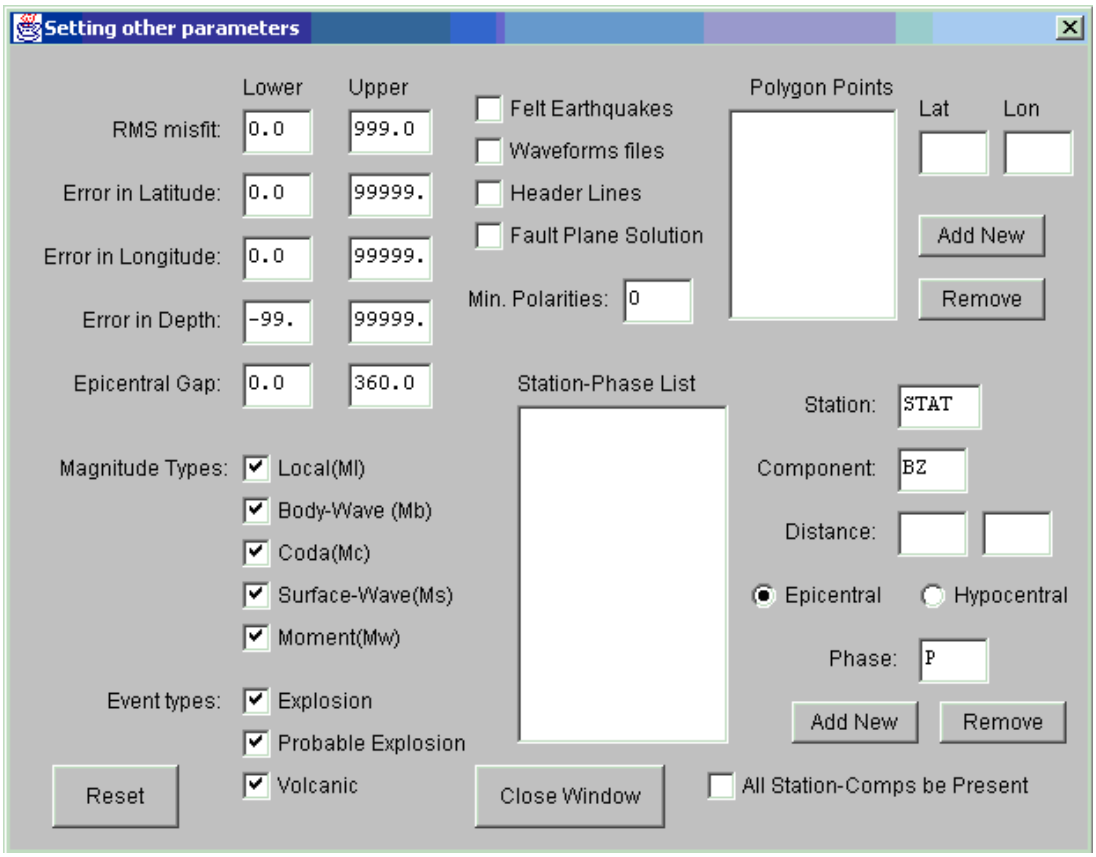


Figure 4.3: Dialog-window for setting other parameters used during search.

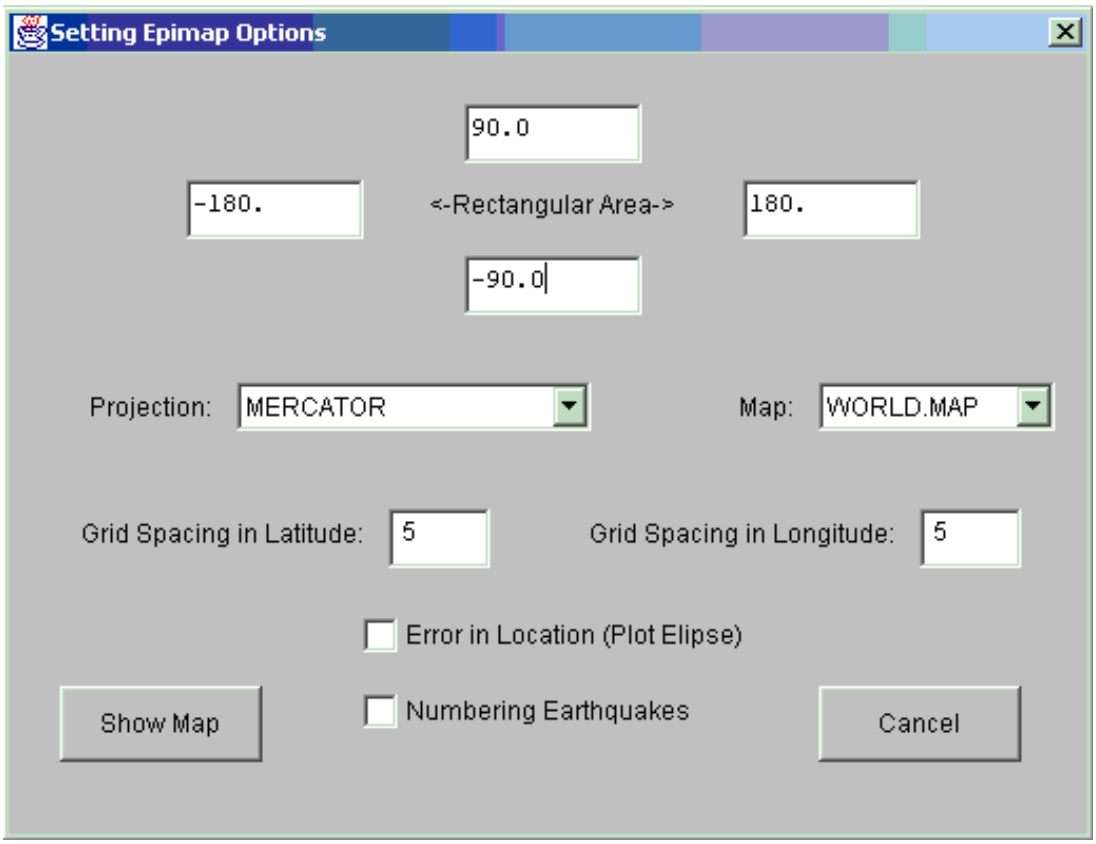


Figure 4.4: Dialog-box for setting the EPIMAP input data. Only the most common EPIMAP parameters are set.

but using only one index. When the map is being retrieved from Internet there is no zooming limitation. You can go as deep as you want. The configuration variable **MAP_SERVER** controls where to retrieve the image from.

Making an epicenter map from the earthquake list

An epicenter map can be made through the SEISAN command "EPIMAP". The steps are:

1. Press the <Epimap> button
2. A new dialog-box for the configuration of the map is shown (Figure 8), then click the <Ok> button.

Working with a particular earthquake

There are four available options for working with a particular earthquake of the list: (1): Plot waveforms, (2) Edit phases, (3) Locate and (4) Update. The associated buttons are located on the right side of the earthquake list (Figure 6).

Plotting and processing traces

Any number of channels, up to 75, can be plotted. In order to visualize a large number of channels, a page system is implemented, when more than 12 channels are selected. For practical reason, two operation modes are given in the plotting process: "multi-traces mode" (Figure 9) and "single-trace mode" (Figure

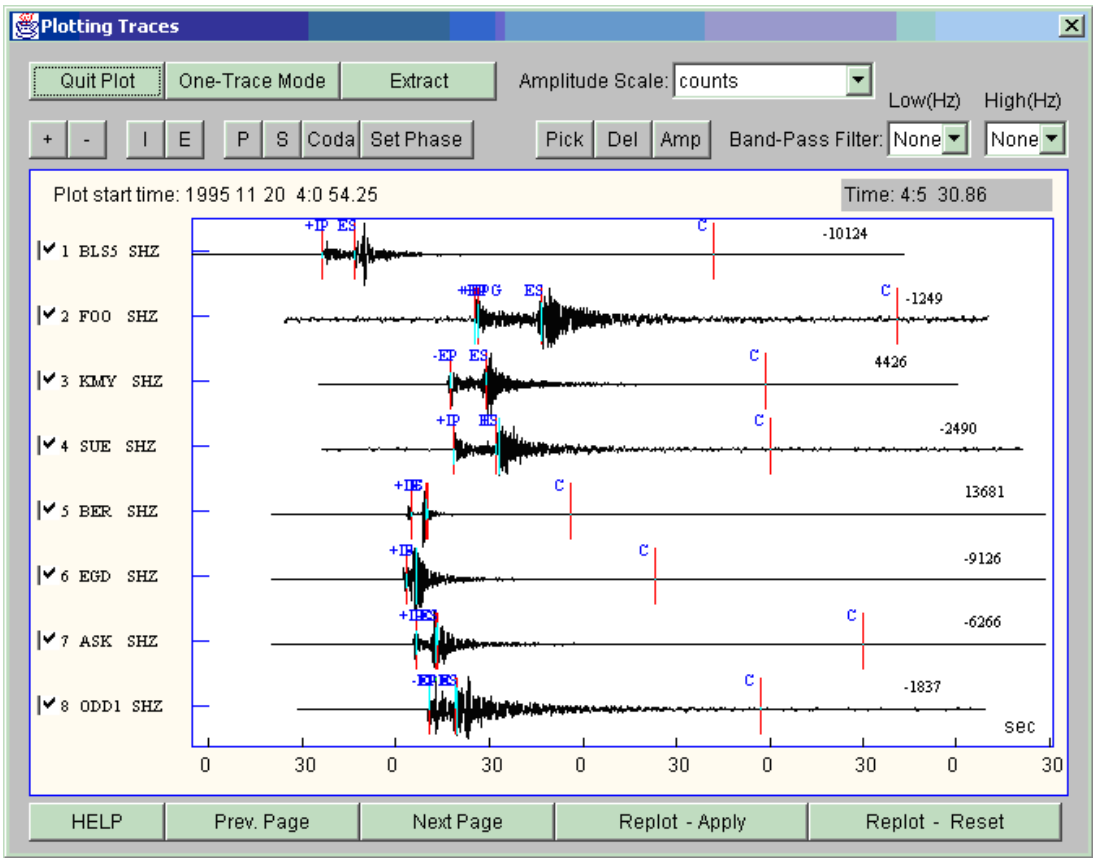


Figure 4.5: Plotting window. Multi-Trace mode.

10). The distinction is made mainly for choosing which channels will be plotted together (multi-trace mode with select/unselect) and to navigate through the selected channels in more detail (single-trace mode). This is similar to MULPLT in SEISAN. In both operational modes, picking phases, zooming and filtering can be done. For "multi-trace mode", a set of additional options are available, e.g. save traces, which is applied to the plotted channels on the screen within a selected time window.

The steps to plot a particular earthquake selected from the list are:

1. Select a desired earthquake from the list by clicking the left button of the mouse on it (Figure 6).
2. Press the <Plot> button.
3. A new dialog-box for the selection of the channels to be plotted is shown (Figure 11), then click the <Plot> button.

Filtering one/several channels

1. Select the low corner frequency of the Band-Pass filter (select "none" if a High-Pass filter is desired) from the combo-box labeled as "Band-Pass filter" (Figure 9).
2. Select the high corner frequency of the Band-Pass filter (select "none" if a Low-Pass filter is desired).

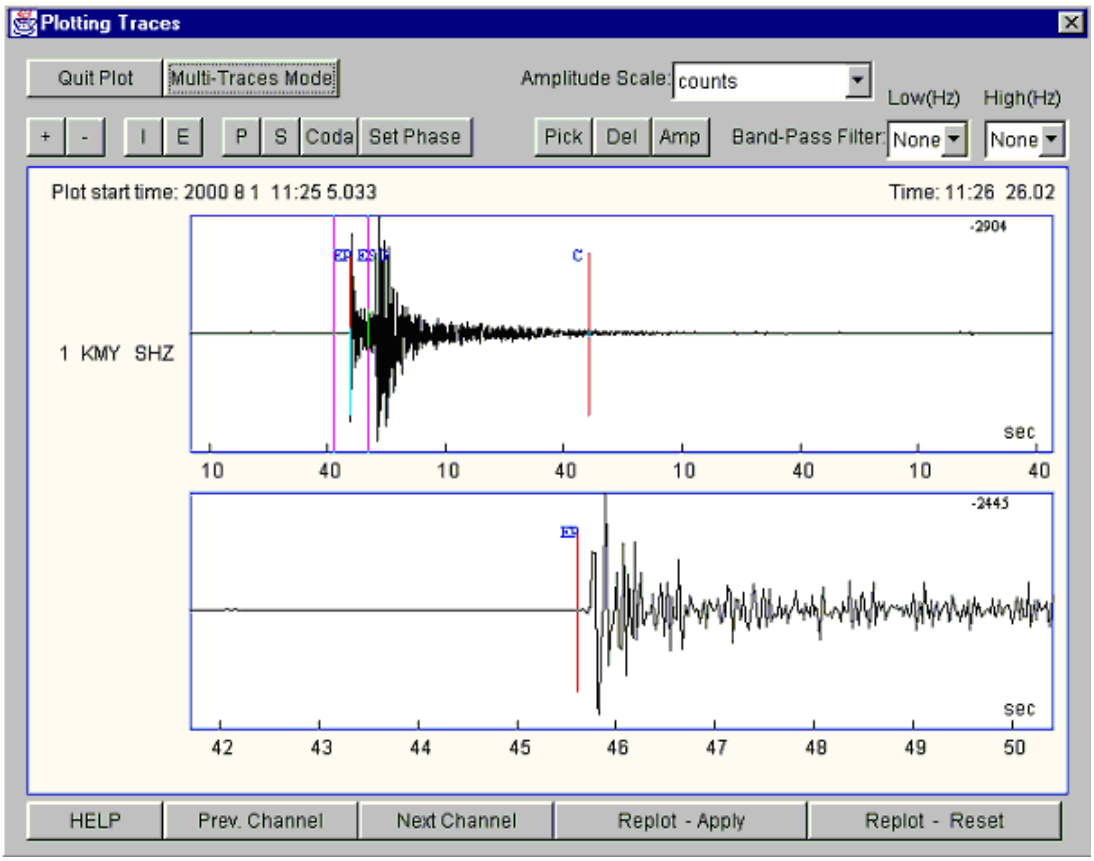


Figure 4.6: Plotting window. Single-Trace mode.

Figure 4.7: Selection of channels.

3. Make a zoom over the traces (**see Zooming section**) or click the button <Replot - Apply>

Instrument corrections
Converting the amplitude of one/several traces into ground velocity, displacement or acceleration and simulating Wood-And., Mb and Ms amplitude.

1. Select the desire amplitude from the combo-box labeled "Amplitude Scale" (Figure 9).
2. Make a zoom over the traces (**see Zooming section**) or click the button <Replot - Apply>

Picking and removing phases

1. Press the <+>/<-> button (Figure 9) in order to select/unselect the polarity (Compression/ Dilatation) of the first impulse (if needed)
2. Press the <I>/<E> button in order to select/unselect the shape (Impulsive/Emergent) of the first impulse (if needed)
3. Press the button identifying the phase to be picked/removed (<P>/<S>,. . .)
4. Press the <Pick>/<Del> button to active the picking/deleting process
5. Move the mouse pointer to the position of the first impulse
6. Click the left button

* The previous buttons will be kept active. Furthermore you can pick the selected phase for the rest of the channels without going through the first 4 steps.

Reading amplitude

1. Press the <Amp> button (Figure 9) to active the reading process
2. Move the mouse pointer to the position of one extreme (upper or lower) of the signal
3. Click the left button (a horizontal line will be drawn)
4. Move the mouse pointer to the position of the opposite extreme of the signal one half period away.
5. Click the left button

* The <Amp> button will be kept active. Furthermore you can continue reading amplitudes for the rest of the channels without going through the first step.

Zooming

1. Move the mouse pointer to the initial time of the zooming window (the time is shown above the right-upper-corner when you move the mouse)
2. Click the left button

3. Move the mouse pointer to the final time of the zooming window
4. Click the left button

Extracting traces from a time window
You can extract either raw or processed traces from the plotting window (Figure 9). The steps are:

1. Select/unselect (see "Other options" section) the channels to be downloaded
2. Make any desire processing (zoom, filter, etc.) if you want to save processed traces or a data selection of the original traces.
3. Press the <DownLoad> button.

The data will now be stored in your working directory in SEISAN format in a file with name specified by the user. Since an integer format is used, processed data with amplitudes less than 1 will have amplitude values 0.

Other options

Switching between "single" and "multi" traces mode

1. Press the <single-trace mode>/<multi-traces mode> button (Figure 9 and Figure 10).

Show/hide channels to be plotter together on the screen

1. Press the <Replot - Reset> button (Figure 9)
2. Click the left button of the mouse over the check-box identifying each channel in order to select/unselect the channel
3. Press the <Replot - Apply> button

Navigating through individual channels

1. Go to the "single-trace" mode (Figure 10).
2. Press the <Next Channel> button to go forward or <Previous Channel> to go backward

Navigating through pages of channels (12 channels as maximum on one page)

1. Go to the "multi-trace" mode (Figure 9)
2. Press the <Next Page> button to go forward or <Previous Page> to go backward

Editing S-files
The editing window (Figure 12) has 3 buttons for closing the window. When you click the button <Save & Exit> the S-file is physically modified on the hard disk. The <close> button keeps the changes in memory until another earthquake is selected on the list. So, you can still perform full processing without making any real change in the S-file. The steps to edit the S-file are:

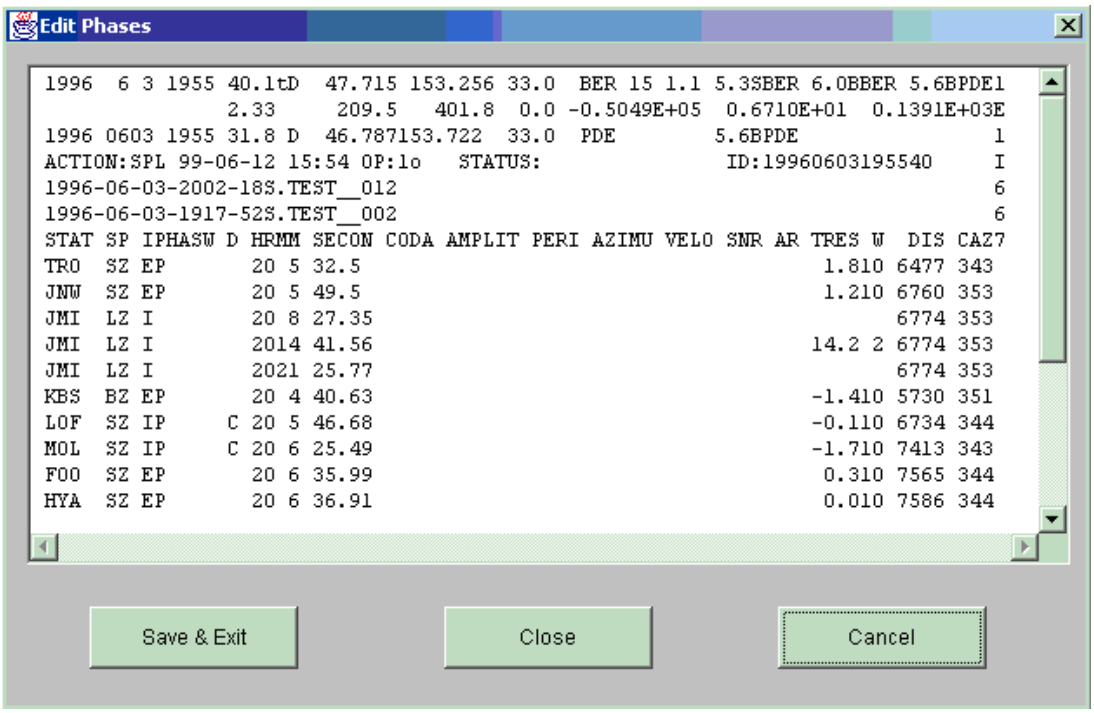


Figure 4.8: Editing window.

1. Select a desired earthquake from the list clicking the left button of the mouse on it (Figure 6)
2. Click the <Edit> button
3. An editing window with the information is shown (Figure 12).

Note that this editing function do not yet, as in SEISAN EEV, check the file for correct editing.
Determining hypocenter and magnitude

1. Select a desired earthquake from the list clicking the left button of the mouse on it
2. Press the <Locate> button
3. A new window with the results is shown (Figure 13)

The usual SEISAN files print.out and hyp.out are generated (see section 6.1.1).
Updating the earthquake in the database

1. Select a desire earthquake from the list by clicking the left button of the mouse.
2. Press the <Update> button (Figure 6)

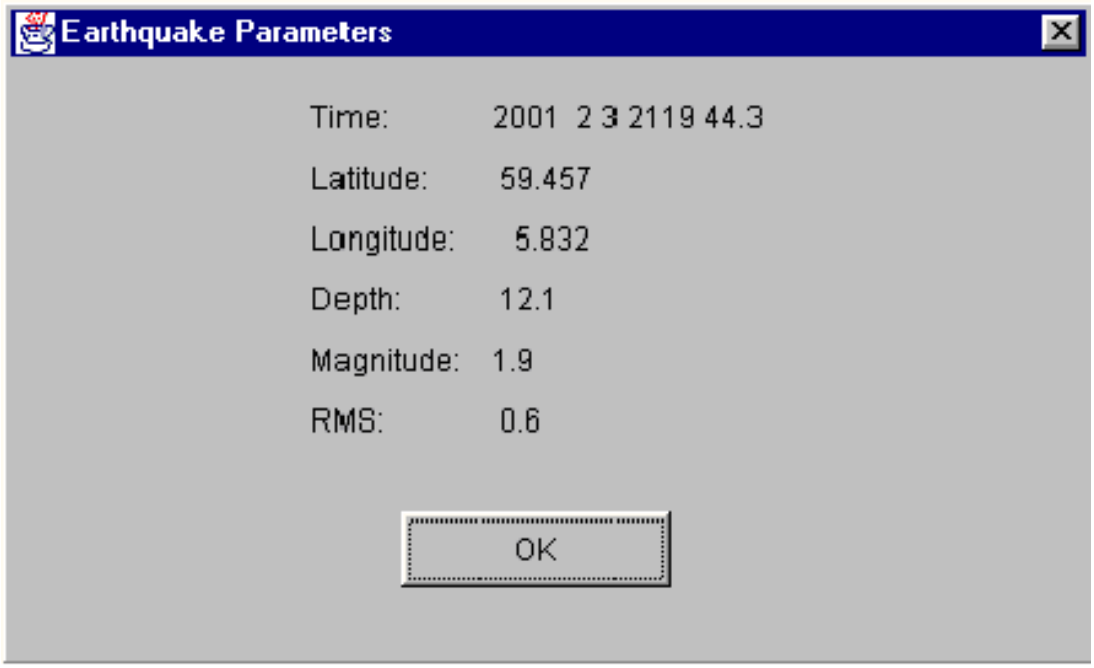


Figure 4.9: Results of the hypocenter determination.

Note: This updates the S-file, but the original CAT file in the database is NOT updated. In order to do so, an update outside JSEISAN must be done (see SEISAN manual).

Help Function
The help system implementation is based on PDF files. When the user clicks the <HELP> button, a list of PDF files is shown (Figure 14). The list is stored in the file "helpmenu.inp", which is found in the DAT directory. You can edit this file for adding new PDF files to the help system

Configuration parameters in JSEISAN

The following parameters used by JSEISAN are stored in the configuration file `SEISAN.DEF`. They can be edited through the program SEISCONF. Most of them are defined to be used only with JSEISAN, but some are already defined in the SEISAN configuration file: `SEISAN.DEF` (EPIMAP_PROJECTION and EPIMAP_MAP_FILE)

OUTPUT_DIR: Output Directory for SEISAN commands results. Default "./"
EPIMAP_MAP_FILE: Name of the map file for EPIMAP command
EPIMAP_PROJECTION: Number of the projection for EPIMAP command (see SEISAN Manual)
INIT_IMGMAP_FILE: File name of the initial map represented as image. Default "/seismo/DAT/IMGWORLD
MAP_SERVER: Type of map retrieved from Internet or locally; 0: Static local image (country boundaries), 1: Static remote image (country boundaries), 2,3,4,5: Dynamic Remote image (2:country boundaries, 3:Relief from GTOPO30 only land, 4:Two minute shaded relief, 5: Combine 3 and 4). Default 0.
IMGMAP_PATH: PATH for the static local maps (images) stored in the local hard disk used for zooming. Default "/seismo/DAT/IMGMAP"
INIT_MAP_LOWER_LATITUDE: Lower latitude of the initial map. Default -90.0
INIT_MAP_UPPER_LATITUDE: Upper latitude of the initial map. Default 90.0
INIT_MAP_LEFT_LONGITUDE: Lower longitude of the initial map. Default -180.0

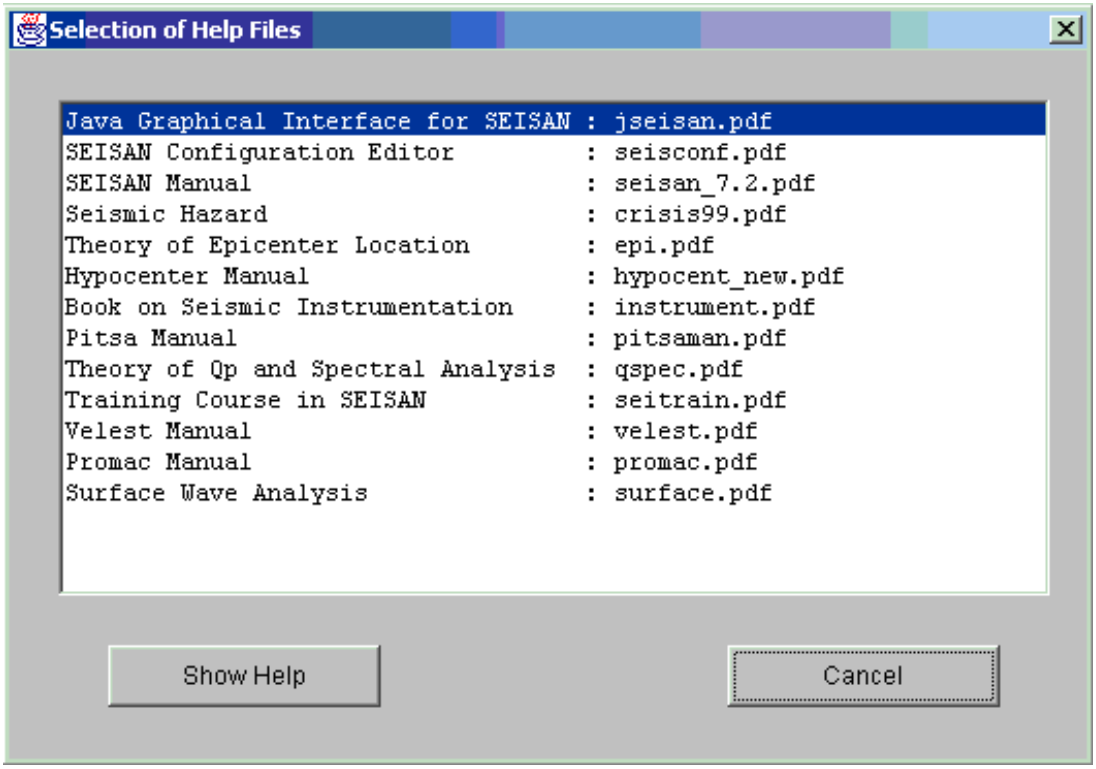


Figure 4.10: Example list of PDF files used for the help system, normally in DAT.

INIT_MAP_RIGHT_LONGITUDE: Upper longitude of the initial map. Default 180.0
ACROBAT_READER: Path for the Adobe Acrobat Reader (needed for the help files). Default "/prog/acroread" for Unix.
INTERNET_BROWSER: Location and place of browser
HELP_DIR: Directory of help files, usually INF

4.5.2 EEV Windows driver program: SEISAN

The program is an alternative to the standard EEV and it has all the functions of EEV. NOTE, the program has not been updated since SEISAN 8.2.1. The main difference compared to EEV is that is has a 'Windows type' selection of events in the database and that the most used commands in EEV and SEISAN can be executed by pressing a Button. The intention is that the majority of routine tasks in SEISAN can be done within the W95 interface without learning all the SEISAN prompt line commands.

Starting SEISAN Windows

When Windows is running, SEISAN can be started by clicking on the SEISAN icon if installed (see section 3) or writing SEISAN on the prompt line. SEISAN will start up and show a figure as shown in Figure 15. In addition to the main SEISAN window, there will also be a console window used for input and output since all underlying programs are started from the prompt line.

Working directory

Most programs read and write to the current working directory. The name of the working directory is displayed on the bottom of the screen. To change the working directory, press file selection at the top lefthand corner.

Database selection

When SEISAN starts up, it will start with the database used when it last was closed. Other databases can be selected with the 'Database Selection' menu, which also displays the current database. The choices are

1. One of the 1-5 letters databases already in existence. New databases are created as usual with MAKEREA on the prompt line.
2. A local database in the current working directory. The current working directory is displayed on the bottom of the SEISAN screen. To change the working directory use file selection at the top left hand corner.
3. An index file. The file name is selected on the menu 'Selected Index File'

Year and month selection: If a 1-5 letter database has been selected, the years and months available are seen under year and month buttons and can be selected there.

Selecting an event

Once the database has been selected, SEISAN will work much like EEV. The event window will show 12 events with the same information as seen in EEV. The total number of events for the month is shown above the event selection box. The first event in the list will be the current event. Any other event on the list can become the active event by clicking on it and it will be highlighted. Events outside the window can be displayed using the scroll bar. In addition, all EEV commands can be used including event selection commands. This works exactly as in EEV. Write e.g. 22 anywhere on the screen, press return (or click on execute button) and event 22 will be highlighted.

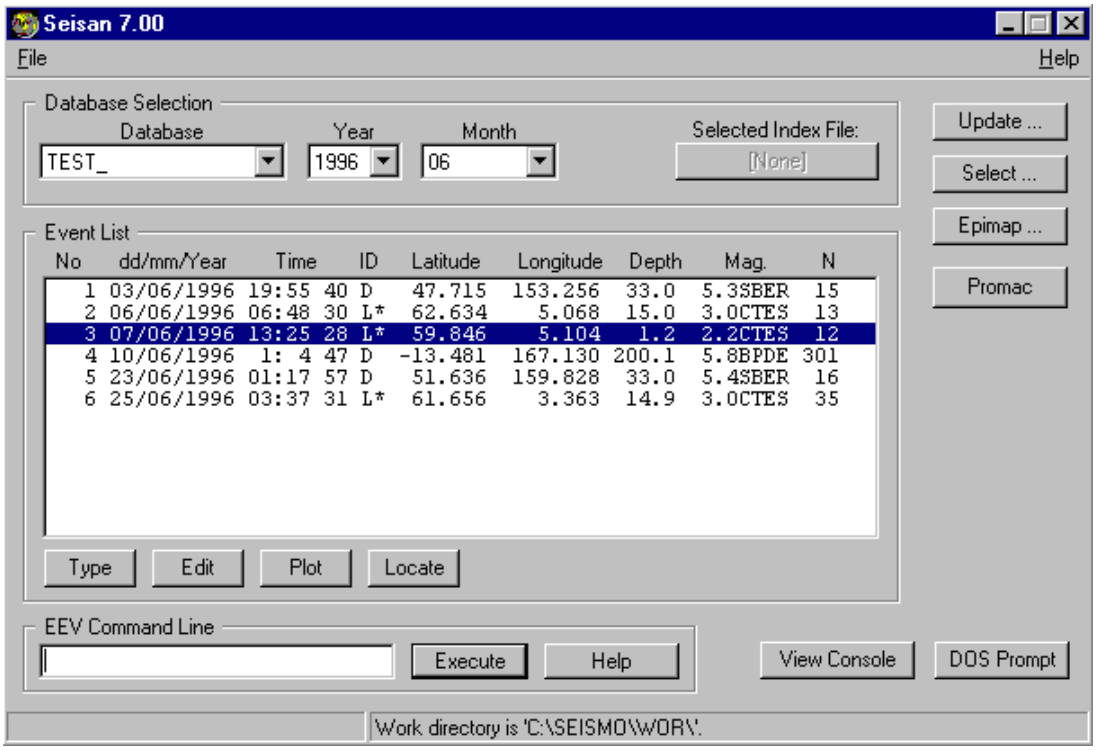


Figure 4.11: Windows SEISAN display.

Commands

All commands from EEV can be used and they are used like in EEV. Typing e.g. 'l' and return (or click on execute button), will locate the event. While typing the command, it will appear in the 'EEV Command Line' window. The command can be edited and a command can be repeated by just pressing the 'Execute' button or hitting 'return' again. However, the 4 most used EEV commands can also be executed by clicking a command button:

Type: Will display the content of the S-file, same as EEV 'T'
Edit: Edit the S-file, same as EEV command 'E'
Plot: Plot the traces, same as EEV command 'P'.
Locate: Hypocenter location, same as EEV command 'L'

In addition it is possible to display the S-file header line by double clicking on the active event. This corresponds to EEV command 'TT'.

Program output and interaction

Since all programs started by SEISAN are console based programs, the screen output and input will appear on the console window. The console window will come in the foreground if data is output or input is required. As soon as the action stops, the SEISAN window comes back to the foreground. With a large screen resolution, it is possible to see both windows at the same time. It is also possible to switch between the two windows by clicking on the 'View Console' button.

Access to the DOS prompt

Since all programs under SEISAN run in the prompt mode, it is often practical to get a DOS window on the screen. The 'DOS Prompt' button will open a DOS window in the current working directory from which SEISAN or other program can be executed. On NT, the equivalent is a console window.

Other programs

UPDATE, SELECT, EPIMAP and W_EMAP can also be started from SEISAN by clicking a button. These programs have been selected since they are often used in routine operation.

4.6 System response

The instrument response can be defined for each channel of digital data in either SEISAN, GSE, SAC or SEED response format. There are three places in the system where it can be stored. Often the instrument response is part of each channel header in the digital waveform file in SEISAN waveform format (see the Appendix B for format description). However, the instrument response is often not available at the moment the data arrives, or it is later discovered that the response given in the waveform file is wrong. There is therefore by default a directory CAL that contains one response file for each channel and for each date from which it is valid. Since the filenames contain the date from which a change in the response was made and the channel code and component code, a directory listing of CAL will give the history in chronological order of the response of a given channel. This is the most common way to use the response information in SEISAN.

Response information can also be kept in any other directory specified with the environmental variable LOCAL_CAL. The variable must be set with the full path to the directory e.g. /home/seismo/WOR/test/, or on PC, \seismo\new\cal\. On Sun it can be set in the SEISAN.csh file and under Windows by using the setting of environmental variables Control panel/system/advances/environmental variables. The variable can also be set from the keyboard (Sun: 'setenv LOCAL_CAL directory', PC, 'set LO-

CAL_CAL=directory'. This is a useful option when testing response files.

The response information gives the gain of a channel in counts/m and to get the correct ground displacement, the count values must be divided by the response values. In the current SEISAN system, only the programs MULPLT, WAVETOOL and SPEC use the response information when doing spectral analysis, generating Wood Anderson or ground motion traces. The programs will look first in the CAL (or alternative) directory for a valid response file and if not found there use the header information in the waveform file. A message will be given if the file header information is used.

If waveform files are generated on the SEISAN system from raw field station files or other input files without response information, the conversion programs (e.g. QNXSEI from a SEISLOG QNX system) will look in the CAL (or alternative) directory to find the response information to include with the Seisan waveform file. The response will be only put into the SEISAN waveform file, if the response is stored in SEISAN format. The response files are generated with RESP, see below and Appendix C.

The RESP program (section 6.32) can be used to generate the response files. The filenames for the response files are STATTCOMP.YYYY-MM-DD-hhmm_FOR where STATT is station code, COMP is component, YYYY is year, MM is month, DD is day, hh is hour, mm is minute and FOR is the format indicator which can be SEI or GSE. If FOR is not given, the format is SEISAN. An example is BER__S__Z.1999-05-05-1244.. You should take a backup of the response files before you run the program (see chapter 3).

The response files can be located in CAL, or, if many files are available optionally also in a subdirectory structure. This optional structure simply consist of a subdirectory for each station and the subdirectory name must have 5 letters so base BER would have the name BER__. The system automatically locates the response files whether all are in CAL or in the subdirectory structure.

The response file can store the response in different ways:

1 SEISAN format:

   a Parameters used for calculating the response: Generator constant, filters etc. In addition, the response (amplitude and phase) at 30 frequencies is listed. In this case the response is calculated from the parameters.

   b Incomplete set of parameters or no parameters and the response at 30 frequencies. In this case the response is calculated by interpolation of the 30 values.

   c Poles and zeros: No discrete values are given and the response is calculated directly from the poles and zeros. The number of poles and zeros in the SEISAN format is limited.

2 GSE CAL2 format:

   a Poles and zeros, number is unlimited, the response is directly calculated from the poles and zeros.

   b Pairs of frequency amplitude phase, number of pairs is unlimited, the response is calculated by interpolation.

3 SEED

   – Poles and zeros (only Transfer function type A (Laplace Transform (Rad/sec)), number of poles and zeros are unlimited, the response is directly calculated from the poles and zeros. Only reads SEED response in ASCII format as written out by rdseed, not dataless SEED volumes. The command to extract the files is `rdseed -R -f seed_filename`. Standard filename such as `RESP.IU.TRIS.10.BHZ` are understood. Files are read from CAL directory. Note that if e.g. the horizontal channels are rotatede the `-R` flag by `rdseed` will not provide this information.

4 SAC format: SEISAN can use SAC PAZ files as created by rdseed. The files have to be names in the standard SEISAN way, but have to end on `_SAC`.

When rotating signals, it is assumed that the response is the same on all 3 channels. !!!!

Response files can be plotted from MULPLT showing the actual response information that is used with a given trace. Response files can also be plotted directly with program PRESP, see below.

All or a subset of the response files can be printed out in a table with program PR_RESP. The program must be executed from the directory with the response files. Make a listing (file filenr.lis) of files to print out with DIRF and run the program. It will produce an output file ready for printing.

A response file can be plotted with the program PRESP. The program is started with commandpresp filename, where filename is the response file name. If no file name is given, the program asks for a filename or number. If a DIRF has been made and the list of files in filenr.lis is available, a response file can then be selected with a number. The program produces a PostScript output file with name `presp.eps`.

4.7 Working with catalogs

It is often convenient to have multiple solutions of hypocenters in the database S-files or the CAT-files. Typically data has been entered from different sources and merged to form a single catalog. The first hypocenter line in the file is then considered the prime hypocenter estimate and this is the one used by e.g. EPIMAP to plot the hypocenters. The order of the hypocenters can be rearranged by CAT_AGA. Several programs use all the hypocenter lines. The magnitude correlation program will search any hypocenter line and the database selection program SELECT will optionally also use all the hypocenter lines. When the data base is updated with a new location and magnitude (UPDATE, section 6.7), it is only the first hypocenter line which is overwritten. If there is a magnitude in the 3 position, it is left unchanged unless it has the same agency as used for updating. This is useful in normal observatory practice, where it is common to put in some external agency magnitude which then must be left unchanged. If more magnitudes than 3 are calculated, they will be placed on a subsequent hypocenter line identified by having the same year, month, day and hypocenter agency as given on the first line. In order to merge different catalogs, it might be an advantage to put all the data into a complete database where each event is one file, even when only hypocenters are available. This is done by first splitting up the catalogs with SPLIT and then using EEV to merge the events. Since there is no requirement for monthly directories to have data, this methodology can also work for historical catalogs. The data can then subsequently be put into the CAT database without relocation using the UPD command.

4.7.1 Explosions in SEISAN

Many catalogs are contaminated by explosions and in SEISAN, explosions can be dealt with in several ways. In the data base, confirmed explosion are marked with E and probable explosions with P. These indicators are mostly put in when the operator first registers the event. However, there is also a possibility to automatically identify events which are probable explosions. This is done with program EXFILTER (section 6.27). In the data base S-files, there is a special format for recording explosion information (command EXP in EEV). The explosion site there can be assigned a three letter code, which can be used by SELECT to find explosions from specific sites. In this format it is also possible to store the explosion charge and explosion location and origin time separately from the calculated location and origin time.

4.8 Printing

All SEISAN programs, that produce graphical output, also generate Postscript files with the file suffix eps (note this was plt before version 8.1). These can be directly sent to a Postscript printer. It seems that programs like Microsoft Word don't like the SEISAN Postscript and you will need to convert your files to another Postscript, this can be done for example with the program ghostscript using pswrite as output device.

Note: On Solaris 7, both the lpr and the lp command for sending files to the printer, don't create a copy of the file before sending it (bug in Solaris). This means that a plot file can be overwritten before being sent to the printer. Therefore when SEISAN on Unix is sending plots, the system waits for 5 seconds after a file is sent to the plotter before continuing. This is most important when plotting continuous data or a large number of files with MULPLT.

4.9 General Work with SEISAN

Once data is in the database and the routine analysis has been finished by running UPDATE (final epicenters recorded in CAT and the S-files), it is possible to go on with general work with the data. This means searching the database, making a bulletin or plotting the epicenters. It is also possible to use some of the more specialized tools of SEISAN which include working on subsets of data or creating other databases, see 4.5. For general use, the basic philosophy is that the user should not enter the REA directories. All commands and programs should be used from the user's own directory or the WOR directory. To access part of the main database, the programs always ask for start and end date as follows:

19880602011001	: including from or to the second
198806020110	: including from or to the minute
1988060201	: including from or to the hour
19880602	: including from or to the day
198806	: including from or to the month
1988	: including from or to the year
BLANK	: only used as end date, means to end of month

Note that the end time is inclusive, this means that e.g. 198806 includes all of June 1988.

Thus most programs will work from any given date-time to any other given date-time. Programs that work directly on the S-files in the database (e.g. COLLECT) can work with any time interval in which the database structure has been created. THERE IS NO REQUIREMENT THAT THERE IS DATA IN THE INDIVIDUAL MONTHLY DIRECTORIES, ONLY THAT THEY EXIST. There are usually 4 options for database, either the standard base (often by default), the user's own subset of the standard base (an INDEX file or S-files in local directory) or another database. If the user has his own database specified by an INDEX file, the event ID's must be in that INDEX file. Since the index file gives complete file name of event files, the index file can work on a subset of the main database.

Note that most of the programs are used as stand-alone programs, disregarding the database structure. If one for example prefers to have all events gathered in one file rather than split into many files and directories most programs will therefore work.

4.10 Graphics in SEISAN

Most programs in SEISAN producing graphics on the screen use the SEISAN graphics system (see also chapter 7). This produces fast and low quality graphics both on the screen and similar PostScript output files. Most of these plots are not suitable for publication and many programs therefore also create output ASCII files of the main results, which then can be put into more professional plotting routines. The GMT (Generic mapping Tools) system is one of the more widely used plotting systems used in seismology. Several programs in SEISAN therefore produce output that can be used with GMT or makes plots directly in GMT. From version 8.0 of SEISAN, a script is included (GMTXY, manual in INF) which will produce nice xy-graphics from specially made output files. So far only programs SPEC, CATSTAT and LSQ produce these output files, the intention is to include this feature in more SEISAN programs.

If there is a need to produce better quality graphics there are several possibilities:

Maps: GMTMAP, Unix (must be installed separately, on CD

W_EMAP: Windows based mapping system
Seismograms: TRACEPLOT (GMT based)
XY-plots: GMTXY (GMT based)
Maps: create GMT input files with SEIGMT
Volcanic event distribution: VOLCSTAT

Chapter 5

SEISAN TRAINING COURSE

The document 'Computer exercises in processing earthquake data using SEISAN and introduction to SEISAN' which is a tutorial for new users as well as experienced users, is included in the distribution. The testdata used in the exercises need to be installed, see chapter 3. Going through the exercises of the tutorial might be the best way to learn SEISAN. The document is given as PDF file (`seitrain.pdf`) in the INF directory.

The main goal of the introductory training course is to become familiar with the database program EEV, the plotting program MULPLT and the location program HYP. Of course additional reading of relevant sections in this manual is required. The basic exercises can be completed within one or two days, while the advanced exercises take more time.

Content of the exercises

1 SEISAN basic exercises

2 Phase reading

3 Response files and seismic formats

4 Signal processing

5 Earthquake location

6 Magnitude

7 Focal mechanism

8 Spectral analysis and Q

9 Operation and earthquake statistics

10 Array analysis

11 Analysis of a data set

12 Data manipulation and import and export of data

Chapter 6

DESCRIPTION OF PROGRAMS AND COMMANDS

This section gives user manuals for programs and command procedures used with SEISAN. Not all are as detailed as one could want, however many questions from programs should be self-explanatory. Most programs will produce output files with the extension .out and proceeding by the name of the program. E.g. output from collect, will be `collect.out`. Running a program twice will erase the earlier output files. If these files are to be used later, remember to rename them before running a program again. There are several programs, which have separate manuals in the INF directory.

6.1 Hypocenter location programs: HYPOCENTER, HYPO71 and HYPOINVERSE

6.1.1 The hypocenter program, HYP

The hypocenter program is a modified version of HYPOCENTER [Lienert et al., 1986; Lienert, 1991; Lienert and Havskov, 1995]. The main modifications are that it can accept more phases, locate teleseismic events and use input in Nordic format directly from the database. A detailed manual (earlier version, `hypocent.pdf`) and some of the later changes
(`hypocent_latest.pdf`) is given in INF directory. The input parameter file with station coordinates, model etc. is `STATION0.HYP`, see later.

Local crustal phases:

The program will accept P, Pg, Pn, S, Sg, Sn, Pb, Sb, Rg, T and Lg phases and when locating teleseismic events most of the IASPEI phases (see below). If only P or S is given, the fastest phase is used as in the original version of the program. The phase used by the program is indcated in output, see later.

Azimuth and single station location:

The program also uses observed station azimuths as given in the Nordic Format. Station azimuths can be obtained with either 3-component stations or array stations or by using a local network as an array (see EEV pfit option) This means that the program can locate with one station if it has at least two phases like P, S and azimuth. Azimuth residuals contribute to the overall rms, see TEST(52) and section

on weight. In order to locate with one station, azimuth and P and S, TEST(56) MUST be set to 1. Note that the depth then will be fixed to the starting depth. So if the starting depth is larger than the hypocentral distance, no solution is possible and the starting depth must be set to a value smaller then the hypocentral distance. This can be done in the `STATIONO.HYP` file or individually in the S-file. Known problem: If Azimuth on one station and P and S on another station, HYP might not locate properly.

Magnitudes:

In SEISAN version 8.3, there are substantial changes in the way amplitudes are read and two new magnitude scales have been added (broad band body and surface wave magnitudes). Furthermore, the Richter attenuation curve is now used be default for the body wave magnitude. The phase names used for amplitudes have also changed. These changes are due to the new standards for magnitude calculation approved by the IASPEI. For more on the application of the different magnitude scales, see Havskov and Ottemöller [2010].

Magnitudes are calculated using coda, amplitude and spectral level. Parameters are given in the station file using the RESET TEST variables. For magnitude based on amplitude, the amplitude must be given in nanometers in the input file (SEISAN standard).

Local magnitude Ml

The formula used to calculate local magnitude is

$$Ml = a * log_{10}(amp) + b * log_{10}(dist) + c * dist + d$$

where a,b,c,d are constants, log_{10} is logarithm to the base 10, amp is maximum ground amplitude (zero−peak) in nm and dist is hypocentral distance in km (RESET TEST 75-78). The default constants are for California [Hutton and Boore, 1987] which gives the following relation

$$Ml = log_{10}(amp) + 1.11log_{10}(dist) + 0.00189dist - 2.09$$

It is here assumed that the gain of the Wood-Anderson instrument is 2080. An amplitude of $1mm$ of the Wood Anderson seismogram is then $10^6nm/2080$ and inserting this amplitude above together with a distance of 100 km gives magnitude 3 as originally defined by Richter. It is assumed that the maximum amplitude is picked on a seismogram simulating the original Wood-Anderson seismogram, see program MULPLT. SEISAN uses hypocentral distance, while the original Ml scale used epicentral distance (no deep earthquakes in California). We use hypocentral distance so Ml also can be used for deep earthquakes, but the user should be aware that the Ml relation for deep earthquakes might be different from the relation for shallow earthquakes.

Local magnitudes are only calculated for events with epicentral distance LESS THAN TEST(57) (default 1500 km) and if the period is less than 5.0 secs. All amplitudes for the phases 'L', 'S ', Sg, SG, AMP, and AML, AMPL or blank are used. This means that if an amplitude is picked on both Lg and Sg, both will be used. The period is not used. The many possible phase names is a result of changes over time and thus to ensure that Ml is calculated correctly with older data. The current version of MULPLT (8.3) produces the standard IASPEI name IAML.

Coda magnitude Mc

The coda magnitude is calculated using

$$Mc = a * log_{10}(coda) + b * dist + c$$

where coda is coda length in secs and a,b and are constants (RESET TEST 7-9). If 'a' is given as a negative number, the following formula will be used

$$Mc = abs(a) * log_{10}(coda) * log_{10}(coda) + b * dist + c$$

If both Mc and Ml are calculated, Ml is written first on the header line.

Coda magnitude is only calculated if the epicentral distance is less than TEST(57).

Surface wave magnitude Ms

Ms is calculated using the standard

$$Ms = log_{10}(amp/T) + 1.66log_{10}(dist) + 3.3$$

where T is period. Amplitude is in micrometer and distance in degrees, however in the Nordic format nm and km are used and the program converts. Ms is only calculated if the period is larger than 10.0 seconds in which case the program automatically assumes that Ms is the wanted magnitude. The phase used can be AMS, AMP, IAMS20 or blank. The current version of MULPLT produces the standard IASPEI name IAMS_20. The many possible phase names are a result of changes over time and thus to ensure that Ms is calculated correctly with older data. It is assumed that the amplitude has been picked on a WWSSN standard LP trace and that the period is in the range $18 - 22s$ (see program MULPLT). Ms will be calculated even if the period is outside this range, but it will not be correct according to the standard.

Broadband surface wave magnitude MS (IASPEI code MS_BB, but SEISAN uses MS for simplicity, new from SEISAN version 8.3)

MS is calculated using the standard

$$MS = log_{10}(amp/T)_{max} + 1.66log_{10}(dist) + 3.3$$

or

$$MS = log_{10}(V_{max}/2\pi) + 1.66log_{10}(dist) + 3.3$$

where V_{max} is the maximum velocity. The IASPEI definition is to use velocity and the period is thus not needed but read for information. The velocity is in micrometer/s and distance in degrees, however in the Nordic format nm/s and km are used and the program converts when calculating magnitudes. MS is only calculated if the period is larger than 3 seconds and less then 60 seconds, distance must be larger than or equal to 222 km (2 degrees) and less or equal to 160 degrees. The depth must be less than 60 km, however there is no check for that in SEISAN. The phase used to report the amplitude and period must be called IAMSBB which the current version of MULPLT produces. The biggest advantage using MS compared to Ms, is that any period in the range $2 - 60s$ can be used.

Body wave magnitude mb

mb is calculated using

$$mb = log_{10}(amp/T) + Q(dist, depth)$$

where Q is a hardwired function of distance and depth and amp is the amplitude in nm. There are two possibilities: The default (set by REST TEST(108) is the standard Gutenberg and Richter (1956) curve while alternatively the Veith-Clawson curve can be used [Veith and Clawson, 1972]. Before SEISAN version 8.3, Veith-Clawson was always used. mb is only calculated if the epicentral distance is less than or equal to 100 degrees and larger than or equal to TEST(57) (IASPEI standard and SEISAN default is 21 degrees) and the period must be smaller than 3 s and the phase is P, AMP, AMb, AMB, AMPB, AMPb, blank character or IAmb. The current version of MULPLT produces the standard IASPEI name IAmb. The many possible phase names are a result of changes over time and thus to ensure that mb is calculated correctly with older data.

Broad band body wave magnitude mB (new from SEISAN version 8.3)

The broad band magnitude mB (official IASPEI name is mB_BB) is calculated using

$$mB = log_{10}(amp/T)_{max} + Q(dist, depth)$$

or

$$mB = log_{10}(V_{max}/2\pi) + Q(dist, depth)$$

where V_{max} is the maximum velocity and Q is a hardwired function of distance and depth. The IASPEI standard is to use velocity and SEISAN store the velociyt in nm/s. There are two possibilities for the atteneuation function: The default (set by RESET TEST(108) is the standard Gutenberg and Richter (1956) curve while alternatively the Veith-Clawson curve can be used [Veith and Clawson, 1972]. mB is only calculated if the epicentral distance is less than or equal to 100 degrees and larger than or equal to TEST(57) (IASPEI standard and SEISAN default 21 degrees) and the period is larger than 0.2s and less than 30s and the phase name is IVmB_BB . The current version of MULPLT produces the standard IASPEI name IVmB_BB. The biggest advantage using mB compared to mb, is that the mB scale does not saturate before magnitude 8.

Moment magnitude Mw

Mw is calculated as

$$Mw = 2/3 * \Big(log_{10}(moment) - 9.1\Big)$$

where moment is in Nm (see also section 6.2.12). When an event is relocated, the moment is also recalculated according to revised hypocentral distance.

NOTE: If an amplitude has a given period between 5 and 10 secs, it is not used for Ml and mb magnitude calculation, see above. If an event is not located, there will normally be no magnitude calculation and all magnitude and distance information is deleted from the output S-file (`hyp.out`) except, the magnitude in the 3rd position on the header line if it has an agency different from the default agency. The only exception is that if a coda is given, the epicentral distance is retained and coda magnitude will therefore be calculated. This means that for events, which cannot be located, it is still possible to calculate coda magnitudes by manually entering the epicentral distance on the line containing the coda length.

On the first header line, there is room for 3 magnitudes. If there is a magnitude in the 3rd position, it is not overwritten unless the default agency is overwritten, so there will often only be room for 2 calculated magnitudes on the first header line. If more magnitudes are calculated, they will be written on a subsequent hypocenter line, which is identified by having the same year, month, day and hypocenter

agency as the first header line. This means that there is room for a total of 6 magnitudes, which can each, be updated when relocating. Hypocenter info and all 6 magnitudes can be printed out on one line with program REPORT.

All magnitudes can have a station dependent correction given in the station file. This correction does not affect the Mc in `print.out` file. Mb and mB use the same correction and Ms and MS use the same correction. Only calculate magnitude: If TEST(106) is set to 1.0, only magnitudes are calculated, provided a distance is given.

Use of S-P and L-S differences:

Uncertainty in absolute times often makes it necessary to be able to use the difference in time between two arrivals such as P and S or P and L. If no absolute times are available, the calculated origin time will be close to that at the first arrival station and is of course meaningless. However, a perfectly good epicenter and depth can still be obtained from P-S or P-L differences alone. To enable this feature, set the weight for the P phase input record to 9. This P is then assigned a weight of 0, effectively disabling its use. However, a time residual and azimuth, etc., will still be calculated for it, enabling an assessment to be made of its absolute time. A search will then be made of the entire input phase set for an S or L phase at the same station. If such a phase is found, its variables are used to store the observed and calculated difference times and their derivatives, and it's weight (0-4) is used for the difference phase. DON'T SET IT TO 9!! If two or more such phases (e.g., SN, SG, LG, etc.) are found, all their differences with the P time will be used instead of their absolute times. Blanks will appear beneath 'hrmn' in the residual summary for all such phases, while the observed and calculated difference times with the first P will appear beneath 't-obs' and 't-cal'.

NB. There must be at least one phase with absolute time to get a location.

Global event location:

When locating globally, the program uses the IASPEI91 travel time software described by Buland and Chapman (1983) and Kennett and Engdahl (1991). HYP evaluates all the IASPEI91 phases (up to 60) at each delta, and searches for the phase specified in the 4-character phase identifier. If no phase is found, the phase is given a weight of -1, which effectively removes it from the phase set. If a phase is labeled as 'P ', 'S ', 'PKP ' or 'SKS ', and this phase is not in the IASPEI91 list, the first arrival phase having P or S as its first letter is used, or PKP, SKS as its first 3 letters. In addition, include the PKiK phases in this search for 'PKP ' and 'SKiK' phases in the search for 'SKP '. The IASPEI91 phase set currently includes: P, Pdiff, PKP, PKiKP, pP, pPdiff, pPKP, pPKiKP, Sp, sPdiff, sPKP, sPKiKP, PP, P'P', S, Sdiff, SKS, sP, pSdiff, pSKS, Ss, sSdiff, sSKS, SS, S'S', PS, PKS, SP, SKP, SKiKP, PcP, PcS, ScP, ScS, PKKP, PKKS, SKKP, and SKKS.

Long phase names:

Normally SEISAN and the Nordic format assume up to 4 character phase names. However, when working with global phases, the phase name length can in a few cases be up to the ISC standard of 8 characters. The program then uses column 9 for weight (normally blank) and column 11-18 for the phase. In this case it is not possible to give a polarity.

Criteria for a solution:

The cases where a solution will not be attempted are as follows:

1. Multiple phases at two stations, but no azimuths. This is a non-unique case, even though four different arrivals are present.

2. Less than three phases from three different stations and no azimuths.

3. A single phase at one station with an azimuth.

Note that if phases are weighted out due to large distance or a bad fit during the first iteration, there might not be a location even if more than 3 stations are available.

Weighting:

A number of different weights may be used to calculate the solution.

1. User specified weights: These are calculated using the HYPO71 style weight number 0 to 4, read with each phase, where 0 corresponds to w1=1.0, 1 to w1=0.75, 2 to w1=0.5, 3 to w1=0.25 and 4 to w1=0. Uncertain time is 9 meaning that absolute time is not used, see also use of S-P times on previous page.
2. Distance weighting: This is given by the formula w2=(xfar-delta)(xfar-xnear) where delta is the distance (km) of the event from the station and xnear and xfar are read from the station file, `STATION0.HYP`.
3. Bisquare weighting: This scheme, described by Anderson [1982] calculates residual weights, see details in HYP manual. Used for distant events.

4 Azimuth weighting: Azimuth residuals are divided by test(52), which is the error in azimuth that corresponds to a one-second error in arrival time. For example, if test(52)=5 (default), a phase residual of 5 degrees will become a residual of 1 (5/test(52)) in the parameter corrections and rms calculation.

All the above weights are multiplied together to calculate the weight used in the inversion. If the user-specified weight, w1, is changed by (2) or (3) above, changed to zero by the consistency check, or set to -1 because the phase is not recognized, an asterisk will appear after the final weight in the residual printout.

Determining which travel time software is used:

The parameter test(57) is used to determine whether a layered model or IASPEI91 software is used to calculate the travel times and their derivatives. For the initial starting location, the distances from each station are calculated and IASPEI91 is used if any of them exceed test(57). However, this can be overridden by the distance indicator in column 22 of the Nordic header record. If this is L, a crustal model is used regardless of distance, whereas if it is D, IASPEI91 is used, while R has no effect i.e., test(57) is still used. So if either a crustal model or IASPEI91 tables are wanted, use either L or D respectively.

Starting epicenter location:

The program uses a starting location algorithm (reset test(56)) which tests the rms of all starting locations and select the minimum rms solution, see HYP manual.

User defined start location: If an S is written in the input S-file at column 45 of the epicenter line, the location starts at the location (epicenter) given on the header line. If an S is written in column 44 on header line, the depth iteration will start at depth given on the header line. If N is written in column 45, the nearest station will be used irrespective of global settings.

Starting depth:

If no event specific start depth is given in S-file, the starting depth is taken from the first number on the control line (see later) in the HYPO71 style. However, there is often problems obtaining a reliable depth due to local minima. This can be manually checked with program RMSDEP from EEV. HYP can also

be set up to locate the same event starting with a range of different start depths, and then choose the one with the lowest RMS. This can significantly improve the reliability of depth determination. Selecting 3 to 5 different start depth is often enough. This option is set on the control line in the station file.

Fixing location:

Using F instead of S, fixes the position (depth and location).

Do not locate event:

If a * is written in column 45, the event is not located, can be used if an external location is to be kept unchanged.

Only calculate magntudes and update spectral values

Set TEST(106) to 1.0

Fixing origin time:

Using an F in column 11 of header line will fix the origin time given on the header line.

If both depth and location are fixed, but not the origin time, new origin time and residuals will be calculated. This can be useful when working with readings from a few stations which should be checked against known locations. If e.g. distant events are read, it is often the practice to put in the PDE location on the header line and calculate residuals relative to the observations. When the UPDATE is made, the agency of the location is NOT changed, assuming that if both depth and epicenter are fixed, the hypocenter must come from an external agency.

Alternative model:

By default, an event is located using the `STATION0.HYP` input file. However, each event can use its own model (with all the location parameters) which is specified with one character in column 21 on the Nordic input file header line. The model then has a corresponding name. If e.g. the model is called W, the corresponding input station file will be called `STATIONW.HYP`. It is therefore possible to have as many different station files, as there are printable characters. Note that if a different model x has been specified and is not present, the program will stop with the message "`STATIONx.HYP does not exist`". The file `MODEL.DEF` in DAT can be used to assign the single character a name, which can be listed from EEV. The format in `MODEL.DEF` is one line per model, the model indicator is given in column 1, column 2 is blank and the model name is given in columns 3 to 80.

Using HYP to determine crustal structure

HYP has an option to locate a data set for a large number of different models and then determined which model gives the lowest average RMS for the data set. This might be a useful option, particularly when a sparse data set is available. In order to use this option, an additional input parameter file `h_models.par` is given. When this file is in the working directory, HYP will switch to multiple model mode SO ONLY HAVE THIS FILE IN WORKING DIRECTORY IF MULTIPLE MODEL MODE IS INTENDED. When using this option, all events must use the same `STATIONx.HYP` file, otherwise the program fails. The input MUST be from a single file, NOT from the data base. Below is an example of an input file

```
layer #  start vp  delta vp   # delta   start h   delta h   # delta
      1      4.55       0.1         5       0.0       1.0         1
      2       6.3       0.1         5       4.0       1.0         1
      3       6.8       0.1         5      22.5       1.0         1
      4      7.90      0.05         3      32.5       1.0         1
      4      8.05      0.05         4      40.0       1.0         1
```

The first line is info only. Layer # is also only for information. For each layer, there is a start P-velocity (start vp), increment in velocity (delta vp) and number of increments (# delta). The following inputs are then the same for layer depths. There must be an entry for each layer even if no variation is used. In the above example, no variation in layer thickness is tested for. An example input file is given in DAT. The parameters for location not set in `h_model.par` like Vp/Vs, Lg velocity etc remain unchanged. When HYP starts up, it will `print out` how many permutations are required. If more than a few thousand, reduce the number of models. In any case it is an advantage to first try with just a few models to get a feeling for how sensitive the data is for model changes.

An output file `h_models.out` is generated, see example below. For each model tested, one output line is given with the RMS and the model. In the example below only the last 5 models are shown. Since many models can have very similar average RMS, the best 10 models are printed at the end.

```
0.946    4.95  0.00  6.70  4.00  7.20 24.50  7.90 32.00  8.20 40.00
2.607    4.95  0.00  6.70  4.00  7.20 24.50  7.90 32.00  8.30 40.00
0.934    4.95  0.00  6.70  4.00  7.20 24.50  8.00 32.00  8.10 40.00
0.994    4.95  0.00  6.70  4.00  7.20 24.50  8.00 32.00  8.20 40.00
2.677    4.95  0.00  6.70  4.00  7.20 24.50  8.00 32.00  8.30 40.00

Minimum rms   0.764057

The best models

0.771    4.95  0.00  6.50  4.00  7.10 24.50  7.80 32.00  8.10 40.00
0.766    4.85  0.00  6.50  4.00  7.00 23.50  7.80 32.00  8.10 40.00
0.767    4.85  0.00  6.50  4.00  7.00 24.50  7.80 32.00  8.10 40.00
0.769    4.85  0.00  6.50  4.00  7.10 23.50  7.80 32.00  8.10 40.00
0.766    4.85  0.00  6.50  4.00  7.10 24.50  7.80 32.00  8.10 40.00
0.772    4.85  0.00  6.50  4.00  7.20 24.50  7.80 32.00  8.10 40.00
0.771    4.95  0.00  6.50  4.00  6.90 22.50  7.80 32.00  8.10 40.00
0.771    4.95  0.00  6.50  4.00  7.00 22.50  7.80 32.00  8.10 40.00
0.770    4.95  0.00  6.50  4.00  7.00 23.50  7.80 32.00  8.10 40.00
0.771    4.95  0.00  6.50  4.00  7.00 24.50  7.80 32.00  8.10 40.00
```

Running HYP:

The program is started with command HYP from the prompt line (interactive mode) or with 'L' in EEV. HYP can also be started with an argument like `hyp input.dat`, where `input.dat` is an S-file. The first event in the S-file will then be located without further user interaction. Below follows an example of running outside EEV, explanations are in lower case. Note that the `STATION0.HYP` file MUST be present in the DAT directory for HYP to know that it is working with a SEISAN database. If not present, HYP will only ask for an input file name, see HYP manual.

```
HYP
Arrival time data input, select one:

  SEISAN database or                              : RETURN
  Alternative database, give 1-5 letter code      :
  Local index file, name must start with index or :
  Local database, write ,, or                     :
  File name for one file in NORDIC format         :

                 Your answer here  determines  the input
source. A return means that you work directly on the BER database. A 1-5 letter
code gives name of database, e.g. NAO. An index file or the name of a readings
file  is used when you want to work on specific subsets.
                   Local database is S-files in local directory.

 Start Time           (YYYYMMDDHHMMSS) : 199012
 End Time, RETURN is to end of month   : 19901205
           Standard formatted time input.

 Interactive operation (N/Y=return)
                 If N, whole time interval or file is located, one line output pr event.


#  1   1992 12 3 0137 40.3 NPHS=   12  T Q L #XXX
#  2   1992 12 3 0237 43.3 NPHS=   14  T Q L #XXX  l   ! now locate
here comes location, see HYP manual*****************************
#  2   1992 12 3 0237 43.3 NPHS=   14  T Q L #XXX  q   ! stop


 PRINT OUTPUT IN FILE print.out
 CAT-FILE IN FILE hyp.out
 Summary file in hypsum.out
```

In interactive mode, as shown above, event date is printed out for each event and action is taken as in EEV for the options available. If HYP run on a single file, the options above are available meaning that HYP can select and locate different events in a single file using the event number. If HYP runs on a database, the EEV options D and B are also available, but not shown. If the option of no interactive input is chosen, the program will locate from beginning to end without any more user interaction. This is a useful option for testing a subset of the database with different models etc. without changing the database. Note that the input file or database is never overwritten by HYP.

ALL TYPE ONE LINES WITH SAME AGENCY AS GIVEN IN `STATIONX.HYP` FILE WILL BE DELETED SO THERE WILL NEVER BE MORE THAN ONE TYPE 1 LINE IN OUTPUT WITH CURRENT AGENCY (except possibly a second magnitude line with a different type magnitude as given on main header line).

Problems: Sometimes HYP will not locate an event, look in the `print.out` file to see what happened. In some cases, the initial location was put beyond the limits set by the parameters. If e.g. an event is defined as a local event and no readings are to be used further away than 2000 km (distance weighting, see following table or TEST(41)) then no location will be attempted. Try to change the event type to D

and see if the event locates. In a few other cases it might be an advantage to use a starting location.

Station and model files:

Station input is given in near standard HYPO71 format in the file `STATION0.HYP` in directory DAT. If however the user wants to try a different model without changing the standard model in DAT, this is possible by having a `STATION0.HYP` file in the working directory, since the program always looks there first for the `STATION0.HYP` file (see example at end of this section). Another possibility is to use another model for just one event by setting a flag in the phase input file, see below.

Below is an example of a `STATION0.HYP` file. The format is close to the HYPO71 format with one extra line at the bottom. The test parameters 2-13 are as in HYPO71, see also HYPOCENTER manual section 4.1.2.

Comments are given after !'s

```
RESET TEST(01)=0.3
RESET TEST(03)=0.6
RESET TEST(06)=0.1
RESET TEST(07)= 3.0
RESET TEST(08)=2.6
RESET TEST(09)=0.001
RESET TEST(11)=50.0
RESET TEST(13)=5.0
RESET TEST(50)=1.0
                                ! one and only one blank line here
  UPP 5951.50N 1737.60E  14     ! station lines
  COP 5541.00N 1226.00E  13
  KBS 7855.08N 1155.44E  46
  EBH 5614890N  330490W 375     ! high accuracy lat-lon
  OSG 6029.80N  252.55E-100
  01A06049.43N 1049.95E 426
 BERGE6057.12N 1133.15E 100     ! 5 char station name
-BEBGE6157.12N 1133.15E1100     ! 5 char station name and at 1100 m

...

                                ! one and only one blank line here
  6.2       0.0         !       ! model lines
  6.6      12.0
  7.1      23.0           3.8       2.2       200.0     300.0 **
  8.05     31.0      N  ! N  indicates location of Moho
  8.25     50.0
  8.5      80.0         !
15.  600. 1300. 1.73    5  5.0  10.0     ! control parameters
BER                     ! Reporting agency    (a3)
```

Format of the station line is 2x,a4,i2,f5.3,a1,i3,f5.3,a1,i4,f6.2,5f5.2,9f6.2 or 1x,a5 .... if the station has 5 characters. The content is:

station code 4-5 chars (see above)
latitude in degrees
latitude in min

north or south (N or S)
longitude in degrees
longitude in minutes
east or west (E or W)
altitude in m, in some rare cases, the station is deeper than 1000 m in which case the minus sign has to be put in column 1
P-delay in secs, S-delay is the same multiplied by Vp/Vs as given below
Magnitude corrections for the magnitudes: Mc, Ml, mb or mB, Msi or MS and Mw
Spherical harmonic station corrections

The magnitude residuals are added to magnitudes calculated for each station but the result is only seen in the final average magnitude. If the magnitude correction is set to 99.0, the magnitude is not used in the average. The magnitude corrections for mb and mB are the same and similarly also for Ms and MS.

Format of model line: 3f7.3,a1. The information is:
P- velocity (km/sec)
Depth to interface (km)
S- velocity (not needed)
Interface indicator: N: Moho, B: Conrad

NB: Moho cannot be the last layer, there MUST be one layer below interface marked with N.

The line with ** indicates optional Vs, density, Qp and Qs. This is information only used with modeling, see section 6.19. Format for additional info is 25x,4f10.1.

Format of control line: 3f5.0,f5.2,i5,2f5.1 Information is:
start depth in km, used if no range of start depths specified (see below)
xnear: distance at which distance weighting start
xfar: distance at which distance weighting is zero, beyond xfar, the phase is not used (local events only)
Vp/Vs ratio
number of start depths
start depth of range of start depths
increment in start depths
NB: If these parameters are used, the fixed initial start depth is not used
The input at the bottom is reporting agency used for both hypocenter and magnitudes.

Since the program locates distant events, max distance, reset test(41) must be set to a large value. To avoid that local events move out in the blue, the parameters xnear and xfar must be set not larger than 2000 to 3000 km. Xnear and xfar are only used for local events (flag L) and regional events if the local crustal model is used.

RESET TEST parameters:

HYP will assign reasonable default values for RESET TEST parameter. Below is shown a summary. For full details see HYP manual. The number to the left is the control parameter and D indicates the default value. The most important parameter are given in bold.

2: Step length damping control, D: 500.0.

7-9: Duration magnitude coefficients used for calculating the coda magnitude, as MAG = TEST(7) + TEST(8) * LOG(T) + TEST(9) * DELTA where T is the coda length in seconds, DELTA is the hypocentral distance in km. D: 7: -0.87, 8: 2.0, 9: 0.0035 [Lee et al., 1972] If test(8) is negative, its positive value will be used and log(T) will be squared. Note however, that the individual stations magnitude values printed out during the run of HYP still will be using the unsquared log(T).

11: Maximum no of iterations in the least-squares rms minimization, D: 99.0

13: Increment in km for auxiliary rms, D: 20.0 km. To disable (save some computation time), set to 0.0.

30: Initial damping factor, D: 0.005

31: Max degs of freedom: Set to 3 for determining origin time and hypocenter, set to 2 for fixed depth solution (depth on phase headers), -2 fix all events to starting depth in `STATIONO.HYP`, 1 to fix all hypocenters to value on phase headers, 0 to fix hypocenters and origin times to values on phase headers. D:3.0

32: Magnitude of parameter changes (km) below which convergence is assumed, D: 0.05

34: Minimum spread to normalize residuals, D: 0.1, do not change

35: Bisquare weighting width, D: 4.685, do not change

36: RMS residual low limit for bisquare weighting for local events, D: 0.0

37: Maximum number of increases in damping before fixing depth, D: 10.0

38: Least squares errors (0.0), damped least squares errors (1.0) with initial test(30) damping value, D: 0.0

39: Factor by which damping is increased when RMS increases, D: 4.0

40: Depth origin of coordinate system, 0: sea level, 1:maximum elevation station in station list, D: 0.0

41: Maximum distance (km) from nearest station at which hypocentral solutions will be generated, D: 20000.

43: Maximum rms for an event to be used in average station residual calculation - doesn't affect the final hypocenter solution, D:1.5

44: Rg phase velocity in km/sec, D: 3.0

45: Minimum rms difference between the location on the header line and the new location for the event to be used for average difference in location, D: 50.0

46: Minimum number of non zereo weight phases for event ot be included in average difference in location, D: 3.0 Prevent depth to go below Moho and Conrad for n and b phases respectively, 1: enabled, 0: disabled, D: 0.0

49: T-phase velocity, D: 1.48 km/sec

50: Flag for using azimuth phases, 0 disables. Disabling the azimuths also means that they are not used for a starting location. A better solution will often be to set the azimuth error, TEST(52) to a large value, effectively disabling them.D: 1.0 (enabled).

51: Lg phase velocity in km/sec, D: 3.5.

52: Relative weighting of error in azimuth used in azimuth inversion (degrees). The default value of 10 means that an error of 10 degrees will give the same contribution to the rms residual as a travel time error of 1 sec, D: 5.0

53: Critical distance phases moved to by start loc. if Pn or Sn, D: 130.0 km

(56): A value of 1.0 enables the starting location algorithm, STARTLOC. Estimates are then obtained from apparent velocity, distance, azimuths, etc. If test(56)=0.0 epicenter is taken 0.2 km from the first arrival station. D: 1.0 MUST BE SET TO 1.0 TO LOCATE WITH ONE STATION ONLY.

57: Distance (geocentric km) beyond which IASPEI91 tables are used to calculate travel times. Can be overridden by the distance letter L in the Nordic format. D: 1500 km

58: Maximum apparent velocity (km/sec) for phase data to be used. This option was added to selectively disable some of the PKP phases, which have large errors due to their steep angle of incidence. Their velocities were almost always ¿ 25 km/s, D: 100.0 (effectively disabled)

59: Critical distance for PKP core phases to be used in starting location, D: 13000 km

60: Seconds by which the arrival time difference between two adjacent stations can exceed the travel time between them. Setting this to 0 disables the initial consistency check. D: 5.0

61: Multiple of apparent velocity regression residual rms at which arrival times are weighted to zero during start location determination. Reducing this value will cause arrivals to be rejected when they do not conform to the plane wave set of arrivals which is characteristic of distant events. Unless you are getting a lot of messages ' xxx removed: Apparent velocity deviation =..', in the output, it is recommend against changing this default value. However, you can disable this feature by setting test(61)=0.0, D: 2.0

62: Use of IASP91 phases.0: Only calculate 'basic' phases, 1: calculate all, D: 1.0

63: Types of phases used when calculating travel time, D: 0.0

64: Allow temporary increase in RMS by this factor, D: 2.0

65: Number of iterations for which increased rms is allowed, D: 3.0

66: Print out of travel time calculation errors (1=y,0=n), D: 0.0

67: Recognize blank phases as P (y=1,n=0), D: 0.0

68: Apparent P-velocity(km/sec) to calculate start depth from pP-P, D: 5.0

69: Distance (deg) beyond which PKiKP or PKP is used as first arrival instead of Pdif D: 110.0

70: Maximum depth that the hypocenter is allowed to move to, D: 700 km

71: Sort output according to distance,(y=1,n=0), D: 1.0

72: Auto phase identification for distant events (y=1,n=0), D: 0.0

73: Number of iterations with first P's before autophase id., D: 3.0

74: Print input phase data in `print.out` (y=1,n=0), 0.0

75-78 Ml magnitude coefficients. Ml = TEST(75)*log10(amp) + TEST(76)*log10(dist) + TEST(77)*dist + TEST(78) where amp is amplitude in nm and dist hypocentral distance in km. The defaults are Ml = 1.0 * log10(amp) + 1.11*log10(dist) + 0.00189*dist - 2.09 which is close to the original Richter definition [Hutton and Boore, 1987].

79: Minimum number of stations to attempt a solution,D: 1.0

80: Minimum number of phases (azimuth is counted as a phase) to attempt a solution, D: 3.0

81: Disable location of local events if 0.0, D: 1.0

82: Disable location of regional events if 0.0, D: 1.0

83: Disable location of distant events if 0.0, D: 1.0

84: Disable ellipticity correction for distant events if 0.0, D: 1.0

85: A priori error(sec) of local events. This affects the error estimates, particularly when few stations are present. D: 0.1. See TEST(91) for distant eqrtquakes.

86: Number of degrees of freedom in estimating test(85) for loc. ev., D: 8.0

87: Confidence level that the solution will lie outside the confidence ellipse defined by the covariance matrix . The default value corresponds to 90 %confidence., D: 0.1

88: RMS residual(sec) at which residual weighting is applied for distant events. Set to 0.0 to disable. D: 10000.0

89: Use depth phases (y=1,n=0), D: 1.0

90: Use of core phases (y=1,n=0), D: 1.0

91: Same as TEST(85) for distant events,D 1.0

92: Number of degrees of freedom for test(91), D: 8.0

93: Output longitude to always be positive (y=1,n=0), 0.0

94: Value of residual below which zero weight phases (w=4) is used again, D. 0.0

95: Disable use of core phases between 135 and 150 deg, 1: disabled, 0: enabled, D: 0.0

96: Variation of depth to find minimum rms, 1: enabled, 0: disabled, D: 0.0

97: Minute error correction 1: enabled, 0: disabled, D: 0.0

98: Enable spherical harmonic station corrections, 1: enabled, 0: disabled, D:0.0 99-101: Lg, Rg and T weights put in permanently: D: 1.0,1.0,0.0

103: Minimum number of depth phases for starting depth, D: 1.0

104: Minimum distance of epicenter from array for distant events, D: 30.0 deg.

105: Enable gradient model, not yet implemented
106: Only calculate magnitudes and update spectral values, 1: enabled, 0: disabled, D: 0.0
107: Use xnear and xfar from sfile, 0: disabled (xnear and xfar from `STATION0.HYP` file), 1 enabled, D:0.0 (see format description)
108: mb attenuation curve, 0.0 Richter, 1.0 Veith and Clawson, D: 0.0

The test parameter defaults are set in file hyposub1.for in LIB.

HYP output:

Output from the program is a CAT-file (`hyp.out`) and the original HYPOCENTER print file (`print.out`) with more detailed information. The `hyp.out` file can be plotted directly using EPIMAP. In addition, there is also the HYPO71 style summary file, hypsum.out. NOTE: In `print.out` and hypsum.out, year is only given with 2 digits. Magnitude in hypsum.out and `print.out` are only coda magnitude and will be different from same magnitude in `hyp.out` if a magnitude correction has been used.

When HYP is executed from EEV, the `print.out` file has no station listing. In all other cases, there is a station listing.

Some explanation is given below, for details see HYP manual

The output in `print.out` first shows the content of the TEST parameters in the `STATION0.HYP` file. After that comes some routine output from the starting location algorithm. Then follows the output from the iterations, which should be self-explanatory. The location is then given on one line containing origin time, latitude longitude (deg min), depth, number of phases, the number of degrees of freedom in the spatial solution (maximum 3), rms damping and errors, error estimates, resolution matrix. Last are the station lines with the following abbreviations:

```
stn  : Station
dist : Distance in km
azm  : Azimuth at the source
ain  : Angle of incidence at the source
phs  : Phase specified by user
calcphs: Phase used by program
w    : Input weight
hrmn : Hour minute
t-sec: Arrival time sec
t-obs: Observed travel time
t-cal: Calculated travel time
res  : Residual
wt   : Weight usedi by program, normalized to 1.0
di   : importance of phase in %
```

A station weight wt=-1 means that the phase travel time could not be calculated. The output phases can be e.g. PN2, where 2 means that the phase calculated has been refracted in layer 2 and PN5 refracted in layer 5. The input phase is then just P and a local model is used.

Any change in the input phase ID is signified by an asterisk (*) before the phase ID.

If amplitudes are available, Ml, Mb. Mw or Ms will be calculated, and all stations calculating Ml, Mb, Mw or MS will additionally be displayed at the end of the interactive printout.

Change of day:

If the origin time of the located event occur on the day before the time in the header line, the time in the

header line is changed to the previous day and all phase arrivals are changed accordingly. This means that some hour values will be more than 23 since phase arrival times refer to the main header.

Seismic moments etc: After locating an event, HYP will check if there is spectral information (Moment etc, see MULPLT) available in the S-file and average values will be calculated and written into the output file.

Problems :

If no location or an obviously wrong location is obtained, check `print.out`. Common problems are:

- Wrong location: Program gets into a local minimum. Use the individual event start location option ('S' in header line). If the problem happen often, try either option for start location (nearest station or start location routine). If dept is the problem, try a range of start depths (set in `STATION0.HYP`)
- No location: The program iterates outside the maximum distance set for a local or regional event (RESET TEST 57) or the initial start location is outside limits. Use a fixed start location or check readings to get a better start location.

6.1.2 HYPO71 (Sun only)

By **Brian Baptie, BGS**

HYPO71 is a computer program for determining hypocenter, magnitude and first motion pattern of local earthquakes written by Lee et al. [1972] using a stepwise statistical regression procedure outlined in Draper and Smith [1966]. The user's manuals were originally released by the authors as a series of open-file reports of the U.S. Geological Survey and contain a full description of input and output parameters and usage. The SEISAN version of the program is essentially the same as the original, the only differences being in the input and output facility. Input data required are phase arrival times, station co-ordinates and a crustal velocity model. SEISAN extracts the arrival information from a Nordic format phase readings file and the station and velocity information from the station input file `STATION0.HYP`, found either in the SEISAN data directory, DAT, or the local directory. The format of the `STATION0.HYP` file is described in this manual in the section on the HYPOCENTER algorithm (6.1.1). HYPO71 supports 13 test variables that influence how the program goes about locating the earthquakes. The default values for these variables were developed for the large and closely spaced networks in central California. These variables are defined at the start of the `STATION0.HYP` file by the values of TEST(01) to TEST(13). Brief definitions for each of these variables can be found below and full definitions can be found in the HYPO71 manual.

SEISAN constructs a HYPO71 format input file called `hypo71.input`, containing the station co-ordinates, thickness and velocity for each layer of the crustal model and phase arrival times, then runs the HYPO71 algorithm. The HYPO71 program generates a single output file called `hypo71.output`. SEISAN reads the information contained in this output file to create two further output files: `hypo71.out`, a Nordic format phase readings file containing the calculated location; and `hypo71.brief`, a summary file containing origin time, epicenter, depth, magnitude and station residuals.

There are a number of limitations to the current version.

- The program is designed to run from eev and can only be used for one event at a time; there is no facility for multiple event or batch location/relocation.
- HYPO71 is not included with the UPDATE command, so the database cannot be updated.

- Errors will result if the input phase readings contain arrivals from two different days, i.e. either side of midnight
- All stations must have the same sign of latitude or longitude, so if stations extend across the Greenwich meridian and/or the equator and an offset should be added to allow for this.

Running the program

HYPO71 is run from within eev by typing hypo71, at the command line. On successful completion, the information from the `hypo71.brief` file is displayed on the screen. Below is an example of the screen output.

```
EXAMPLE RUN

#   29 13 May 2001  8:26 59  L 55.1020 -3.6388 12.3 H      2.9L      16  ?
#   29 13 May 2001  8:26 59  L 55.1020 -3.6388 12.3 H      2.9L      16  ? hypo71
 HYPO71 completed successfully
Date        : 13/05/01
Origin time :  8:26:59.78
Epicentre   : 55- 5.45 deg N   3-37.11 deg W   (  55.0908 /   -3.6185 )
Grid Ref    : 296.716 East /  578.581 North
Depth       :   2.13  Quality     : B B*B

              NO DM GAP M  RMS  ERH  ERZ
Statistics  : 22 10 127 1 0.29  0.8  1.8
Magnitude   :  2.8 ML     (from  6 readings)
Magnitude   : No valid coda readings

 STN   DIST AZM AIN   P-RES P-WT   S-RES  S-WT
 BWH    9.7 346  43   0.08  7.22
 BHH   25.6  89  43  -0.03  1.03   -0.29  0.51
 BCC   26.8 108  43   0.26  0.71    0.01  0.36
 GCD   32.7 219  43  -0.40  1.75
 ECK   32.8  72  43  -0.28  1.02
 ESK   36.4  46  43  -0.39  1.00   -0.91  0.26
 BBH   44.2  84  43  -0.29  0.77
 BBO   46.1 149  43   0.20  0.54    0.01  0.36
 BDL   54.1 126  38   0.34  0.70
 BTA   63.2 109  38   0.42  0.69    1.04  0.09
 CKE   65.0 150  38   0.27  0.54
 XAL   93.4 106  38   0.16  0.54
 EDI   96.6  16  38   0.33  1.00
 GIM  104.4 212  38   0.02  1.35    0.36  0.87
 WCB  200.0 198  30  -0.91  0.45
 CWF  302.7 150  30  -0.08  0.24
```

Phase names

Only single character phase names are supported, denoted by P or S.

Weighting

Two weighting options may be used.

1 User specified weights assigned by a single integer value in the range 0 to 4 for a given phase. These will assign a weighting factor of 1, 0.75, 0.5, 0.25 or 0.0 to that phase. Also, a weighting of 9 will assign the absolute time a weighting of 0.0 but will allow the use of relative times if a valid S-arrival is found for that station. The relative arrival time will be assigned the weight of the S-phase.

2 Distance weighting as given by the relationship $w = (xfar - \Delta)/(xfar - xnear)$. By default the parameters xnear and xfar are read from the `STATION0.HYP` file. However, they can also be defined in the s-file and are used if RESET TEST(107) is set to 1. The paramters are specified in the s-file by a type-3 line, e.g.
`XNEAR 150.0 XFAR 300.0 SDEP 7.5`

Using a starting location

The user can specify the use of a starting depth and epicenter by entering the character 'S' in columns 44 and/or 45 respectively, in the header line of the input readings file. The starting depth and epicenter are given by the values in the header line of the readings files. Otherwise, the starting epicenter is set to be the latitude and longitude of the station with the earliest P-arrival.

Fixing the location Using the character 'F' instead of 'S' in columns 44 and 45 of the header line fixes the depth and/or epicenter to the values given in the header line.

Errors

The standard error output from the HYPO71 program is contained in an additional line in the Nordic format readings output, hypo71.out, defined by the characters '83' in columns 79 and 80.

The HYPO71 error line format is defined as follows:

Columns	Format	Description
2-14	A13	'HYPO71 errors'
19	A1	Location quality, Q
21-23	A1*A1	QS and QD rating
25-27	I3	Number phases used
28-30	I3	Distance to closest station
32-34	I3	Azimuthal gap
36	A1	'1'. (Always output?)
38-41	F4.2	RMS
43-46	F4.1	ERH (km)
48-51	F4.1	ERZ (km)
79-80	A2	'83'

RMS is defined as $\sqrt{[\sum_i R^2/N]}$ where R_i is the time residual at the i^{th} station. ERH is the standard error in the epicenter in km given by $\sqrt{[SDX^2 + SDY^2]}$, where SDX and SDY are the standard errors in latitude and longitude. ERZ is the standard error in the focal depth in km. The location quality, Q, is a measure intended to indicate the general quality of the solution and is defined by a single character.

Q	Epicenter	Focal Depth
A	excellent	good
B	good	fair
C	fair	poor
D	poor	poor

Q is taken as the average of QS and QD, where QS is a statistical measure of the solution and QD is rated according to the station distribution.

QS	RMS (S)	ERH (km)	ERZ (km)
A	< 0.15	≤ 1.0	≤ 2.0
B	< 0.30	≤ 2.5	≤ 5.0
C	< 0.50	≤ 5.0	
D	Other		

QD	N	Gap	DMIN
A	≥ 6	≤ 90	$\leq$ Depth or 5 km
B	≥ 6	≤ 135	$\leq$ 2*Depth or 10 km
C	≥ 6	≤ 180	$\leq$ 50 km
D	Other		

Magnitude

Both duration and amplitude can be used to calculate magnitudes as with HYPOCENTER (see above for details). Duration, amplitude and period for each station are used to give a magnitude value for each station. These values are averaged to give the event magnitudes.

The test variables

Test Variable	Default Value	Definition
TEST(01)	0.1 S	TEST(01) is the cut-off value below which Jeffreys' weighting of residuals is not used. It should be set to a value approximately equal to the overall timing accuracy of P-arrivals in seconds.
TEST(02)	10 km	For each iteration, if the epicentral adjustment is greater than TEST(02), this step is recalculated without focal depth adjustment. TEST(02) should be set to a value approximately equal to the station spacing in km.
TEST(03)	2.	Critical F-value for the stepwise multiple regression. TEST(03) A value between 0.5 and 2 is recommended.
TEST(04)	0.05 km	If the hypocentral adjustment is less than TEST(04) then Geiger's iteration is terminated.
TEST(05)	5.0 km	If the focal depth adjustment, DZ, is greater than TEST(05), DZ is reset to DZ/(K+1), where K= DZ/TEST(05). TEST(05) should be set to a value approximately half the range of focal depth expected.
TEST(06)	4.	If no significant variation is found in the stepwise multiple regression, the critical F-value, TEST(03) is reduced to TEST(03)/TEST(06) and the regression is repeated.
TEST(07)	-0.87	Coda magnitude constant a, where $Mc = a + blog10(T) + c\Delta + \Delta M$
TEST(08)	2.0	Coda magnitude constant b.
TEST(09)	0.0035	Coda magnitude constant c.
TEST(10)	100 km	If the latitude or longitude adjustment (DX or DY) is greater than TEST(10) then DX is reset to DX/(J+1), and DY is reset to DY/(J+1), where J=D/TEST(10), D being the larger of DX or DY.
TEST(11)	8.0	Maximum number of iterations in the hypocentral adjustment.
TEST(12)	0.5	If the focal depth adjustment (DZ) would place the hypocenter in the air, the DZ is reset to DZ= -Z * TEST(12), where Z is the focal depth.
TEST(13)	1.0 km	Parameter for auxiliary RMS values

6.1.3 The Hypoinverse program, HYPINV (SUN and PC)

The Hypoinverse program has been implemented in a simple way and is mostly intended to be operated interactively from EEV in order to compare locations. The main program has seen very few changes and can be run according to the original manual [Klein, 1984] and will not be described here. The program

does not work well at large distances(> 1000km) so use it only for local earthquakes. If original data, station and control files are available, it is just typing HYPINV and it will run according to the manual. If none of these files are available, they can be made with the conversion programs. The steps to run HYPINV without EEV are as follows:

1. Convert a CAT file to Hypoinverse file by typing norhin input file. The input file in Nordic format will now be converted to a file norhin.out in Hypoinverse format.

2. Make the control files by typing makehin. This creates the instruction file hypinst, station file hypinv.sta and model file hypinv.mod. These files are standard Hypoinverse files. The information is taken from the `STATION0.HYP` file in either the working directory or DAT. Makehin cannot work with an alternative `STATIONx.HYP` file.

3. Type hypinv and the program runs. There is a one-line output per event on the screen and the full output is in a file called `print.out`.

Running HYPINV from EEV, the above 3 steps are done automatically when using the command H and in addition, the `print.out` file is printed out on the screen.

6.1.4 HYP_ISC (Unix and Linux only)

Program written by **Richard Luckett**

ISC has for many years used a standard procedure to locate earthquakes and the ISC locations have often been used as a reference. The earth model used is the Bullen tables. ISC has recently rewritten the old location program and it was therefore possible to also port it to SEISAN. The purpose is that it should be possible to compare standard ISC locations with location using other programs and models. The implementation in SEISAN was done using the standard hyp program where only the location routines have been changed. The program then behaves almost identical to HYP and uses the same format input and output files.

Parameter files: STASTION0.HYP is used for station coordinates, magnitude scales and agency code. The crustal model information is not used and only the RESET TEST parameters related to magnitude are used. In addition, there is a new parameter file (in DAT) `iscloc.def` with parameters specific for the ISC location routines, see file for explanation of parameters.

Input data files: Just like for HYP

Output files: Hyp.out is like before, `print.out` is different.

Not all crustal phases used with HYP may be available. The weights used in SEISAN do not apply since the program uses residual weighing only, see parameter file.

Magnitudes are calculated exactly like in SEISAN.

In eev, the command to locate with HYP_ISC is 'il'.

For more information about the ISC location program, see `http://www.isc.ac.uk/Documents/Location/`

6.2 Trace plotting, phase picking and spectral analysis, MULPLT

This program is the general plotting and signal analysis program. The program is capable of doing general phase picking, correct for instrument response, and produce Wood-Anderson seismograms for determining Ml, synthetic traces for Mb and Ms, determine azimuth of arrival for 3 component stations, do spectral analysis and particle motion. The program can also read in theoretical arrival times for global phases for help in identifying phases. If a quick location is needed based on a waveform file only, mulplt can both pick the phases and locate the event. MULPLT operates either as a database independent program (started with command MULPLT) or in connection with the database (started from EEV with command P or PO). If the program works independently of EEV, it will create an output file mulplt.out in Nordic format with the readings and results of spectral analysis. This file can directly be used with e.g. HYP. MULPLT reads and plots one channel at a time. This can be very time consuming when replotting traces and from SEISAN version 8.2, a certain number traces are kept in a memory buffer in order to speed up replotting and reprocessing The data is stored in one large array the size of which is determined at the time of compilation so for systems with many or long traces, it might have to be larger and for systems with little memory it might have to be smaller. The dimension is set in file seidim.inc in directory ./INC using variable max_mem_sample. A typical value is 30 000 000.

Starting MULPLT from prompt line

Giving command `mulplt`, the question is

```
Filename, number, filenr.lis (all)
Continuous SEISAN data base: cont
Large SEED volume: conts
Bud archive: bud
SEISCOMP archive: scp
```

Filename, number, filenr.lis(all): The program asks for a file name or file number of a waveform file. To use the number, it is assumed that a list of files has first been created and numbered in a file `filenr.lis` using command DIRF, see section 6.8. By giving the number, the file corresponding to the selected number is used. By giving a ?, the list with numbers is displayed and a new number can be given. If many files are to be plotted with one command (hard copy only), give `filenr.lis` for file name and all events in FILENR.LIS will be plotted. There will only be one question about filter and then all events are plotted with all channels and the chosen filter.

Cont: Plot from a continuous data base. The program will use all data bases defined in `SEISAN.DEF`. A question will be given for absolute start time and window length. See also 6.2.4. Conts: Plot from a large SEED volume. A SEED file too large to be read in can be plotted in parts. A question will be given for file name or number. The file is then read and available data is displayed. Start time and window length is then entered.
bud: Plot for a BUD archive
scp: Plot from a SeisComp archive

6.2.1 MULPLT main functions

The program has 7 main functions irrespective of type of input as illustrated below with the questions given by the program:

```
Plot options:          Interactive picking          Return
                       Multi trace plot on screen, def (0)
                       Multi trace plot on screen      (1)
                       Multi trace plot on screen+laser(2)
                       Multi trace plot on laser       (3)
                       Continuous on screen            (4)
                       Continuous on screen + laser    (5)
                       Continuous on laser             (6)
                       Stop                            (9)
```

Return:	Picking phases, spectral analysis and 3 component analysis
0:	Initial plotting of new waveform data and registration in database using predefined defaults. Phase picking.
1:	Initial plotting of new waveform data and registration in database, or general plotting of multi-trace data. Phase picking. In this mode, the user is asked to select which channels to work with and a graphical window will be shown for selection. If more than 250 channels are available, selection will have to be made on several screens. A maximum of 1000 channels can be used. Optimally, channels can be show in alphabetical order (see `MULPLT.DEF`).
2:	Same as 1, only hardcopies may be made at the same time.
3:	Making hardcopies of many waveform files with one command. No screen output.
4,5,6:	Plotting one channel continuously like on a seismogram with several traces from left to right on top of each other. One channel can be selected. Time windows can be selected for event extraction.

For continuous data, see 2.2.2 and 6.2.4.

Commands f and q: With a plot on the screen, q will always quit mulplt. F will, in single trace mode, bring up the next channel, in multi trace mode bring up the next event.

Option 0 is particular useful for checking many new events, since the program does not ask question about station choice (uses definition in `MULPLT.DEF` file in DAT, or working directory) and typing f (when the plot is on the screen) automatically goes to the next file in `filenr.lis`.

If option 1,2 or 3 is used, a display will show the available channels and the user can click to select. If MULPLT is operated from EEV, a * in a channel box will indicate that readings are available. When the channel selection is shown, it is possible to quit the program with q or continue with f.

If option 4,5 or 6 is selected, continuous data is plotted, see below.

Running MULPLT using command MULPLT, the program asks for a file name or file number of a waveform file. To use the number, it is assumed that a list of interesting files has first been created and numbered in a file `filenr.lis` using command DIRF, see section 6.8. By giving the number, the file corresponding to the number is used. By giving a ?, the list with numbers is displayed and a new number can be given. If many files are to be plotted with one command (hard copy only), give `filenr.lis` for

file name and all events in FILENR.LIS will be plotted. There will only be one question about filter and then all events are plotted with all channels and the chosen filter.

Hardcopies assume a PostScript printer. For each event plotted, a plot-file called `mulplt.eps` is generated. The plot files are sent directly to the printer from within the program with the seisan print command as soon as the plot is finished for one event but before the program is finished. In Unix, this is lpr or lp while on PC, the command is given in a .bat file in COM (see installation section). This means that the same plot-file is overwritten for each event plot. For setting up the printer, see installation, chapter 3.

In multitrace mode, many traces (number limited by the SEISAN system definitions, see chapter 3) can be plotted. If the plot is made via EEV, all picks are also displayed.

Location of waveform file:

Mulplt will search in current directory first. If not found there, the WAV directory is searched. If the `SEISAN.DEF` file has been set up, MULPLT will thereafter try to locate the waveform file in one of the databases or specific directory given in the `SEISAN.DEF` file (located in DAT or working directory).

Format of waveform files:MULPLT can, on all platforms, use SEISAN, GSE, SAC ASCII and binary and SEED. If started from EEV, files in different formats can be used at the same time.

Time gaps in waveform files:

Only SEED format has running time headers and therefore the only format where a possible time gap, in one file, can exist. SEISAN will replace missing data intervals with zeros. For a SEISAN continuous data base, it is assumed that a gap of less than 2 s between files is not a gap. Larger gaps will be replaced by zeros.

6.2.2 Use of MULPLT from EEV

In order to process events more easily using SEISNET, EEV and MULPLT have been tightly integrated. When MULPLT is called from EEV, command f will plot the next event in the database, to go back to the same event in EEV, use quit. The event can be plotted with default parameters from EEV using PO. If PO has been selected, the f command in MULPLT multi trace mode will show the plot of the next event with default options. This is a fast way of plotting waveform files going through S-files in a database.

If several waveform files are available, the user will graphically be shown the files and can select one or several Files can optionally be displayed in alphabetical order (see `MULPLT.DEF`). If more than 75 files, several selection screens will be shown. A maximum of 1000 files can be used If the PO option has been used, all default channels will be used.

6.2.3 Continuous plotting of one channel

This option is used to plot one channel in a multi line display and can therefore simulate a helicorder plot. This is a very different option from the plotting from a continuous data base (see 6.2.4). Interactive processing is not possible in this mode except for selecting time windows for event extraction, see below.

Using this option the program asks for the following input:

Low and high cut for filter: Give values or return for no filter. Sometimes low cut filters are unstable (a different routine used for continuous plotting) so if e.g. a LP record is to be simulated, use filter limit 0

to 0.1 Hz. The zero means it is a low pass filter, not bandpass. A filter 10 to 0 would mean a high pass filter. High and low pass filters are only available with continuous plotting, unless the FILTER TYPE (`MULPLT.DEF`) is set to 1.0=RECFIL. Seconds pr line: Number of seconds on each line.

End time: This question only appear if plotting from EEV. The list of files is then the list of events belonging to the data base used. Give end time as e.g. 2000050203

Max count: The absolute maximum count to be used for full scale. Since many lines and possibly many pagers are plotted, it is not possible to use autoscaling, and like on a seismogram, a fixed value must be set.

Lines pr page: Number of lines per page.

Station code: Station code (max 5 characters)

Componet code: The component code, max 4 characters.

MULPLT will plot from the first file given from the `filenr.lis` file and then continue to plot as long as more file names are given in `filenr.lis`. Alternatively if plotting from EEV, it will start with the current event and continue until the end time. So if a month of data files are given, a month of seismograms will be displayed. There is no requirement that the input files follow each other in time (no time gaps) since each file is plotted on the page where it belongs in time. However, the files must be time ordered. The continuous option can therefore be used to check availability and timing of continuous data. Discrete events can also be plotted in this mode if one want to get a display of when the events occurred. However, if filtering, it is assumed that the files follow each other in time since a few points are carried over from one file to the next to make the filtering continuous. Figure 20 shows an example.

Selecting time windows for event extraction in continuous one channel mode:

To extract events shown in the continuous plot one can mark a given time window by typing 's' and 'e' at the start and end of the required time window (see Figure 6.1). Selected time windows are, when plotting is finished, written to the file mulplt.ext and to the screen as wavetool command lines. By executing `mulplt.ext` (type `sh ./mulplt.ext` or `source mulplt.exe` in UNIX, on PC the name will be `mulplt.ext.bat` so just typing the name `mulplt.ext` will start the extract process) the time windows given in `mulplt.ext`, are extracted from the continuous database defined in `SEISAN.DEF`. The data files are extracted in the local directory. These files can now be registrated in a REA database with autoreg. One cannot delete a "Start" or "End" time mark. So, if another time window is required pick the new window and delete the line in `mulplt.ext` with the old time window before data is extracted.

The time marks is written to the file mulplt.ext for data to be extracted and processed. Type "s" and "e" to add time marks.

6.2.4 Commands in MULPLT, overview

When the trace(s) are on the screen and the cursor is displayed, then several options are available. Most options can be displayed by pressing the MENU button in the upper right hand corner. Pressing MENU again removes the option boxes. Commands can be given by either pressing a letter or clicking on a box in the menu (Figure 21). By pressing ? or clicking on the Help button, the following help menu will be displayed:

Help on MULPLT

```
MULPLT has 3 modes:
```

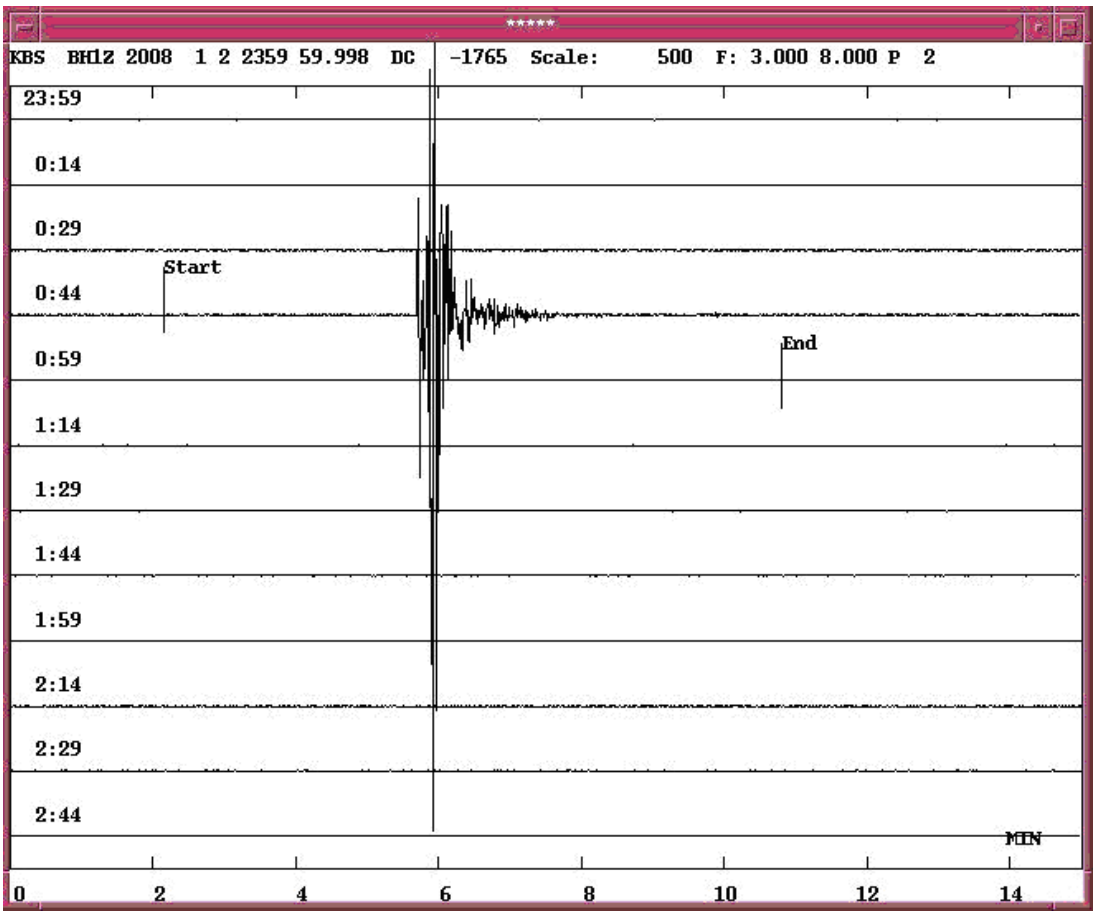


Figure 6.1: Example of time marks at the "Start" and "End" of an event recorded at the KBS station. The time marks is witten to the file `mulplt.ext` for data to be extracted and processed. Type "s" and "e" to add time marks.

```
  multi trace mode:  Displaying many traces, some picking options
  single trace mode: One trace, all picking options
  all channel mode:  All channels for selected stations
```

Most commands are given by pressing one key, however several commands can also be given by clicking in the appropriate menu box on top of the screen. If the full menu is not displayed, select the menu on top right part of the screen. The following commands are available, first given by the single letter command and if also available as a mouse click, the letters in the menu box is shown. COMMANDS ARE CASE SENSITIVE.

Channel selection: In multitrace mode, one or several channels can be selected by clicking on the station code. If only one channel is selected, and toggl (t) to single trace mode is done, a new toggl in single trace mode will display all previous channels again. If All channels is selected, toogle to all channles (y) is used. To get back to either Single trac or Multi trace mode from All channel mode, press y ot t.

Zooming: Select a window with the mouse and a zoomed window will appear below in single mode and replace the plot in multi mode. In Single mode, it is also possible replot the zoomed window in place of the original by placing the cursor above the trace when selecting the zoom window. This makes it possible to zoom in the zoomed window. In order to go back to the original window in multi mode, do an opposite zoom, meaning picking the last point first.

a: Read amplitude. Position cursor at the top of a wave and press a. Position cursor at the bottom of the wave and press a. Amplitude(0-p) and period are now stored. These values will be stored with the NEXT phase pick. Amplitude and period are displayed.

A: Automatic amplitude reading, phase become AMP or read phase AMP but only after using a twice for reading amplitude.

b: Filter 5-10 hz, see below

B: Back Go back one trace in single trace mode
From eev, multi: Go back one event
Continous data base: Go back one window

c: Read coda.

C: Read end of coda automatically

```
d:        Delete phase. Position cursor near phase and press d.

d: Del W: Delete waveform file, cursor outside area of trace plot, file must
          be in working directory, else no files displayed

D: Del S: Delete S-file if operated from EEV, multi mode

e:        Phase E

f: Next:  Single: Go to next channel
          Multi, One event: Go to next event
          multi, continous data base: Go to next window

F: FK:    FK analysis of array data

g: Groun: Make a ground motion seismogram(s).

h: Azim:  Make 3 component analysis (single mode ONLY) to
          determine azimuth of arrival. Select a window around
          the P-phase on the Z-component. Azimuth and apparent
          velocity  will enter the S-file with the next phase
          reading.

i:        Phase I

I: Iasp   Calculate IASPEI synthetic arrival times, which are then
          Displayed, multitrace only.

j: mb:    Generate a synthetic SP seismogram for reading amplitudes
          for determining mb.

J: mB:    Generate a synthetic BB velocity seismogram for reading
          amplitudes for determining mB.

k: Ms:    Generate a synthetic LP seismogram for reading amplitudes
          for determining Ms.

K: MS:    Generate a synthetic BB velocity seismogram for reading
          amplitudes for determining MS.

l: Locat: Locate event, only multi trace mode

m:        Filter 15-24 hz, see below

M: Merge: Merge waveform file, mulplt called from EEV, only if files are
          In working directory.

n:        Filter 10- 15 hz, see below
```

```
o: Oth C: Select other channels

O: Out:   Makes an output waveform file of current
          data on screen. Only multi mode. Response info not saved.
          Standard name with YYYY_MM... is used, name is written in
          text window.

p: Regis: PUT (Register) event in database, you will be
          prompted for event type and waveform file will be copied
          to WAV.

P: PartM  Particle motion plot, requires the three components in
          multi trace plot

q: Quit:  Quit program

r: Plot:  Replot same event, useful when screen is cluttered
          up with e.g. many picks. Also used when a replot is
          wanted with new parameters e.g. filter.


s: Spec   Make a spectrum, single mode ONLY. Press s, select
          window with cursor, answer questions and the spectrum
          will appear below with noise spectrum.

S:        Same as s without the noise spectrum.

t: Toggl: Toggle between multi and single mode

u: Rotat: Rotate components. Backazimuth is calculated from location
          in header or as a second option, taken from observed + azimuth
          residual for respective station. R is plotted instead of N and
          T instead of E. Combining 3 component option (h) with u, the
          user is asked for the backazimuth angle.

v:        Filter 1-5 Hz, see below.

y: AllC:  Toggle to and from all channel mode

Y:        Pick a theoretical phase if displayed. Place cursor
          where the phase might be. Press y and the program will
          select nearest synthetic phase.

w: WA:    Remove system response and display synthetic
          Wood-Anderson ground motion in nanometers (nm)
          on next plot (using R or zoom).

W: Oth W: Select other waveform file, same event
```

```
z:        Filter 0.001 to 0.1, see below
x:        Filter 0.1 to 1.0 Hz, see below
Z: <W>    Increase window length in plotting from a continous data base
X: >W<    Decrease window lenght when plotting from a continous data base

>: Print: Will make a hardcopy of all channels of current event with
          the last selected filter, only in multitrace mode.

<:        Same as D

*: Scale: Fixed scaling of trace amplitudes.

_: Dist:  Select plotting channels in distance order.

:  Resp:  Plot response file, single trace mode only.

TAB: NextW: Next window if multiple windows

Filter options: The fixed filters (4/8 pole Butterworth) are placed
on keys z,x,v,b,n,m with the following frequencies:

z: 01-.1: 0.01 -  0.1 Hz  Pressing key once gives a 4 pole filter one way
x: .1-1.: 0.10 -  1.0 Hz  Pressing the key twice, and the filter also
v: 1 - 5:  1.0 -  5.0 Hz  go the other way and it is now an 8 pole filter.
;  2 - 4:  2.0 -  4.0 Hz
b: 5-10 :  5.0 - 10.0 Hz  From menu, only 4 pole filters are used.
n: 10-15: 10.0 - 15.0 Hz
m: 15-23: 15.0 - 23.0 Hz

.: Filt: Variable filter, question of filter limits is given in text
         window.
,: FixF  Fix filter. If pressed aftrer selecting as filter, the filter
         remains fixed until pressing ',' again.

Phase picking: This is possible in both modes. In Single mode, the phases
               defined are shown on top with some of the options, while
               they are not shown in Multi mode, but have the same
               definitions.

Combining options: Note that you can select several options together.
               E.g. V and S will first filter the signal and then
               make the spectrum.

Saving observations: When you go to the next trace or another event (F),
               the readings are saved in the S-file. They are also
               saved when you pick the next trace in Multi mode.

               This is also true for spectral parameters and azimuth
               determination.
```

jh 4-2-11

Filters in MULPLT

MULPLT uses a 4 pole Butterworth filter that can be used forward and backwards. Normally when a filter character or filter menu press is selected, the filter is only run one way and the number of poles is then 4. This will make a small phase delay where the first onset might appear a bit later, so if possible, read on unfiltered traces. If an 8-pole filter is desired, press the filter key twice and the filter will also run backwards. This gives theoretically a zero phase shift filter, however in practice, some of the onset energy is seen well before the first arrival, so it seems to distort the arrival times much more than using the 4 pole filter. When the program asks for a non fixed filter like when using the '.'(Filt) command, the filter is always 4 poles. When doing spectral analysis and specifying a filter before the spectral analysis, the filtering is done in frequency domain and the filter is 8 pole Butterworth. When reading polarities, DO NOT USE FILTER, if possible. If a filter is chosen from the menu, it is always 4 pole.

The filter pass-band limits can be changed in `MULPLT.DEF`.. The user can also chose between two filter routines: bndpas (default) and recfil.

The filter used in continuous mode can be either bandpass, low pass or high pass. Specifying a filter limit of zero, means that the filter is low pass or high pass. Limits of 0 10 Hz means a 10 Hz low pass filter.

Displaying uncertain time

In each trace header in the SEISAN waveform file, there is a flag to indicate if the time might be uncertain (see Appendix B). If that flag has been set, the message 'UNCERTAIN TIME' will be displayed on top of the trace. Currently this flag is only put into the waveform files if the data comes from a SEISLOG system that has detected a timing error or if the data is converted from SEED/MiniSEED data. Simlarly plotting SEED/MiniSeed data, uncertain time will be displayed if that flag is set in any block in the time window read in for a particular trace.

Below is some more detailed description of some of the options. The one letter command is given with the menu command in parenthesis:

To apply filters, first make a selection of options (filter, window, channel selection) and then execute by pressing R(Plot) (or selecting a zoom window). Figure 17 shows an example.

Single trace mode:

In this mode, one trace is initially displayed on top of the screen, see example on Figure 22. The traces used are the ones earlier selected and will be displayed one by one. Several options are now possible as can be seen on the menu. Normally no hardcopies are made in single trace mode since it is intended for fast routine work. However, by starting MULPLT in multitrace mode (option 2) and then go to single trace mode (command T(Toggl)), hard copy files are made.

Multitrace mode:

In this mode hard copies can be made. If option 2 is used, both screen plot and hard copy files are made. If replot is made, only the last plot is available in the hard copy file. If option 3 is used, which is only hardcopy, there will be additional questions about, window length, start time, scaling and filters. If the scaling is set so that the plot occupies more than one page, several pages will be printed. If in this mode, `filenr.lis` is given as file name, the program assumes that all the files should be plotted and the only questions will be about the scaling and filters. All channels in each file will be plotted. This option is useful for plotting a large number of events with a single command.

All channel mode
In this mode, all channels for selected stations are displayed in a new window. This mode is particularly

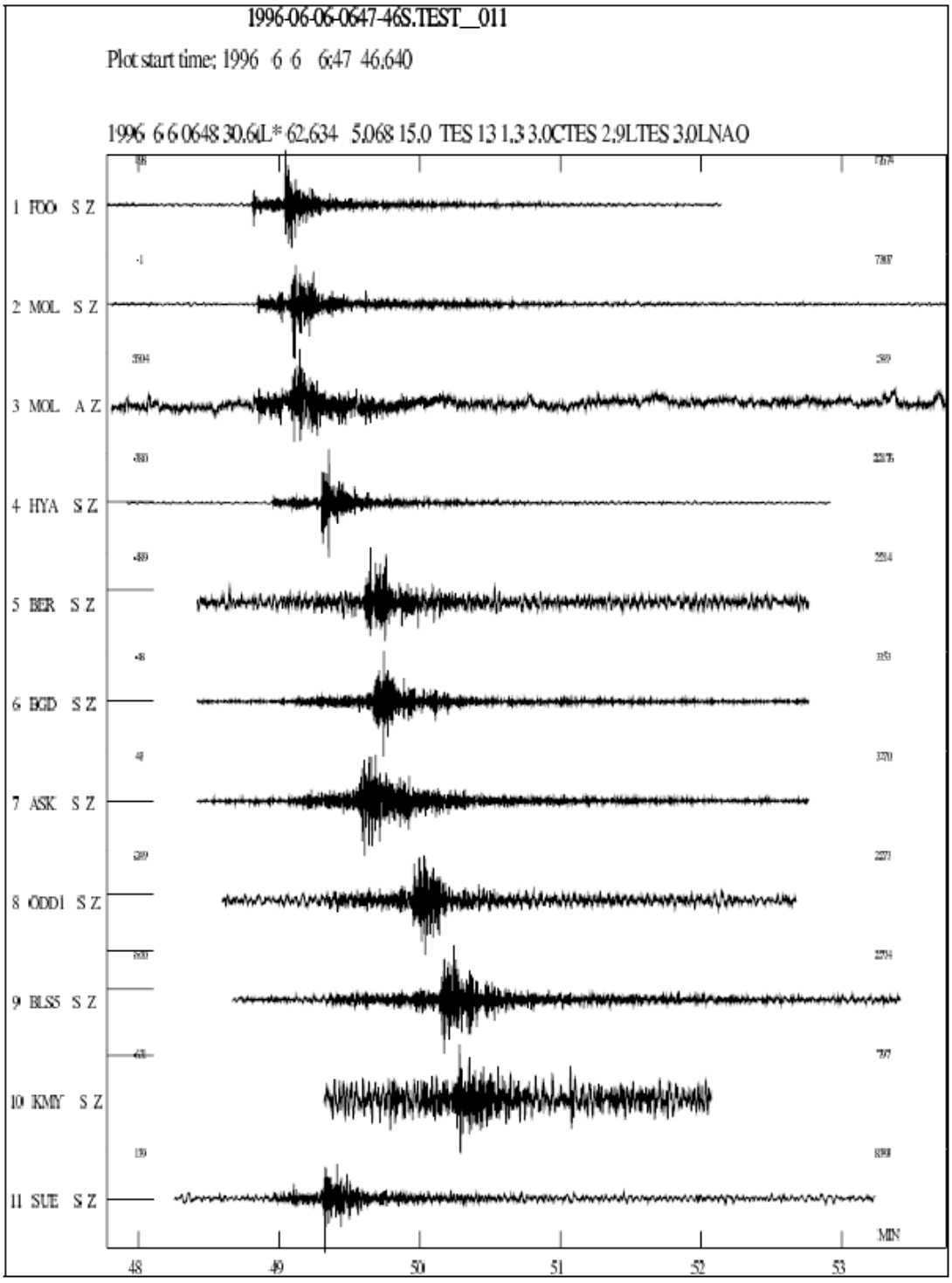


Figure 6.2: An example of using MULPLT in multitrace plot mode. Notice that start and stop times are different for different channels. The horizontal line at the start of the plot is the DC level. The small number above each trace to the right is the max absolute count with the DC-level subtracted and the small number to the left above the trace is the DC level. If plotting from EEV, the phase picks available are shown.

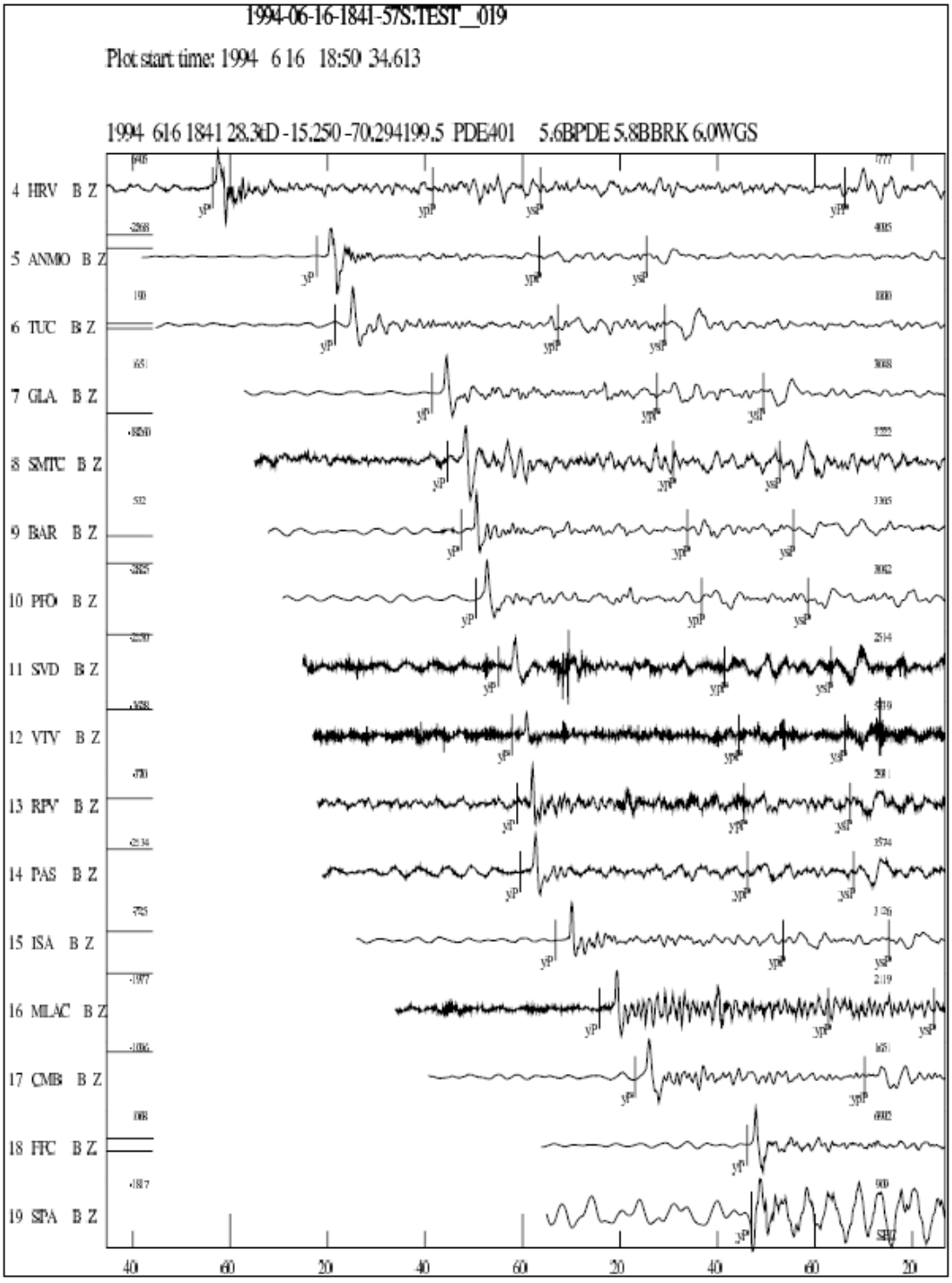


Figure 6.3: Examples of MULPLT with theoretical arrival times of some global phases. Short period seismograms are shown. The theoretical phases are marked with onset y below the trace and the read phases are marked normally above the trace.

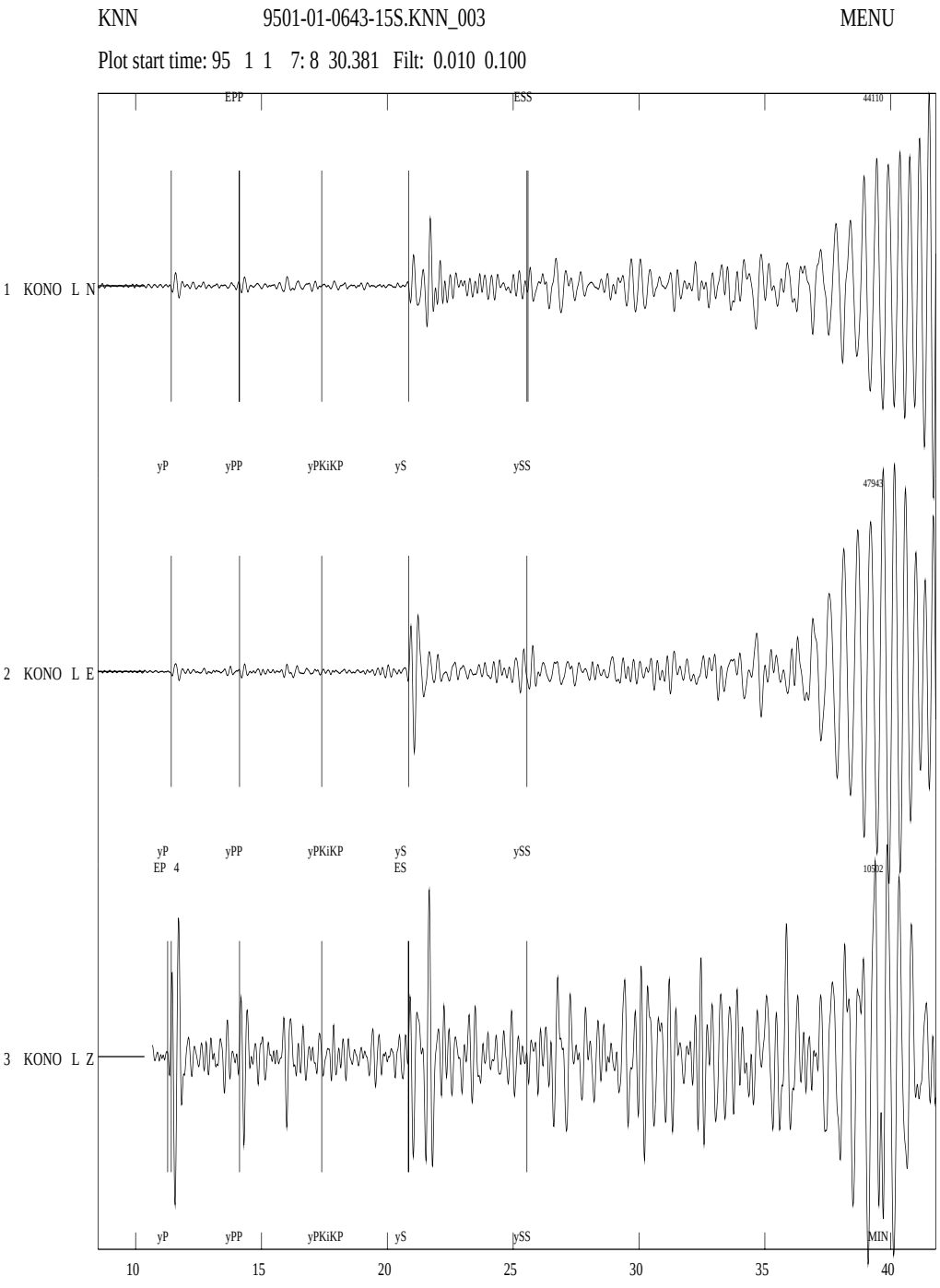


Figure 6.4: Example of MULPLT with theoretical arrival times showing global phases on a long period seismogram. The filter used from 0.01 to 0.1 Hz. Without filtering, almost nothing would have been seen on this broadband station.

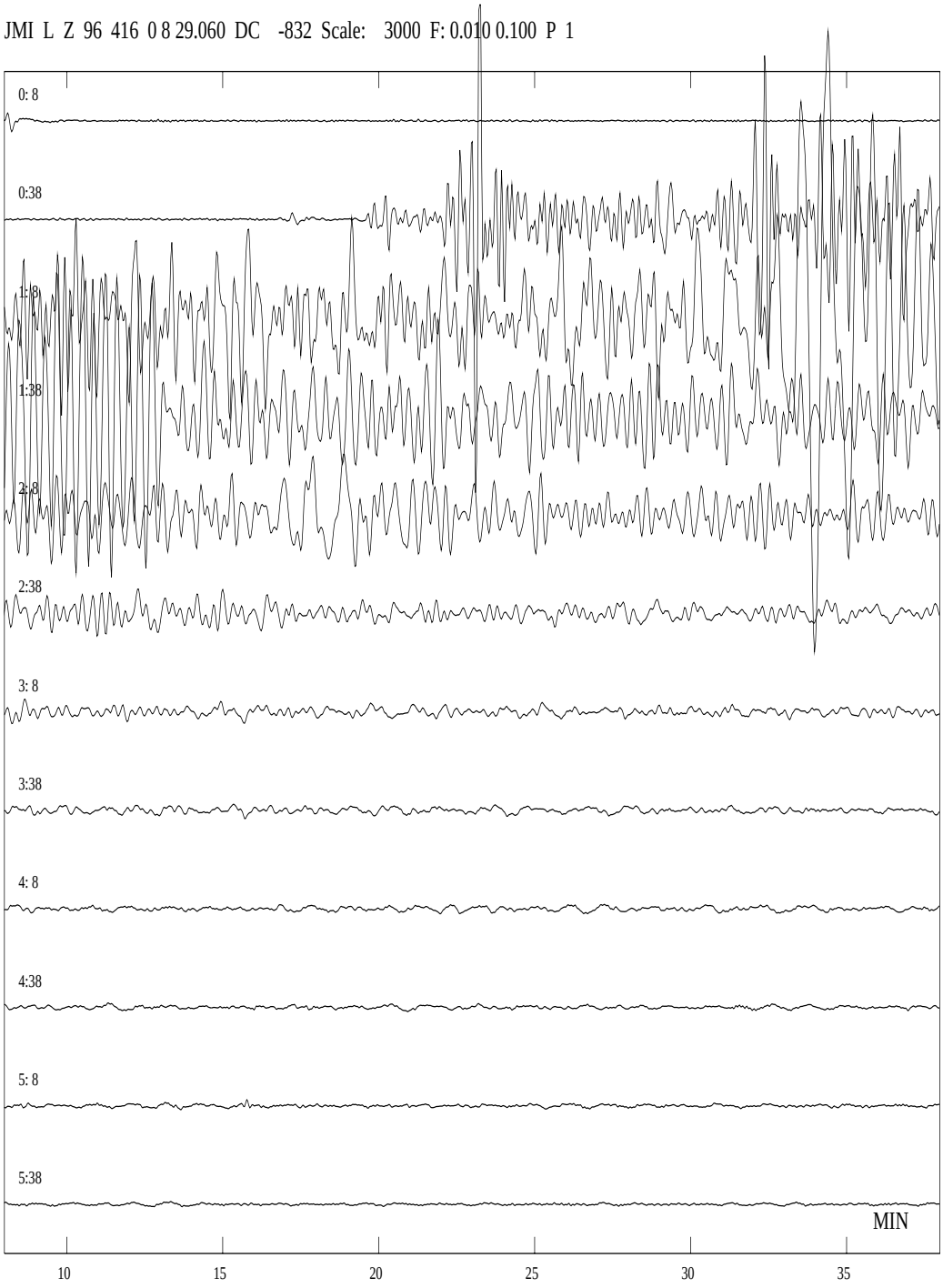


Figure 6.5: MULPLT in continous mode.
The plot shows 6 hours of long period data. The scale is 3000 counts between the traces and the filter used is from 0.01 to 0.1 Hz. The trace start time in hours and minutes is given on top of each trace. On the header line, P1 means the first page and DC is the DC level subtracted. Note that the numbers on the time scale at the bottom only are valid for the first trace unless all traces are 60 sec or 60 min long.

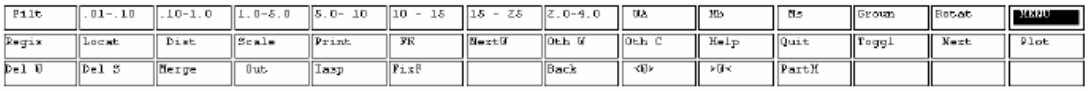


Figure 6.6: Example of the menu, which can be displayed on top of the plot.

useful for working with three component data. By selecting one station in either single or multrace mode, all components for that station will be displayed in new window. The user can then go back to e.g. multitrace window and select another station to work with in three component mode.

Multiple screens in multitrace mode

If many channels are available (like more than 30), it might be difficult to distinguish all and the channels can be displayed in multiple screens. The number of channels per screen is set in `MULPLT.DEF`. The number of windows or screens for a particular data set is given in top left hand corner as e.g. 'Win 2 of 7' meaning current window is number 2 of 7 windows. To move to the next window, use TAB or NextW in menu. In each window, normal operation can be done. Channels selected will be kept. Using a large data set, the user can then view each window separately, select the channels of interest and when all channels have been viewed, only the selected channels will remain for display.

Channel order in multitrace mode:

Normally channels are plotted in alphabetical order according to station name, see parameter CHANNEL SORTING in `MULPLT.DEF`. They can also be plotted in the order they are stored in the waveform file(s) (option NSORT_DISTANCE set in `MULPLT.DEF`). By setting the channel order parameter in the `MULPLT.DEF` file, it is also possible to plot the channels in distance or time order. If MULPLT is started from EEV (and distance ordering is set), the channels will be plotted in distance order provided distances are given in file. . Since there is no consideration for channels for the same station, the channels for one station, will be plotted in the same order as given in the waveform file. If a station is not found in the S-file, it will be plotted last. If plotting is done with MULPLT directly with a waveform file, the plotting order will be the start times as given in the waveform file header. Channel ordering can be turned on with the key "-" or pressing (Dist). If set in the `MULPLT.DEF` file, it is set when MULPLT starts up. It cannot be turned off for a given event when set from MULPLT but the flag is returned to the default value for the next event.

Plotting from continuous data base

If a continuous data base is set up (see section 2.2.3), it is then possible to plot all traces from the continuous data base with MULPLT. When MULPLT starts up, use option cont and the user is prompted for a start time and interval. MULPLT will now check all continuous data bases for available data in required interval and display the available data. The forward (next) or back option will display previous or next window respectively. There is an 25 % overlap between windows. If no data is available for the whole window, no trace is shown. If the beginning and the end is available, a line will join the two segments. If only end or beginning is available, only the available data is shown. All normal operation can be done on the window plotted so it is possible to e.g. extract data. If the register option is used, the whole window is extracted from the continuous data base as one file, copied to WAV and the S-file created.

6.2.5 Registering new events into SEISAN

Mulplt is the main tool for checking and putting new events into SEISAN. New events with waveform data can appear in two ways in SEISAN:

1. Unprocessed waveform files are available in a work directory and have to be inspected and possibly put into the database. No S-files have been made.
2. Raw data has already been put into a SEISAN database with S-files and corresponding waveform files in some work directory, the data has not been checked. This process has most likely been done with the automatic data collection software SEISNET [Ottemöller and Havskov, 1999], however, events can also have been auto registered with program AUTOREG.

In both cases above, the aim is to inspect an event and decide if the event is real and should be put into the database using option 'p'. All work must be done from the directory where the raw waveform files are located. The process of putting an event into the database results in creating the S-file (option1), giving the event identifiers and copying the waveform files of registered events to the waveform directory.By pushing p(Regis), the user will be prompted for distance indicator, which has to be L, R or D for local, regional or distant event. It is possible here to enter 2 characters like LE or LV for local explosion or local volcanic event. The event type or event ID can be any character. Three characters are predefined and should only be used if the following definition correspond: P(probable explosion), E(explosion) or V(Volcanic event). The volcanic events have a sub classification which can be entered when registering an event as volcanic, see section 6.29. The process of registering the event into the database implies that a new S-file is created or registered and in the S-file. An operator ID will be asked and the operator ID will be put on the ID-line. The question about operator will only be asked for the first event since it is assumed that all subsequent events are put into the same database by the same operator. The event ID, can later be used with the SELECT program to select out particular event types. When first putting an event into the database, the user is also prompted for database.

Option (1)
Data is available as waveform files only and a list of files must be made first with DIRF. Main option 0, 1 or 2 can be used for plotting. The 'p' option creates the S-file and copies the waveform file to the WAV directory. The waveform file remains in the working directory. Unwanted waveform files can also be deleted so that when all events have been put in, only waveform files of 'real' events remain in working directory. These can then be plotted with one run of MULPLT, see section 6.2.1.

Option (2)
Data is available already in a database, however since the data has not been inspected, the waveform files are still in a work directory. In EEV, the first unprocessed event in the month is found with command 'ss' and MULPLT is started with command 'po' to invoke all defaults. If the event is to remain in the database, it must be registered with option 'p'. The process and the questions are the same as in option (1) except that the S-file is not created since it is already there. The S-file is cleaned for all processing information from SEISNET if present. This normally also includes automatic phase picks. However, they can be kept if parameter REG_KEEP_AUTO is set in the `SEISAN.DEF` file. The status of the files also changes to being newly registered as under option (1) (see definition of processing codes in Appendix 1) and waveform file(s) copied to WAV. Before registering, it might be an advantage to merge waveform files and delete unwanted files (could be false triggers), see section 6.2.4. Files can only be merged and deleted in working directory with commands Delw and Merge (Menu). In this process of putting new events into the database, it is also an advantage to delete unwanted events. This is done with option 'S'(Del S)'. The S-file is deleted, but the waveform files remain in the working directory.

Preprocessing of data while registering new events, option (1)

Normally a series of events are registered first and MULPLT terminated. Then EEV is started up for interactive picking and location. However, if preliminary processing is desired while registering the event, this is also possible.

Phase picks:If phases are picked before the event is registered, these readings are saved in the database at the time of registration. After the event has been registered, MULPLT automatically goes to the next event in FILENR.LIS and no more phase picking can be done.

Processing with a given program: Optionally MULPLT can, after registration, start any program processing the newly registered event. E.g. the AUTOPIC program can be started or a program reading amplitudes etc. The program name is defined in `MULPLT.DEF`.

Locating the event: As the final step after registration, the event can optionally be located and the location optionally placed in the database.
The above options have been put in on the suggestion of Brian Baptie, who is using it for rapid processing of volcanic events, where in most cases the operator only wants to look at the event once.

6.2.6 Adding BUD archive waveform data to nordic file, WAV2BUD

The program WAV2BUD reads nordic files like `collect.out` or `select.out` and add lines (type 6) to each event in the input file that link to the waveform data in a BUD archive. Note, it is only the data given in `SEISAN.DEF` by the ARC_CHAN parameter that are linked to. The events with the new data link is listed in the `budfile.out` file.
The program is written by **Ruben Soares Luís**.

6.2.7 Phase picking, amplitude, weight and polarity

Picking phases:

The plot will display any pick present in the database (current S-file). To pick new phases, position cursor at phase, and press the key as indicated on top of the screen (if in Single mode). E.g. pressing 1 will read IP. Pressing the same key again with the cursor at a different place will delete the old one (indicated with a D) and display the new one. Additional default phases, which can be picked, are i for I, e for E and A for AMP (note upper or lower case). Keys for phases have default definitions, but can be redefined using the file `MULPLT.DEF`, see below. The end of the coda is picked as a phase (C) and the program calculates coda length IF AND ONLY IF A P-READING IS PRESENT.

Picking amplitudes:

Position the cursor at the bottom or top of a wave and press a, then at the other extreme (bottom or top) and press a (do not use upper case, see below). There is no requirement for going left to right or top to bottom, it can be done in any order as long as the two extremes are marked. At each press, a cross is marking where the pick was made. In case a filter, like WA, MS or Mb is applied, the program will associate the amplitude with the respective amplitude reading (AML, AMS or Amb). Amplitude and period are calculated and stored with the phase. Otherwise, if none of these filters are applied, a menu pops up and the user needs to select a phase name to which the amplitude and period readings are associated. It is often a good idea to store amplitudes with the nondescript phase E, I or AMP since it then will remain even if the phase is deleted or changed. If an attempt is made to pick amplitude on a trace which is not in nm, the reading must be confirmed since SEISAN assumes all amplitudes to be in nm (see section on instrument correction). If no phase is picked, no amplitude is stored. The amplitudes are always assumed to come in pairs so if e.g. 3 amplitude values have been picked, and the

user tries to pick a phase or quit the program, it will appear frozen since the program is still waiting for the next amplitude measurement. It is always the last pair of amplitude measurements, which are used. Amplitudes can be picked on both corrected and uncorrected traces.

If A is pressed instead of a, the amplitude is read and marked automatically. It works in most cases, but sometimes two subsequent peaks are not correctly chosen and the amplitude reading has to be done manually. The method is to find the absolute extreme and then the largest amplitude before or after is selected in order to obtain the peak to peak amplitude, from which the amplitude is calculated by dividing by 2. For more information, see. ../LIB/auto_amp.for.

Component names when picking phases:

In the S-file, the component only has 2 letters while in the waveform file it has 4 letters. There must therefore be a unique translation between the two. This definition is given in the subroutine componen.for in LIB. Most common combinations are now defined, however if a new one is defined in the waveform file which does not exist in componen.for, the first and last letter of the input component will be used. If e.g. an input component is called SS Z, then the code in the S-file will be SZ. This means that picks for stations with components, which do not differ in first and last character, cannot be separated in the S-file. Component names for rotated channels will be e.g. SR and ST for short period radial and transverse components respectively.

Reading polarity:

If the cursor is above or below the trace at a distance marked by horizontal tics on the sides of the plot, the first motion is also picked and displayed. Do not use a filter if possible. Assigning weight:

A phase can be assigned a weight. Move the cursor close to a pick and press one of the keys 1-9 in UPPER case thus using e.g. !"# (default, can also be changed), and a HYPO style weight is assigned and displayed. Although weights 0 to 9 can be put in, HYP only uses 0-4 and 9 (see section 6.1.1). Phases with associated amplitude, period, azimuth or apparent velocity are displayed with a hat below on the phase indicator line. The default keys for the weights might not be correct on all keyboards, if not, set keys in `MULPLT.DEF`.

Automatic determination of coda length (C or c):

The coda length can be quite variable among different operators and a function has been made to automatically determine the coda length. The signal is bandpass filtered and the end of the coda is determined by a standard STA/LTA procedure. The parameters are set in the `MULPLT.DEF` file. Press C to find coda length automatically or c to determine manually. If parameter CODA AUTO is set in `MULPLT.DEF`, c I sused. The coda length can only be determined if a P-phase is present.

6.2.8 Theoretical arrival times for global and local phases and location

In order to assist in identifying seismic phases, there is an option for displaying the theoretical arrival times of several global and regional phases while picking phases. The steps to do so are the following:

1 Before entering MULPLT from EEV, the theoretical travel times have to be calculated for the current event. This assumes that the origin time and hypocenter is given in the header line or a subsequent type one line. If not, enter manually (from e.g. PDE) or use the EEV command INPUTEPI or INPUTONE. Then proceed to calculate the theoretical arrival times using EEV command IASP with the IASPEI91 traveltime tables (for more details, see section 6.20.4). The same command is also available inside MULPLT in multitrace mode. All arrival times (or a subset, see 6.20.4) for all stations in current S-file will now be calculated with program IASP and stored

in file iasp.out (no importance for the use, just for information). See Figure 17 for an example in multitrace mode. Note that very many theoretical phases can be generated if the S-file has many stations. MULPLT will stop if more phases are used than the dimensions are set up for (include file seidim.inc), and you must use fewer phases (a warning is given when 500 phases are generated) or set up SEISAN with larger dimensions, see section 3. Theoretical local crustal phases for the current model can be calculated with program WKBJ and displayed, see section 6.19. Theoretical phases can also be calculated when using the location option, see next section.

2 Pick phases: When a trace is displayed on the screen, all theoretical phases inside the time window will also be shown. To distinguish the theoretical phases, they are prefixed with a y and displayed below the trace (normal phases have I, E or blank and are displayed above the trace). Position cursor where you see a phase which you think corresponds to a theoretical phase and press y. The nearest theoretical phase will now be placed at that position with a prefix E. Only theoretical phases selected in this way will be written in the S-file. Note that the phase names can be up to 8 characters long, see Appendix 1 for the definition of long phase names.

If the phases fit badly, start looking at the P-phase. If that does not fit the theoretical P-phase, change the origin time in the S-file so that the P-arrival fits, and recalculate the theoretical phases.

PROBLEM: In multitrace mode, only one theoretical phase can be picked. Replot must be made before picking the next.

Locate earthquake

If several phases have been read and saved in the S-file, the event can, in multitrace mode, be located with command l (Locat), just as in EEV. The screen is cleared and the usual location rolls over the screen. When the location is finished, the plot will reappear and the calculated travel times will be displayed as synthetic phases (see previous section). In this way it is possible to immediately visualize the differences between the read and calculated phases. The output files are `hyp.out` and `print.out` as usual.

6.2.9 Instrument correction and magnitudes Ml, mb and Ms

The correction for instrument response is done by taking the spectrum of the selected window of the trace, dividing with the response function and converting back to the time domain. Any filtering specified is done in the frequency domain. Filtering is needed in most cases.

Ground motion

Option g(Groun) removes the effect of the instrument and displays a ground motion seismogram. After selecting g and the zoom window, there is a question of which type of seismogram to calculate: Displacement (d), Velocity (v) or Acceleration (a). The corrected trace is shown below in nanometers(nm), nm/sec or nm/(sec*sec) (if response information is available). Note that this might produce strange seismograms, since e.g. a SP seismograph has very low gain at low frequencies so noise might be amplified very strongly. It is therefore recommended to also do some filtering when using the g option.

Amplitude for determining Ml

For the w(WA)-option (Wood Anderson), the trace is corrected for the instrument to produce displacement. The displacement trace is then multiplied with the response of the Wood-Anderson instrument to produce a signal to look exactly like it would have been seen on a Wood-Anderson seismograph. The maximum amplitude (nm) is read and saved to the S-file with name IAML. The Wood-Anderson response (PAZ) is hardwired in SEISAN and it is similar to a 2 pole Butterworth high-pass filter at 3 Hz. In SEISAN versions prior to 8.3, a fixed 8 pole bandpass filter was used (1.25 Hz - 20 Hz). Filtering is

done in the frequency domain. For noisy traces it might also be required to put a filter at the high end. This can be specified in the `MULPLT.DEF` file. Unfortunately, the correct low cut filter with 2 poles will often result in the seismogram blowing up at low frequencies and might be quite useless for earthquakes with magnitude below 2.0 - 2.5 So in addition to the PAZ filter, a fixed bandpass filter can be added (see `MULPLT.DEF`). In the standard distribution of SEISAN, this additional filter is set to an 8 pole filter at 1.25 - 20 Hz. In all cases where an additional filter is used, the read amplitude is corrected for the filter gain and the true ground motion written in the S-file will be larger than the amplitude seen on the screen. The additional default filter probably only makes a difference for very large events (Ml > 5). Other filters at a higher frequency should only be used for small events (M<1) . NOTE: In SEISAN version 7.1.1 and earlier, the low cut filter was set by mistake to 0.8 Hz. Repicking amplitudes with the correct filter might change magnitudes of larger events slightly.

Displaying response information

The response function for the current channel can be shown with option ':' (Resp), see Figure 26. If no response function is given, a message is shown. If the response function is taken from the waveform file header instead of from the CAL directory, a message is given.

Amplitude for determining mb:

Determining mb assumes that the maximum amplitudes are measured on classical 1 Hz WWSSN instruments having a peak gain around 1.5 Hz. This in reality means a band limited measurement. To pick ground amplitudes for determining Mb on instruments with a broader or more narrow frequency band, like most high frequency SP instruments, some filtering must first be done. Using the j(mb)-option, the trace is corrected for the instrument to produce displacement. The displacement trace is then multiplied with the response of the SP WWSSN instrument to produce a signal to look exactly like it would have been seen on a SP WWSSN seismograph. The unit of the amplitudes seen on the screen is nm, however the amplitude will only represent the ground motion correctly at the frequency of the maximum gain at 1.5 Hz and for all other frequencies, the true ground motion will be larger than seen on the screen. The maximum amplitude is now picked and displayed below the trace, corrected for the gain relative to the gain at 1.5 Hz and written to the S-file with name IAmb. This means that the amplitude written to the S-file generally will be larger than the amplitude displayed on the plot. The SP WWSSN response (PAZ) is hardwired in SEISAN and cannot be modified with filters.

In SEISAN version to 8.2.1, the default filters used to simulate SP WWSSN were, by mistake, in the band 0.9 (2 pole) to 1.8 Hz (3 poles). This will result in slightly wrong magnitudes unless the user had put in correct new filter contants... Prior to SEISAN version 8.2, the default filters used were 0.5 Hz (8 pole) and 5.0 Hz (8 pole filter), which was close to the correct values. No correction for relative gain was used in SEISAN versions prior to 8.3.. All of these changes could have resulted in smaller errors in mb, which only can be corrected by repicking the amplitudes.

Amplitude for determining mB

Amplitude for mB is defined as the maximum velocity on a wide band instrument (0.2 -30 sec or 0.033 - 5 Hz). The maximum amplitude Vmax is measured on a velocity trace. Using the J(mB) option, a velocity trace (nm/s) in the frequency band 0.033 - 5 Hz is displayed. The maximum amplitude in nm/s (irrespective of frequency) is picked and displayed below the trace. This amplitude is now written to the S-file with phase name IVmB_BB. In principle, mB can be calculated using any instrument, but in practice it can only be used if the P-signal is seen clearly on an unfiltered broad band velocity record. The Butterworth filter 0.033 - 5 Hz , 8 poles, is hardwired and it cannot be modified with additional filters.

Amplitude for determining Ms:

The attenuation function for determining Ms assumes that the amplitudes are measured on classical LP WWSSN instruments having a peak gain around 15 second. To pick ground amplitudes for determining Ms on instruments with a broader or more narrow frequency band, like most broad band instruments, some filtering must first be done. Using the k(Ms)-option, the trace is corrected for the instrument to produce displacement. The displacement trace is then multiplied with the response of the LP WWSSN instrument to produce a signal to look exactly like it would have been seen on a LP WWSSN seismograph. The unit of the amplitudes seen on the screen is nm, however the amplitude will only represent the ground motion correctly at the frequency of the maximum gain at 15 seconds and for all other periods, the true ground motion will be larger than seen on the screen. The maximum amplitude is now picked and displayed below the trace. This amplitude is then corrected for the gain relative to the gain at 15 seconds and written to the S-file with name IAMs_20. This means that the amplitude written to the S-file generally will be larger than the amplitude displayed on the plot. The LP WWSSN response (PAZ) is hardwired in SEISAN and no additional filters can be used.

The attenuation function for determining Ms assumes that the amplitudes are measured in the period range 18 - 22 sec and it is up to the user to make sure that the the amplitude is in the correct range..

For SEISAN 8.2.1, the default filters used were in the band 0.038 (2 pole) to 0.1 Hz (1 pole). Prior to SEISAN 8.2 default filters were 0.042 to 0.063 Hz (8 pole filter). No correction for relative gain was used in SEISAN versions prior to 8.3. These changes might have resulted in small errors ins Ms and can only be corrected by repicking the amplitudes.

Amplitude for determining MS

Amplitude for MS is defined as the maximum velocity on a wide band instrument (3 -60 sec or 0.017 - 0.3 Hz). Using the K(MS) option, a velocity trace (nm/s) in the frequency band 0.017 - 0.3 Hz is displayed. The maximum amplitude in nm/s (irrespective of frequency) is picked and displayed below the trace and written to the S-file with phase name IVMs_BB. In principle, MS can be calculated using any instrument, but in practice it can only be used if the surface wave is seen clearly on an unfiltered broad band velocity record. The big advantage with using MS is to avoid the 18-22 s limitation needed for Ms. The Butterworth filter 0.017 - 0.3 Hz , 8 poles, is hardwired and cannot be modified with additional filters.

Problem: If a long trace (large number of samples) is used, the instrument correction might fail (funny result seen) due to numerical overflow in the spectral conversion. Choose a shorter window.

6.2.10 Determine azimuth of arrival (3 comp or array) and component rotation

Azimuth of arrival from 3-component stations, h(Azim)

If a 3 component station is available, the azimuth of arrival can be determined using the method developed by Roberts et al. [1989]. Display any of the 3 components and press h (Azim). Then select a zoom window around the P-arrival of a few secs duration for the analysis. The 3 components will now be displayed below in order Z, N and E and the calculated azimuth, apparent velocity and correlation will be displayed at the bottom line. In order to check the stability of the estimate, try different windows and filters. Often, a filter must be used to get reliable results. The displayed azimuth and apparent velocity is only saved in the S-file when an associated phase is picked. THAT PHASE MUST BE PICKED ON THE SINGLE UPPER TRACE SEEN ON THE SAME SCREEN. If there is none, use I or E. The velocity estimate is not very reliable and is dependent on the local velocities. In order to calculate the apparent velocity, the P-velocity of the top layer must be given. The default value is 5.0 km/sec, but another value can be set in the `MULPLT.DEF` file. To get a good estimate, the correlation coefficient should be as high as

possible and positive. The quality of the obtained azimuth can be tested by locating the event with the calculated azimuth weighted out and observe the azimuth residual. Figure 25 shows an example.

Azimuth and apparent velocity from array data, FK analysis f(FK)

Using this command, the traces seen on the screen will be put into the FK program and an FK plot will be displayed. The azimuth and apparent velocity with the highest correlation is selected by the program, however any other value can be manually selected. The values will ONLY enter the S-file if associated with a phase in the same way as amplitudes are picked. For more details, see section 6.30.

Rotated seismograms

Option u(Rotat) will rotate the horizontal components for the next plot if the two horizontal components are available. The rotation will display the radial component instead of the N-component and the transverse component instead of the E-component. The back-azimuth used is displayed above the trace. All channels will be displayed rotated until u(Rotat) is pressed again This means that phases can be picked and spectra made with the rotated channel. When picking phases on rotated signals, these will appear in the S-file with components R or T instead of N and E respectively. This also means that only if the rotated signals are shown, will the phases read on rotated channels appear on the plot. The station back-azimuth is obtained in the following way: If a hypocenter is given in the header line, the angles are calculated using the current `STATIONx.HYP` file. If no hypocenter is available, the angle will be read from the S-file under column observed azimuth (47-51) (if not blank) and the azimuth residual will be added. This option permits the user to first determine the azimuth with the 3-component option and then rotate the signals with the determined azimuth. Finally, if no observed azimuth is available, the event to station azimuth + 180 deg. will be used if available (column 77-79). If no back-azimuth can be found, no rotation is done and an angle of 999 deg. is displayed. If in single trace mode and choosing the 3-component option AND the rotate option, the user will be prompted for a rotation angle and the rotated channels will be shown in the usual 3-component plot, however, the azimuth determined is done with the unrotated channels.

PROBLEM: In general, the R-channel will use the response of the N-channel and the T-channel will use the response of the E-channel so for instrument response removal to be correct, the 2 channels must have the same response curve.

6.2.11 Data manipulation commands

Select other channels: o(Oth)

The channel selection menu comes up again.

Go back one channel in single trace mode, go back one event in multitrace mode if MULPLT is started from EEV: B(Back)

Select other waveform files from S-file: W(OthW)

If more than one waveform file available for the event, one or several others can be selected.

Delete waveform files:

This can only be done in multitrace mode: The command is d(DelW) and the cursor must be above the top frame of the plot. There are two possibilities:

1 Input is from `filenr.lis`: The current file is deleted and if in default mode, the plot moves on to the next event.

2 MULPLT is started from EEV: If only one waveform file is available, the program proceeds as under (1). The waveform file is deleted and the waveform file entry in the S-file remains. However, if more than one waveform file is available, the user can use a menu to select which files to delete. Only the waveform file entries in the S-file are deleted, the waveform files remain. This option is mostly used with SEISNET.

Delete S-files D(Del S)

This command deletes the current S-file. It can only be used if MULPLT is called from EEV. No waveform files are deleted.

Merge waveform files given in S-file M(Merge)

The files will be merged to one waveform file and the old individual file names removed from the S-file and replaced by the new file name of the merged file. The original waveform files remain. Files to be merged will be shown on a menu. Mostly used with SEISNET. The user MUST have files in working directory. If files are in the data base, they will not be shown on the merge menu.

Output of binary waveform file, O(Out)

It is often useful to be able to select part of a waveform file and save it. The Out option makes an output file of the traces AS DISPLAYED ON THE SCREEN with exactly the same channels, and time window in a file with a standard SEISAN waveform name. The output format is always SEISAN, even if some input files have a different format. The network code in the file name will ALWAYS be the station code if all channels are from the same station Otherwise the network code has the default name MERGE. Alternatively the parameter MERGE_WAVEFORM can be set in `SEISAN.DEF`. The data is output exactly as displayed on the last screen, so if filtering or instrument response has been made, the output file will also be filtered or instrument corrected. If any filtering or instrument response correction has been done, a note will be inserted in the SEISAN waveform header so the user can see that this is no longer the original data. The note could be e.g. ‘ Displacement 1.250- 20.0 Hz’ indicating that output has been filtered and converted to displacement (nm). Note that numbers have been scaled so only if the SEISAN file is read with standard SEISAN routines, will the numbers be correct. If the output file is converted to ASCII by SEIASC, the number shown must be multiplied with a given scaling factor, see SEISAN binary format description (Appendix B). There is no response information in the header other than the short text. Since the station code is still the same, it is technically possible to correct for the response again using the response information in the CAL directory, **however, be aware that this will give wrong results**.

Output of ASCII waveform file

This option only works if parameter SPECTRAL_OUTPUT has been set in `MULPLT.DEF`. The output file signal.out contains the last data displayed in the single trace zoom window (in ASCII and real numbers). This option is a another way (see option O(Out) above) of getting an output file that has been filtered or instrument response corrected. The main difference is that this file is only for one trace written in ASCII.

Fixed scaling

Normally all traces are plotted with autoscaling. However, it is sometimes useful to be able to scale the traces with a fixed scale in order to e.g. compare traces or override the autoscale in case a spike distorts the autoscaling. Option *(Scale) will prompt the user for a maximum count to use for the scaling of all traces.

Example of using MULPLT on SUN:

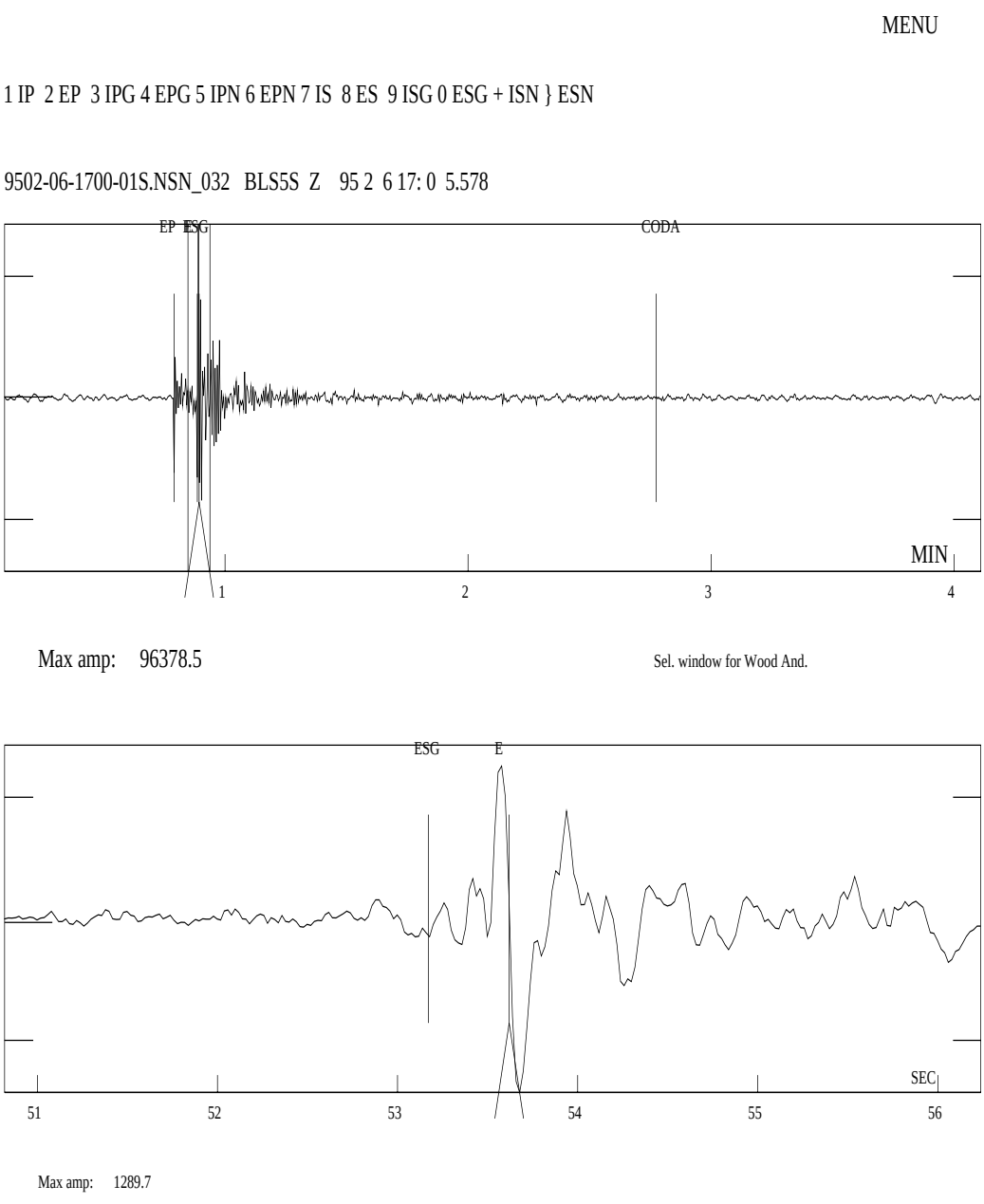


Figure 6.7: Using MULPLT for picking phases. The top shows the original trace and the bottom the zoomed part. Note that the amplitude has been associated with the phase E and not the ESg. This means that if the S-phase is deleted, the amplitude will remain.

Comments are given with ! in front

This example shows how running MULPLT from EEV would look.

```
/top/seismo/REA/BER__/1991/01/01 0557 12L.S199101 ! S-file name
Read header from file /top/seismo/WAV/9101-01-0557-12.WNN_13

 Plot options:     Interactive picking          Return  ! first choice
                   Multi trace plot on screen, def (0)
                   Multi trace plot on screen      (1)
                   Multi trace plot on screen+laser(2)
                   Multi trace plot on laser       (3)
                   Multi trace plot on laser       (3)
                   Continuous on screen            (4)
                   Continuous on screen + laser    (5)
                   Continuous on laser             (6)
                   Stop                            (9)
                       ! now comes a menu for selection and then
                       ! the plot appear in Single mode since a
                       ! return was made
```

The next example shows how to plot many events in one go, first make a list with DIRF.

```
dirf 9101-10*                       ! events from January 10, 1991
#  1  9101-10-0915-15S.KMY_03
#  2  9101-10-1510-55S.NSS_12
#  3  9101-10-2333-44S.NNN_11

mulplt
file name, number, filenr.lis for all
filenr.lis             ! plot all events in filenr.lis
 Resolution in cm/sec, 0: plot all on one page (default)
0                      ! scale will be different for each plot!!!
Read header from file:9101-10-0915-15S.KMY_03
 Page           1
 Channel:           1

Plotfile sent
Read header from file:9101-10-1510-55S.NSS_12       ! next event in list
 Page           1
 Channel:           1
 Channel:           2
 Channel:           3
 Channel:           4
 Channel:           5
 Channel:           6
 Channel:           7
 Channel:           8
 Channel:           9
 Channel:          10
 Channel:          11
 Channel:          12

Read header from file:9101-10-2333-44S.NNN_11
 Page           1
 Channel:           1

 ! etc.


Plotfile sent
Read header from file:9101-10-1510-55S.NSS_12       ! next event in list
 Page           1
 Channel:           1
 Channel:           2
 Channel:           3
 Channel:           4
 Channel:           5
 Channel:           6
 Channel:           7
 Channel:           8
 Channel:           9
 Channel:          10
 Channel:          11
 Channel:          12

Read header from file:9101-10-2333-44S.NNN_11
 Page           1
 Channel:           1

 ! etc.
```

6.2.12 Spectral analysis, s(Spec)

The spectral analysis option for local and teleseismic events is selected in single trace mode. The spectral analysis is based on the Brune [1970] model and various assumptions about the geometrical spreading and anelastic attenuation.

The theoretical displacement spectrum d(f)[Brune, 1970] is:

$$d(f) = G(r,h) * D(f) * Moment * KK/(1 + f**2/f0**2) * (4 * pi * DE * V**3))$$

where G(r,h) is geometrical spreading, r is epicentral distance, h is hypocentral depth, D(f) the diminution functiondue to anelastic attenuation, f is the frequency, DE the density, V the velocity at the source, f0 the corner frequency and KK a factor of 2.0*0.6 to correct for the free surface effect and radiation pattern.

The diminution function D(f) is written as

$$D(f) = P(f) * exp(-pi * f * trtime/(q0 * f**qalpha))where$$

trtime is the travel time from the origin time to the start of the spectral window and

$$P(f) = exp(-pi * kappa * f)$$

is meant to account for near surface losses [Singh et al., 1982] with the constant kappa having a value of the order 0.02 sec. Anelastic attenuation Q is assumed to be frequency dependent following the relation $Q = q0 * f**qalpha$.

For teleseismic events, only t* is used and Q must be set to zero (not used). The t* parameter is the same as kappa and is usually set to 1.0 (same value is used for P and S)..

The geometrical spreading has been defined to be dependent on the wave type with several possibilities, all made equivalent to a distance called geo_distance (GD) such that geometrical spreading is expressed as 1/GD. There are several possibilities for GD:

Local and regional events geometrical spreading

P-waves:
GD is the hypocentral distance $(HD) = sqrt(r * r + h * h)$ so body wave spreading is assumed.

S-waves:
The geometrical spreading has been made dependent on distance and depth. At short distances, the geometrical spreading is assumed to be body wave spreading. For distances beyond the Herrmann-Kijko distance (default of 100 km) and a shallow focus, the following relation is used:

$$G(r,h) = 1/r = 1/GDforr < 100km$$

$$G(r,h) = 1/sqrt(100 * r) = 1/GDforr > 100km$$

which is commonly used [Herrmann, 1985; Herrmann and Kijko, 1983]. This relation assumes surface wave dispersion for epicentral distances larger than 100 km. In SEISAN 100 km is the default, however it can also be set to any other value by the parameter HERKIJ_DISTANCE (see later).

The above relation breaks down if the depth is large or comparable to the epicentral distance and in that case body wave spreading is again assumed. In order to get a smooth transition from surface wave to body wave spreading, it is assumed that the relation changes nearly linearly from surface wave spreading to body wave spreading between the depths GEO_DEPTH1 to GEO_DEPTH2. For depth less than GEO_DEPTH1(default 50 km), Herrmann-Kijko spreading is assumed, for depths larger than GEO_DEPTH2 (default 100 km), body wave spreading is assumed with the transition in between. In each case the geometrical spreading term is given as the equivalent GD, which is also recorded in the database. These 3 parameters can be used to change geometrical spreading. If e.g. HERKIJ_DISTANCE is 10 000 km, body wave spreading is always used. For more info, see [Havskov and Ottemöller, 2010].

Geometrical spreading for teleseismic events

The geometrical spreading is approximated with [Havskov and Ottemöller, 2010]

$$G(r) = 1/GDwhereGD = (27 + .)/0.0048$$

where . is epicentral distance in degrees. This approximation is only valid for h ¡ 100 km and . ¿ 30 degrees.

From the spectral parameters, source radius and stress drop can be calculated as follows:

$$Sourceradius = 0.37 * V/f0$$

where f0 is the corner frequency and V the P or S-velocity at the source for P and S-spectra, respectively. The velocities are set in `MULPLT.DEF`.

$$Stressdrop = 0.44 * Moment/(sourceradius) * *3$$

The spectral analysis is used in two ways. The first and most common is to make the attenuation and instrument corrected displacement spectrum and determine the flat spectral level OM0, and corner frequency f0 from which the seismic moment, source radius and stress drop can be calculated. The second option is to display the instrument corrected spectrum (displacement, velocity or acceleration) and model the spectrum for corner frequency and attenuation parameters. In this case no correction for attenuation should be made.

Spectral analysis to determine moment, source radius and stress drop:

Select the spectral option, s(Spec). Before the spectrum comes up, you will get a question of the type of spectrum wanted. The possibilities are displacement (d), velocity (v), acceleration (a) or raw spectrum (r). For determination of Moment etc, the displacement spectrum MUST be selected. Unless raw spectrum is selected, the spectrum will be instrument corrected. If no response file is available in CAL, a message will be displayed on the screen and the raw spectrum calculated. At this stage it is also possible to change the velocity from the `MULPLT.DEF` value or the moment given in the S-file (see spectral fitting below). The spectrum shown will normally show both the spectrum from the selected time window as well as a noise spectrum from an identical length time window at the start of the trace. IF NO NOISE SPECTRUM is desired, select spectrum with capital S instead of s.

The spectral analysis produces two output files:

`com_spec.out`: The complex spectrum with some additional information needed for surface wave analysis, must be displacement spectrum.
`amp_spec.out` : The real spectrum given as frequencies and amplitudes. The files are only generated

if parameter SPECTRAL_OUTPUT is set in `MULPLT.DEF`. Setting this parameter will also generate an ASCII waveform file with the input signal used.

Power spectra: The above spectra can also be displayed as power spectra if capital letters are used. Using e.g. 'V' instead of 'v' will show the power velocity spectrum.

When the spectrum comes up (see example in Figure 24, the axis units are log amplitude in nanometers-sec (displacement) versus log frequency (Hz). The cursor can be used to select the level, corner frequency and slope by defining the spectrum with a 3 point selection. This 3-point selection is finished with f, q or r with the same meaning as in picking mode. The spectral values are displayed on the screen once q, f or r is pressed. The abbreviations are

General parameters

Vel:	Velocity used (km/sec) (Vp or Vs)
Dens:	Density (g/(cm**3)
Dist:	Hypocentral distance (km)
q0:	q0 for spectral amplitude correction
qalpha:	qalpha for spectral amplitude correction
k:	kappa

NOTE: The veleocity is the velocity at the source, so if the dataset contains both shallow and deep earthquakes, a single velocity will be an approximation. There are two solutions to this problem. A: Use different MULPLT.DEF for deep and shallow events. Since the the MULPLT.DEF used is taken from working directory if available, you can have different directories when working with different depth earthquakes. B: Use the general MULPLT.DEF in DAT and change the velocity after making the spectum in the S-file. At the next update, the moment etc will be recalculated with the new velocity. On top of the general parameter is indicated which kind of spectrum is assumed, P or S. In order for the program to automatically determine which kind of spectrum to assume, there must be a P or S reading displayed on the screen near the time window analyzed. The reading must be within 10 sec of the start of the window. If both a P and S-reading is within 10 secs, the nearest phase is chosen. If it cannot be determined which kind of phase is analyzed, the user will get a question to select type of phase (can also be changed later when spectral choices come up) The determination of which phase influences the further calculation of geometrical spreading and moment (uses P or S-velocity).

If f is selected, the spectral values together with calculated moment etc are stored in the S-file at the next key press (see parameters below). Spectral values in S-files accumulate, since no old values are deleted !!!. This is because the spectrum might be made under different conditions (start time, time window etc). The input parameters for the spectral analysis is given in file `MULPLT.DEF`, which can be in either DAT or the working directory, see below. Additional parameters for geometrical spreading are given in `SEISAN.DEF` in DAT.

The spectral parameters are calculated using the relations

$$Moment = 4 * pi * DE * V**3 * 10**OM/(G(r,h) * KK)$$

where V is the seismic wave velocity at the source (P or S if P or S-spectrum respectively) and OM the spectral flat level on the attenuation corrected displacement spectrum.

$$Momentmagnitude = 2/3 * log10(moment) - 6.06 which is equivalent to the relation$$

$$Momentmagnitude = 2/3 * log10(moment) - 10.73 if momentisindynes - cm$$

[Kanamori, 1977].

The moment is calculated in Nm, the source radius in km and the stress drop in bars. All results are written to the S-file. Below is an example:

```
SPEC ITK S  Z MO 13.0 ST  4.2 OM  1.5 f0 9.45 R   .22 AL 2.50 WI  4.0 MW  2.6 3
SPEC ITK S  Z K 0.002 T    7  GD   52 VP 6.00 DE 3.00 Q0   .0 QA 1.00 VS  3.5 3
```

Note that no special line has been created in the Nordic format. Comment lines are used with SPEC at the start of the line followed by station and component. Only the first 4 characters of the 5 character station name is used. The station and component names are given at the start of the line. In case of a 5 character station name, the station name is shifted one character to the left. The information is:

MO:	log of moment, unit Newton*m
ST:	Stress drop in bars
OM:	log spectral level (nm-sec) not distance corrected
F0:	Corner frequency (Hz)
R :	Source radius (km)
AL:	Decay of log spectrum
WI:	Spectral window used (secs)
MW:	Moment magnitude
T :	Start time of window for spectrum in hr, min, sec
K:	Kappa
GD:	GEO_DISTANCE in km
VP or VS:	Velocity in km/sec at source for P and S-spectra respectively. The P or S in this line indicated if the spectrum is a P-spectrum or an S-spectrum. It MUST be P or S to be used for magnitude determination. A '?' is put in if MULPLT does not know which kind of spectrum is (no P or S reading near start of spectral window). This can be changed by editing the S-file afterwards.
DE:	Density in g/cm**3
Q0:	q0 in relation Q = q0 * f ** qalpha
QA:	qalpha

Note: The component codes have not been adjusted for SEED so the location code is not included.

Note: In earlier versions (before version 7.0), the field for kappa was used for the travel time to start of window. This can be calculated from origin time and the start time of the window.

NOTE: MOMENT IS NOT CALCULATED IF THE SPECTRUM IS NOT IN DISPLACEMENT.

When doing an UPDATE of the database or just a location with HYP, all distance dependent spectral values are recalculated and average values written into the output file. Mw will be calculated from the average value and written in the header line. **However, the original distance dependent Q and kappa correction is not changed**, since this correction was used to modify the spectrum used for reading parameters. Normally a small distance change has insignificant influence on the spectral level or the corner frequency so the Q-correction should be no problem. Spectra of the same type (P, S or ?) and from the same channel are overwritten. Only in case of UPDATE are the values written back into the database.

Display of spectral parameters: Program MAG can read and plot relations between spectral and source parameters. Program REPORT can read spectral parameters and combine in a table.

Potential problem with Q-correction: If the origin time in header is wrong, the Q-correction can be very wrong.

There must be a phase line in the S-file with component and distance corresponding to the spectra made in order for the spectral values to be calculated.

Spectral fitting

Once the spectrum has been shown (displacement, velocity or acceleration), a theoretical spectrum can be calculated and superimposed on the observed spectrum in order to forward model either source parameters or attenuation.

Entering constants and modeling: The modeling can only take place when the spectrum is seen on the screen.

Press s or S and a question will appear to enter the constants f0, k, Q0 and qa which are as defined above except qa is Qalpha. Once these parameters have been entered (terminate with return), the theoretical spectrum (displacement, velocity or acceleration depending on what is used for the spectrum) is calculated and superimposed on the observed spectrum. The parameters used or calculated are displayed. The level of the theoretical spectrum is adjusted so it approximately passes through the observed spectrum and the level difference is printed out on the screen (see below). S or s can now be pressed again and a new theoretical spectrum calculated and plotted. To get out of the spectral fitting loop, type r or q as usual.

Which constants and parameters are used: The moment is taken from the S-file if an average moment has been calculated (see UPDATE command). If no moment is available, it can also be entered the first time the spectrum is shown. If no moment is given, a log moment of 1.0 is used. The distance and depth is likewise taken from the S-file. If no distance is available, a distance of 1 km is used. If all 4 parameters f0, k, Q, qa are entered, stress drop is calculated with the relation given above. If the corner frequency is given as zero, the user will be asked to enter the stress drop and the corner frequency is calculated from the stress drop. If Q is zero, no Q-correction is made. **IMPORTANT: The Q and qa used here are distinct from the Q0 and Qalpha used for making the amplitude spectrum and both should not be used when modeling since this would imply a Q-correction two times. The best way is to use Q0=0 and kappa=0, so that Q is only corrected for when modeling**. The distance used is everywhere is GEO_DISTANCE.

The spectral parameters shown are:

Obs - calculated level:	The difference in log absolute level of the observed and calculated spectra. If a correct moment is used it should be small, in the order of 1.
Moment:	Moment used
Geo dist:	Geo distance used
Stress drop:	Stress drop in bars
f0:	Corner frequency
k:	Constant used in diminution function
q:	q0 used in spectral fitting
qa:	qalpha used in spectral fitting

Power spectrum and noise spectrum

The 3 types of spectra (displacement, velocity and acceleration) can optionally be made as power spectra. Instead of selecting the type of spectrum by pressing d, v or a, just press the same characters in upper

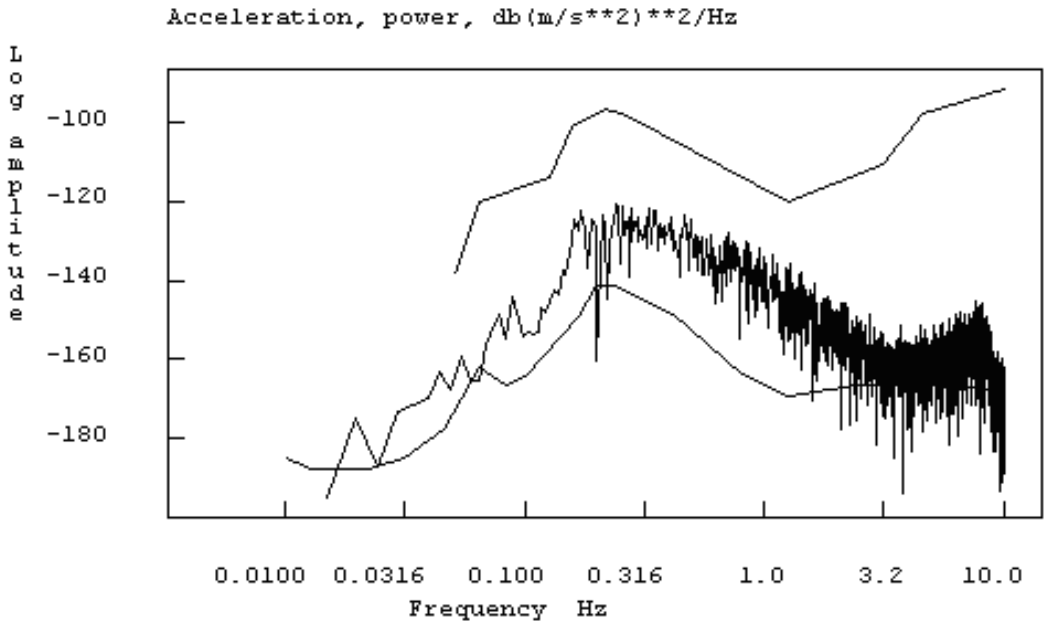


Figure 6.8: Example of a noise spectrum.

case and the power spectrum will be shown.

In seismic noise studies, the seismic background noise is often displayed as acceleration power spectral density in dB relative to ((1m/s**2)**2)/Hz. Instead of selecting d, v or a, press n instead. The plot shows the Peterson [1993] new global high and low noise models superimposed on the observed spectrum (Figure 6.8). When doing noise spectra, no attenuation correction is done. The normalization of the spectrum is as follows

$$P = |F^{DFT}|^2 \times \frac{\Delta t^2}{T} \times 2$$

where P is the Peterson Power spectrum, F^{DFT} is the discrete Fourier transform, Δt is the sample interval and T is the length of the time window. The factor 2 comes from the fact that only the positive frequencies are used so only half the energy is accounted for. The total power is proportional to the length of the time window since the noise is considered stationary, so by normalizing by T, the length of the time window should not influence the results. This noise option is a handy method of checking the noise characteristics of a given seismic station and compare it to global standards. This kind of analysis can also be done with the SPEC program (section 6.23). For more information, see instrument.pdf in INF.

Problems: There is currently no check if a displacement seismogram has been calculated when calculating the spectral parameters. If spectral analysis is done outside EEV (output in MULPLT.OUT) or with EEV when there is no origin time and/or epicentral distance, the output results are wrong for moment etc. Before calculating moment etc, the S-file MUST HAVE BEEN UPDATED SINCE BOTH THE DISTANCE AND ORIGIN TIMES ARE USED. If the spectra get very high amplitude levels when correcting for instrument, this might be caused by correcting for Q. With a Q of 100 and a distance of 10 000 km, this gives a very large correction. The Q-correction can be disabled in the `MULPLT.DEF` file.

If picks are made, but no readings appear in the S-file or readings appear with wrong component, the waveform file component might not have been defined in subroutine componen.for. If poles and zeros are used to remove the response, rotation cannot be used at the same time.

6.2.13 Particle motion plots

Particle motion plots can be made in multi trace mode when three components from one station are selected. The particle motion is plotted below the rescaled trace plot. The particle motion plots are made for the time window shown in the trace plot. The trace plot has all the normal functionality, so it is still possible to zoom and filter. The particle motion plots can be useful when determining phase types. No readings can be made from the trace plots.

6.2.14 The MULPLT.DEF and SEISAN.DEF files

In the MULPLT.DEF and the SEISAN.DEF files, it is possible to set the various parameters for MULPLT. Nearly all parameters are set in the MULPLT.DEF except geometrical distance parameters, which are set in SEISAN.DEF since these parameters also are used by HYP. MULPLT will operate without DEF-files using hardwired defaults. The MULPLT.DEF can be located in the working directory and or in DAT. The if a DEF-file is present in the working directory, it overrides the file in DAT. In MULPLT.DEF, several groups of parameters can be set: The keyboard, default channels to use and analysis parameters (e.g. for spectral analysis). The parameters are identified by keywords, see example file below for explanation. Note that all numbers given in file are real and must be given a '.'

Example file:

This file is for defaults for MULPLT and called MULPLT.DEF. The name must be in upper case on SUN. The following shows the parameters, which can be set. The file can contain any number of lines in any order, only the lines with recognized keywords and a non blank field under Par 1 will be read. Numbers under Par1 and Par2 must be given as reals. The comments have no importance.

```
KEYWORD.............Comments.............Par 1.....Par 2
X_SCREEN_SIZE       Size in pixels         90.0
PHASE NAME KEY      Phase key and phase
PHASE WEIGHT KEY    Weight key and weight
PHASE MOUSE KEY     Mouse key character
SPECTRAL Q0         Q0                     440.0
SPECTRAL QALPHA     Q = Q0**Qalpha         0.70
SPECTRAL KAPPA
SPECTRAL P-VELOCITY P velocity              6.2
SPECTRAL S-VELOCITY S velocity              3.6
SPECTRAL DENSITY    Density
3COMP VELOCITY      velocity for 3 comp
RESOLUTIONX         # points pl. screen   1500.0
RESOLUTIONHC        # points pl. hc       3000.0
NSORT_DISTANCE      0: no sort, min ph.
NCHAN PER SCREEN    max # chan screen
CHANNEL SORTING                            1.0
SPECTRAL F-BAND                            0.01      20.0
AUTO_LOCATE         0,1,2       0,1,2      0.0       2.0
```

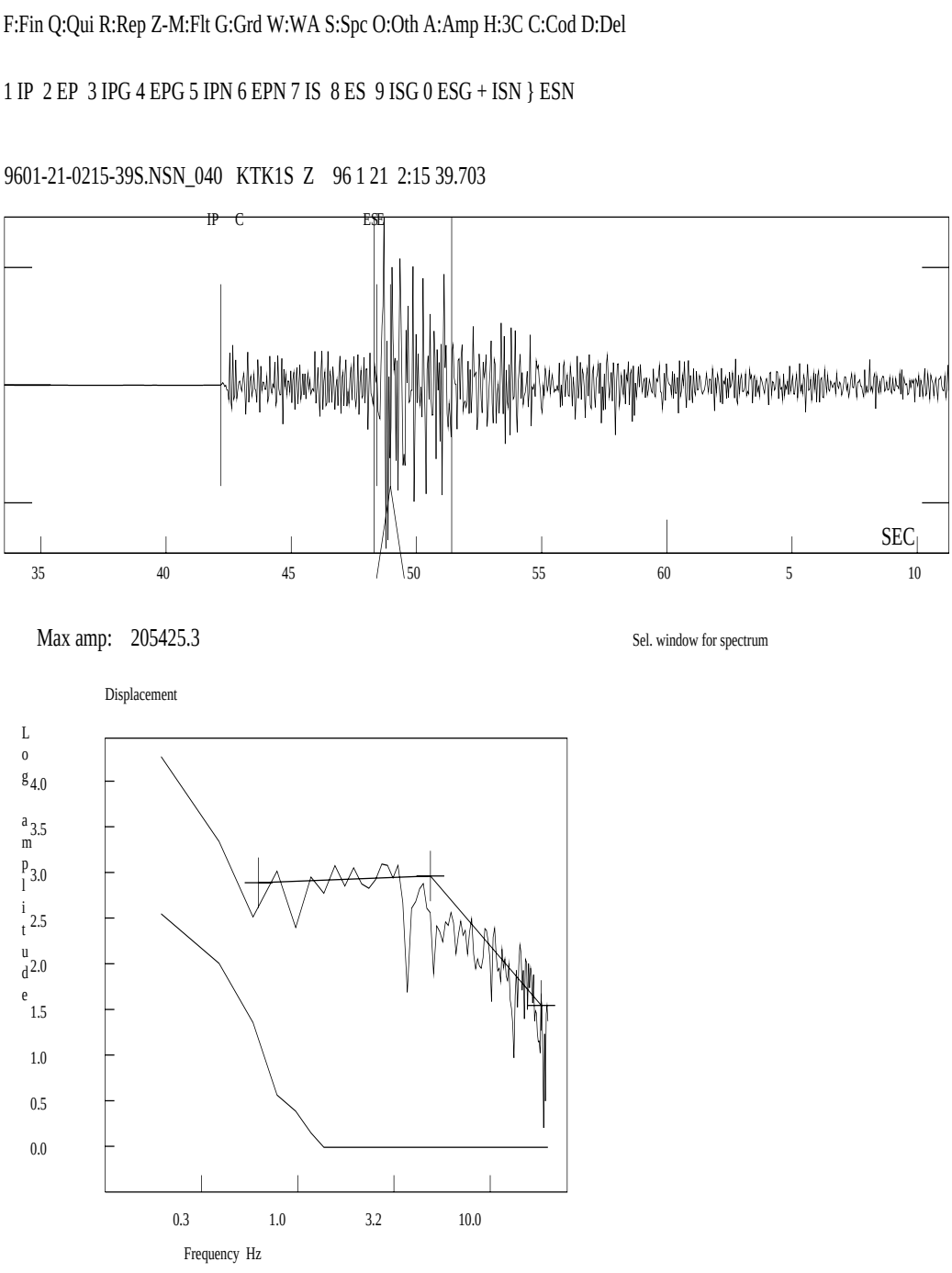


Figure 6.9: Spectral analysis
On top the original trace is seen and on the bottom the displacement spectrum (log -log, unit nm-sec and Hz). The level and slope has been indicated interactively. Note the noise spectrum at the bottom of the figure.

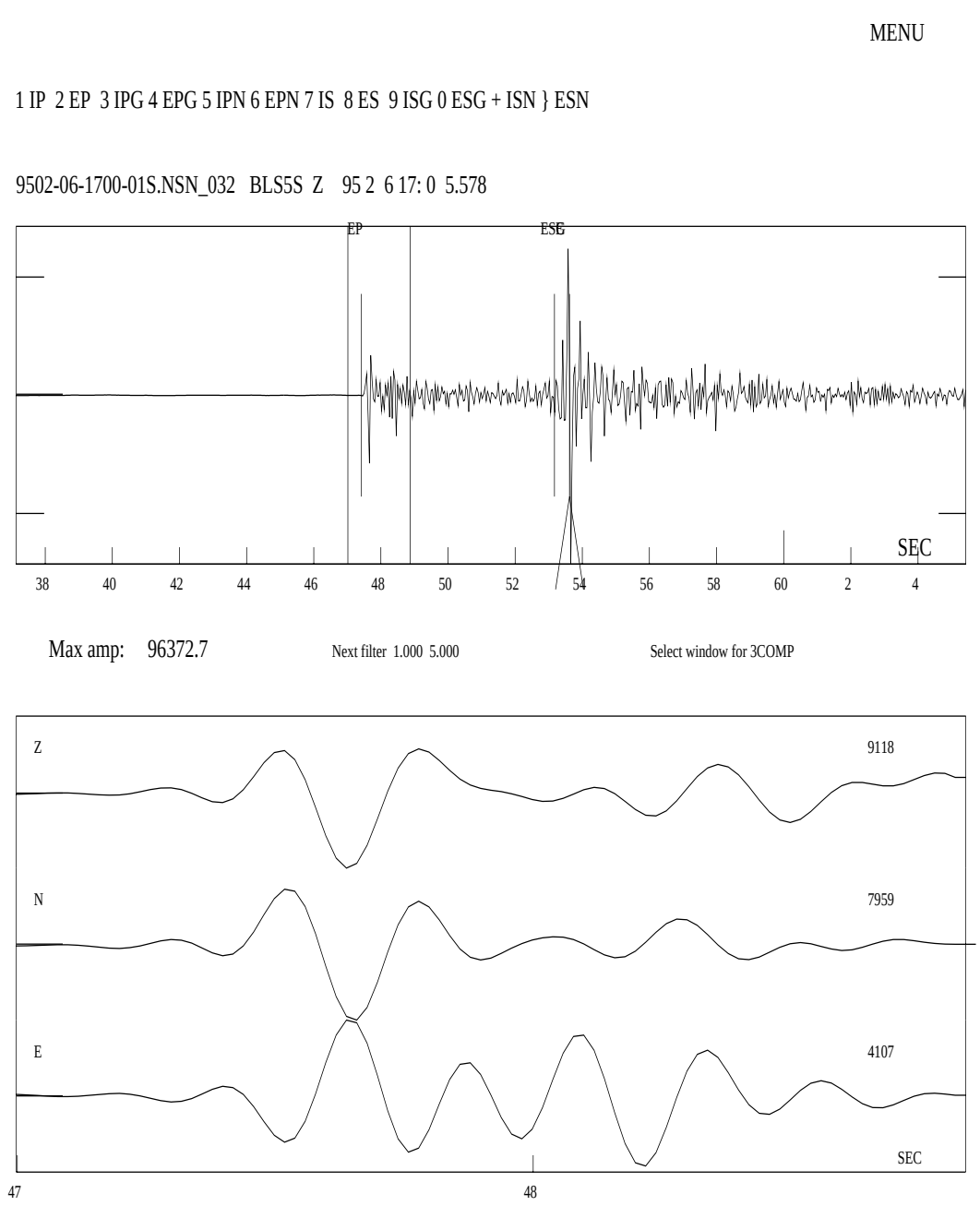


Figure 6.10: Three component analysis
On top the Z-channel is shown together with the window used for the 3 channels Z, N and E shown below. The signals below has been filtered between 1 and 5 Hz and the resulting azimuth of arrival is 160 degrees and a correlation coefficient of 0.2. The apparent velocity is 9.8 km/sec.

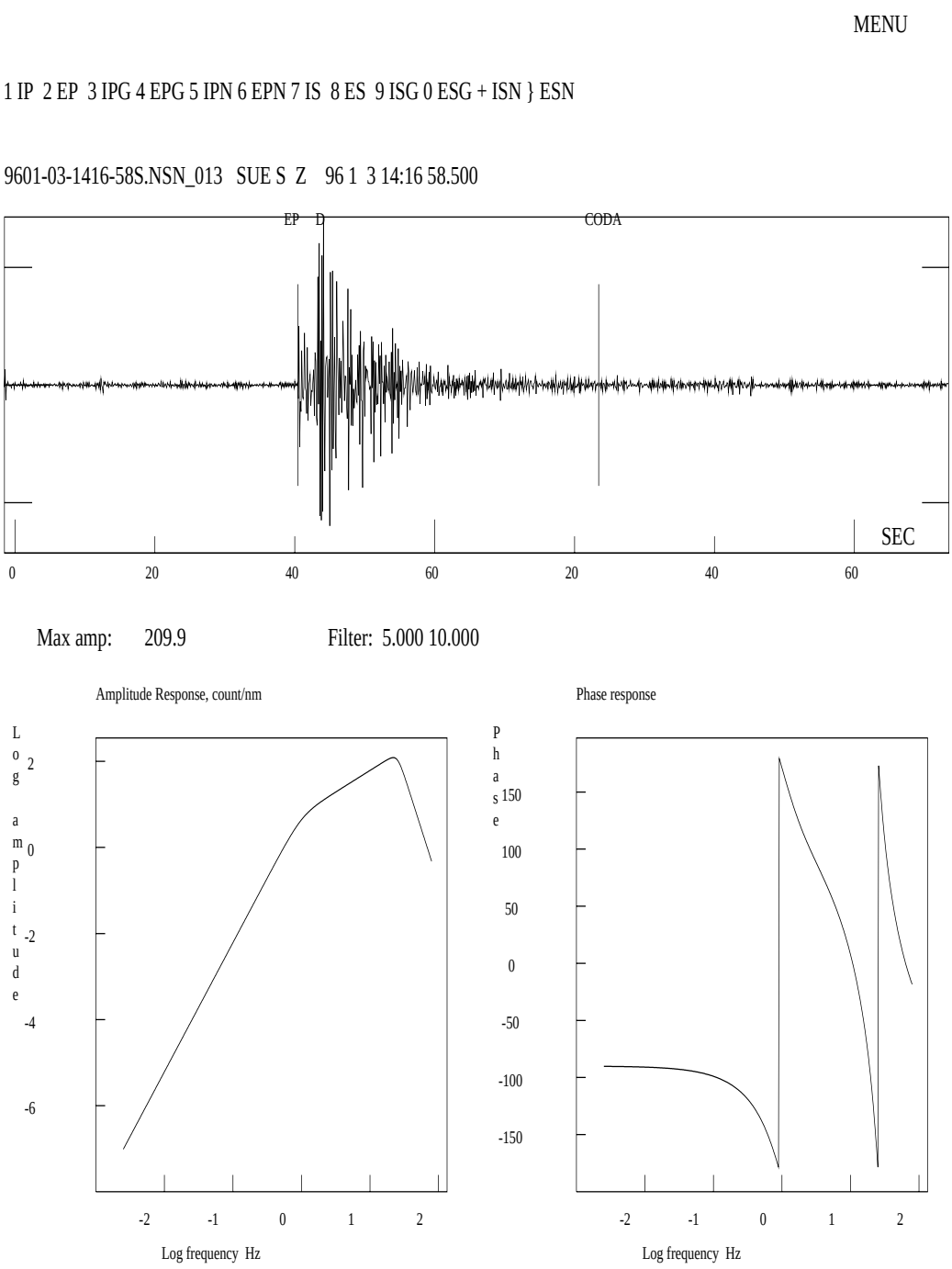


Figure 6.11: Plotting response curves
The figure shows the amplitude and phase response for station SUE, component S Z. The response is the one which will be used in analysis irrespective of whether it is taken from the file header or the CAL directory.

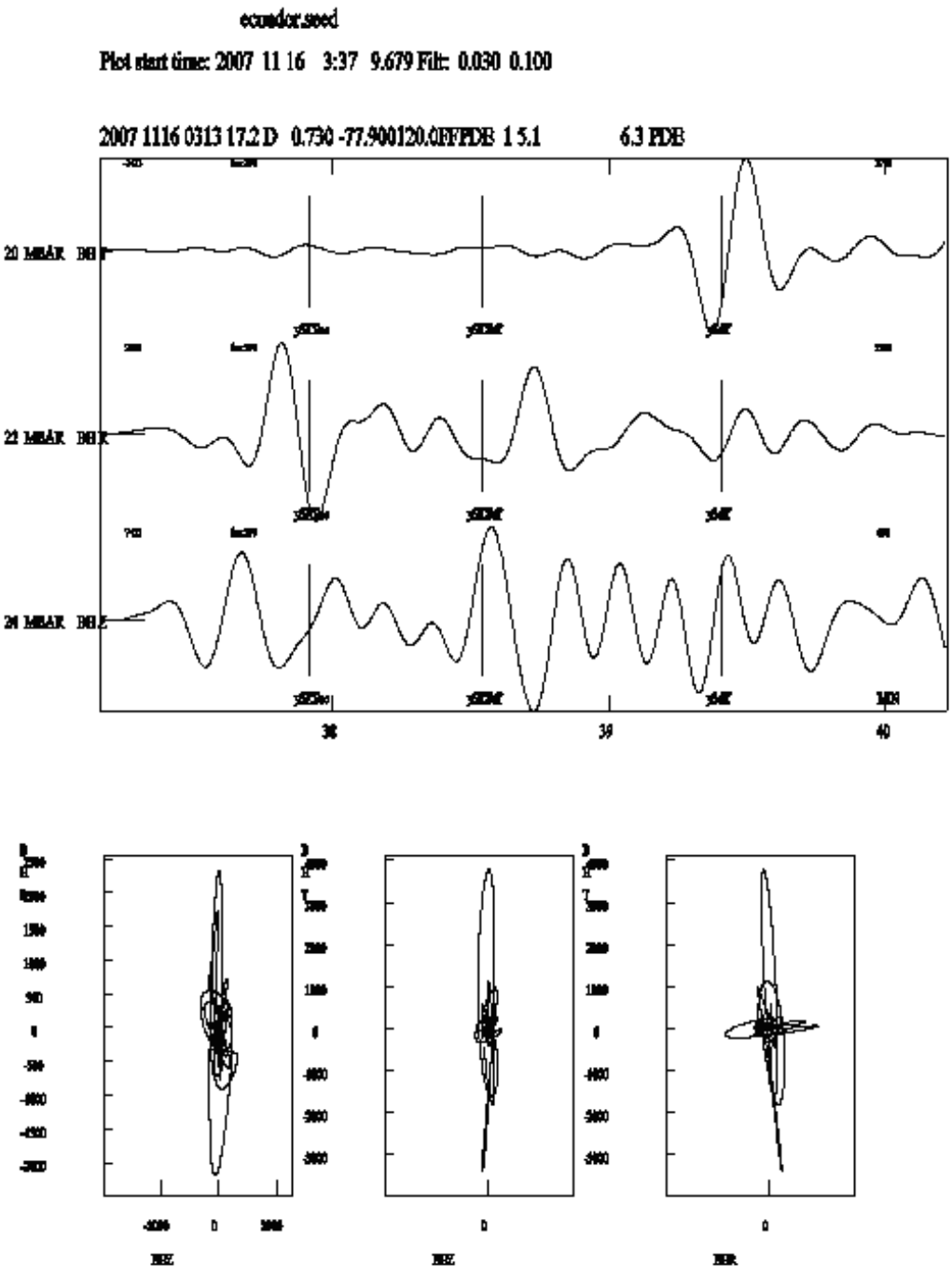


Figure 6.12: Example of particle motion plot.

```
AUTO_PROCESS       0,1,2       name      0.0      ls
SPECTRAL OUTPUT                          1.0
WOOD ANDERSON HIGH CUT                  20.0
### Set Filter type 0 for bndpas 1 for recfil routine
FILTER TYPE                             0.0
### Setup user-defined filters
FILTER 1           z-key                0.01     0.1.
FILTER 2           x-key                0.1      1.0
FILTER 3           v-key                1.0      5.0
FILTER 4           b-key                5.0      10.0
FILTER 5           n-key                10.0     15.0
FILTER 6           m-key                15.0     25.0
FILTER 7           ;-key                2.0      4.0
ML LOW CUT AND POLES
ML HIGH CUT AND POLES
MS LOW CUT AND POLES                     0.05      2.0
MS HIGH CUT AND POLES                    0.1       2.0
MB LOW CUT AND POLES                     0.5       2.0
MB HIGH CUT AND POLES                    4.0       2.0
BANDPASS FILTER                          5.0      10.0
CODA AUTO                                1.0
AUTOCODA FILTER                          0.1      10.0
AUTOCODA STA                             3.0
AUTOCODA RATIO                           1.2
DEFAULT CHANNEL                         KONO      LU Z
DEFAULT CHANNEL    Station and channel  ITK       S  Z

--------------------------------------------------------------------------------

In SEISAN.DEF, the following parameters can be set, format as in SEISAN.DEF
SPECTRAL GEO_DEPTHS                     10.0       14.0
HERKIJ_DISTANCE                         100.0
```

All parameters are within column 41 and 60 and each occupying up to 10 characters.

NOTE: If any of the phase or weight keys are redefined, all previous defaults disappear.

DEFAULT CHANNEL: All channels are default if not given. For routine display, it is useful to only select some channels.

PHASE NAME KEY: The keys associated with given phases. Remember that I, E or a blank MUST be part of the name so it is not possible to chose a name like "P", it must then be " P" (note the blank in front of P). About 10 phase combinations are currently default as seen on the pick display. If a new phase key is selected, you must define all the keys you want to use for phases including all the predefined phases. The combined onset/phase key can be up to 9 characters.

PHASE WEIGHT KEY: The defaults are upper case 1,2,... to 0 for weights 1,2,... to 0 . Again, choosing just one other key, and all must be redefined. The symbol must be in column 41 and the weight in column 51. The weight is an integer 0, 1,2,3, 4 or 9.

PHASE MOUSE KEY: The default is blank. Normally no redefinition is needed since the mouse character is defined in SEISAN. The key can be defined as a character or the ASCII code written as a

real number.

SPECTRAL P-VELOCITY: P-velocity in km/sec, default 6 km/sec

SPECTRAL S-VELOCITY: S-velocity in km/sec, default 3.5 km/sec Both above parameters must be set separately, the Vp/Vs in `STATION0.HYP` is not used to calculate one from the other. The values go into the S-file the first time spectra are calculated. if values are changed later in the `MULPLT.DEF` file, no change will be made in the S-file, old values remain.

SPECTRAL Q0: Q is defined as $q0 * f * *qalpha$, default 0 meaning no Q-correction

SPECTRAL QALPHA : See above, default 1.0, NOTE: Q is only used when doing spectral analysis and has no effect on the displacement seismograms.

SPECTRAL DENSITY: Density for spectral analysis (g/cm**3), default 3.5 g/cm**3

SPECTRAL KAPPA: Near surface attenuation, default 0.0 meaning no attenuation. For teleseismic events, this is t*.

SPECTRAL GEO_DEPTHS: Depth range where geometrical spreading changes from surface wave to body wave spreading, S-waves only. Default 50 and 100 km. This is only used if distance is larger than HERKIJ_DISTANCE. THIS PARAMETER IS NOT SET IN `MULPLT.DEF`, BUT IN `SEISAN.DEF`, MENTIONED HERE SINCE IT IS IMPORTANT FOR SPECTRA.

HERKIJ_DISTANCE: Epicentral distance at which geometrical spreading changes from body wave spreading to surface wave spreading, S-waves only. Default 100 km. THIS PARAMETER IS NOT SET IN `MULPLT.DEF`, BUT IN `SEISAN.DEF`, MENTIONED HERE SINCE IMPORTANT FOR SPECTRA

3COMPVELOCITY: Velocity used (km/sec) in 3 component azimuth analysis. Default is 5 km/sec.

CHANNEL SORTING: If set to 1.0, channels and filenames are sorted alphabetically, if 0.0, no sorting. Default = blank is sorting.

NCHAN PER SCREEN: The number of channels to be displayed per screen. Default = blank is 99 channels. It may conflict with **DEFAULT CHANNEL**.

NSORT_DISTANCE: If blank or zero, channels are plotted in the order as they appear in the waveform file or in alphabetical order if flag CHANNEL SORTING is set. If set to 1.0, the channels are plotted in distance order if a distance is given in S-file. If not plotted from EEV, 1.0 will indicate sorting in waveform file header time order. Default 0.

X_SCREEN_SIZE: Size of initial X-window in % of total screen. Default 90 %.

RESOLUTIONX and RESOLUTIONHC is the number of points plotted on the screen or laser printer respectively. If e.g. 1000 points are plotted, this means that the remaining points are skipped although some primitive smoothing is done. Choosing too few points can lead to funny looking seismograms with aliasing effects and using all points will slow down the plotting. Resolutionx is for the screen and resolutionhc for the hardcopy. NOTE: If using MULPLT mode where both screen and hardcopy is used, it is the hardcopy resolution, which is used for both. Default 1000 and 3000 respectively.

SPECTRAL F-BAND: Spectral range (Hz) used for spectral plots. Default values are 0.05 to 20.0 Hz.

AUTO_PROCESS: Immediately following registration, MULPLT can run any program specified here. Since the event name has been put into memory, the program can operate on the newly registered S-file. Parameter one has the options: 0: Do not auto process, 2: Ask the user if autoprocess, 3: Autoprocess without asking the user. Parameter 2 gives the name of the process to run. The name is limited to 10

characters. Default, no auto processing.

AUTO_LOCATE: Immediately following registration, MULPLT can locate the newly registered event and put the location into the database. Parameter one has the options: 0: Do not locate, 1: Ask the user if locate, 2: Locate without asking the user. Parameter 2: 0: Do not save in database, 1: Ask if saving in database, 2: Automatically save in database. Default, no auto locate.

SPECTRAL OUTPUT: If parameter set to 1, two output files are created for each signal spectrum. `com_spec.out` is the complex spectrum and `amp_spec.out` is the real spectrum. Default 0.0. In addition, the single trace zoom window is saved in signal.out.

FILTER: Change definition of filters 1 to 7. The settings affect both the shortcut keys and the menu boxes.

FILTER TYPE: Choice between two different filter routines that basically do the same. Use 0.0 for routine 'bndpas' (default) and 1.0 for 'recfil'. Bndpas is a recursive Butterworth bandpass filter routine. Recfil is an IIR filter routine (various filter types supported, but in SEISAN only Butterworth is used), which works as low-, high- or band-pass, however, gives problems for bandpass at low frequencies. Low- and high-pass when using recfil are given by setting either the upper or lower frequency limit to 0.0, respectively.

ML LOW CUT AND POLES and ML HIGH CUT AND POLES: Filter band for Wood Anderson additional filter. Default values are 1.25 Hz to 20 Hz and 8 poles.

BANDPASS FILTER: When using all defaults from EEV (option PO), a bandpass filter can be set. Default is no filter. The parameters are lowcut and highcut for parameter one and two respectively.

CODA AUTO: Enable automatic coda determination (YES or NO). Default is NO. If enabled, auto coda is read with c instead of C.

AUTOCODA FILTER : Filter band for automatic coda: Default 5 to 10 Hz. AUTOCODA STA : Auto coda short term average: Default 5.0 secs.

AUTOCODA RATIO : Autocoda ratio. Default 1.5.

6.2.15 Distance trace plot with GMT, TRACE_PLOT (Unix only)

TRACE_PLOT is a simple program to create a distance trace plot using GMT programs (Generic Mapping Tools, `http://gmt.soest.hawaii.edu/`). The axes of the plot are time and distance, and the traces are centered on the respective epicentral distance. The input to the program is a single event in Nordic format (S-file). From the S-file, the program reads the origin time, epicenter location and the names of the associated waveform files. TRACE_PLOT reads the waveform data and writes the x-y coordinates of the lines in the plot to a file that is then used as input to the GMT program psxy. The TRACE_PLOT program removes the DC from the data and as an option can apply a band-pass filter. The output of the program is a Postscript file (`trace_plot.ps`) and a batch file that can be modified and used to rerun the GMT programs (`trace_plot.bat`). The parameters are set in the `trace_plot.par` file, which can be located either in the DAT or in the working directory. An example is seen in Figure 25.

The parameters in `trace_plot.par` are:

FILTER: The pass-band filter limits can be specified through the FILTER parameter.
DISTANCE: The distance range (y-axis) for the plot.
TIME: The time range in seconds (x-axis).
AMPLITUDE_SCALE: The amplitudes are scaled for every trace individually, by [amplitude/(max am-

plitude) * AMPLITUDE_SCALE].
STATION_SFILE_ONLY: This variable can be set to 1.0 to only plot traces that are listed in the S-file, the default is 0., which plots all traces without checking if they are present in the S-file.
TIME_ORIGIN: In the current version, the origin of the time axis corresponds to the origin time of the event.
COMPONENT: This can be used to select components for plotting, in case no component is defined, TRACE_PLOT will show all vertical component traces.

Example of `trace_plot.par`:

```
KEYWORD.............Comments..............Par 1.....Par 2

FILTER              filter range          0.1       5.0
DISTANCE            dist range            0.        440.
TIME                time window           0.        250.
AMPLITUDE_SCALE     amplitude/max*scale   15.
STATION_SFILE_ONLY  0. if any station     0.
                    1. if station has
                       to be in s-file
TIME_ORIGIN         1.=origin time        1.
                    0.=file start time
COMPONENT                                 BH Z
COMPONENT                                 HH Z
```

6.3 Plotting epicenters

In SEISAN there are three different programs for plotting epicenters, EPIMAP, W_EMAP and GMAP. EPIMAP is the basic program for plotting epicenters in SEISAN and is the program that is called from EEV, when you type `map`. W_EMAP is a windows based program. GMAP is linked to the Google Earth and Google Map Internet mapping tool.

6.3.1 EPIMAP

The command for plotting epicenters is EPIMAP ¡file¿, where the optional file is a file with EPIMAP commands. If file is not given, the user will be prompted for the input. The program can plot land contours, epicenters, macroseismic intensities, stations and level contours as well as depth profiles. It is possible to zoom in on selected areas (option by **Mario Villagrán**). The program has been much revised by **Jim Bolton**.

Input files: Land contours and other contours

The program will look for all files ending with .MAP located in the DAT directory. The user can then choose any one or a combination of files. The users own contour files (e.g. faults) can be added to the DAT directory. A very detailed world map is available on the SEISAN CD and on the SEISAN web site. Areas can be selected out of these files with program SELMAP.

Stations

Epimap will look in `STATION0.HYP` for station coordinates. It will search first in the working directory, then in DAT.

Epicenters

The user will be prompted for epicenter input files. The format can be Nordic or Nordic compact. Magnitudes are plotted proportional with symbol size unless the ellipticity option is selected in which case the error ellipses are plotted (if smaller than 100 km). Fault plane solutions can optionally be plotted instead of error ellipses. The first fault plan esolution found in file will be used. Name of intensity files (SEISAN standard format, see Appendix A) are also entered here. The file name must have the 3 letters 'mac' after the '.' See also section 6.33.

Input files for EPIMAP can be made e.g. with the COLLECT command which collects S-files into one file or with the SELECT command selecting data from the database using several criteria. HYP also generates a CAT-file (`hyp.out`) which may be used as input to EPIMAP.

Macoseismic information

EPIMAP can plot SEISAN macroseismic observations, see section 6.33.

Magnitudes

The program will read all 3 magnitudes (magnitude1, magnitude2 and magnitude3) in the header line. It will use the first non-zero magnitude in the order magnitude1, magnitude3 and magnitude2. Epimap will search the first header line only. If it is desired to use a particular magnitude from any header line for plotting, use MAG program first to select particular magnitude type which is then placed in first header line magnitude position one. Program NORHEAD can move magnitudes from following header lines to the first line. Program REPORT can move magnitudes around on the header line.

A typical run is as follows, comment after !:

```
Projection menu
===============

   1: POLAR STEREOGRAPHIC     conformal, azimuthal
   2: ORTHOGRAPHIC            view from infinity, azimuthal
   3: MERCATOR                cylindrical, conformal
   4: LAMBERT EQUAL AREA      azimuthal
   5: GNOMONIC                Great Circles are straight lines
   6: AZIMUTHAL EQUIDISTANT   distance from origin is to SCALE$
   7: STEREOGRAPHIC           conformal, azimuthal
   8: EQUIDISTANT CYLINDRICAL
   9: OBLIQUE MERCATOR        cylindrical, conformal
  10: MOLLWEIDE ELLIPTICAL    pseudocylindrical, equal area
  11: SANSON'S SINUSOIDAL     pseudocylindrical, equal Area

Please enter projection number   :  3
Enter latitude  range of the map :  60 70    ! N  positive
Enter longitude range of the map :  0 30     ! E  positive

Center of geographical map space is ( 65.0,  15.0) degrees.
         ! can be used e.g. to make an accurate MERCATOR at high latitude
 Press <return> to accept these as the reference latitude
 longitude for the projection or <N>o to enter your own
co ordinates :
```

```
 Enter latitude of any grid line and also the grid spacing
:  60 2      ! possible to have grid spacing at any value
 Enter longitude of any grid line and also the grid spacing
:  0 4

DO YOU WANT THE EVENTS NUMERATED? (Y/N=RETURN)
  ! a sequential number will be plotted besides each hypocenter, the     !
corresponding hypocenters are found in output file epimap.out.

Plot title (max 60 chars), or press <return> for none:
Plot error ellipses: e or fps: f (n=return)?:       ! from HYP or fps programs
File name for contour levels, or press <return> for none: ! format below
Plot place names (P) or                             ! file format below
Plot all (a) or some (s) stations with a label, ! if s, question given about
                                ! which station, give in UPPER CASE all stations
without a label (X), or none <return>
...Enter in uppercase if you wish the symbols to be filled
and this facility is available...     ! only filled out in Postscript
:

Available colour index values are:
           1) Blue
           2) Green
           3) Red
           4) Yellow
           5) White
           6) Black (default)


 Enter epicenter filename and colour index,
 separated by a blank, otherwise press <return>
collect.out 2              ! plot first file green

 Input file is Nordic      ! full Nordic format of input file

 Enter epicenter filename and colour index,
 separated by a blank, otherwise press <return>
select.out  3              ! plot second file red

 Input file is Compact     ! compact file format of input file


 Enter epicenter filename and colour index,
 separated by a blank, otherwise press <return>

 Enter the following in uppercase if you wish the symbols
 to be filled and this facility is available...
 By default, symbols will be plotted according to
```

```
 magnitude, do you wish them to be plotted according
 to magnitude range ?

==== Loading Epicenters...

! now comes the plot, see below for options

 Earthquake locations inside the window are in epimap.out
 Coordinates of the surrounding area are in epimap.cor
 Area selected epicenters are in epimap.are
 Plot file is called epimap.eps
```

Interactive options:

When the plot is shown, there appears in the lower left-hand corner a menu of several options:

Q: Quit
P: Profile
A: Area
Z: Zoom

Press one of the letters to continue.

P: Profile

One or several depth section windows can now be selected with the cursor. First move the cursor to where the section shall start (from where distances are calculated), press any character to select point, move cursor to end of profile, press any character to select. A line between the two points is now plotted. Move the cursor to a point on the side of the line and press any character. A rectangle defined by the three points is now drawn, which defines the area used for the section. If more than one section is wanted (up to 9), press the number of sections instead. The selected number of profile boxes will now be plotted, all the same size. Pressing any character will draw the depth sections auto scaled, while PRESSING THE CHARACTER F, THE X AND Y SCALES ARE EQUAL and determined by the horizontal extension. When the first section appears, you can either press q to quit or any other character to plot next profile or, if the last profile, replot epicenter map and select new sections. IF YOU WANT ALL SECTIONS TO REMAIN IN PLOT FILE, QUIT AFTER PLOTTING THE LAST PROFILE. The plot file always stores what has been plotted so far, and is overwritten when a replot is made. It is also possible to plot a previously defined profile by entering O. The parameters are then taken from file profile.out. This file stores the last parameters selected by EPIMAP, but can also be edited by the user.

A: Area

Select, by clicking with the cursor, at least 3 points defining a polygon within which epicenters are selected. A new plot is made enclosing only the polygon and showing the epicenters within the polygon. The corresponding epicenters (S-files) are in file epimap.are. Known bug: Sometimes epicenters are still left outside, SELECT can be used instead.

Z: Zoom

Similar to Area, however a rectangle is selected by defining just the 2 diagonal corners.

MAP files The map files consist of blocks of coordinate pairs. Each block starts with the number of pairs in the block. The format of the header line is i4 and the following lines 10f8.3. Thus each block can at most have 9999 pairs.

Plotting place names If option P is used when the program asks for place names or station codes, the user will be prompted for one or several files with place names. The place name file format is:

name latitude_degrees longitude degrees

eg:

```
Edinburgh    55.94422  3.20096   or
Edinburgh   55.94422   3.20096   etc.
```

The only requirement is that at least 2 blanks separate the place name and the geographical co-ordinates. Note that the place name can contain one or more blanks, however each blank must occur singly. An example of a place name file is place_names.macro located in DAT. Epimap contour file EPIMAP has a simple contouring routine accepting a regular spaced grid. Below is an example (output from EQRSEI). The top part of the file is just comments, the data starts at "Fields to use". The data must come in longitude, latitude pairs (+ value of contour) in order as shown below. The contour value is plotted exactly as shown below. E. g. the value 117 is plotted as __117_____ where "_" is blank. By specifying ____117.0_, the value would be plotted as 117.0 and moved one space to the left on the plot. Currently only programs EQRSEI (version 7.0) and CRISEI from SEISAN version 6.0 make contour files. In the DAT directory, there is an example of an EQRSEI.OUT file

NB: In the input file shown below, the FIRST COLUMN MUST BE blank.

```
         Test Case 1.

 NSTEP =   10 JCALC =    0 JPRNT =    1 IATTE =    0

 LIST OF EXAMINED INTENSITIES     3.91    4.61    5.01    5.30    5.52    5.70

 RISKS DESIRED    .1000   .0200   .0100   .0050   .0000   .0000   .0000   .0000

 ATTENUATION DATA=   C1      C2      C3    SIGMA   RZERO    RONE     AAA     BBB
                   6.16     .64    1.30     .50   25.00    5.0010000.00     .00

 NO. OF GROSS SOURCES     3
 NO. OF SUBSOURCES IN GROSS SOURCES     3    2    1

 This file is input for epimap contour plotting. The input of
 parameters must appear as listed below with the actual data
 starting after the line Fields to use. There can be any
 number of header lines. The contour values plotted are
 plotted with the number of decimals given below. The last
 part of the file is the actual longitudes, latitudes and
 levels. Color use is optional and the field can be left
 blank, the codes are as follows:

 1: blue   2: green   3: red   4: yellow   5: white   6: black

 Min and max level  115.1800  129.6700
```

```
Fields to use                          I.........I.........I.........I
Latitude range and number of values         40.00     49.00       10
Longitude range and number of values         5.00     14.00       10
Contour level to plot and color           117           2.0
Contour level to plot and color           119           3.0
Contour level to plot and color           121
Contour level to plot and color           123
Contour level to plot and color           125
Contour level to plot and color           127
Contour level to plot and color           129
Contour level to plot and color           131
Contour level to plot and color           133
Contour level to plot and color           135
   5.00000    40.0000    117.620
   6.00000    40.0000    118.490
   7.00000    40.0000    119.080
   8.00000    40.0000    119.390
   9.00000    40.0000    119.390
  10.00000    40.0000    119.080
   11.0000    40.0000    118.490
   12.0000    40.0000    117.630
   13.0000    40.0000    116.510
   14.0000    40.0000    115.180
   5.00000    41.0000    119.680
   6.00000    41.0000    120.620
   7.00000    41.0000    121.260
   8.00000    41.0000    121.600
   etc
```

EPIMAP output files:

`epimap.out`: Gives a numbered list of all events within main window. This can be used in connection with the number option.

`epimap.cor` and `epimap.are`: If option A (selecting area) has been used, the coordinates of the corners will be given in `epimap.cor` and the complete events (S-files) within the selected area, in `epimap.are`.

`epimap.num`: A compact file of `epimap.out` with the numbers plotted.

`epimap.eps`: Postscript plot file of epicenters and possible profiles. If only one profile has been selected, all is on one page. If several profiles are selected, there will be two profiles per page up to a maximum of 6 pages (one with map and 5 with profiles).

`epimap.inp`: This file is storing all input parameters of the run and can be used to run epimap again without entering any parameters. The file can be edited if a run has to be repeated with e.g. a new epicenter file. The file can have any name so several predefined plot definitions can be stored and thereby automate map production.

`profile.out`: The file stores the parameters used with the profiles. The file is overwritten for each new profile parameter selection. An example is:

```
  60.93583   7.21519  63.29655   1.36709  63.39875   5.01266
```

```
27.8
   3
```

The first line gives latitude and longitude of the 3 points used for selecting profile (see explanation for interactive section), next line the azimuth calculated for the profile and the last line gives the number of profiles. The file can be used to repeat the same profile as in an earlier run or to predefine a more exact profile than can be selected with the cursor.

`profile.num`: Output of distance and depth of the profile in km. Distance is only correct in unzoomed plots.

Problems: Known bug: When selecting events with polygon, sometimes some events remain outside

Figure 29 and Figure 30 shows examples of plots made with EPIMAP.

6.3.2 W_EMAP, Version Windows based map program

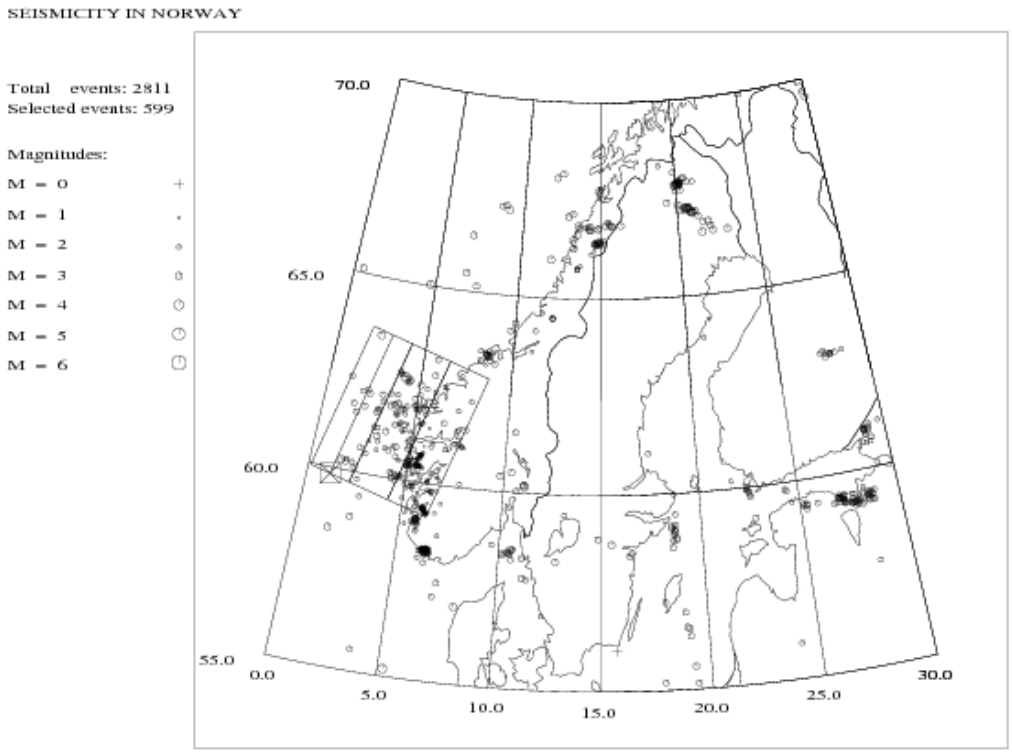
Program and documentation by **Fernando Carrilho**, fernando.carrilho@meteo.ptu
Program must be installed in addition to SEISAN.

This program was developed to be used on seismic routine processing. Its main features are the capability of allowing visualization of epicentre locations, seismic stations, error ellipses, coastlines, macroseismic data, focal mechanisms (one or many) and simplified tectonics. From the previous public version (4.1), some bugs were corrected and new features added. In particular: cartographic deformation is taken into account in error ellipses and station-epicentral path draws; some bugs on printing were corrected; compacted seisan files can be used within 'additional events' representation; travel time curves can now be displayed; simplified relief can be displayed (if available as a MAP file); more than one file can be used for each category layer (coastlines, tectonic, relief and places names).

The program can be integrated within the SEISAN environment, since it uses SEISAN parameter files, macrosseismic files, MAP files and station/model files.

If the program is called from EEV or from the command line as W_EMAP, then it displays information contained in `hyp.out` file, generated by the HYPOCENT [Lienert, 1994] location program, included in SEISAN, in the settable working directory.
During the first run, user is driven to edit the configuration file w_emap.def that is created in the users personal directory (SEISAN_TOP/DAT/users/¡username¿), where most of the program parameters can be changed.
The program can automatically detect changes in the `hyp.out` file so the user doesn't need to restart the program each time the epicentre changes.
The program can also display epicentres contained in any SEISAN parameter file, where the user may choose between one single epicentre and all epicentres at the same time. Double clicking the right mouse button will change the active epicentre to the one picked.

Multi-user individual configurations (color schemes, additional event files, tectonic files, coastline files, relief files, cartographic projections, etc.) are supported. The program also has an option for Google Earth and Google map.. Installation All the files are included in the distribution file w_emap.exe. To install it, you just execute this install script and make sure W_EMAP is installed under _SEISMO_TOP (Usually
seismo). The manual will only be found in INF after installation. The current version of W_EMAP is 4.8.2 but the manual has not been updated since version 4.6. An important new option is be able to directly plot a file with many hypocenters with the command wemap filename. Known problem: Thre



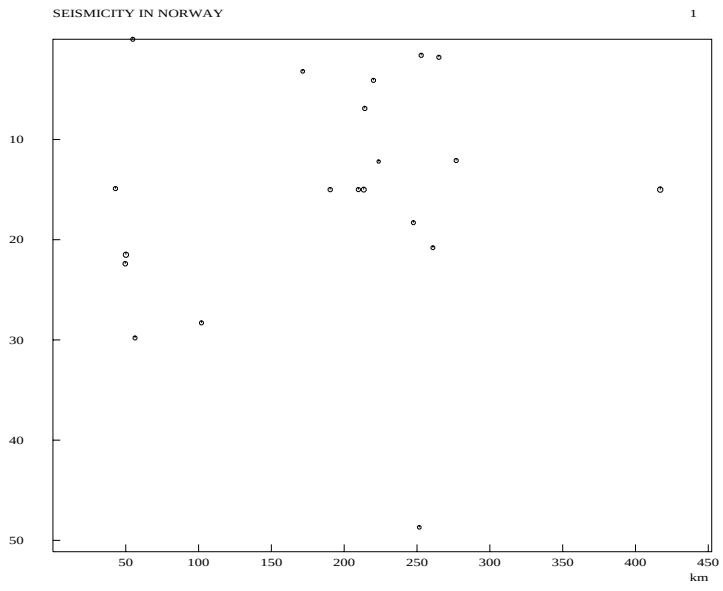


Figure 6.13: An example of using EPIMAP. The top shows epicenters plotted and the bottom the first of a series of profiles. The frames on the top plot show the location of the profiles.

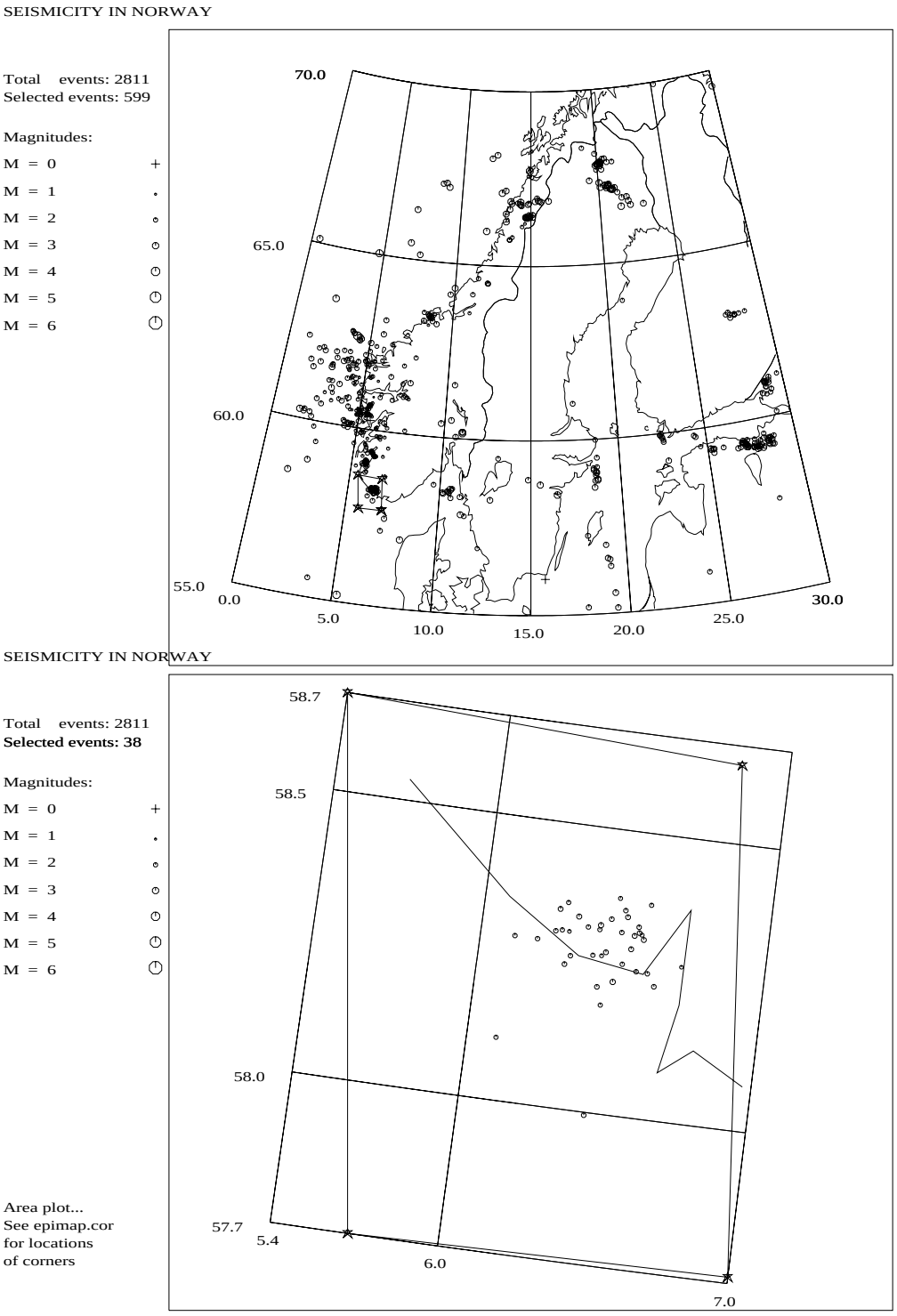


Figure 6.14: An example of using EPIMAP with area selection. The top plot shows where the area is selected, while the bottom plot shows the selected area.

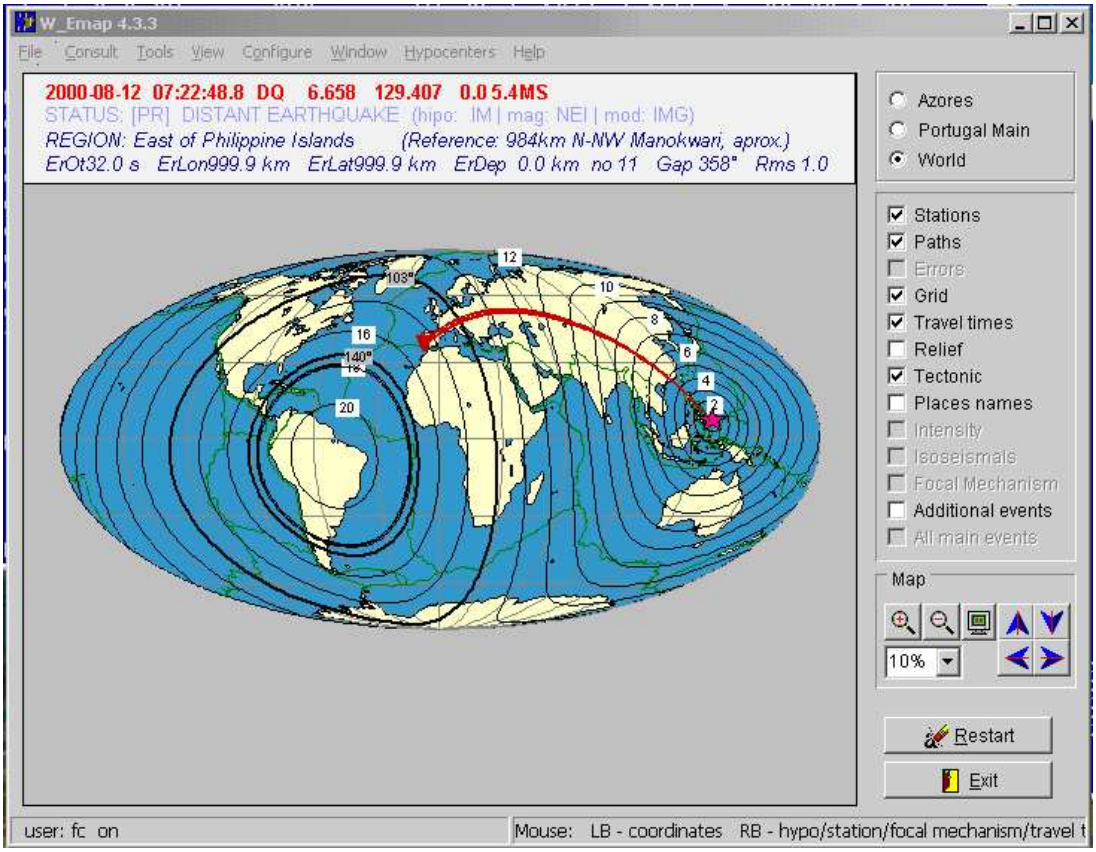


Figure 6.15: Example for plot using w_emap (example not latest version).

must be a type 7 line in S-file in order to be able to plot a focal mechanism. Some versions of w_emap plot some of the fault plane solutions with inverted colors.

6.3.3 GMAP, Plotting epicentres with Google maps or Google Earth

Google maps and Google Earth seem to quickly establish themselves as commonly used mapping tools. GMAP provides the conversion from Nordic data to the input format required by these systems. GMAP also convert SEISAN station and polygon files. , which is the Keyhole Markup Language (KML). The input format of Google Earth is described on `http://earth.google.com/kml/`. GMAP required Google Earth installed on your system to plot the output file. Download Google Earth here `http://earth.google.com/download-earth.html` (note the terms and conditions on `http://pack.google.com/intl/en/eula_print_us.html`).

GMAP can run in two modes:

The simple GMAP:
Type gmap in eev, a file `gmap.html` is created and copied to you GMAP_DIR directory. When you open the `gmap.html` with your browser, you will be redirected to Google Maps and a green arrow will show the epicentre. The following parameters in `SEISAN.DEF` are used: GMAP_DIR /home/seismo/www
GMAP_TYPE MAP [MAP, SATELLITE, HYBRID or TERRAIN]
GMAP_TYPE determines which type of map Google MAPS will use, you can choose between: MAP, SATELLITE, HYBRID and TERRAIN.

The advanced GMAP:

- Type `gmap` on the command line to start gmap.
- Type `gmap -help` to get help.
- Type `gmap -stat` to convert SEISAN station files.
- Type `gmap -poly` to convert SEISAN polygon files.

In `gmap` you can:

1. Set the colour of the icons.
2. Show events as a animation over time.
3. Set the scale of the icons (default: scale=0.2*mag**0.5).
4. Set the type of icon used for earthquakes, explosions, probable explosions and for other events.
5. Include or exclude the S-file.
6. Rename the output file.
7. Append text in KML format to the output file (See `SEISAN.DEF` parameters below).

Example:

1. Use select to grap data from your Seisan database.

2. Run the GMAP program:

```
unix:/home/seismo/WOR: gmap
 INPUT FILE NAME
select.out
Title:
West Greenland [2000;2008]

Number of Earthquakes : 945 Explosions : 0 Probable Explosions : 2 Other events : 2
Output file is
gmap.kml
```

3. In Google Earth open the output file gmap.kml. See Fig. 6.16.

To make an animation for events over time use the -timespan flag. As an example explosions in the south of Norway from 1983 to 2007 can be downloaded here: `http://seis.geus.net/ber-exp.kml` Press the play button at the time slider at the top of the Google Earth. Use the ruler to control how the animation is displayed (speed, days shown, etc.).

GMAP parameters added to `SEISAN.DEF`: Icon used for earthquake, explosion, probable explosion and other events:

```
GMAP_ICON_QUAKE                 http://maps.google.com/mapfiles/kml/pal2/icon26.png
GMAP_ICON_EXPLOSION             http://maps.google.com/mapfiles/kml/shapes/star.png
GMAP_ICON_PROB_EXPL             http://maps.google.com/mapfiles/kml/shapes/open-diamond.png
GMAP_ICON_OTHER_EVENTS          http://maps.google.com/mapfiles/kml/shapes/square.png
```

Events with magnitude smaller than GMAP_ICON_MSIZE will be plottet with size of GMAP_ICON_MSIZE:

```
GMAP_ICON_MSIZE                 0.5
```

The scale of the earthquake icons is give by GMAP_ICON_XSIZE*magnitude**GMAP_ICON_YSIZE:

```
GMAP_ICON_XSIZE                 0.2
GMAP_ICON_YSIZE                 0.5
```

The scale of other events is furthermore multiplied by 2 Text can be added to the KML file, see this example, note the text must be placed from character no 41 to no 120:

```
#GMAP_APPEND_KML                 <!-- lines to be appended to the gmap.kml file : -->
GMAP_APPEND_KML                 <ScreenOverlay id="LOGO">
GMAP_APPEND_KML                 <name>Info and links</name>
GMAP_APPEND_KML                 <description> <![CDATA[
GMAP_APPEND_KML                 Data is in Nordic format. The format is <br>
GMAP_APPEND_KML                 described in the Seisan manual at UIB.<br>
GMAP_APPEND_KML                 UIB:  http://www.geo.uib.no/seismo/<br> ]]>
GMAP_APPEND_KML                 </description>
GMAP_APPEND_KML                 <Icon>
GMAP_APPEND_KML                  <href>http://seis.geus.net/geus.png</href>
```

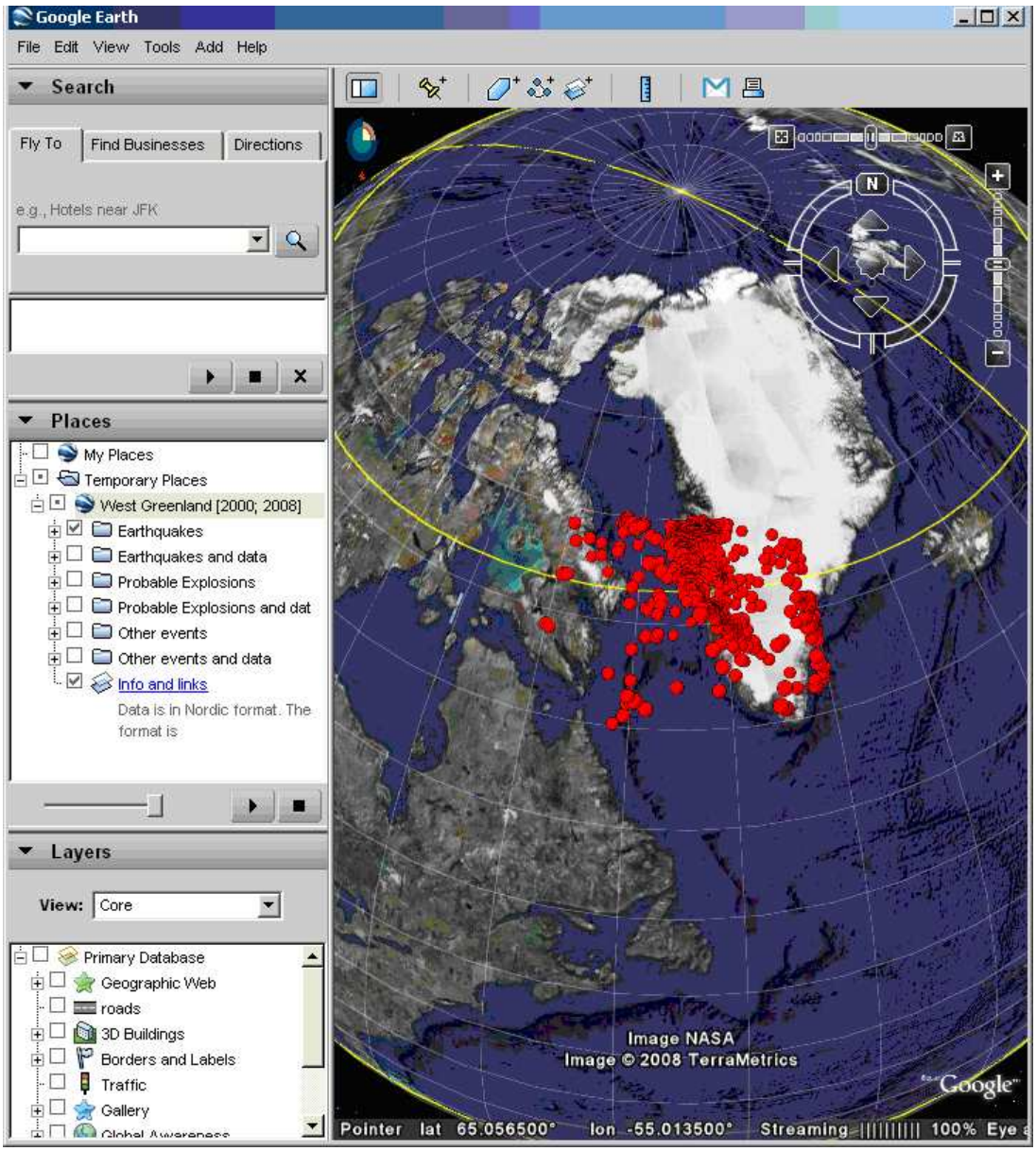


Figure 6.16: Example of mapping with gmap. Events in West Greenland. Note the folder and the subfolders in the `Places` window.

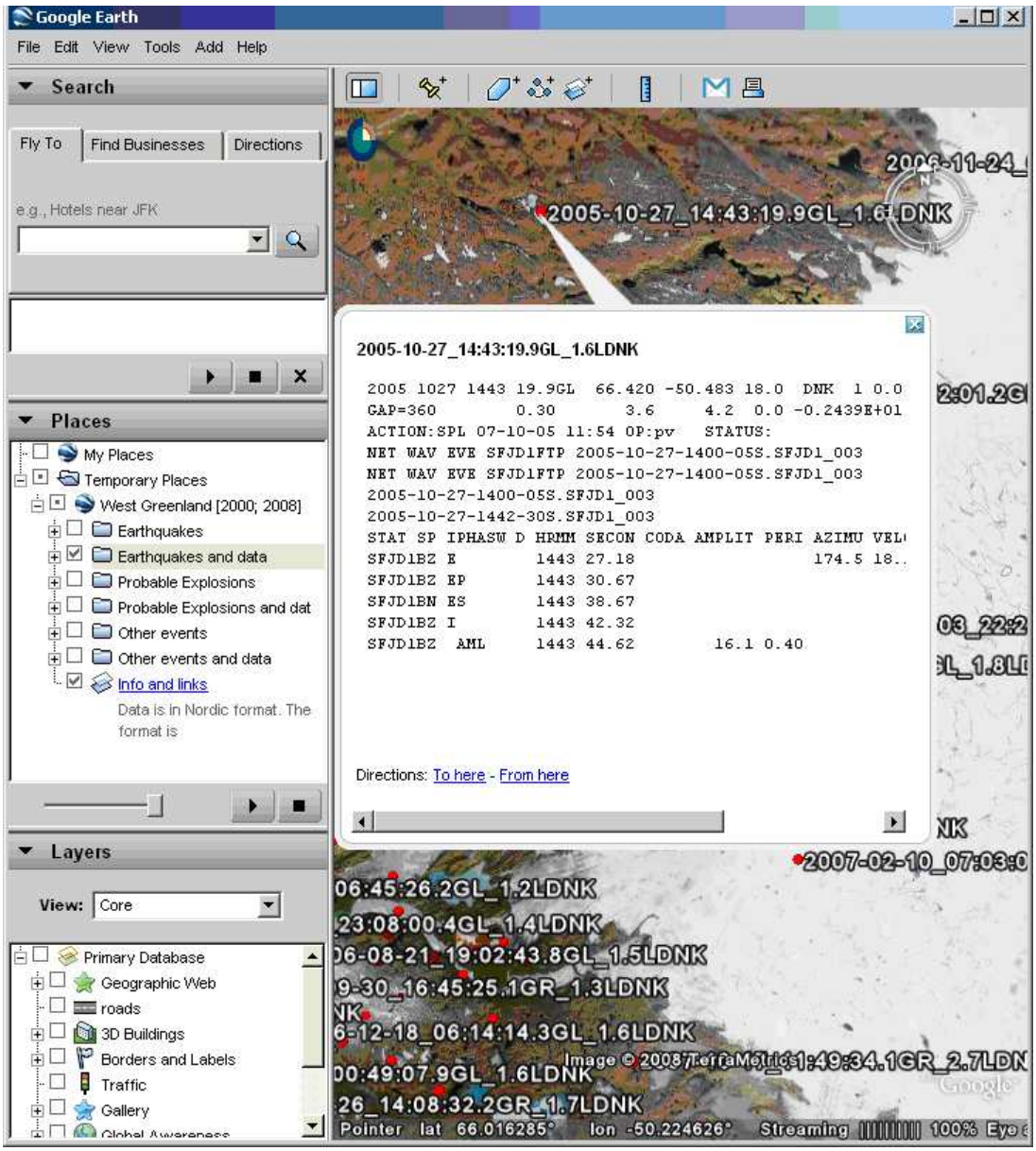


Figure 6.17: If the `Earthquakes and data` folder is selected in the `Places` window S-files can be shown.

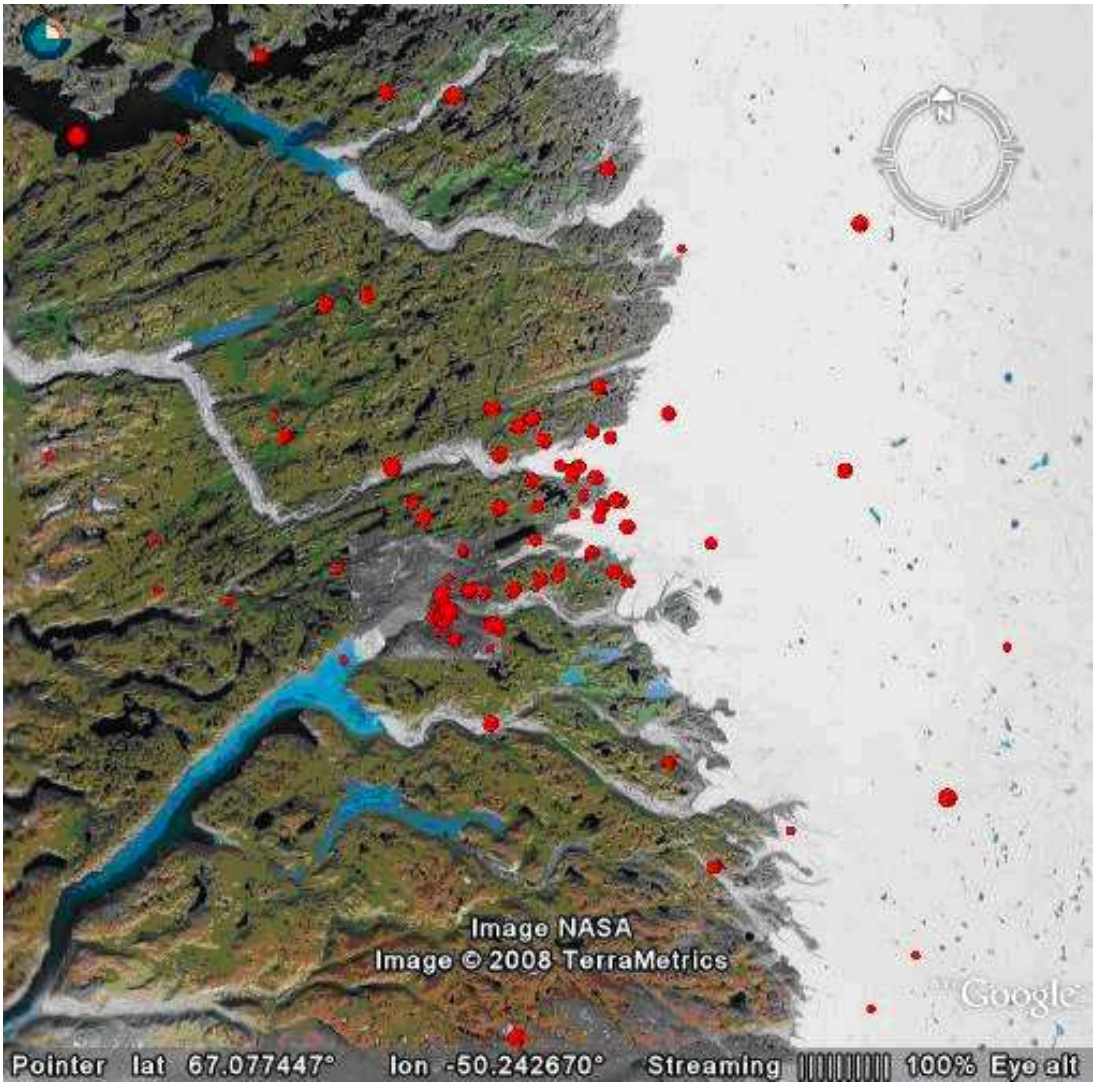


Figure 6.18: A map can be saved as raster file (File-Save-Save Image).

```
GMAP_APPEND_KML                         </Icon>
GMAP_APPEND_KML                         <overlayXY x="0" y="1" xunits="fraction" yunits="fraction"/>
GMAP_APPEND_KML                         <screenXY x="-0.01" y="0.99" xunits="fraction" yunits="fraction"/>
GMAP_APPEND_KML                         <rotationXY x="0" y="0" xunits="fraction" yunits="fraction"/>
GMAP_APPEND_KML                         <size x="0.1" y="0.1" xunits="fraction" yunits="fraction"/>
GMAP_APPEND_KML                         </ScreenOverlay>
```

GMAP -help:

```
unix:/home/seismo/WOR$ gmap -help

  The GMAP program converts Nordic format to the KML format.
  The output file gmap.kml can be opened with Google Earth.
  The program prompts for a input file in nordic format, the
  input file can be compact.

  Usage: gmap [options]
         input file
         Title used for kml folder

  ## Options ##
    -help        Print this list
    -h           Same as -help
    -color       Define color of epicenters [blue/green/yellow/
                 black/white] Default color is red
                 To uses other colors see describtion below
    -timespan    Events gets timetag scroll in time domain
    -nodata      kml file will only contain header infomation
    -out_file    Define name of output file (default is gmap.kml)
    -verbose     Be more verbose
    -version     Seisan version

  Scale:
  The scale of the icons is set by the SEISAN.DEF
  parameters GMAP_ICON_MSIZE, GMAP_ICON_XSIZE and
  GMAP_ICON_YSIZE, see the manual for details.

  Color:
  Color and opacity (alpha) values are expressed in
  hexadecimal notation. The range of values for any
  one color is 0 to 255 (00 to ff). For alpha, 00 is
  fully transparent and ff is fully opaque. The order
  of expression is aabbggrr, where aa=alpha (00 to ff);
  bb=blue (00 to ff); gg=green (00 to ff);
  rr=red (00 to ff). For example, if you want to apply
  a blue color with 50 percent opacity to an overlay,
  you would specify the following:
  <color>7fff0000</color>, where alpha=0x7f, blue=0xff,
  green=0x00, and red=0x00. See also:
  http://code.google.com/apis/kml/
  documentation/kml_tags_21.html#color

  Examples:
            gmap -color blue -nodata
```

```
gmap -timespan -color 7eee00ee
echo "collect.out\nDK events\n" | gmap -out_file dk.kml
```

6.3.4 SEIS2VIEWER, Plotting hypocentres in 3D

The program is written by **Ruben Soares Luís** (ruben.so.luis@gmail.com) and use the SeismicityViewer50 program by **Anthony Lomax**. To program has been aliased to smap.

Overview

seis2viewer is a wrapper for the application SeismicityViewer, developed by Anthony Lomax for rapid mapping of seismic events. SeismicityViewer displays hypocenter locations in 3D as well as station locations and P/S residuals. Geographic and geologic features can also be displayed as well as focal mechanisms. Please find details here:
`http://alomax.free.fr/seismicity/`

seis2viewer has been created to take information from a nordic file from Seisan and convert it in a format usable by SeismicityViewer 5.0 (NLLoc Format). It automatically launches Seismicity Viewer, generating a set of files required for its operation.

In its current version, seis2viewer allows the visualization of hypocenters and magnitudes in SeismicityViewer 5.0. Other information, such as station locations, P and S residuals, etc. is not displayed.

Installation and Configuration

1: **Configuration Files:**

- `seis2viewer.def` (mandatory): seis2viewer uses a **single** configuration file: `seis2viewer.def`, which may contain references to other files containing information on maps, elevation, localities, etc. The file `seis2viewer.def` is actually passed directly to Seismicity Viewer adding only a reference to an automatically created file. This operation is transparent to the user. As such, the configurations can be obtained from the Seismicity Viewer 5.0 website: `http://alomax.free.fr/seismicity/`
- Other files: `seis2viewer.def` may reference other files containing map information, elevation data, locality names, etc. These files should follow the definitions presented in the Seismicity Viewer 5.0 website.

2: **Installation**

seis2viewer is distributed as a single .jar file (`seis2viewer.jar`), containing all the necessary classes to work, including Seismicity Viewer 5.0. This file should be placed in the PRO directory of seisan. To facilitate the usage of seis2viewer, the administrator should create an executable script file to call seis2viewer. As an example, the executable script should contain the following line:

```
java -jar $SEISAN_TOP/PRO/seis2viewer.jar $1        (for unix/linux)
java -jar %SEISAN_TOP%/PRO/seis2viewer.jar %1%      (for windows)
```

The configuration file, `seis2viewer.def`, and any other required files should be placed in the DAT directory of seisan. Alternatively, the user may have its configurations on the working directory.

2: **Automatically Generated Files** seis2viewer generates a set of files to be used by Seismicity Viewer 5.0. Although these files are, in principle, irrelevant to the user, it is nevertheless important to mention them for reference purposes.

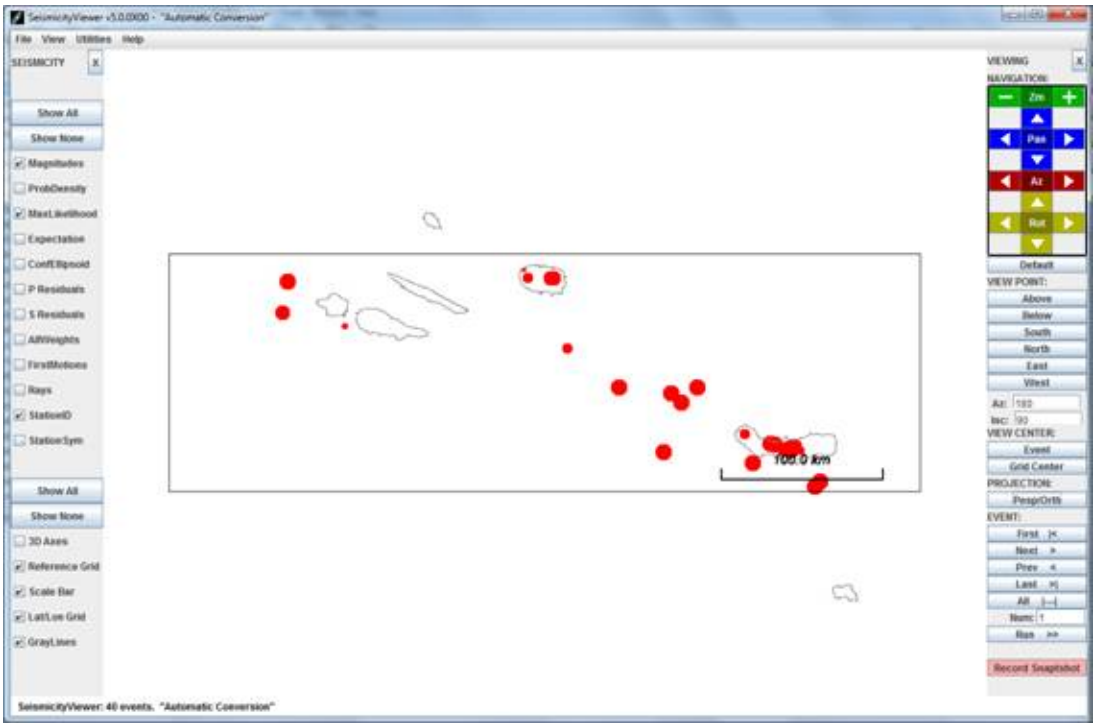


Figure 6.19: Epicenter map by SEIS2VIEWER.

- `seismicitydefaults`: This is the default configuration file for Seismicity Viewer. It contains a reference to the converted events file, `seis2viewer.hyp`.
- `seis2viewer.hyp`: This file contains a translation of the event file selected by the user for visualization in NonLinLoc format, appropriate for Seismicity Viewer.
- `seis2viewer.hdr`: This is an automatically generated grid file for the area surrounding the events to be visualized. It is only created if the maximum distance in longitude or latitude between events is less than 20 degrees.

Usage

seis2viewer can be used directly with a nordic file containing one or more events (e.g. `select.out`) as:

```
java -jar %SEISAN_TOP%/PRO/seis2viewer.jar select.out     (windows)
java -jar $SEISAN_TOP/PRO/seis2viewer.jar select.out      (unix/linux)
```

or, if the executable script has been created, as (assuming the script is called '`smap`')
`smap select.out`
seis2viewer can also be called directly from eev, as an external program, using the prefix 'o' as:
`osmap`
In this case, smap will try to find the file eev.cur.out, which is automatically generated by eev and contains a reference to the event that is currently under work.

6.4 Searching in the database, SELECT

Whenever selective search and extraction is wanted SELECT is used. The program can run on the CAT database, single CAT files (Nordic or Nordic compact) or the S-file data base. The output file, `select.out`, will also be in Nordic format. Since the input CAT database can contain both normal and compact files, the output will always be a normal file with blank lines between events. If however the input is one compact file, the output will also be a compact file. Note: If SELECT is used on the CAT database (normal operation), you need to UPDATE your S-file database in order to transfer changes from the S-files to the CAT database. Select can work with input in 3 different ways:

1. The user is asked for selections
2. The selection parameters are in a file
3. Parameters are given on the prompt line

The program is started by typing SELECT (parameters from screen), SELECT 'input file' (parameters from input file) or SELECT -options. A typical user interactive run is shown below. Comments following !

```
POSSIBLE INPUT IS:
     STANDARD CAT DATABASE: RETURN
     ALTERNATIVE DATABASE, GIVE 1-5 LETTER CODE:
     FILENAME FOR ONE FILE, MUST BE 6 OR MORE CHARACTERS:
     Type of base: CAT (Return) or Sfiles (s):
 Updating database TEST_              ! standard base
 The database TEST_ has  10 files
 The first file starts: 199309
 The  last file starts: 199909

  Start time (blank is 1980), yyyymmddhhmmss: 199309! time range
  End time, enter for 2015                  : 199607

             PARAMETERS

      1   - Fault Plane Solution
      2   - Earthquake Felt
      3   - Magnitude Type(s)
      4   - Distance ID(s)
      5   - Event ID(s)
      6   - Magnitude Limits
      7   - Latitude Limits
      8   - Longitude Limits
      9   - Depth Limits
      10  - RMS Limits
      11  - Number of Stations Limits
      12  - Hypocenter Errors Latitude Limits
      13  - Hypocenter Errors Longitude Limits
      14  - Hypocenter Errors Depth Limits
      15  - Minimum Number of Polarities
```

```
     16  - Hypocenter Agencies
     17  - Magnitude Agencies
     18  - Station Codes, components and distance range
     19  - Polygon
     20  - Use all header lines
     21  - Look for wave form file names
     22  - Gap range
     23  - Phases
     24  - Volcanic subclasses


 SELECT NUMBER TO CHANGE PARAMETER, RETURN TO SEARCH: 6
 Minimum Magnitude, return for default:   5
 Maximum Magnitude, return for default:   7

            PARAMETERS

     1   - Fault Plane Solution
     2   - Earthquake Felt
     3   - Magnitude Type(s)
     4   - Distance ID(s)
     5   - Event ID(s)
     6   - Magnitude Limits                          5.0        7.0
     7   - Latitude Limits
     8   - Longitude Limits
     9   - Depth Limits
     10  - RMS Limits
     11  - Number of Stations Limits
     12  - Hypocenter Errors Latitude Limits
     13  - Hypocenter Errors Longitude Limits
     14  - Hypocenter Errors Depth Limits
     15  - Minimum Number of Polarities
     16  - Hypocenter Agencies
     17  - Magnitude Agencies
     18  - Station Codes, components and distance range
     19  - Polygon
     20  - Use all header lines
     21  - Look for wave form file names
     22  - Gap range
     23  - Phases
     24  - Volcanic subclasses

 Ok. Input realized successfully.
 SELECT NUMBER TO CHANGE PARAMETER, RETURN TO SEARCH:
199309.CAT  No of events:     1  Selected:     1  Selected total:     1
199406.CAT  No of events:     1  Selected:     1  Selected total:     2
199410.CAT  No of events:     1  Selected:     0  Selected total:     2
199508.CAT  No of events:     1  Selected:     0  Selected total:     2
199511.CAT  No of events:     1  Selected:     0  Selected total:     2
```

```
199606.CAT  No of events:     6  Selected:     3  Selected total:     5
199607.CAT  No of events:     5  Selected:     1  Selected total:     6

 TOTAL NUMBER OF EVENTS IN TIME INTERVAL   16
 NUMBER OF DISTANT EVENTS   - - - - -      7
 NUMBER OF REGIONAL EVENTS  - - - - -      0
 NUMBER OF LOCAL EVENTS     - - - - -      9
 ------------------------------------------------
 NUMBER OF EVENTS SELECTED ************   6
 NUMBER OF WAVEFORM FILES SELECTED         9
 NUMBER OF INDEXES SELECTED                6
 SELECTED EARTHQUAKES ARE IN FILE: select.out
 LOCAL INDEX FILE IN:              index.out
 NAMES FOR WAVEFORM FILES IN FILE: waveform_names.out
 SELECT COMMANDS IN FILE:          select.inp
```

Note above, that the second time the menu is shown, the choice of magnitude limits is shown. For each CAT file in the catalog, the number of events in file, number of events selected from that file and the accumulated number are listed. The last file might not show the correct number of events in file since SELECT might stop before reading the whole file if the end time is in the middle of the file. If start time is blank, 1980 is used. The end time can also be blank, and 2015 is used. This option is useful when selection on whole data base or whole file. Input parameters:

In the input database (or file) a time window must always be given. If no more selection is done, all data in time window is selected. Further selection can be done by choosing a number and giving parameters. The chosen parameters are then shown on the next parameter selection menu as shown above for magnitude. Parameters can be reentered. Parameters not entered will have no influence in the selection. If several parameters (numbered selections below) are entered, conditions for all must be true for the event to be selected. Within each numbered selection, usually only one of the entered conditions must be fulfilled for the event to be selected. If e.g. Ml and Mb are selected, events, which have either magnitude, will be selected. When no more parameters are desired, press enter.

1. - Fault Plane Solution
   Selects events with a fault plane solution (F- line in S-file). There will also be the question: "Give quality, e.g. A or ABC, enter for all", in this way different qualities can be selected.

2. - Earthquake Felt
   Events felt indicated by a type 2 line

3. - Magnitude Type(s)
   Normally, all magnitudes for one event are searched to see if any magnitude fits the selection criteria. With option 3 it is possible to use one or a combination of magnitude types e.g. L and B. If magnitudes without type are to be selected, use underscore "_" for magnitude type. If there is no magnitude in the first magnitude position, chose "N" for one of the magnitude types to be able to select the other 2 magnitudes on the line. Magnitude types are: C: Coda magnitude, L: Local magnitude, b: mb, B: mB, s: Ms, S: MB and W: Moment magnitude. N: Find events with no magnitude in first position. An event is selected if any one of the types of magntudes are found. Magnitudes are only searched on first header line unless "Use all header lines is set".

4. - Distance ID(s)
   Restricting the search to be for one or a combination of the distance id's L, R and D.

5. - Event ID(s)
   Restricting the search to one or a combination of event id's, e.g. E and V for explosion and volcanic events. The letters used for selection are not limited to the examples shown above, they are however the ones used currently. It is thus e.g. possibly to label events as X for unknown type (column 23 in header line) and then later on select out all those events by specifying X for event ID. For the 3 questions about types, up to 5 letters can be used. The currently used codes are: E: Explosion, P: Probable explosion, V: Volcanic, S: Sonic boom, Q: Earthquakes which is equivalent to blank for type. However, if blank is selected, all event types are selected, while if Q is used as input, only events with no ID or Q ID are selected. So if all earthquakes and volcanic event are to be selected, use QV. Without the Q, only volcanic events are selected. Selection is made if either one of criteria is met.

6. - Magnitude Limits
   Range of magnitudes to select. Note that if no magnitude type is given, the extreme of all magnitude types reported is used. Magnitudes are only searched on first header line unless Üse all header lines is set:.

7. - Latitude Limits
   Range of latitude. NOTE: If no latitude or longitude values are chosen, SELECT will include an event even when it is not located if the remaining criteria are OK. If it is required that only located events are searched for, enter at least one value like an upper latitude limit of 95.

8. - Longitude Limits
   Range of longitude.

9. - Depth Limits
   Range of depths.

10. - RMS Limits
    Range of rms travel time residuals.

11. - Number of Stations Limits
    Range of number of stations.

12. - Hypocenter Errors Latitude Limits
    Range of hypocenter latitude errors. Works only if error line (E-type) is present in S-file. Currently error lines are generated by HYP and the ISC conversion program ISCNOR. There should only be one error line in file associated with the prime solution in first header line. However, if more than one error line is present, all are checked and if one fulfills the selection criteria, the event can be selected.

13. - Hypocenter Errors Longitude Limits, See 12.

14. - Hypocenter Errors Depth Limits, See 12.

15. - Minimum Number of Polarities, only P-phases are used
    Counts all polarities, useful to find potential events for fault plane solutions.

16. - Hypocenter Agencies
    Selects events only with given hypocenter agencies as indicated on header line.

17. - Magnitude Agencies
    Select only events with given magnitude agencies as indicated on header line. Magnitudes are only searched on first header line unless Üse all header lines is set.

18. - Station Codes, components, distance range and phase
    Selects only events with given stations, component, distance range and phase. A formatted help line comes up for selecting items. Any one or a combination can be selected, however, a station code or component code must be selected. The distance can be hypocentral or epicentral. Distances are integers right justified.

19. - Polygon
    Selects events within a given polygon of at least 3 latitude-longitude pairs.

20. - Use all header lines
    All header lines are searched for relevant information

21. - Look for waveform file names
    Search the database for particular waveform files, input can use a fraction of file name or * for any name. No wildcards can be used in the string so e.g. ASK* will select all due to the *. Use just ASK in this case to select all filenames with the string ASK.

22. - Gap range
    The range of gap as given on the E-line (normally 2. header line). Only hypocenters calculated with SEISAN version 7.0 have gap.

23. - Phase
    Look for events with particular phases. Up to 6, 4 character phase names can be selected. The event is selected if at least one of the phases is present for the event. For a more selective selection based on phase, see option 18.

24. - Volcanic subclasses
    Search for events of given subclasses given by up to 10 codes. Any code can be given, however, normally they will be as defined in VOLCANO.DEF. The program searches for lines starting with 'VOLC MAIN'.

Historical data: When working with historical data, it can be useful to work with catalogs of several centuries. The century is available in the Nordic Format, so catalogs can go back to year 0. Output:

`select.out`: A CAT-file or compact file (depending on input) of selected events.

`index.out`: A list of event id's of selected events can be used with EEV or other programs accepting index files. This could be used e.g. to work on only distant events in the database by first selecting all distant events and then working with these directly in the database using command EEV index.out. Index files can have any name (must contain a '.') so different subsets can be available with different index files.

`Waveform_names.out`: A list of corresponding waveform files. It is mainly intended for copying to or from tape specific waveform files. It has the format of the `filenr.lis` files and can be used directly with e.g. MULPLT. See also program get_wav for selecting waveform files from the database.

`select.inp`: A file with all the parameters used for the run. The file can be renamed, edited and used as input for select. This is particularly an advantage if a complex set of selection parameters are used and the selection is wanted again with just a small change. An example file is shown below

```
Base or file name            : TEST_
Start time                   : 19930300000000
End time                     : 19961231235959
Minimum number of stations   :              0
Maximum number of stations   :            999
Minimum latitude             :        -90.000
Maximum latitude             :         90.000
Minimum longitude            :       -360.000
Maximum longitude            :        360.000
Minimum magnitude            :          5.000
Maximum magnitude            :          7.000
Magnitude agencies           :
Hypocenter agencies          :
Minimum rms                  :          0.000
Maximum rms                  :        999.000
Minimum depth                :        -99.000
Maximum depth                :      99999.000
Minimum error in latitude    :          0.000
Maximum error in latitude    :      99999.000
Minimum error in longitude   :          0.000
Maximum error in longitude   :      99999.000
Minimum error in depth       :        -99.000
Maximum error in depth       :      99999.000
Magnitude types (L,C,B,S,W)  :
Distance (ID) types (L,R,D)  :
Event types (e.g. E,V,P)     :
Minimum number of polarities :              0
Felt earthquakes             :              F
Fault plane solution         :              F
Check all header lines       :              F
Waveform files to check      :
Minimum gap                  :          0.000
Maximum gap                  :        360.000
Phases                       : P   SSS PP
Volcanic subclasses          :
Stat., comp. dist range, phase (1x,a5,a2,2i7,1x,a4) one pr line, end blank line:
STAT CO Mindis Maxdis Phas All stat hdist->TT
BER  SZ      1    999 P

Polygon points (lat,lon), one pair pr line, end with blank line :
```

Note: The TT at STAT line indicates that all stations must be present (True) and hypocentral distance is used (True)

Select with input from the prompt line

This option is particular useful when using select with automated operations and has been made specifically to deal with extracting data out of the data bases using WEB based software. This option do not have all of the above options. The following are implemented:

-base :	5 letter data base
-seisweb:	if set, WEB output parameters
-time :	time interval (2 variables)
-web_out:	complete path to where data is placed, only active if seisweb set. 3 files are made: `web_out.id` : id's, like `index.out` without `web_out.all` : like `select.out` `web_out.head` : header lines
-area :	lat-lon grid, minlat,maxlat,minlon,maxlon
-depth :	depth range, mindepth,maxdepth
-mag :	magnitude range, minmag,maxmag
-nstat :	range of number of stations, min,max
-gap :	range of gap, min,max
-rms :	range of rms, min,max
-magtypes :	up to 5 mag types, one string, e.g L
-disttype :	distance type, e.g D
-eventtype :	event type, e.g. E

Problems: An event might be found and listed in `index.out`, but when looking for it with EEV, it is not there. This can happen if an event has been deleted with EEV and no UPDATE has been made, so that the event is still present in the CAT part of the database.

6.4.1 Searching for text string in nordic files, SELECTC

The command SELECTC is used to search for text stings in nordic files like `collect.out` or `select.out`. Events with the maching text string is listed in the output file `selectc.out`. The program is written by **Ruben Soares Luís**. Below is an example :

```
selectc
Input file: collect.out
Search comment: Bergen
Found 634 events. Saving output file: selectc.out
```

6.5 Extracting events from the database, COLLECT

The command COLLECT is used for collecting many event files from the database S-files into a single file. This may be split into individual event files later using SPLIT. The file can be used for exchanging data with other agencies or be used with the epicenter plotting program. The questions are:

```
Base  name, ,, for local directory, name of index file
or return for default base:

Start time                          :
End time, return for end of month:
Compact output file (Y/N=default)
```

If a local data base is input, default start time is 1980 and default end time 2015. In this way it is fast to collect all data from a local data base. At the end, the program will give statistics of collected data, and file name. For getting data out of the database represented by the monthly CAT files, use SELECT. If an update has been made, SELECT will always be the fastest program to use. COLLECT and SELECT are the only programs that can make a CAT file from the individual S-files. Program input can also be on the prompt line, below is an example:

```
collect -start_time 19910912 -end_time 19911015 -base_name BER -compact
```

This means that a CAT-file (default) is collected from BER and is written in compact format (-compact has no arguments). The time interval is between 19910912 and 19911015. Only start_time is required, the other arguments are optional. The syntax is: -"keyword" value -"keyword" value etc.

6.6 Inserting events into the database, SPLIT

The program splits up a multiple event S-file in Nordic format (usually made by COLLECT or NEWEVE) or compact file to single files in the database or in the users own directory. Type SPLIT to start program and questions are:

```
 INPUT FILE NAME                          !  Give file name
COLLECT.OUT                               !  File name
 INPUT BASE NAME:                         !  Indicate which type
   BER FOR STANDARD DATABASE:            !  - database
   THREE LETTER CODE FOR OTHER BASE
   RETURN FOR SPLIT UP IN LOCAL DIRECTORY
BER                                       !  Choice was standard base
   OPERATOR ID, MAX 4 CHARS               !  Operator id logged in file
jh

1988  2  5  13:51 35.0  L       RECORDS:  4 ! Listing of events split up
File already exists, options are:         ! try to make a file with same
                                          !  id
Ignore (leave old event):              Return
Ignore all                             I
Overwrite duplicate:                   O
Overwrite all duplicates               A
Create a new event, different ID:      N
Create new events for ALL duplicates: * O

1988  2  5  14:15 25.2  D       RECORDS:  4
1988  2  5  19: 4 10.0  D       RECORDS:  3

1992 11  1   1:32  1.0  D       RECORDS: 55


NUMBER OF LOCAL EVENTS:       1            ! Statistics of events
NUMBER OF REGIONAL EVENTS:    0            ! - split up
NUMBER OF DISTANT EVENTS:     2
```

```
NUMBER OF EXPLOSIONS:          0
NUMBER OF PROB. EXPLOSIONS     0

TOTAL NUMBER OF EVENTS:        3

TOTAL NUMBER OF RECORDS:      14

FORTRAN STOP
```

In the above example, there was already an event in the database with the same file name and therefore the same id. It is up to the user to decide if this is the same event in which case it should be ignored or if it is a new event which happens to have the same id (start time or origin time to the same second and same event type). In case of a new event, a new id with one second different will be tried. Sometimes it can be desirable to overwrite the whole database event by event. If e.g. a station code is wrong in all events, this can be corrected by making a collect to extract all events, edit the `collect.out` file using a global substitute, and finally use split to put the events back in. In that case the option of overwriting all should be chosen.

Compact files can also be split up. Since this is unusual to do, the user will be prompted 2 times to confirm the split up. Since there is no ID line in a compact file, the database name will be generated from the header time. This option to be able to split up compact files has been made to facilitate work with seismic catalogs where it is often desirable to be able to access individual events even when no readings are available.

6.7 Updating final locations in database, UPDATE and UPD

UPDATE

Both the monthly epicenter files in \SEISMO\REA\BER__\CAT and the updated S-files are generated with program UPDATE which is a special version of HYP. Type UPDATE to start the program and there will be questions about time period and database. The program will also ask for operator ID (4 chars), which is stored in the updated log file and the S-file, see below.
By updating, both the S-files and the CAT-files in the CAT-directory are updated. The reason for updating both at the same time is to ensure that there is a correspondence between the two.

The program will go through as many months as specified by the user. When the program is running, one line will be printed out for each event. The S-files will be overwritten with the updated location, residuals etc. At the same time, a monthly CAT file is created in the CAT directory containing all events, also events not located. If a monthly file is already present, it is overwritten.

Update can also work on a local database. The S-files are updated as described above. Since there is no CAT database, the Update program makes a CAT file in the local directory called hyp.cat with events in chronological order.

At this time, an S-file might contain several old ID-lines which in an append process have been converted to comment lines. These are deleted when doing an update. The remaining ID-line is updated with the action UPD, the operator ID and the time. At the same time, all the error lines are deleted and only the one belonging to the prime location is kept.

The update process can also change all S-file names according to the origin time and the ID's are changed correspondingly. This is done in order for the database to be in chronological order according to origin

time and not the more random times used when the events were first registered into the database. Even if the event is marked not to be located with a * in header line column 45, the ID will still be updated (same for program UPD). Like with the SPLIT program, if two events of the same type (L, R or D) have the same origin time to the second, one second is added to the file name part indicating seconds (see also section 6.6). The event will also be in chronological order in the CAT database.

*****VERY IMPORTANT ******

The first time an update is done, the S-files get a new name according to the origin time now calculated and the internal ID is changed accordingly. The ID is then locked indicated by an L in column 76 of the ID line. For all future updates, by default, the ID will remain the same, the S-file name will also be the same irrespective if the origin time changes. This is VERY important in case data is taken out of the database for some special analysis and then put back in to overwrite the original data. If the ID is the same, the correct event will be replaced. Optionally, Update can make a new ID each time the program runs (not recommended). It might be necessary sometimes to allow this in case the events are no longer in chronological order according to origin time (e.g. a teleseismic event is put in with the ID corresponding to the recording time, when located, the origin time is many minutes before and it will appear too late in the database). However, this is rarely a problem after the first location is done and it is recommended to use the default option of locking the ID.

NOTE: When an update takes place, the old location, magnitudes (except 3. if a different agency from the default agency), residuals etc are removed. If an event cannot be located, the old location etc is lost. This is intentional since the updated database should represent the data available. If a location should be retained, special flags must be set, see section 6.1, "Fixing location" (a '*' in column 45 in header line).

In order to keep track of how and when the database has been updated, every run of UPDATE creates a log file of the update process. This file is located in a subdirectory of the database directory (default BER__). If e.g. updating REA, the logfiles will be in ../REA/BER__/LOG/ (unix). Filenames are similar to S-files. Below is an example of a logfile with name `01-0000-00L.S199606`:

```
1996 06 kk   99-09-08 14:30 03-1955-35D. 25-0337-29L.     6
1996 06 jh   98-09-08 14:29 03-1955-40D. 25-0337-31L.     5
```

The content is as follows: date and time of file updated, operator ID, time of update, event id of first and last event of the month, number of events for month. The example above shows that June 96 has been updated 2 times, the last time on September 10, 1999. For each update, one line is added to the top of the file, so the update history is saved.

Note: If the command UPDATE is used from EEV, only one S-file is updated (name stays the same), and a general update should be made.
UPDATE recalculate moments if distances (or depths) change, however it does not change the Vp or Vs velocities used if a change is made in MULPLT.DEF.

Problem: If UPDATE crash, there will not be a correspondence between S-files and the CAT data base: Redo UPDATE.

UPD

The command UPD is very similar to the UPDATE command, however there is no modification of the S-file except the ID line. The program is used to simply move single S-files into the monthly CAT-files without relocating. It is mainly used to manipulate database events already processed. E. g. if ISC data a available and it is desirable to have it in individual files to be able to use EEV, the same data can then be copied into the CAT part of the database using UPD without modifying the original solutions. The

data must be in the CAT part of the database in order for SELECT to work fast. KNOWN BUG: On Sun OS, it seems that UPD can only operate on up to a 4 year time period.

6.8 Using filenr.lis, DIRF and DELF

DIRF

The DIRF command is a useful program for making a file with a numbered list of files from a directory. The command makes a file with file name `filenr.lis` e.g. when working with many waveform files with long names, a DIRF is first made, and subsequent programs then get file names from `filenr.lis`, either by using the whole list, or just a given number. This is handled with routine filename (in LIB). Below are some examples of using DIRF with SEISAN data files.

```
dirf 9101-10*
#  1  9101-10-0915-15S.KMY_01
#  2  9101-10-1510-55S.N2F_08
#  3  9101-10-2333-44S.N3F_06


dirf 9101-10-0915-15S.KMY_01 9101-10-2333-44S.N3F_06          (Unix only)

#  1  9101-10-0915-15S.KMY_01
#  2  9101-10-2333-44S.N3F_06
```

The wildcard '*' above indicates that all files from the 10'th is wanted. Many programs use the same subroutine to get the file name from `filenr.lis`. This means that most programs using `filenr.lis` assume that if a name given is less than or equal to 4 characters, it is a number so file names less than 5 characters cannot be used when the program asks for "Filename or number".

DELF

DELF is a simple program that allows the user to delete a file that is listed in a `filenr.lis` file or another index file. First run DIRF to list the files that you want to delete. Then start DELF and choose the number of the file to delete, '?' shows the contents of `filenr.lis`. In addition, DELF also has an option to delete all the files in the `filenr.lis` or index file. This is a useful option if selected files in a data has to be deleted. If e.g. all S-files from a particular agency has to be removed, run SELECT first and then DELF.

6.9 Making a bulletin, BUL

The bulletin program BUL is intended for writing seismic bulletins in a nice format. The output is written to a PostScript file.

Input files:

1. A monthly data file: This file can be made by COLLECT or SELECT
2. `BUL.INP` : This file must be in DAT or in the local directory. In this file the layout of the front pages are decided, as well as the font selection for the main bulletin. There are ample comments in

the file on how the commands are written.

Some special format features:

Type 3 line: If the first 5 columns in a type 3 line are:" Bul:", then the rest of the line is interpreted as text line that is written in the bulletin. In this way comments to certain earthquakes can be written into the bulletin. Type 2 line: Maximum intensity and casualty/damage reports are included in the bulletin if found in the S-file.

How to run the program:

Type `bul -h` this gives you a list of the different options like this:

Options:	
-frontpage:	Only frontages are printed.
-nofrontpage:	No frontages are printed. ll
-onlyhypo:	Only hypocenter solutions are printed.
-minmag x.x :	Only hypocenter solutions with magnitude than the requested are printed.

The last option may be used in cases where the number of earthquakes is very high, so that it is preferable to report phases only for events above a given magnitude.

You can also run the program without any options, in which case the default values used are:

i) All phases are reported.

ii) Front pages are printed.

You will always be asked for the name of the S-file.

Output file:

The output file is called `bul.ps` and is a PostScript file that you can print. Optionally, a limited number of pages can be selected from the `bul.ps` file for printing. The header page is still included and the page numbers correspond to the original page numbers.

6.10 Reports and statistics

SEISAN has several programs for extracting and writing out data for plotting or printing statistics, most of which will be listed in this section.

Report

The program extracts parameter data from all header lines in a CAT file and rearranges the data in a table. In additions, there is an option to rearrange order and location of magnitudes on the header line. Below is an example of a run where the input CAT file is called `collect.out` :

```
report collect.out

Below is shown parameters which can be chosen for output.
 A return will chose all, placing any character under a field
 will chose that parameter in the output. Each field starts
 with a capital letter and ends within the following blank.
```

```
 The order of the output can be changed by placing a number
 under the field and fields will be written out in the order
 of the numbers. E after time, lat, lon and dep are errors,
 L E is distance and event id s, F is both fix flags and A is
 agency for magnitude.
 The following example shows that Mc, Depth(Dep) and Time with
 error are selected and written out in given order.
Date TimeE L E LatE LonE Dep E F Aga Nsta Rms Gap McA MlA MbA MsA MwA Fp Spec
     30  45              20                       10

Date TimeE L E LatE LonE Dep E F Aga Nsta Rms Gap McA MlA MbA MsA MwA Fp Spec
x               x    x    x           x       x                       x
 Number of output fields            8

 Number of events                                    12
 Number of events with spectra:                       1
 Number of events with fault plane solution:          3
 Number of events with error estimates:              16
 Number of events with mc             :              10
 Number of events with ml             :              10
 Number of events with mb             :               8
 Number of events with ms             :               5
 Number of events with mw             :              12

 Output report file is report.out
 Output nordic file is report_n.out
 Output of choices used in report.inp
```

The `report.inp` is a file with the choices used. Report can use that file (or a file with the same format and a different name) as second argument:

```
report collect.out report.inp
```

in order to use a fixed set of choices.

Content of `report.out`

```
 Year Date Latitud Longitud Depth NST GAP   Ml STRIK  DIP RAKE
 1996  6 3  47.776  153.222   0.1  12 348
 1996  6 6  62.652    4.940  15.0  13 270  2.9    28   61  -41
 1996  6 6  62.634    5.068  15.0  13      2.9
 1996  6 6  62.652    4.940  15.0  13 270  2.9    28   61  -41
 1996  6 7  59.841    5.127  12.0  12      1.9     8   41   75
 1996  610 -13.481  167.130 200.1 301
 1996  625  61.656    3.363  14.9  35      3.2
 1996  7 5  61.282    4.825   7.1  10      2.0
 1996  713  61.416    3.870  12.1   9      1.5
 1996  718  60.156    2.070  15.0   9      1.8
 1996  718  51.438  157.737  29.9  18
 1996  726  61.755    2.293  22.1   9      1.8
```

The file `report_n.out` contains the input data with the only difference that the magnitudes have been moved around on the header line. This can be practical for later plotting with EPIMAP. If no magnitude selection has been made, the magnitudes will come in the order Mc, Ml and Mb. If no magnitude of that type is available, the output field is blank. The magnitude selected is the first to occur of the corresponding type. If other magnitudes are to be selected, numbers can be used to select any 3 magnitudes in any order. If it is important to select magnitudes by agency also, use program MAG. REPORT can also give a numbered output by adding the second or third argument `-n`.

NORHEAD, making a compact Nordic file from a Nordic file

You must give arguments: First is input file, optional second is output file, if an optional second or third argument is -mag, magnitudes from following header lines are moved up to empty magnitude spaces on first line. The program was earlier called COMPACT (version 7.2 and earlier).

STATIS, statistics of databases

This is a simple program for making statistics of stations used in the database or in a file. The program will ask the following questions:

1. Information about which stations should be searched for in the database. There are several options for entry:
   - a: Give a filename with the stations listed one per line. The format is a5. The file name MUST have a '.' not to be confused with option (b) below.
   - b: Give stations, one pr line, enter to finish, enter for def file statis.def
   - c: Just make a return and the stations given in file statis.def will be used. The file has one station per line an dcan be located in either the working directory or DAT.
2. Standard questions about base or filename and time interval
3. Question about counting all phases. This means counting the occurrence of a station for each phase for that particular station. This can give the total number of phases read at a particular station in a given time interval which can be more than the number of events. If not counting all phases, the program gives the number of events recorded at the station.

The output from the program could be as follows:

Station	Local Ev.	Local S.	Distant E	Distant S
KONO	0		21	8
KMY	24	10	6	0
ODD	0			
EGD	28	0	3	1
ASK	29	0	3	0
HYA	16	1		0
SUE	16	4	2	0
FOO	18	1	3	0
NRAO	86	9	0	
MOL	38	17	14	1
NSS	9	6	7	0
MOR	0			
LOF	25	13	12	0

```
   TRO          12          2         13          1
   BJO           0
   KBS           3          2          8          6
   JMI          16          0
   KTK1         22          1          2          0
   ARA0         66          6          0

NET  NWAW
NSN   147
JMI    21
KNN    10
W_L     2
W_E     1
W_S     2

Number of events selected with given stations     222
Number of events selected with more than ---       98
Number of events with no phases                     0
Number of events with waveforms                   168
Number of events with only waveforms                0
Number of events with 2 or more waveforms           9
Total number of waveform files                    183
Total number of local events                      200
Total number of regional events                     0
Total number of distant events                     34
Total number of events                            234
Total number of records                          2830

Output files are: statis.out
                  statab.out (station statistics only)
```

The top part shows the event statistics by station. Local Ev is number of local events (readings if so specified above) (type L and R) at the station, Local S means number of local events ONLY recorded at that station, Distant E and distant S is the same for distant events (type D). The middle parts shows the number of waveform files NWAV from different networks NET as indicated by the first 3 letters of the waveform file name after the "." At the bottom is a summary statistics most of which should be self-explanatory. The information about ".. more than given stations" means that in addition to the stations searched for, the event had additional stations not used in the statistics.

CATSTAT

This program calculates the yearly, monthly and daily number of events from a given earthquake catalogue and plots the results (written by **Mario Villagrán**). The input is a standard Nordic file containing only the header lines (compact file). The output is given in three different files with following default file names:

`catyear.out` :	Output catalogue of the yearly number of events. This file contains two columns of data corresponding to year and the number of events.
`catmonth.out` :	Output catalogue of the monthly number of events. This file contains three columns of data, corresponding to the year, month and the number of events, respectively.
`catday.out` :	Output catalogue of the daily number of events. This file contains four columns of data corresponding to the year, month, day and the number of events, respectively.
`cathour.out` :	Hourly distribution of events within a day interval.

In addition, a series of files with gmt in name give similar output for use with gmtxy (only Unix). The output files can then be used for plotting the histograms for the desired time interval at yearly, monthly or daily intervals. If desired, the corresponding histograms can be plotted interactively on the screen or can be printed. Several other routine programs such as grapher, xyplot, gnuplot or GMT, etc., can also be used for this purpose. The general purpose of this program is to evaluate the catalogue completeness. When run for different magnitude intervals, one can detect the magnitude thresholds above which the catalogue can be considered complete.

SWARM, finding earthquake swarms

The program is used to identify seismic swarms in a catalog. Input to the program is a CAT file with many events and some manually entered parameters. Output is identified swarms. The output file swarm.out contains all swarms organized as 'events'. In the header line is given the center for area identified and the 'magnitude' is the number of events in the area divided by 10. The rest of the line is information from first event in swarm.

Principle of selection:

The area is divided into a lat-lon grid. Around each grid point, there is a cell with radius small_r. The program first checks the number of events in each cell for the whole catalog. It then checks each cell to find which has more than the minimum number of events to constitute a swarm under the condition that enough events are within the required time window. For each time window, with enough events, a swarm is declared so a swarm lasting e.g. twice the time window will be declared as two swarms. An additional condition is that the number of events is larger than the normalized background activity. The normalized activity is calculated as the activity in the large cell normalized for area to the small cell, and normalized in time to the window for the swarm.

STATSTAT, number of events per seismic station in catalog

The program reads Nordic file input data and writes out text files giving the number of events per station.

LSQ

A simple program to make and plot a least squares relation between two parameters. Input is from a file with two columns x and y. The program also makes an output used with GMT in order to make nice plots. The PostScript output file is `lsq.eps` and the GMT file is `lsq_gmt.out`. In order to produce the GMT plot (only Unix), use command `gmtxy lsq_gmt.out`.

6.11 Waveform file management tools

This section describes the programs used for modifying and checking waveform files. The most important features are to add or subtract channels and modify headers. A special program in this group is GET_WAV which checks data bases for availability of waveform files. New from version 7.1 is that SEISAN also can handle other waveform formats, however not all programs can work with all formats. This will be indicated with each program. The following programs are available:

APPEND:	Append two or more waveform files following each other in time
AUTOREG:	Automatically register events
CONGAP:	Check completeness of continuous waveform database
CONNOI and EVANOI:	Compute noise power spectral density and evaluate out to produce plots
DATABASE2MSEED:	Convert waveform data to miniseed based on parametric database
GET_WAV:	Check for available waveform files
MSCUT:	Cuts MiniSEED files into 1 hr or 15 min files (Unix only)
RDSEED_MANY:	Simple way to chop up a seed volume
RESAMP:	Resample waveform files
SEIASC:	Convert SEISAN waveform files between ASCII and binary form
SEICUT:	Extract an interval of a waveform file
SEIDEL:	Splitting up a SEISAN waveform file in 2
SEISEI:	Split and merge SEISAN, GSE and MiniSEED waveform files
SELSEI:	Find waveform files with given stations
P_ALIGN:	Time shifting waveform data to align P-phase arrival times
WAVETOOL:	Extract waveform data
WAVFIX:	Fix waveform file header time correction, make standard file names, change headers etc.
WAVFULLNAME	Print full file name including path for waveform file.

APPEND, Append two or more waveform files

The program uses a `filenr.lis` input file. All files are read, and then written out as one new file. The maximum number of channels is max_chan_out which is set as a parameter (currently 7). Only the first max_chan_out channels are used or less if fewer channels in file. A blank line followed by a new group of files will make a new output file. The output file cannot have more than standard SEISAN dimension number of samples (more than 1 200 000, see file `../INC/seidim.inc` for exact number) per channel.

It is assumed that all channels have the same sample rate.

AUTOREG, automatic registering of events

When a large number of waveform files are available and it is known that they are real events, it might be an advantage to automatically register them into a database. Remember, the database can be made beforehand with MAKEREA. If the filename follows the SEISAN filename convention, the date and time used to generate the S-file are taken from the filename. Otherwise, the file is read to get the date and time

from waveform headers. Obviously, the first option is faster. It is possible to register events both to the default database, any other database or the local directory. To run the program, make a `filenr.lis` of the waveform files and run AUTOREG. It is possible to put blank lines into the `filenr.lis` to separate into events, in case there is more than one waveform file from the same event. All waveform files before a blank line are put together into one S-file. Optionally, the waveform files can also be moved or copied to WAV or a WAV database subdirectory (including year and month). This can either be the default parameter COPY_WAV_DIR (in `SEISAN.DEF`) if different from blank. COPY_WAV_DIR should be the same as the data base used by the S-files. However an optionally data base directory entered interactively can also be used.

You get the questions:

```
 autoreg
  Event type for all events: Local:    L (default)
                             Regional: R
                             Distant:  D

  Move (m) or copy (c) waveform files to WAV (enter=n) ?
m
 Files will be moved to default WAV base: TEST_
 Enter new WAV data base to copy or move to
 enter for default base or m for main WAV directory
LVC
  Operator, max 4 chars
jh
 2010.018.12.26.15.0695.IU.LVC.10.LH1.R.SAC
 /home/s2000/seismo/REA/LVC__/2010/01/18-1226-15L.S201001
 mv   2010.018.12.26.15.0695.IU.LVC.10.LH1.R.SAC /home/s2000/seismo/WAV/LVC__/2010/01/
  File transferred to WAV base LVC__ **********
 2010.018.12.26.52.0695.IU.LVC.10.LHZ.R.SAC
.....
.....

In this case wavform files were moved to  data base LVC instead of the default TEST.
```

Now comes a listing of waveform file names and S-file names. The program will check if the event is already registered and the same options are available as in program SPLIT (section 6.6). Since AUTOREG automatically creates S-files for all events in `filenr.lis`, they will all be given an event type.

CONGAP, check completeness of continuous waveform database

This program checks for completeness of continuous data for a given time interval. The program reads the waveform data to see what data are available and checks for gaps, defined by a constant amplitude value (e.g. 0). The input can come either from an input file (`congap.par`) or the command line.

Parameters in the input file are:

CONT BASE:	name of database, you can have more than one
START DATE:	start time and date of interval to be read (yyyymmddhhmmss)
STOP DATE:	stop time and date of interval to be read (yyyymmddhhmmss)
INTERVAL:	duration of intervals read at a time in minutes (e.g. 60. for one hour)

When started from the command line, the same parameters can be given:

```
congap -start <yyyymmddhhmmss> -stop <yyyymmddhhmmss> -cbase <text> -interval <number>
```

The output file (`congap.out`) looks like this:

```
EDI   HHZ 20080101 0000 0.00   3600.00   3600.00
EDI   HHN 20080101 0000 0.00   3600.00   3600.00
EDI   HHE 20080101 0000 0.00   3600.00   3600.00
EDI   HHZ 20080101 0100 0.00   3600.00   3600.00
EDI   HHN 20080101 0100 0.00   3600.00   3600.00
EDI   HHE 20080101 0100 0.00   3600.00   3600.00
...
```

The fields are station and component code, date and time, expected duration and actual time with data. The output file can be used to produce plots showing data completeness (tool for this not included). When the program runs it also produces a summary output at the end, where the last column gives the percantage of data completeness, and the actual and expected times are in seconds:

```
-------------------------------------------
 # stat comp       actual     expected     %
-------------------------------------------
 1 EDI   HHZ     86400.00     86400.00 100.0
 2 EDI   HHN     86400.00     86400.00 100.0
 3 EDI   HHE     86400.00     86400.00 100.0
-------------------------------------------
```

CONNOI and EVANOI (does not compile with gfortran), noise power spectral density

These two programs with the help of GMT allow to produce noise power spectral density (PSD) plots similar to the ones produced by the PQLX software. CONNOI is the tool that reads the continuous database and produces output files that are evaluated by EVANOI. The computation of the noise PSD follows the method described by McNamara and Buland [2004].

To run CONNOI use for example:

```
connoi -start 20100501 -stop 20100502 -cbase BER
```

In this example BER is the database, you can also specify 'def' and the program will take all default continuous databases defined in SEISAN.DEF. The default output filename is connoi.out.

Example of output:

```
stat comp date and time       duration frequency noise PSD
-----------------------------------------------------------
BER   HHZ 20100501 0000 0.00   3600.00  0.00200    -159.14
BER   HHZ 20100501 0000 0.00   3600.00  0.00204    -159.14
BER   HHZ 20100501 0000 0.00   3600.00  0.00209    -159.14
...
```

The output from CONNOI can then be used as input to EVANOI. You can enter station and component, give a time interval, select a time of day interval, and chose a reference station. EVANOI produces GMT plotting scripts files that are named after the station. Then simply run the script file to get a plot.

DATABASE2MSEED, convert database to miniseed

This program can be used to convert the waveform data that is linked to from a parametric database to miniseed format. The user is asked to enter the database name, and start and end time for the operation.

One should do a small test with a copy of parts of the database before runnung it through the complete database.

WAVETOOL, extract and convert waveform data

The program was called extract in SEISAN 7.2 and earlier. The program extracts all or selected time sections of waveform data, optionally applies some signal processing and then creates the output file(s). The input formats supported are SEISAN, SEED, MINISEED, GSE and SAC (all platforms)and output formats are SEISAN, MINISEED, GSE and SAC (all platforms). The program can be used as a conversion program between these formats, instead of using e.g. SACSEI. It would also be possible to convert for example SAC and GSE files to SEISAN in one go. When creating GSE, MINISEED or SAC files, the respective format code is added to the filenames. The program can also extract data from a SEISAN continuous data base or a large SEED file.

If an output file already exisits, "_SEI" is added to file name.

Note that for MINISEED writing, only Steim1 compression is used. Integer format is also possible, but requires a parameter change in WAVETOOL and recompilation.

There are two input options: (1) a single S-file or a list of S-files created with DIRF (is also an index file), which points to the waveform data or a `filenr.lis` type of file that gives the waveform file names; (2) a waveform file or a list of waveform files. The program can be started either interactively, without arguments, or non-interactive by specifying the commands as arguments.

The arguments are: `-sfile <sfile-name>` : The S-file name of the event you want to extract waveform data from.

-wav_files <file-name>: Extract from a list of waveform files in `filenr.lis` format. Input from S-file will be ignored. This option merges all files from list (if within a 'reasonable time window') and program is then partly doing what SEISEI is doing for only SEISAN format files. If file-name is given as 'SEISAN', then the output file name follows SEISAN convention.

-wav_in_file: input waveform file.

-maxpoints : Skip points to get approximately maxpoints

-wav_out_file <file-name>: Name of waveform output file, not used if output format is SAC.

-start <time>: Start time can be used to set start time the same for all channels instead of using chansel file, time can be absolute or relative to beginning of the first trace. Options are Start time relative earliest channnel, Abs start time string yy...s.sss for all channels.If ABS time used, string MUST contain a '.'.Abs start time string yyyymmddhhmmss (integer), used to define cont start time.

-duration <time>: Select duration of time window in seconds if -start is used.

-interactive : enable interactive mode

-cbase: name of file with selected continuous databases,all bases is default

-command_file: give arguments to wavetool in a file rather than on the command line (used by Seisweb)

-cwav: Input is from the SEISAN continuous data base, useful for extracting intervals

-cseed: Input from large SEED file, similar to -cwav -wav_in_file <file-name>: Input of one waveform file

-bud: Input from BUD archive.

-scp: Input from SeisComp archive.

-sfile <file-name>: Input of S-file name

-format <output format>: The output formats supported are SEISAN, SAC, GSE (GSECM6) and GSEINT. In case of SEISAN or GSE, multi-trace files including all selected traces are created, while for SAC, single trace files are generated.

-chansel <file-name>: Input file to select channels and time windows. The first line contains number of channels. The following lines give station code, start time (both absolute and relative to earliest trace allowed) and duration. If start time and duration are set to 0, complete traces are selected.

-chan_out_file: file name of file with channel description if this option is given, program terminates after writing out the file.

-ground : compute displacement, velocity or acceleration (1,2,3)

-seed_location: seed location code NOT USED.

-rsam: comput 1 min rsam data.

Example

Select the first three complete traces
3
1 0 0.0
2 0 0.0
3 0 0.0

Use absolute start time, and duration of 180 seconds

3
1 19991001124500.000 180
2 19991001124500.000 180
3 19991001124500.000 180

Use relative start time of 60 seconds from beginning of earliest trace, and 300 seconds duration

3
1 60 300
2 60 300
3 60 300

The program assumes that a large number is absolute time.

`-chan_out_file <file-name>` : Name of text file containing a list of available channels from a list of waveform files. If wav_out_file is not specified, program terminates after creating the list.

Example

```
 1 KBS   BV Z  1996  6  3 20  2 18.991  6000   20.000 2666.400  299.950
 2 LOF   S  Z  1996  6  3 20  5  5.531  5800   50.000 2832.940  115.980
 3 MOL   S  Z  1996  6  3 20  5 24.984 10000   50.000 2852.393  199.980
 4 FOO   S  Z  1996  6  3 20  5 34.156  9650   50.000 2861.565  192.980
 5 HYA   S  Z  1996  6  3 20  5 36.078  9900   50.000 2863.487  197.980
```

`-filter <flow> <fhigh>` : bandpass filter limits

If only a filter is used, a 4 pole Butterworth filter is used one way. If both a filter and response removal is used, an 8 pole Butterworth filter is used in the frequency domain

`-npole <n>` : number of poles used for filter, if not given, 4 is used

`-ground <1,2,3>` : compute displacement, velocity or acceleration (1,2,3)

`-ichan <id>` : select one channel only

`-interactive` : Flag to specify non-interactive use, in which case the program does not ask any questions (for example given by SEISAN-autodrm interface), default is interactive.

`-seisweb` : Flag to indicate that the program is started by SEISWEB.

`-maxpoints <number>` : Number specifies the total number of points desired for the total time window covering all selected traces. This option is meant to reduce the number of points to what is needed to visual correctly plot the traces. When plotting the trace using a number of pixels, which is smaller than the number of points on the trace, samples are plotted on top of each other for one time sample. This results in the maximum and minimum being plotted at the same place on the x-axis. The idea now is to reduce the trace to these maxima and minima only. Then using twice the number of samples than pixels will allow to visually correctly show the trace. Note that this is not a resample routine. Option mainly used with SEISWEB.

`-stat_out` : write out station location file, simple xy output file (`station_list.out`)

`-rsam` : convert data to RSAM (1-minute absolute average), which is commonly used in volcano seismology

`-resp_out` : write out list of all response files for channels given in waveform files (`respfile_list.out`)

Intercative input

The interactive input has less options as the mon interactive input, however the option are as above. The questions are:

Filename of s-file or waveform file, number or filenr.lis Maximum number of points in output trace, return for all same as -maxpoints Ground motion output (dis = 1, vel = 2, acc = 3, none = return) Filter low and high, return for no filter Select all data, y=return,n If answer is n, then the following 2 questions come: Number of channels to read Channel number, start time and window

Output formats (SEISAN, GSE (def=CM6), GSEINT, SAC, MSEED) Default is SEISAN=return

Another way of extracting waveform data is using MULPLT where many traces can be extracted as a binary SEISAN file (using WAVETOOL in the background) or a single trace as an ASCII file.

Accuracy of extracted data

If the data is filtered or corrected for instrument response, the number out can be less than one and an output file of zeros can be made. If the output format is SEISAN, the values will always be scaled to avoid this and the appropriate scaling factor is included in the waveform file. Subsequently reading of these files in SEISAN will produce the correct values. For this reason, it is advised to use SEISAN as output format when filtering or correcting for instrument response. SAC data can have values less than 1.0 so only if written in SAC or SEISAN will the values be correctly represented. SAC input data is checked for max values. If smnaller then 10, output will be scaled in SEISAN format. Note: Parameter MERGE_WAVFORM in `SEISAN.DEF` sets the network extension of extracted files.

GET_WAV, get listing of available waveform and response files

The program uses a CAT file as input and checks for availability of all waveform files listed. For each

channel, there is a check on existence of corresponding response files. A typical run is shown below:

```
get_wav
 INPUT FILE NAME
select.out
 Full path name : /net/seismo/users/jens/TD/WAV/1996-06-03-2002-18S.TEST__012
 Full path name : /net/seismo/users/jens/TD/WAV/1996-06-03-1917-52S.TEST__002
 Full path name : /net/seismo/users/jens/TD/WAV/1996-06-06-0647-46S.TEST__011
 Full path name : /net/seismo/users/jens/TD/WAV/1996-06-07-1324-51S.TEST__009
 Full path name : /net/seismo/users/jens/TD/WAV/1996-06-10-0105-42S.TEST__014
 Full path name : /net/seismo/users/jens/TD/WAV/1996-06-23-0126-27S.TEST__013
 Full path name : /net/seismo/users/jens/TD/WAV/1996-06-23-0059-47S.TEST__001
 Full path name : /net/seismo/users/jens/TD/WAV/1996-06-25-0336-34S.TEST__032

  Total number of events 6
  Number of events without waveform files 0
  Number of waveform files 8
  Number of waveform files present 8
  Number of waveform files missing 0
  Number of cal files found 28
  Maximum number of cal files missing 29
  Output file with events is get_wav.out
  Output file with waveform file names is copy_wav.out
  Output file with cal files is copy_cal.out
  Output file with waveform file names missing is copy_wav_missing.out
  Output file with missing calibration channels is copy_cal_missing.out
```

Note: On PC the files `copy_wav` and `copy_cal` have names `copy_wav.bat` and `copy_cal.bat`, respectively

In the above example, a `select.out` was used. For each file, it is checked if the waveform and response files are available in the system. All waveform data bases and directories specified in `SEISAN.DEF` are searched. Calibration files are seached for in working directory and CAL. In order to extract the waveform files corresponding to the input CAT file, the output file `copy_wav.out` can be used to copy the files out of the data base to working directory. On Unix, just source the `copy_wav.out` file, on Windows, change the file to a `.bat` file (e.g. `copy_get_wav.out wav.bat`) and run it. For the calibration files there is similarly a file called `copy_cal.out`.

MSCUT chop up MiniSEED files

The program cuts up MiniSEED files into 1 h or 15 min files. The program is compiled for Unix but probably also works under Windows (not tested). To compile the program, the miniseed library ***libmseed*** by **Chad Trabant** is required. The options are

-H	Cut into one hour files (default)
-Q	Cut into 15 min files
-V	Report program version
-h	Show this usage message
-v	Be more verbose, multiple flags can be used
-p	Print details of header, multiple flags can be used
-s	Print a basic summary after processing a file
-r bytes	Specify record length in bytes, required if no Blockette 1000
file	File of Mini-SEED records

RDSEED_MANY, chop up seed file

The program reads a large SEED volume and divides it up into several files of the same size. It calls rdseed, so rdseed must be installed. Rdseed can do the same, but RDSEED_MANY is simpler to use. Example:

rdseed_many
Seed file name
test.seed
start time YYYY,MM,DD,HH,MM
2005 01 01 01 01
Interval in minutes
20
Number of intervals to extract
200

The output format is SAC, other format require a change of the program.

RESAMP, resampling waveform files

RESAMP is a simple resampling program, which can resample one or several waveform files. It only works with SEISAN format. All files are read, filtered and resampled. Then written out as one new file with the data from one or several input files. The maximum number of channels is max_chan_out, which is set as a parameter in the program, currently it is set to 7. Only the first max_chan_out channels are used or less if fewer channels in input file. It is assumed that all channels have the same sample rate and will be resampled to the same lower sample rate, which is an integer fraction of the original sample rate. If e.g. the original sample rate is 50, new rates of 25,10,5,2 etc can be obtained. The anti-alias filter is a single pass Butterworth with 6 poles. The user specifies manually both the decimation rate (2,5,10,25 in the above example) and the filter frequency. The new file(s) can have a new component specification, which is asked for interactively. Finally the user is asked for a new network code.

The input files(s) come from a `filenr.lis` file generated with DIRF. If more than one file is given in the `filenr.lis`, these will be put together in one file and some samples are saved from one file to the next in order to assure that there are no overlap problems when using the filter. IT IS ASSUMED THAT ALL FILES HAVE THE SAME LENGTH OF TIME. The program will check if a following file has the correct header time based on the length of the previous file. If the following file starts before the end of the previous file (err_samp samples, default 70), it is assumed that the timing is wrong and that the files should follow each other. A warning is given and the program continues. If the following file has a header time that is more that a given err_samp samples after where it should be, it is assumed that the next file is missing and zeros are inserted in the channel data. The number of sample errors, err_samp, is hardwired in the program, currently 70. The program will continue to put data together in one file until there are no more file names in the `filenr.lis` file or a blank line is encountered. After a blank line in the input file, a new output file will be created. This can be used to make daily files of e.g. 2 weeks on continuous data by manually placing a blank line in the `filenr.lis` file for every 24 hours. The program recalculates the sample rate based on time in first and last file. The output file name will be given the standard waveform file name with type R for resampled like `1999-07-02-1112-22R.BERGE_005`.

Works ONLY with SEISAN format

SEIASC, converting SEISAN waveform files to or from ASCII

A simple program to make an ASCII equivalent of a binary SEISAN file, or vice versa. It is the same call to use the program both ways. By using a `filenr.lis` file as input, many files are converted and the original filenames are kept with the addition of an A for ASCII or B for binary. If the files are converted

back, the A or B is removed.
The program is useful for manually editing a waveform file or checking the content in case of problems. The program is also useful for moving binary files between different types of computer platforms (moved as ASCII files, not needed between platforms running SEISAN). Between PC, Sun, Linux and MacOSX, SEISAN programs will automatically adjust for differences in binary structure. The header format is exactly like the binary SEISAN files and the sample values are written in multicolumn format.

Works ONLY with SEISAN format

SEICUT, extract part of a waveform file

A simple program to extract out a section of a waveform file (any seisan primary format). A similar job can be done with wavetool. Syntax is:

```
seicut filename yyyymmddhhmmss.s interval
```

The first sample to use is the first sample found before the start time, the output time interval (in seconds) will be the time from first to last sample, so if e.g. one second of data is asked for at a sample rate of 100 Hz, the time interval in header will be 0.99 sec and the number of samples output will be 100. If the interval is not available in any of the channels, the program will stop. The output file name will use a network code reflecting station code of first channel in input file and 'CUT' is added to the end of the file name. The same time window must be available in all channels.

SEIDEL, splitting a SEISAN binary file into 2 files

The program splits up waveform file into 2 files. Input can be file or list of files (`filenr.lis` created with DIRF). The questions are:

```
Filename, ?, number or filenr.lis for all
filenr.lis
No of channels to remove
3
Channels to remove
1 3 6
```

The program will generate 2 new files, one with the channels removed and one with the remaining channels. The original file is still present.

Works ONLY with SEISAN format

SEISEI, splitting and merging SEISAN and MiniSEED binary files

The program can merge several SEISAN or MiniSEED waveform files to one file or take one SEISAN or MiniSEED file and split it into single channel files. The program is intended for editing waveform files and merging files from different networks to one file. In order to use SEISEI for merging files, a DIRF must be made to make a `filenr.lis` file containing the files to be merged. The program will sequentially read `filenr.lis` and merge files which have start times within the time interval specified (3 minutes default). Once a gap of more than 3 minutes occur, a new output file is made. Merging to a new file can be forced by editing `filenr.lis` so the groups of files to be merged are separated by a blank line, however, within the group, the time difference can still only be the given time interval.

If two channels to be merged have the same station and channel codes and the same start time, the second occurrence will be ignored. If the station and channel codes are the same, but start time different, the user will be asked to confirm merging.

The program can also split up a multichannel file to files with only one channel. This can be used to remove unwanted channels by deleting selected channels and merging again. When the file is split up, the channel component is added to the file name. A `filenr.lis` file can also be used for splitting many files in one go. If a file is only to be split into only 2 files, it is more convenient to use the program SEIDEL (only works on SEISAN format), see above. SEISEI is also used in connection with MULPLT for merging files automatically based on waveform file names in an S-file.

Note: The network extension of merged files will be set by default to the value of parameter MERGE_WAVEFORM in `SEISAN.DEF`.

SELSEI, searching headers in SEISAN waveform files

A simple program to search headers in waveform files for files containing a particular station.

Works ONLY with SEISAN format

P_ALIGN: Time shifting waveform data to align on P-phase arrival times.

If one wishes to compare signals (align in time) from different earthquakes observed at the same station, the P_ALIGN program can be used. The program works by time shifting the waveform header times to a common time and then putting all the new waveform file names into an S-file. First use e.g. the SELECT program to extract information of earthquakes in a defined area that have been observed by a given station, and use GET_WAV to copy the waveform files to your working directory. Then execute P_ALIGN. The input is the Nordic file (e.g. `select.out`) and the station name for data to be compared. The output is:

- Waveform files with time shifted headers, all have the same time, but station names are labeled STA01, STA02 etc in the same order as given in input file. Only first 3 letters of station code is used.
- `tsd.out` : A file in Nordic format with the new waveform file names of the time shifted data. This file can be split and then used with EEV for plotting all traces. It can be split into a local data base or any other data base. Or copy `tsd.out` to a file e.g. named `27-1200-00L.S207011` and then start EEV.

The arrival time of the P-phase in the new waveform files is the pseudo date and time : 2070-11-27 12:00.

Note: The station name is renamed in the output. If there are more than one P-phase observation for a single event (e.g. Pn and Pg, or P read on two different channels) it is the first P that is time shifted. The program can only time shift 99 waveform files and they have to be in SEISAN format. The waveform file must be present in the working directory.

Figure 6.20 show an example.

WAVFIX, fixing time correction and channel names in SEISAN waveform file headers and make standard file names

It can easily happen that a waveform file has a wrong time in the headers, or that individual channels have wrong timing, for example introduced by different delays in the acquisition system that are not accounted for. WAVFIX can change all header times with a given constant time delay, or correct individual channels as specified in a parameter file (`wavfix.tim`). In addition, the file name will also be changed to reflect the header time change. Waveform file names were shorter in SEISAN version 6.0 so when using older files, the user might want to use standard file names. In cases where only channel names or timing of individual channels are changed, the filename can be kept the same. In this case a temporary file is created, which is later renamed to the original name.

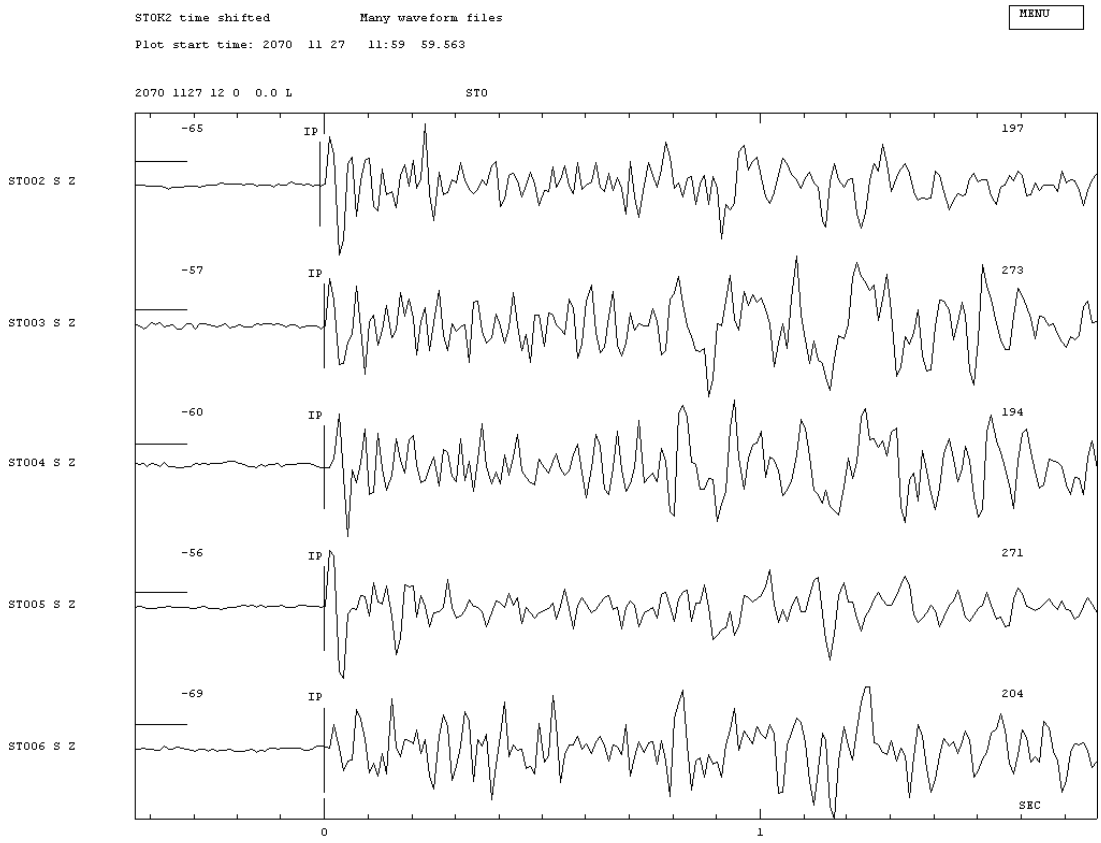


Figure 6.20: Example of aligning traces from 5 events for the same station. Note that the alignment is critically dependent on the original P-picks.

WAVFIX can also change polarity. This is done by setting the output channel and station codes to the same as the input values in `wavfix.def`.

In case channel names are to be changed, this can also be done with WAVFIX. A definition file is needed for changing station, component or both. The parameter file name is `wavfix.def` and an example is given in DAT. For definition of the `wavfix.def`, see next section 6.12 on "Conversion programs definition file".

WAVFIX can change header times and/or file names for one or many files. Before running the program, a list of file names must be made with DIRF. Below is an example where the header time is changed by 120 secs. No `wavfix.def` file is present (current or DAT directory).

To correct the timing of individual channels, you need to create the file `wavfix.tim` in either the DAT or working directory. WAVFIX checks if the file is present and applies the correction from the file as default. The format of this file is as follows:

Column 1-5: station code
Column 7-10: component code
Column 12-25: start date and time for time correction (can be empty)
Column 27-39: end date and time for time correction (can be empty)
Column 41:60: time correction to be added

Example:

```
 wavfix time correction applied to individual components
 stat  comp start time     end time       correction in
 a5    a4   yyyymmddhhmmss yyyymmddhhmmss seconds f20.3
 -----|----|--------------|--------------|--------------------
 TEST  SH Z 19500101120000 19600101120000 -0.015
```

File names of waveform files can be given to WAVFIX directly, from a `filenr.lis` file or from a Nordic format file. In case you choose the Nordic input, the waveform file names will be changed in the Nordic file (output file nordic.fix). This option is useful if you are correcting file names, since the entries in the S-files are otherwise not fixed.

Example of running WAVFIX

```
No wavfix.def file, will use internal information for channel codes
 This program will change header times in all headers
 with the same amount. The waveform file name will be
 changed at the same time and adjusted to the standard
 name.
 If no time correction is given, only the waveform
 names are adjusted.
 In addition, channels names can be changed if a
 wavfix.def file is available

 Time correction in seconds, return for no correction
 Input options: (1) filenr.lis or waveform file name
                (2) Nordic file
Filename or number, filenr.lis for all
Input file name: 1994-06-16-1841-57S.TEST__019
```

```
Output file name 1994-06-16-1843-57S._____019

Input file name: 1994-10-04-1324-00S.TEST__016
Output file name 1994-10-04-1326-00S._____016

Input file name: 1994-10-04-1324-24S.TEST__016
Output file name 1994-10-04-1326-24S._____016
```

Note: WAVFIX ONLY works with SEISAN format

WAVFULLNAME

Prints full filename including path for a waveform file by searching directories and databases specified in `SEISAN.DEF`. Filename is to be given on prompt line, e.g. `wavfullname 1996-06-13-1248-15S.NSN__003`.

6.12 File conversion and modification programs

There are mainly two types of files to convert, parameter files with readings and related parameters and binary waveform files.

PARAMETER FILES

CAT_AGA:	Records the S-file header lines according to agency
EDRNOR:	Converts USGS monthly bulletins (EDR files) to Nordic format
GETPDE:	Grap PDE bulletin from USGS web page and add to SEISAN database
GIINOR:	Converts from Geophysical Institute of Israel parameter format to Nordic
HARNOR:	Converts standard Haward fault plane solutions to Nordic format
HYPNOR:	Converts from Hypo71 readings files to Nordic format files
HINNOR:	Similar to HYPNOR for Hypoinverse files
HSUMNOR:	Converts from Hypo71 summary file format to SEISAN format
ISCNOR:	Converts from ISC 96 column format to Nordic format
ISCSTA:	Converts ISC station list to SEISAN station list selecting specific stations.
ISFNOR:	Converts between ISF1.0 and Nordic
KINNOR:	Converts from Kinemetrics to NORDIC
NORGSE:	Converts between Nordic format and GSE parametric format
NORHIN:	Converts from Nordic format to Hypoinverse format
NORIMS:	Converts from Nordic to and from IMS1.0
NORHYP:	Converts from Nordic to HYPO71 format
PDENOR:	Converts a PDE bulletin file to NORDIC format
RSANOR:	Converts Andalucian Seismic Network data to NORDIC format
SEIGMT:	Converts from NORDIC file to input for GMT
SELMAP:	Selects out a part of a MAP file
STASEI:	USGS station file to SEISAN
USGSNOR:	USGS/NEIC CDROM catalog conversion to NORDIC format

CAT_AGA, reordering of CAT file header lines

When plotting hypocenters or doing seismic hazard work, it is the first header line in an S-file or CAT-file

that is used since it is assumed that it is the prime estimate. When making compact files it is also the first header line, which is used. However, there can be a need for resorting the many type 1 header lines for one or several events so that they are ordered according to agency. It could e.g. be needed to put priority on all the ISC solutions, which then should be the first line in the file. CAT_AGA will reorder the type 1 lines in a CAT file according to the order in which the agencies (3 character codes) are given by the user. If there are many agencies, they can be given in an input file named `cat_aga.par`, format is one agency per line in the first 3 columns. If the file is not present, the program will ask the user to enter the agencies manually. The output file `cat_aga.out` will contain the sorted events.

EDRNOR: USGS monthly bulletins (EDR files) to Nordic format
Program to convert USGS weekly EDR files (ftp://hazards.cr.usgs.gov/weekly/mchedr*) to Nordic format. The program is written by **Mohammad Raeesi** (email Mohammad.Raeesi@student.uib.no).

GETPDE, USGS Preliminary bulletin to SEISAN
This program will get the PDE events from the USGS web page and store them in a SEISAN database named PDE. The program is written by **Ruben Soares Luís** (ruben.so.luis@gmail.com). Contact the author for more information or consult our web pare for new documentation.

GIINOR, Geophysical Institute of Israel to SEISAN
The input files are the bulletin type files.

HARNOR, Harvard to Nordic
The standard moment tensor solutions given by Harvard (`http://www.globalcmt.org/CMTsearch.html`) are converted to Nordic format. Only strike, dip and rake is written out.

HYPNOR, converting HYPO71 files to Nordic files
Input is just filename of HYPO71 file. A similar program for HYPOINVERSE files is HINNOR.

HINNOR, converts from Hypoinverse to NORDIC format
This program works like HYPNOR.

HSUMNOR, HYPO71 summary file format to NORDIC format
Note that the program only converts to header lines.

ISCNOR, converting ISC bulletin file to Nordic format
This program works with the ISC fixed 96-column format as e.g. distributed on CDROM (files of type FFB). The program can select out subsets of ISC data using a latitude-longitude window, depth and prime magnitude. Any of the magnitudes Ms and mb are used. Before 1978, there was only mb on the CD's. More detailed selection can be done on the output file later with SELECT. Since the amount of data is very large it is also possible to write out only the hypocenters. Note that ISC now writes in ISF format also, which can be converted with ISFNOR.

Newer CD's have compressed data and cannot be used directly. files must be copied to disk first, decompressed and then handled as single files.

The program will first check if a file with agency codes called agency.isc is present. If so the station codes are read from this file (same format as files on CDROM). The program will also check the beginning of the data input file for a possible list of agencies and station coordinates. If present, the stations coordinates are read and converted to SEISAN format and additional codes read in. The agency codes are needed in order to identify in plain text the various agencies used.

Principles in conversion:

Phases:	The phases out can be either the phase ID's sent to ISC or the ISC reinterpreted phases (given with a number code in the input file). If the user supplied phases are used, parenthesizes are removed, and if P/PKP etc is given, it is replaced by P.
Times:	If day is incremented relative to origin time day, it is carried into the hours, which can be more than 24.
Agency:	It is assumed that it is the same agency for hypocenter and first magnitude. Magnitude is checked for agency, if blank, it is assumed also to be the same as for hypocenter. Only first 3 characters of code is used.
Stations:	Only first 4 characters of code are used.
Depth:	If no error on depth, a depth fix flag is set.
First motion:	Only C or D are used, ISC codes J and B are ignored.
Hypocenter orders:	ISC put the best solution last, here the order is reversed, and the prime estimate is first.
Duration magnitude:	Change D to C for type.
Distance indicator:	If station furthest away is less than 1000 km indicator is L, between 1000 and 3000 km indicator is R and if more than 3000 km indicator is D. If no stations are present the type is set to D.

In order to relocate an event and compare to ISC location, the ISC reidentified phases must be used (option 2, see below). This has the disadvantage that phases not used by ISC (mainly S-phases of local earthquakes) are weighted out in the output file. If option 3 is used, the ISC identified phases are selected if there and if no ISC identification is given, the local reported phase is used. The output file for option 2 and 3 looks the same except that for option 2, the user-defined phases are weighted out. The residuals given in the output file are always relative to the ISC identified phases.

Running ISCNOR:

Below is an example of a run where a latitude - longitude window has been used.

```
 Phases selected can be:
   User reported phases (default=return)          : 1
   ISC identified phases only                     : 2
   ISC identified phases and user reported phases
       when not identified by ISC                 : 3
3
 Output: All hypocenters and phases : Return
         All hypocenters            : 1
         Prime hypocenter           : 2

 Latitude range, return for all
60.2 70.5

 Longitude range, return for all
10, 20

 Depth range, return for all

Magnitude range, return for all
```

```
Write selected events on screen (y/n=return)

 No agency.isc file present
 If ISC CDROM, give drive letter, else return
d

Give first year and month, e.g. 199501 198601
Give  last year and month, e.g. 199602 198602

 Now opening d:\1986\198601.FFB
 Number of agencies in input file        244

etc, for each month


        493 events converted
 Output file name is iscnor.out
 File with stations is isc.sta
```

The file input can be from a CDROM as in the example above. In that case, the whole CDROM can be read or a smaller time interval can be given. The input can also be from a single file and the program will then ask for the next file when the first has been converted. If many files are to be converted, a list of file names can be made with DIRF and `filenr.lis` entered as an input file name. The Nordic format output file is `iscnor.out` and the station list is in `isc.sta` which has the format used by SEISAN. Optionally, output can also be in the original isc format, however that requires setting a flag in the program and recompiling, see program source code.

ISCSTA, selecting stations in the complete ISC station file
The complete station list in the ISC list is very large and it is often an advantage to use a smaller subset, although HYP can use the whole list. The program can select out subsets of stations in both SEISAN and an older ISC format. The program will read an S-file, find how many different stations there are and select those stations out of a station file, which can either be in SEISAN (=HYPO71) format or ISC format (automatically determined). The output is in SEISAN format. If no S-file is given the input station file is assumed to be in ISC format and the whole file will be converted to SEISAN format.

KINNOR, Kinemetrics to NORDIC
Converts .PCK file output of EDPPICK to file in SEISAN format. Many events are converted from one file. The program is based on program from Kinemetrics by **Christopher S. Lim**. For info on how conversion is made, see program source code.

ISFNOR, ISF1.0 to and from Nordic
The ISF format is used by the ISC and is an extension to the IMS format. The program is based on the routines provided by the ISC for reading and writing ISF, and the SEISAN standard routines for reading and writing Nordic data. The program converts in both directions. All possible information is converted. Information on the ISF format can be found on the ISC website (`http://www.isc.ac.uk`). It is recommended to use ISF format for data exchange with ISC.

NORIMS, IMS1.0 to NORDIC format.
The IMS1.0 (International Monitoring System) is a new version of the GSE format and very similar. The program can partly be used for the new ISF (IASPEI Seismic Format) which will include all of the IMS format an additional information needed by ISC and NEIC. The program and the following description is by **Mario Villagrán**. The program works with the IMS1.0:SHORT format (phase-readings/origin files)

and the program works both ways.

IMS1.0:SHORT ⇒ Nordic
Nordic ⇒ IMS1.0:SHORT

The IMS1.0:SHORT format is exactly the one used at the IDC International Data Center (Vienna, Austria). In addition some features used by the ISC International Data Center and the Spanish NDC National Data Center had been added. Magnitudes in IMS format use many characters, the Nordic format allows only one; the following rule is followed:

IMS Nordic
For mb → 'b'
For MS → 'S'
For ML → 'L'
For MD → 'D'
For Ml → 'l'
For MN → 'N'
For mblg→ 'G'
For ms → 'S'
For MB → 'B'

IMSNOR do not include code magnitude.

The maximum likelihood magnitudes mb1, mb1mx, ms1, ms1mx, etc are pending. IDC still does not have documentation and they may be changed.
Single measurements of magnitude/station are parsed as comment lines (type 3) starting with symbol "$". When importing data from IMS format, only the "Event IDC" number is parsed and included into a comment line (type 3) of Nordic, together with the ellipse dimensions orientation and the mb standard deviation.
All parameter values read that exceed field limits of Nordic (Amplitude, velocity, snr, etc) have been set to the maximum or minimum possible, example: if snr > 999.9 then snr=999. For conversion from Nordic to IMS it is necessary to use both the `hyp.out` and `print.out` files; The reason is that IMS includes many parameters that need to be searched in both files.
When converting to IMS format, the user can specify the start numbering for the first event and phase in the file; ignoring will assume (1,1). It is optionally also possible to set the no location flag in the output header lines.

NORGSE, NORDIC from and to GSE parametric format
The program (written by **Mario Villagrán**) converts parametric data between Nordic and GSE2 format. It can be used interactively or by giving the options as arguments. Type `norgse -help` to see the options.

NORHIN, From Nordic to Hypoinverse format
The program is started by typing `norhin input-file`. The output file is norhin.out.

NORHYP, From Nordic to HYPO71 format (SUN and PC)
The program is written by **F. Courboulex**. The program asks for the input file name and the output file name is `norhyp.out`.

PDENOR, converting PDE bulletin file to NORDIC format
PDE distributes bulletins on e-mail, both a monthly bulletin and a weekly bulletin (different formats). The program converts one of these files to Nordic format and put the file into a standard SEISAN database called PDE for the monthly files and PDEWE for the weekly files. This database must have been created before running the program. Both CAT and S-files are made and SELECT and EEV can be used afterwards

RSANOR
Program converts between format used by "Red Sismologica de Andalucia" and a few others in Spain.

SEIGMT, Nordic to GMT input
The program SEIGMT reads information from Nordic or compact files and writes the parametric data to files that can be used as input for GMT(Generic Mapping Tools, `http://gmt.soest.hawaii.edu/`). The user can choose a scaling for the magnitudes and also select a magnitude type order. The scaling option is useful if you wish to scale the symbol size of your epicenters with magnitude. The magnitude type order defines, which magnitude should be taken in case several magnitudes have been determined for one event. If you don't give a magnitude order, the program chooses the largest magnitude.

The files written by SEIGMT are:

`gmtxy.out` - event locations, to be plotted with psxy
`gmtxyz.out` - event locations and depths, to be plotted with psxy
`gmtstxy.out` - station coordinates (longitude, latitude and station code)
`gmtpath.out` - travel path data, to be plotted with psxy
`psmeca.out` - fault plane solutions, to be plotted with psmeca (Aki and Richards convention)

SELMAP, selecting a subsection of a MAP file
The program can retrieve parts of a large MAP file written in SEISAN map format. On the SEISAN web site or on the SEISAN CDROM, very detailed global mapfiles are available in SEISAN format. The file originally comes from the USGS. SELMAP can select out part of a MAP file in a latitude-longitude grid. The MAP files consist of several small segments and a segment is selected if at least one point is inside the specified grid.

STASEI
Converts the official global station file from USGS (comma format) to SEISAN station format (same as HYPO71 format). A list of most global stations are now found on the SEISAN CD.

USGSNOR, USGS catalog to NORDIC format
The program converts USGS CDROM hypocenters to NORDIC format. Most of the information is used. If more than 3 magnitudes are available, only the 3 first are used. The number of stations is included when available. The depth is indicated as fixed in all cases where the operator has been used (A,N,G). Macroseismic information is included with max intensity. The residual standard deviation is put into rms column. Event types are set to R. Magnitude types are converted as follows:

UK is made blank
b is replaced by B
s is replaced by S
D is replaced by C
w is replaced by W

WAVEFORM CONVERSION PROGRAMS

This group of programs are mostly converting waveform files from some format to SEISAN although a few also convert from SEISAN to some other, mostly standard, formats. Most programs convert from binary to binary formats.
Many instruments come with conversion programs to some standard format like PCSUDS or MINISEED, and these have often been used to convert to SEISAN instead of writing programs reading the original files directly. Many such conversion programs work on PC so the corresponding SEISAN programs only work on PC. However, since the PC files can be read directly on Sun, this should not present a problem. Many programs have VERY LITTLE documentation, look in source codes for more information.

The number of programs are forever increasing with new recorders coming onto the market and new

formats coming in use and others going out of use and it is becoming increasingly difficult to keep track of it all. For this release of SEISAN it has not been possible to test all programs on all platforms. An attempt has been made to standardize the programs. A general problem is that many seismic recorders and formats do not provide proper identification of the channels. In the worst cases, there are no station codes, only channel numbers and in very many cases, there is no room for proper component information. This is being taken care of by having a definition file, and only one format for the definition file is used, see below. This is also used with program WAVFIX.
Most programs work in the standard way with a `filenr.lis` file as input (made with DIRF).
The response information is seldom in the original files and in most conversion programs, the response information is taken from the CAL directory. If no response information is available, a message will be given. For each program, a comment will be given as to the status of testing and on which platforms they operate. If the channel definition file option is implemented, the array dimensions will be SEISAN standard.
The program SEIPITSA might be an easy way to convert between 1-column ASCII data and SEISAN (see below).

When converting between the major analysis format (MiniSEED, SEISAN, SAC and GSE) mostly using program WAVETOOL, only SEISAN and MiniSEED will preserve the network and location codes as well as the flag for uncertain timing since the other formats only partly have room for this information.

Conversion programs definition file

The conversion programs use a common format for the definition file for naming station and channels. The definition file is named programname.def as e.g. sudsei.def. The definition file can be in the working directory or the DAT directory. The conversion program will first look in the working directory for the file and then in DAT. The conversion of codes can take place in 2 ways (see below for details): (1) An input station and component code is converted to an output station code and component, (2) an input channel number is assigned a station and component code. The advantage of (1) is that the conversion is independent of the channel number or order, however, the user must then know the default station and component names generated by the conversion program.

Default assignment of station code and component:
This is very much dependent on the conversion program used since some data files have complete

information and others very little, see description of individual programs in manual or at start of source codes. In all cases, the conversion program will make both station and component codes based on what is available of information in the input files. IT IS THESE CODES THAT are used for input code as described below. In order to find out what they are, it is easiest to run the conversion program once (without a def file) and see what codes the program assign. Alternatively, some of the programs have documentation in the manual. Some of the station codes might be instrument serial numbers, which are not always known. Therefore, running a test might be the best way to find out.
In addition to converting channel codes, the def file can also give SEISAN waveform file header information and network code as it appears in the file name. If no network code is given, the network code will be the station code of the first channel.

Principle of conversion in order of precedence:

1. Both station and component given on input: Converted to what is given for output station and component.

2. If both are not present, the channel number is used.

```
Header line text (29 char)... NetCd (5 chars), Comment for next line
```

```
Header for REFTEK             NEWNT
chan stati  comi stato  como, In and output definitions, comment for next line
   1 BO11   S  Z BOM    B  Z
     BO12   S  N BOM    B  N
     BO13   S  E BOM    B  E
```

The first line is just a comment line, must be there in any format. Here it shows where the network code is positioned as indicated by NetCd.

The second line gives the header information for the SEISAN main header, which are the first 29 characters. The file name network code is also given and is here NEWNT. Format a29,1x,a5.

The third line is just comment to indicate the position of the columns in the following lines (max 200). A line must be there. The abbreviations are:

chan:	Channel number, optional unless no input station and component given.
stati:	Input station code, 1-5 chars
comi:	Input component code, 4 characters
stato:	Output station code, 1-5 characters
como:	Outut component code, 4 characters. First character MUST be S, L, B, A, or I, last character MUST be Z, N or E, all upper case.

Format i5,1x,a5,2x,a4,1x,a5,2x,a4

The conversion programs are listed below

AHSEI:	AH ASCII to Seisan, little tested
ARCSEI:	Reftek archive to Seisan, windows only
ASCSEI:	ASCII to SEISAN
BGISEI:	Beijing Geodevice Institue to SEISAN
CITSEI:	CityShark to SEISAN
CNVSSA:	Kinemetrics SSA to Kinemetrics Dataseis
CNVSSR:	Kinemetrics SSR to Kinemetrics Dataseis
DIMASSEI:	USGS Dimas to Seisan
DRSEI:	Sprengnether recorders to SEISAN
GIISEI:	Geophysical Institute of Israel to SEISAN
GSRSEI:	GeoSig to SEISAN
GSESEI:	See WAVETOOL
GSERESP:	Conversion between GSE and SEISAN response files
GURSEI:	Güralp to SEISAN
IRISEI:	From IRIS ASCII to SEISAN
ISMSEI:	ISMES to SEISAN
KACSEI:	Kinemetrics ASCII acceleration to SEISAN
KINSEI:	Kinemetrics Dataseis to SEISAN
K2SEI:	Kinemetrics K2 to SEISAN
LEESEI:	Willy Lee system to SEISAN
LENPCQ:	Converts from Lennartz to PCEQ to PCEQ format
LENSEI:	Lennarts ASCII to SEISAN
M88SEI:	Lennartz MARS88 to SEISAN
MSFIX:	Rewrite MiniSEED files
NANSEI:	Converts from Nanometrics to SEISAN
NEISEI:	Converts from NEIC CDROM waveform data to SEISAN
NRWSEI:	Geol. Survey. of Northrhine-Westphalia format to SEISAN format
OS9SEI:	Converts SEISLOG files to SEISAN waveform files
PITSA:	Conversion programs described with program PITSA
PCQSEI:	Converts from PCEQ to SEISAN
PDASEI:	Geotech Instruments PDAS to SEISAN
PSNSEI:	Public Seismic Networks to SEISAN
QNXSEI:	SEISLOG QNX to SEISAN
RDSEED:	IRIS program to read SEED volumes
RSASEI:	Conversion from Andalucian Seismic network to SEISAN
RT_SEIS:	Reftek Passcal format to SEISAN conversion
SEI2PSXY:	Convert waveform data to trace input for psxy
SEIM88A:	Conversion from SEISAN to MARS88 ASCII format
SEISAN2MSEED:	From SEISAN to MiniSEED
SEISAF:	SEISAN to SESAME ASCII
SEIPITSA:	SEISAN $\Leftrightarrow$ PITSA ASCII
SGRSEI:	SeisGram to SEISAN
SISSEI:	Sismalp format to SEISAN format
SILSEI:	SIL network ASCII files to SEISAN
SUDSEI:	PCSUDS to SEISAN
TERSEI:	Terra ASCII to SEISAN
WGSSEI:	WGSN format to SEISAN

For each program, a summary of capabilities is mentioned: The platforms available (all for all or specific

name), channel definition file available (chan. def. yes or no) and if the program will look for response files in the CAL directory to insert in the headers (resp. yes or no).
If you do not find the conversion program here, look on the ORFEUS website for other programs that might convert to one of the formats used above.
(`http://orfeus.knmi.nl/other.services/conversion.shtml`).

AHSEI, AH ASCII to SEISAN all, chan. def. yes, resp yes
Converts AH ASCII files to Seisan format.

ARCSEI, Reftek archive to SEISAN (windows only)
ARCSEI is a program to automate the extraction of data from a RefTek data archive and the conversion to SEISAN format. The program works interactively and with a simple text interface that asks the user to enter the details for the data extract. Based on the user selected criteria the program (1) extracts the data from the archive in Passcal format using ARCFETCH, (2) converts the Passcal data files to SEISAN format using RT_SEIS, and (3) merges the SEISAN files if merging is activated by the user, using SEISEI. The program is written in Fortran and works on Windows only.

The ARCSEI program can be used in various ways:

- to extract a single time window from one or several stations
- to extract several time windows from one or several stations
- to extract sequential time windows from one or several stations

The ARCFETCH and RT_SEIS programs, both part of the RefTek software package, have to be installed (see RefTek documentation) and the PATH variable set to include the directory where the programs are stored. It is assumed that the RefTek data archive exists and that the user is familiar with the content of the archive. The archive content can be shown with the command ARCINFO. To test that the program is installed correctly, open the Windows command tool (from the menu, or by selecting Start . Run . cmd) and type ARCSEI <RETURN>.

The definition file: `arcsei.def`

The purpose of the definition file is to set some parameters needed to run ARCSEI, however, the program also works without. The arcsei.def file can either be stored in the seismo/DAT directory, or the current working directory. The program first checks in the current directory. The arcsei.def file should be adjusted to the user's set-up, before ARCSEI is started.

The parameters are:

ARCHIVE:	The path of the RefTek data archive, can also be entered manually at run time.
OUTPATH:	The directory in which the SEISAN files are to be stored. The default is '.\' (the current directory).
MERGE:	Select if SEISAN files from several stations for the same time interval should be merged (Y), or not (N).
NETWORK_CODE:	Network code used in case SEISAN files are merged.
CHANNEL:	Data channel in RefTek archive consisting of the unit, stream and channel (unit,stream,channel). The * can be used as wildcard to select all streams or channels, BUT not to select all units (since ARCFETCH is used in cooked mode, which means that the time interval extracted matches the input start- and end-time.

Example of the `arcsei.def` file

```
KEYWORD.....Comments.....Par1.....

ARCHIVE            Path to archive      G:\CTBTO\ARCHIVE
OUTPATH            Path to put          C:\test
                   converted SEISAN
                   files
MERGE              Y or N for           Y
                   merging Seisan files
NETWORK_CODE       used for merging     ARCSE
CHANNEL            Name of channel      8020,1,*
CHANNEL            Name of channel      8021,1,*
CHANNEL            Name of channel      8022,1,*
CHANNEL            Name of channel      8023,1,*
CHANNEL            Name of channel      8024,1,*
CHANNEL            Name of channel      8025,1,*
CHANNEL            Name of channel      8028,1,*
```

How to run the program:

ARCSEI is started from the Windows Command Tool (cmd) by typing

```
arcsei <RETURN>

 ARCSEI - DATA EXTRACTION FROM REFTEK ARCHIVE AND CONVERSION TO SEISAN FORMAT

 SEISAN FILES ARE MOVED TO DIRECTORY: C:\test

 ENTER ARCHIVE, OR <RETURN> TO USE DEAULT (default: G:\CTBTO\ARCHIVE)

Return to accept default, which is set in the arcsei.def file, or give the archive path.

     AVAILABLE CHANNELS ARE:
     8020,1,*
     8021,1,*
     8022,1,*
     8023,1,*
     8024,1,*
     8025,1,*
     8028,1,*

 ENTER CHANNEL SELECTION (UNIT,STREAM,CHANNEL)
       OR TYPE ALL TO USE ALL AVAILABLE CHANNELS
```

Type channel and <RETURN>, if defined in arcsei.def channels are listed, otherwise an example is shown. The channel is given as unit,stream,channel. Wildcards can be used for stream and channel, but not for the unit.

```
NEXT CHANNEL OR RETURN TO CONTINUE
```

Additional channels can be entered, to continue press <RETURN>.

```
ENTER START-TIME (YEAR:DAY-OF-YEAR:HOUR:MINUTE:SECOND)
          EXAMPLES: 2000:200:12
                    2000:200:12:15
                    2000:200:12:33:15
```

Type start time as year:day-of-year:hour:minute:second. Minute and second can be omitted.

```
NEXT START-TIME OR RETURN TO CONTINUE
```

Additional start times can be entered, to continue press <RETURN>.

```
ENTER END TIME USING ONE OF 3 OPTIONS:
- ABSOLUTE TIME AS YYYY:DDD:HH:MM:SS (LIKE START-TIME)
- +SECONDS FOR TIME INTERVAL (e.g. +300)
- ++SECONDS FOR MULTIPLE INTERVALS(CONTINUOUS EXTRACT, e.g. ++300)
```

Specify the end time, either in the same style as for the start time (only if one start time), or in some cases more useful, the desired time window in seconds, by entering +seconds. If sequential time windows are to be extracted, use ++ seconds. The user is then asked how many time windows should be extracted. It is thus possible e.g. to extract 10 consecutive windows of 600 seconds. Only if sequential extract windows specified:

```
ENTER NUMBER OF CONTINUOUS WINDOWS
```

After the program has finished, the data in SEISAN format can be found in either the current directory (default) or in the OUTPATH directory if the variable is specified in `arcsei.def`. The temporarily created files are deleted automatically.

How it works

ARCSEI reads the user input that specifies what should be extracted from the RefTek archive and then calls the programs ARCFETCH, RT_SEIS and SEISEI. For temporary data storage ARCSEI creates the directory arcsei_temp under the current directory. The arcsei_temp directory is automatically deleted upon program completion.

1. Create empty arcsei_temp directory
2. Arcfetch

   The arcfetch program performs the data extraction from the RefTek archive. A complete list of the command line input of arcfetch can be obtained by starting the program without additional options. ARCSEI starts arcfetch in the following way:

   arcfetch archive channel,start-time,end-time -o OUTPATH -c

   Where:

-o OUTPATH: Specifies the output path for arcfetch, always arcsei_temp
-c: Specifies cooked mode, which means that the time interval extracted matches the input start- and end-time (this is not the case, when not running in cooked mode)

Example:
```
arcfetch G:
ARCHIVE 8020,1,*,2000:200:12,+10 -oarcsei_temp -c
```

3. rt_seis

   RT_SEIS converts all files with the suffix 'rt' in arcsei_temp to SEISAN format. RT_SEIS reads the `RTU.INI` file for station definition, if the environmental variable RTU is set to point to the `RTU.INI` file (see RT_SEIS section below).

4. SEISEI

   SEISEI, if merge is selected, merges all SEISAN files in the arcsei_temp directory.

5. move

   Finally all files (single or merged) are moved to the OUTPATH directory or the current directory if OUTPATH is not defined. In case multiple stations are selected, ARCSEI runs steps (1) and (2) in a loop, before the data is merged and moved. In case several time windows are selected, ARCSEI runs steps (1) to (4) in a loop, and in addition a second loop over multiple station (1) and (2). If sequential time windows are specified, ARCSEI computes multiple start times and works as if these time windows were user specified. All, def. File yes, resp yes

ASCSEI, ASCII to SEISAN all, chan. def. yes, resp yes
Converts a single column file to a one channel SEISAN file. Date, time, sample rate, station and component must be entered manually.

BGISEI, Beijing GEODEVICE FORMAT (BGI) to SEISAN. Linux, PC, chan. def. yes, resp yes
The program to convert waveform files from BGI to SEISAN format is called BGISEI. The instrument response in the original files is not used. The program has only been tested with data recorded in Cuba. The program is written by **Bladimir Moreno**.

CITSEI, CityShark to SEISAN all, chan. def. yes, resp yes
Converts from CityShark ASCII to SEISAN. Components S Z, S N, S E are assumed for the 3 channels. Assume 3 channels files only, all channels same sample rate and number of samples.

CNVSSA and CNVSSR Kinemetrics accelerometers to Kinemetrics Dataseis PC
The programs are supplied by Kinemetrics to convert from SSA and SSR formats to Kinemetrics Dataseis. To further convert to SEISAN, use program KINSEI. Only PC executable programs are available. The data is 16 bit.

CSS
At the moment there is no direct conversion from CSS to SEISAN. It is possible to convert CSS data to SAC or GSE using other tools like codeco, Geotool and SAC, and then convert to SEISAN format.

DIMASSEI, USGS DIMAS to SEISAN all, chan. def. yes, resp yes
Converts Dimas files to Seisan format.

DRSEI, Sprengnether data recorders to SEISAN all, chan. def. yes, resp yes
Converts Sprengnether DR3024 and DR3016 to SEISAN format. These two formats are slightly different, but the program makes the adjustment. Only essential information is read in and only 4 lowest digits of

serial number are used. If station codes are set up, these are used, else the serial numbers are used for station codes.

GIISEI, Geophysical Institute of Israel to SEISAN all, chan. def. yes, resp yes
Converts Geophysical Institute of Israel imported DAQ files to SEISAN format. The initial station codes are as defined in file, can be converted with the normal .def file. If 4.th character of station name indicates component (N or E), that is blanked out and transferred to 4.th character of component name BEFORE using the def file conversions.

GURSEI, Güralp to SEISAN all, chan. def. yes, resp yes
Converts Güralp GCF files to SEISAN format, only works with one channel data. Maximum number of samples as defined in seisan, at least 1 200 000, channels codes can be defined using the gursei.def definition file. If no definition file, the station name is the first 4 letters from internal station name and the component is B Z.

GSERESP, conversion between GSE and SEISAN response files all
The program provides conversion between SEISAN, GSE1 and GSE2 response files. The response can be given in frequency, amplitude and phase (FAP) triplets or in poles and zeros (PAZ). Since the number of values in the GSE format is unlimited the conversion from SEISAN to GSE only changes the format, whereas converting from GSE to SEISAN, if the number of FAP triplets is more than 30 or the number of poles and zeros larger than 37, the response in SEISAN format will be approximated by 30 FAP triplets. The output files in SEISAN format will have the default SEISAN response filenames (see RESP program and SEISAN response format). Output files in GSE format will include the station name, the component, number 1 or 2 for GSE1 and GSE2 respectively and end on '.CAL' (e.g. MOR_SHZ2.CAL (GSE2), KONO_BZ_1.CAL (GSE1).

GSRSEI, GeoSig to SEISAN all, chan. def. yes, resp yes
Converts from GBV recorders to SEISAN. GeoSig was earlier GeoSys. Before version 8.1, there was a bug in program so start time was wrong by the amount of the prevent time.

IRISEI, IRIS ASCII to SEISAN all, chan. def. no, resp yes
The input format is the variable ASCII download format used on the GSN Quanterra stations. The format is used in connection with SEISNET. The program only works if input file has more than 1000 samples. ISMSEI, ISMES to SEISAN PC, chan. def. no, resp no ISMES is an Italian seismic recorder. This is the first version of the program made by IIEES in Iran. The program can convert one file with up to 3 channels.

KACSEI: Kinemetrics ASCII acceleration to SEISAN all, chan. def. yes, resp yes
Kinemetrics ASCII film record acceleration files (type *.v1) to SEISAN. It is assumed that:

- channel 1 is N, 2 is Z and 3 is E
- there are always 3 channels in file
- input values are in 1/10 g, the output is in 1/1 000 000 g
- station code is taken from file name as given in first line of input file
- the 3 channels can have different number of samples, however it is assumed that they all start at the same time

KINSEI, Kinemetrics DATASEIS to SEISAN PC, chan. def. yes, resp yes
The program takes the station code from the input files. The component codes are also taken from the input file as far as Z, N and is E is concerned, but the first letter is always set to S, like 'S Z'. The program is also used if CNVSSR or CNVSSA have been used first.

K2SEI, Kinemetrics K2 to SEISAN PC,Linux, chan. def. yes, resp yes
Program for K2 binary files. The program works by first converting the binary files to ASCII by internally running the Kinemetrics program kw2asc (PC only). If no definition file is present, channel 1-3 will be A Z, A N and A E. If more channels they will be called A 04, A 05, etc.

LEESEI, Willy Lee binary files to SEISAN PC, chan. def. no, resp no
The number of channels is fixed to 16 and the time information is not read, it must be entered when converting the file.

LENSEI, Lennartz ASCII to SEISAN all, chan. def. yes, resp yes

LENPCQ, converting Lennartz to PCEQ format PC
Only executable code for this program and only PC (made by the Royal Belgian Observatory). The format is used by an older version Lennartz tape recorder. The output files have the same names as the input files and are placed in a directory c:
qcoda, WHICH MUST BE THERE.

MSFIX MiniSEED to MiniSEED all chan. def. no, resp no
SEISAN might have problems with reading some Steim2 MiniSEED files. Msfix rewrites the file to Steim1.

M88SEI, Lennartz MARS88 to SEISAN all, chan. def. yes, resp yes

NANSEI, Nanometrics to SEISAN PC, Sun, chan. def. yes, resp yes
The program converts from the Y-file format to SEISAN. This is done by first making an ASCII file with Nanometrics y5dump program (done internally in NANSEI). NOTE: The y5dump program requires some special Nanometrics libraries (Solaris) or *.DLL files (PC), which are included and installed with SEISAN (see installation section). The program converts single channel files only.

NEISEI, NEIC digital data to SEISAN PC, chan. def. no, resp no
NEIC earthquake digital data comes on CDROM. The data is extracted with a program coming with the data and then converted to SEISAN binary waveform data. The response information is given as poles and zeros in the SEISAN waveform file header.

OS9SEI, converting SEISLOG files to SEISAN PC, SUN, chan. def. no, resp yes
The program takes a SEISLOG ASCII (downloaded in CMP6 format) or binary file and converts to a SEISAN file. The input can be several files from a `filenr.lis` or an ASCII downloaded file either compressed or uncompressed. The program will look for the calibration file in the CAL directory and add it to the SEISAN file, or give a message if it is not there. The program will work with SEISLOG files recorded under operating system OS9 or QNX up to version 7.6. For QNX version 7.0, use program QNXSEI.

PCQSEI, converting PCEQ format to SEISAN PC, chan. def. yes, resp no
PCEQ format to SEISAN. Earlier used with IASPEI software libraries.

PDASEI, converting PDAS files to SEISAN all, chan. def. yes, resp yes
The program converts a single channel PDAS file to a single channel file in SEISAN format. Several of these files can then be merged with SEISEI. PDASEI in previous SEISAN versions (before version 6.0) only worked with PDAS in 16-bit format, so if 32 bit or gain ranged format was input, the output would have been in error. The current version of PDASEI should be able to convert all 3 types of input files. A description of the PDAS format is found in the PDASEI program.

PSNSEI, Public Seismic Networks to SEISAN all, chan. def. yes, resp yes

The Public Seismic Network recording system makes one file pr channel. Since component is not well defined, several files from the same recording system might get the same SEISAN file name. Do some testing when setting up the recording system. The one component files can be assembled into multichannel files with SEISEI. There might be a newer version of PSN format not supported.

QNXSEI, SEISLOG QNX version to SEISAN all, chan. def. no, resp yes
This program works as OS9SEI except that it does not read the ASCII files. The program must be used with Seislog 8.0. The program is currently the only program that put in the time synchronization flag in SEISAN waveform files except for data logging programs under Seislog Windows. See format description in Appendix B. The program recalculates the sample rate based on the time in the first blocks in the file and the last blocks in the file (each block is one second long). For very long files, this might be of importance since the digitizer might not have exactly the nominal sample rate.

RSASEI, Andalucian Seismic Network to SEISAN all, chan. def. yes, resp yes
Conversion of network and broad band files to SEISAN format. Covers several versions of the DTS format also used by other institutions in Spain. Not tested on Linux.

RT_SEIS, Reftek Passcal to SEISAN PC, chan. def. no, resp no
The RT_SEIS program converts Reftek Passcal format to SEISAN. This program is provided by Refraction Technology Inc. The program does not use the `filenr.lis` as input file. To see the options of RT_SEIS, start the program without any arguments. In order to make use of the `RTU.INI` definition file, the environmental variable RTU needs to be set to for example c:
seismo
dat (see RefTek documentation for more details). This file can be used to set station names for respective unit IDs.

Example of `RTU.INI`:

[8020]
Station=SB00
Network=CTBTO
CH1Band=
CH1Type=
CH1Axis=a
CH1Loc =
CH2Band=
CH2Type=
CH2Axis=b
CH2Loc=
CH3Band=
CH3Type=
CH3Axis=c
CH3Loc =

[8021]
Station=SB01
Network=CTBTO
CH1Band=
...

SEI2PSXY
Converts waveform file to GMT psxy trace plotting ASCII file. The output files have one line for each sample giving the date and time and amplitude value, e.g.:

2005/06/16T00:59:59.51 -40.0000
To plot the trace data with psxy, use projection '-JX<xsize>T<ysize>' and option '-R' giving time range in the same style as the data. To plot the data the gmtdefaults should be set to 'gmtset INPUT_DATE_FORMAT yyyy/mm/dd INPUT_CLOCK_FORMAT hh:mm:ss.xx'. See psxy man pages for more details.

SGRSEI PC, chan. def. yes, resp yes
SeisGram binary to SEISAN. Only 3 component data has been tested. Channel order is assumed to be Z, N, E. The input real values have been multiplied by 100 000 before being converted to integers. Program little tested.

SEED
The Standard for Exchange of Earthquake Data (SEED) format is defined by the Federation of Digital Seismographic Networks (FDSN). The rdseed program is distributed with SEISAN to extract data from SEED volumes. RDSEED is an IRIS program to read SEED volumes. The program provides conversions to SAC (ASCII and binary), AH, CSS and miniseed. It is described in the file 'rdseed.txt' in the INF directory. Updated versions of rdseed will be available at `http://www.iris.washington.edu/pub/programs`. A PC version (rdseed.exe) is distributed with SEISAN CD (also on home page). SEED volumes contain the complete response information, details on how to convert the SEED response to GSE response format can be found in Havskov and Alguacil [2004].

SEIM88A, conversion from SEISAN to MARS88 ASCII format all, chan. def. no, resp no
The program converts SEISAN waveform files to Lennartz-ASCII MARS88 format. The program will write one file per channel. Output files are either mars.xxx if a single file is converted or marsxxx.yyy if the '`filenr.lis`' file is used as input.

SEIPITSA all, chan. def. yes, resp yes
The program converts from SEISAN to PITSA ASCII format and back. The ASCII format has one file per channel. The user will be asked for a name of the output file-system. If a single file is converted, the channel number will be added to the output file-system name (e.g. data.001). If the '`filenr.lis`' file is used the filenumber will be added to the file-system name (e.g. pitsa001.004, first file and fourth channel). The program is no longer used for conversion when PITSA is started from EEV, but might be useful, since it creates one column ASCII data and can easily be modified.

SEISAF, SEISAN to SESAME ASCII all, chan. def. no, resp no
The 3 first channels in SEISAN file are read. There is no check if from same station. It is assumed that the order in SEISAN file is Z,N,E, that all 3 channels have the same start time, number of samples and sample rate. These values are taken from the first trace.

SEISAN2MSEED All chan.def. no resp no
By **Chad Trabant**, IRIS Data Management Center

Program developed at IRIS to convert from SEISAN to mseed, all platforms and all mseed formats. This program can be used as alternative to converting data with wavetool, advantage is that SEISAN2MSEED supports STEIM2 compression.

Source code can be found at `http://www.iris.edu/chad/`

SYNOPSIS

seisan2mseed [options] file1 [file2 file3 ...]

Seisan2mseed converts SeisAn waveform data files to Mini-SEED. One or more input files may be specified on the command line. If an input file name is prefixed with an '@' character or explicitly named '`filenr.lis`' the file is assumed to contain a list of input data files, see LIST FILES below. The default

translation of SeisAn components to SEED channel codes is as follows: a 3 character SEED channel is composed of the first, second and fourth characters of the component; furthermore if the second character is a space and the first and fourth are not spaces an 'H' is substituted for the 2nd character (i.e. 'S_Z' → 'SHZ'). The default SEED location code is '00', if the third character of the SeisAn component is not a space it will be placed in the first character of the SEED location code. Other translations may be explicitly specified using the -T command line option. If the input file name is a standard SeisAn file name the default output file name will be the same with the 'S' at character 19 replaced by an 'M'. Otherwise the output file name is the input file name with a "_MSEED" suffix. The output data may be redirected to a single file or stdout using the -o option.

OPTIONS
"-V "
Print program version and exit.
"-h "
Print program usage and exit.
"-v "
Be more verbose. This flag can be used multiple times ("-v -v" or "-vv") for more verbosity.
"-S "
Include SEED blockette 100 in each output record with the sample rate in floating point format. The basic format for storing sample rates in SEED data records is a rational approximation (numerator/denominator). Precision will be lost if a given sample rate cannot be well approximated. This option should be used in those cases.
"-B "
Buffer all input data into memory before packing it into Mini-SEED records. The host computer must have enough memory to store all of the data. By default the program will flush it's data buffers after each input block is read. An output file must be specified with the -o option when using this option.
"-n Inetcode P"
Specify the SEED network code to use, if not specified the network code will be blank. It is highly recommended to specify a network code.
"-l Iloccode P"
Specify the SEED location code to use, if not specified the location code will be blank.
"-r Ibytes P"
Specify the Mini-SEED record length in Ibytes P, default is 4096.
"-e Iencoding P"
Specify the Mini-SEED data encoding format, default is 11 (Steim-2 compression). Other supported encoding formats include 10 (Steim-1 compression), 1 (16-bit 3 integers) and 3 (32-bit integers). The 16-bit integers encoding should only be used if all data samples can be represented in 16 bits.
"-b Ibyteorder P"
Specify the Mini-SEED byte order, default is 1 (big-endian or most significant byte first). The other option is 0 (little-endian or least significant byte first). It is highly recommended to always create big-endian SEED.
"-o Ioutfile P"
Write all Mini-SEED records to Ioutfile P, if Ioutfile P is a single dash (-) then all Mini-SEED output will go to stdout. All diagnostic output from the program is written to stderr and should never get mixed with data going to stdout.
"-T Icomp=chan P"
Specify an explicit SeisAn component to SEED channel mapping, this option may be used several times (e.g. "-T SBIZ=SHZ -T SBIN=SHN -T SBIE=SHE"). Spaces in components must be quoted, i.e. "-T 'S Z'=SHZ".
LIST FILES

If an input file is prefixed with an '@' character the file is assumed to contain a list of file for input. As a special case an input file named '`filenr.lis`' is always assumed to be a list file. Multiple list files can be combined with multiple input files on the command line.
The last, space separated field on each line is assumed to be the file name to be read. This accommodates both simple text, with one file per line, or the formats created by the SeisAn dirf command (`filenr.lis`). An example of a simple text list:
2003-06-20-0643-41S.EDI___003
2005-07-23-1452-04S.CER___030
An example of an equivalent list in the dirf (`filenr.lis`) format:
1 2003-06-20-0643-41S.EDI___003
2 2005-07-23-1452-04S.CER___030

SILSEI all, chan. def. no, resp no
Conversion from the Icelandic SIL system to SEISAN. Only conversion from ASCII files.

SISSEI, Sismalp to SEISAN all, chan. def. yes, resp yes
The program converts from Sismalp to SEISAN. Sismalp is a French field recording system. The input consists of 2 files pr event, a header file and a data file. It is assumed that the Sismalp ndx files have the same file name as the header file except for the file extension. It is also assumed that the file names are 12 characters long.

SUDSEI, PCSUDS to SEISAN PC, chan. def. yes, resp yes
The program converts from PCSUDS to SEISAN. This is done by first running the program SUD2ASC (included) and then converting to SEISAN. The SUD2ASC program and test data was supplied by REFTEK through the distribution of PC-SUDS Utilities by Banfill [1996].

TERSEI, Terra ASCII to SEISAN all, chan. def. yes, resp yes
Program converts from Terra Technology ASCII files to SEISAN. Only tested with 1-3 channel files

WGSSEI to SEISAN all, chan. def. yes, resp yes
Program converts from WGSN files to SEISAN. The format is used on IRIS stations as processing format. Little tested.

6.13 Signal processing programs

The two processing systems PITSA and SAC are interfaced to SEISAN and can be started directly from EEV. This is done since both systems support functions that SEISAN does not have. They only operate on Unix. The Degtra program only operates in Windows. It is not interfaced directly to SEISAN but operates on SEISAN format.

6.13.1 PITSA

PITSA (Programmable Interactive Toolbox for Seismological Analysis) is a program written by **Frank Scherbaum** and **James Johnson**. The program is included in the SEISAN package, updated versions are available at `http://lbutler.geo.uni-potsdam.de/service.htm`. From this version, PITSA is interfaced with the SEISAN system through the program WAVETOOL, which converts waveform files from SEISAN to the GSE2 format. PITSA since version 5.0 supports reading multi channel GSE2 files. PITSA can be started from EEV by typing 'pitsa' on the prompt line. All waveform files listed in the S-file will be converted to multi-channel GSE2 files. The multi-converted files are put into your local directory and are named 'gse1', 'gse2' etc. The response is converted to GSE format. When PITSA is

started, the waveform files have to be loaded using the GSE2 input format. The response file names will be given as described in the GSERESP section.

6.13.2 SAC2000

SAC2000 (seismic analysis code) is currently developed by **Lee Minner** and **Peter Goldstein** [Goldstein, 1999]. SAC is not distributed with SEISAN, information on SAC can be obtained from the SAC homepage (`http://www-ep.es.llnl.gov/www-ep/esd/seismic/sac.html`). The main features of SAC include general arithmetic operations, Fourier transforms, three spectral estimation techniques, IIR and FIR filtering, signal stacking, decimation, interpolation, correlation, and seismic phase picking. SAC also contains an extensive graphics capability. With SAC it is possible to write macros, which helps to process large amounts of data. The SAC format is used in several research oriented programs. SAC can be started from EEV using the command 'sac'. EEV will start the WAVETOOL program to convert the data to SAC and then execute the command sac. In case your sac executable is called sac2000, it is necessary to rename it (to sac) or alternatively to create a link in either the SEISAN PRO directory or the SAC bin directory. This is done for example by the command :

```
ln -s /sac/bin/sac2000 /sac/bin/sac
```

Since the SAC format is a single trace format, the SEISAN multichannel files are split into single trace files. The station and component names are included in the file name and the suffix 'SAC' is added to all SAC files. For both systems, waveform data can be converted to the respective format outside EEV using WAVETOOL, GSESEI or SACSEI, and the programs can be started without using EEV.

6.13.3 Degtra A4

Degtra A4 is a Windows-based computer system designed to process ground-motion time histories for seismology, engineering seismology and earthquake engineering applications. It has been developed at the Institute of Engineering, UNAM, Mexico, where it has been used as a professional and teaching tool for more than 10 years. Figure 6.21 shows a typical view of Degtra A4.

Degtra A4 accepts input ground-motion recordings written in several formats: simple ASCII or binary files, SEISAN format and the standard format of the Mexican Strong-Motion Database.

The program has a nice and user-friendly interface, in which several time-histories can be manipulated at a time. It includes zoom-in/zoom-out capabilities that can be used if only a portion of a recording is of interest.

There are two types of operations that Degtra A4 can perform using strong-motion recordings: operations involving a single recording and operations involving two recordings.

Among the first group, the main functions are the following:

- Scaling. Multiplies the recording by a constant
- Decimation. Decimates the recording by a user-given factor
- Base-line correction: Includes several methods.
- Differentiation: Numerical differentiation with respect to time.
- Integration: Numerical integration with respect to time.
- Computation of Arias' intensity.

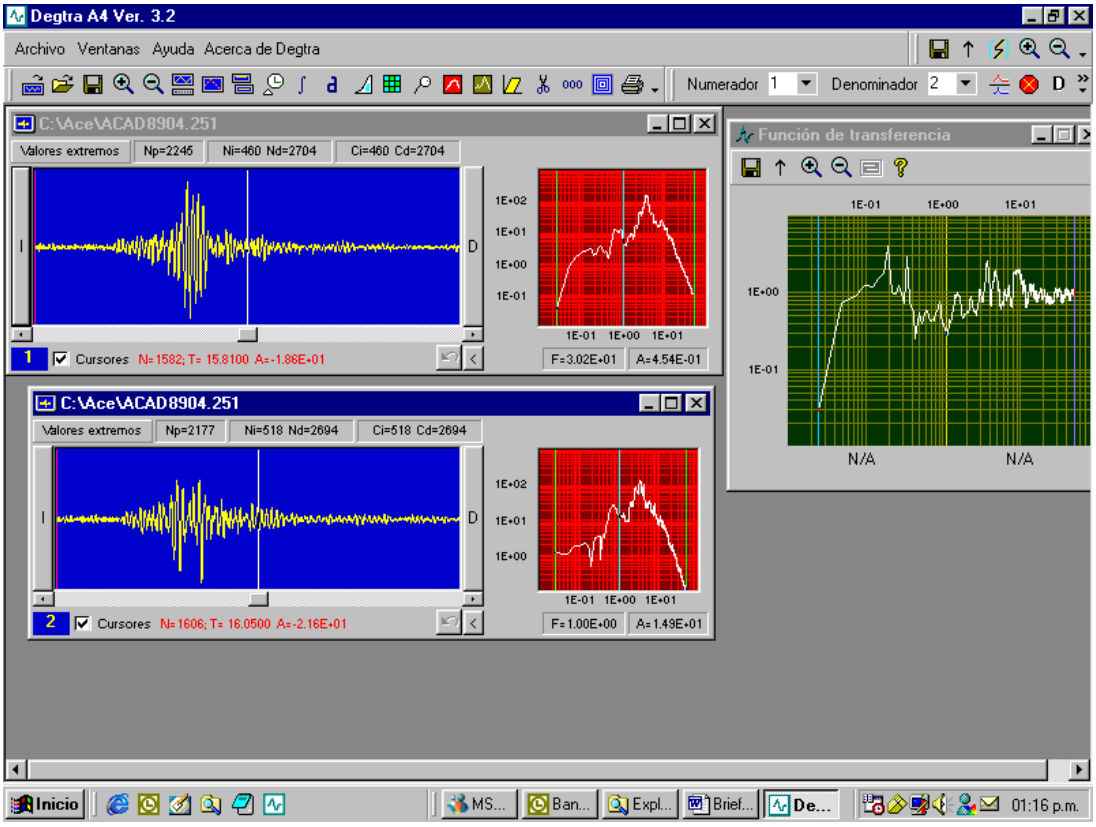


Figure 6.21: Example of DEGTRA A4.

- Filtering: Includes high-pass, low-pass, band-pass and band-stop Butterworth filters, as well as Gaussian and Futterman filters.
- Computation of elastic response spectra. Computes absolute acceleration, relative velocity, relative displacement and pseudo acceleration response spectra.
- Computation of inelastic response spectra. Computes required-strength response spectra for fixed ductility demand of bilinear systems.
- Computation of Fourier spectra. Computes and (optionally) smoothes FFT of a time signal.
- Computation of response (acceleration, velocity, displacement) of elastic and inelastic (bilinear) single-degree-of-freedom oscillators.
- Computation of the S-wave response of a soil column of given properties assuming that the time-history is the incident wave field at the base of the column.

The main operations involving two recordings are:

- Spectral ratio. Computes the (Fourier) spectral ratio between two recordings.
- Odogram. Displays an X-Y parametric representation (time is the parameter) of two recordings, typically ground displacements, in order to observe particle trajectories.
- Addition/Difference. Computes the sum or difference of two recordings. This is useful, for instance, to identify rocking or torsional motions in buildings.
- Rotation: Rotates two components of the ground motion to arbitrary angles.
- Cross-correlation: Computes cross-correlation between two signals.
- Coherence: Computes coherence between two signals.

Degtra A4 includes an on-line help file with details about the meaning of required parameters, techniques used, and use of Degtra A4 itself. It is available in two versions: one for Windows 2000 or lower and another for Windows XP. Installation is made with a typical Windows executable setup file. The distribution is found in directory SUP.

Degtra A4 is currently in Spanish, English version in preparation.

For questions please contact:

Mario Ordaz, Institute of Engineering, UNAM
mors@pumas.iingen.unam.mx

6.14 Calculating b-value, BVALUE

BVALUE is a program to make b-value plots using a NORDIC input file (also compact). A postscript plot file is generated.

The questions are:

```
 Input file name, select.out or collect.out are defaults
    ! Give filename or return
 Which magnitude type, C,L,B,W or S, return for no type
    ! C: coda, L: Ml, B: mb and S: surface wave magnitude, W: Moment mag.
         blank: no magnitude type
 Output:
 Number of events selected from file:    91
Duration of catalog in years: 0.502

    ! Output number of events selected and duration of catalog
 New input:
 Magnitude step 1.0, 0.5, 0.25, 0.2 or 0.1
    ! Magnitude step for summing number of events, MUST be one of the above.
 Magnitude range m1,m2 for b value and fixed b-value
    ! Range for calculating b value, and the fixed b-value for which a-value is calculated. The

 Output is now:
      n    m1    m2 maxl a maxl b   sd  lsq a  lsq b   cor    rms   bfix   afix   sd
     52    2.0  4.0 3.25     0.68 0.46   3.77   0.93  0.93   0.14    1.0   4.02 0.16
 Normalized         3.55                 4.07                              4.32
 Normalized m1      2.19                 2.21                              2.32
 Norm. lin. m1     154.9                162.2                             208.9

! Normalized means normalized to one year (m=0), Normalized m1 (m=m1) and Norm. lin. m1 is just

   mag  nmag  cmag
    0.4     1    91
    0.6     1    90
    0.8     3    89
    1.0     6    86
    1.2     6    80
    1.4     7    74
    1.6     4    67
    1.8    11    63
    2.0     8    52
    2.2    11    44
    2.4     6    33
    2.6     7    27
    2.8     5    20
    3.0     4    15
    3.2     7    11
    3.4

 n: Number of values used
 m1 and m2: Magnitude range used for b value calculation
 maxl a and b: Maximum likelihood a and b
 sd: Standard deviation
 lsq a and b: Least squares a and b ( the one plotted)
```

```
 cor and rms: Correlation coefficient and rms of above
 bfix: fixed b-value given at input
 afix: a-value for above
 sd: standard deviation for above
 mag: magnitude
 nmag: number of events within mag interval
 cmag: cumulated number of events

 Questions:

 Plot(y/n)
   ! This will make a plot and a postscript plot file.
     Note that only the last plot on screen is saved. The plot
     must be sent manually to the printer.

 Which b-value 1: Least squares (default)
               2: Fixed bvalue
               3: Maximum likelihood
   ! Choice of b-value to plot

 Run again with other parameters (y/n)
   ! Another step length or range can be selected

The final output is:

Output file in bvalue.out
Last plot in file bvalue.eps
```

The output file bvalue.out contains the same information in the same format as shown in the example above. The file can be used with other plotting programs to make 'nicer looking' b-value plots. An example is shown in Figure 6.22.

6.15 Automatic phase picking, AUTO, AUTOPIC, AUTOSIG, CONDET

6.15.1 AUTO and AUTOPIC

AUTOPIC is a tool to automatically pick phases on events registered into the database. The AUTO program will go through a series of events in the usual way using start time and end time and start AUTOPIC for each event. If an event file (S-file) has any readings, the program will not reread in order to not destroy old picks. The automatic readings in the file are marked with an A after the weight column to indicate automatic pick. Each pick is evaluated by using the signal to noise ratio and an indication of the quality is given with the weight. The program will run on all waveform files given in an S-file. Each time the program runs, there is a file called autopic.out containing information about the run. If there are any 3-component stations, an azimuth will also be calculated, and the S-phase will be more reliable. The pick program can also be used from EEV by typing Z (will run program AUTOPIC). When

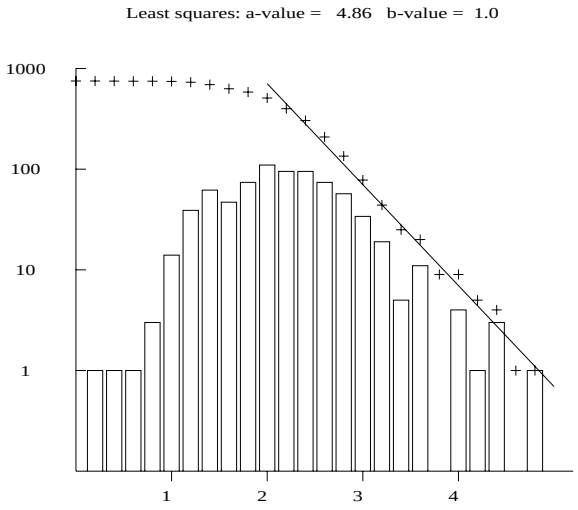


Figure 6.22: An example of a b-value plot. The bars are number of events and crosses the accumulated number of events.

it is used from EEV, there is always an output in the S-file, which will be grouped at the bottom of the file, making it possible to compare manual and automatic readings. THE S-FILE MUST THEN BE EDITED MANUALLY IN ORDER TO REMOVE DOUBLE READINGS. The program requires an input parameter file in the working directory or DAT with the name AUTOPIC.INP. The program will first look in the working directory. The parameters in that file are explained in file AUTOPIC.INF. Both files are shown below. The program uses a 4-pole filter running one way. This might result in phases being picked a bit late. However, it seems more accurate than the earlier version where the filter run both ways and picks were often far too early. The program is made mainly by Bent Ruud. For more information about how it works, see Ruud et al. [1988]; Ruud and Husebye [1992]. Description of parameters

```
%
% Input parameters common to all filters:
%
% LWIND  : used to define step length (DELTA=WINDOW/LWIND)
% ISHIFT : defines time shift between STA and LTA window (ISHIFT*DELTA)
%          Delay for LTA window (15 * 0.1) ==> 1.5 sec. after STA window.
% ISIGMA : defines fall off rate of LTA window (larger values   longer windows)
%          LTA(i) = (1   2^(isigma)) * LTA(i 1) + 2^( isigma) * STA(j)
% COHMIN : Polarization threshold.
%          Minimum coherence (see thresh_1 and thresh_2)
% NDMIN  : Mimimum number of consecutive triggered windows in a detection
% SVELO  : S wave velocity of the medium below the station (used for 3 comp)
% NFILT  : number of filters
% CRAT   : Ratio for calculation of coda duration ( range 1   4)
% LWIN   : Window used in coda duration routines (range 20   50 seconds)
% THRES  : Quality threshold (range 2   5). Used on the maximum to average
%          amplitude ratio in order to sort out the most noisy traces.
%
% Input parameters defined for each filter:
%
% WINDOW : length of the moving time window (sec)
% F1     : lower cutoff frequency (Hz) of band pass filter
% F2     : higher cutoff frequency (Hz) of band pass filter
% THRSH1 : STA/LTA threshold for polarized signals
% THRSH2 : STA/LTA threshold for unpolarized signals
%          If coherence > cohmin then detection is made on thresh_1
%          If coherence < cohmin then detection is made on  thresh_2
%
% Output parameters:
%
% D    : day of year
% H    : hour
% M    : minute
% SEC  : second
% DUR  : duration, i.e. time in detection state (sec)
% FRQ  : centre frequency of filter giving the best detection (Hz)
% SNR  : signal to noise ratio (SNR=STA/LTA)
% STA  : short time average (root mean square of amplitude)
% NT   : total number of triggered time windows in the detection
```

```
% NH   : number of windows with best SNR on one of the horizontal comp.
% NV   : number of windows with best SNR on the vertical comp.
% NC   : number of windows with acceptable polarization
% Q    : quality class, 1(best)   4(worst)
% PS   : P/S wave discriminator, 0(S)   10(P)
% AZI  : backazimuth in degrees measured from North through East
% DA   : variability in azimuth (deg)
% VEL  : apparent velocity (km/s)
% DV   : variability in apparent velocity (km/s)
%
% Note : azimuth and apparent velocity calculations are based on the
%        assumption of P wave, so that these variables should be
%        neglected for S waves.
%
```

Example of input file AUTOPIC.INP for AUTO

```
% This is the parameter file needed by program:     AUTO
%
% The following rules apply:
% 1. All lines with % in the first column are comment lines
% 2. Lines with a blank in column 1 are read for fixed parameters.
% 3. All lines starting with "filter_x", where x is a number,
%    are read for filter variable parameters
% 4. All lines with * in the first column are read for stations to process
% 5. A brief explanation of all parameters is given in preprocess.inf
%
% FIXED PARAMETERS THAT ARE USED THROUGHOUT THE PROGRAM
%
% Lwind Ishift Isigma Cohmin  Ndmin  Svelo  Nfilt   Crat   Lwin  Thres
%     !      !      !      !      !      !      !      !      !      !
%
    4.0   15.0    6.0   .4   .0   2.75    3.0    1.6   30.0    3.0
%
%
% PARAMETERS THAT ARE FILTER DEPENDANT
%
%Filter_nr    Window        F1        F2    Thrsh1    Thrsh2
%        !         !         !         !         !         !
%
filter_1        .0       1.0       4.0      1.70      2.55
filter_2         0.6       4.0       8.0      1.80      2.60
filter_3         0.4       8.0      16.0      1.90      2.75
%
%
% STATIONS TO USE IN THE PROCESSING
%
*SUE  S  Z
```

```
*BER  S  Z
*HYA  S  Z
*KMY  S  Z
*ODD  S  Z
*ODD1 S  Z
*BLS  S  Z
*BLS1 S  Z
*ESG  S  Z
*EGD  S  Z
*KTK1 S  Z
*NSS  S  Z
*MOL  S  Z
*JNW  S  Z
*JNE  S  Z
*FRO  S  Z
*JMI  S  Z 3 component
*ASK1 S  Z 3 component
*ASK  S  Z 3 component
*MOR7 S  Z 3 component
*LOF  S  Z 3 component
*OSG  S  Z 3 component
*TRO  S  Z 3 component
*FOO  S  Z 3 component
```

6.15.2 AUTOSIG

AUTOSIG is a program to perform some automatic processing. The program includes routines for P-phase picking, determination of signal duration, amplitude determination, determination of spectral parameters [Ottemöller and Havskov, 2003] and determination of distance type (local, teleseismic, noise). The program can still do with improvement. The input to the program can be either a parametric Nordic file (with one or several events) or waveform files. In both these cases, the output is written to the `autosig.out` file. Additional output files are `autosig.trace` and `autosig.err`, which will help to find potential problems. Alternatively, the program can also be started from EEV; the output is then directly written to the S-file. The input parameters are defined in the file `autosig.par`, which is located either in the DAT or the working directory.

Following are descriptions of the automatic processing routines:

P-phase picking:
The phase picking is based on changes in the STA/LTA ratio. A band pass filter can be specified. The routine gives options to enhance the changes in the signal before computing the STA/LTA ratio. It is recommended to first remove the linear trend and then to compute the characteristic function which is given by y**2+k*(dy/dt)**2, which enhances changes in both amplitude and frequency content. Then the STA/LTA ratio is computed to detect changes in the signal. The routine can also compute the squared STA/LTA. When a change is detected (STA/LTA ratio above trigger level), it is tested whether the signal spectral amplitudes are significantly higher (factor of 2 in amplitude) than the pre-signal noise spectral amplitudes. This is done to avoid triggering on spikes.

Signal duration:
The signal duration is determined by comparing the signal amplitudes with the amplitudes of the pre-

signal noise. The duration is determined by the point from which the ratio of these amplitudes is lower than a given value. A filter is applied if specified in the parameter file.

Amplitude:
Routine finds maximum amplitude between two peaks.

Spectral parameters:
The routine computes the displacement amplitude spectrum for P or S waves (see section 6.2.12) and, using either a converging grid search or a genetic algorithm determines the seismic moment and the corner frequency by minimizing the difference between observed and synthetic source spectra. The frequency band is determined by comparison with the pre-signal spectrum. The grid search is generally more cost effective and produces better results. The method is described in Ottemöller and Havskov [2003]. The displacement spectrum is corrected for geometrical spreading and attenuation (both along the travel path and near surface). Therefore, the hypocentral distance has to be known. The time domain window for extracting the data from the trace can be given by either a group velocity (Vg=distance/travel time) window or a fixed window in seconds around the phase pick.

Distance type:
The routine determines whether the signal is from a local or teleseismic event, or noise. If signal spectral amplitudes are not significantly higher than pre-signal noise amplitudes, it is assumed that the signal is noise. Otherwise the amplitudes at two selected frequencies given by 'DIST FREQ SELECT' are compared, the rules are (f1<f2):

Spec signal amp(f1) - Spec noise amp(f1) > Spec signal amp(f2) - Spec noise amp(f2): teleseismic
Spec signal amp(f2) > Spec noise amp(f2): local

There are a few command line options that can be used to run autosig in non-interactive mode, syntax is

```
autosig -infile <filename> [-spec on/off -phase on/off -clear on/off]
```

where

-spec on/off: determine spectral parameters if option given
-clear on/off: remove phases from input S-file before start if option given
-phase on/off: detect phases if option given
-infile <file>: give name of input file, either S-file or waveform file
-help: get help

Note: When running the program the first time and the hypocenter location is not known, the determination of spectral parameters is not done. To run the determination of spectral parameters, the hypocenter location has to be given in the S-file.

The meaning of most parameters in the parameter file is clear from the keyword. The spectral parameters are as described in the MULPLT section. Other parameters that need explanation are:

AUTO PHASE, AUTO SPECTRUM and AUTO AMPLITUDE: Logical flag to activate phase picking, spectral analysis and amplitude reading, respectively (1. for true)

GA POPULATION SIZE: Number of elements in the population, used only if SEARCH ALGORITHM is 1.

GA GENERATIONS: Number of generations in one run, used only if SEARCH ALGORITHM is 1.

Note: Increasing GA POPULATION SIZE and GA GENERATIONS will increase the computation time.

GRID NLOOP: Number of loops in converging grid search for spectral parameters, used only if SEARCH ALGORITHM is 2. Resolution increases with every loop.

NGRID FREQUENCY: Number of grid points in search for corner frequency, used only if SEARCH ALGORITHM is 2.

NGRID SPECTRAL AMP: Number of grid points in search for spectral amplitude, used only if SEARCH ALGORITHM is 2.

NORM: Norm for computation of residuals in spectral fitting can be set, however, tests show that 1 or 2 produce the same result, and generally default of 1 can be used.

SEARCH ALGORITHM: Defines whether genetic algorithm (1) or converging grid search (2) should be used. Converging grid search is recommended.

SELECT PHASE: Defines, which phase to use for spectral analysis, choices are: 0 for P by AUTOSIG, 1 for computed P arrival for given location, 2 for computed S arrival, 3 for P from s-file, 4 for S from s-file or 5 for S or P from s-file.

SEPCTRUM F LOW: Lower limit of frequency band to be used.

SPECDURATION CHOICE: The time window for computation of the spectrum can be given either as a time window starting from the phase onset (0.) or can be defined by a group velocity window (1.).

SPECTRUM P LENGTH: Duration in seconds of signal starting from P arrival.

SPECTRUM S LENGTH: Duration in seconds of signal starting from S arrival.

SPECTRUM PRE LENGTH: Duration in seconds of signal to be included prior to phase arrival.

GROUP VEL WINDOW P: Range of group velocities defining time window to be used for P spectrum. Time window is given by (distance/group velocity)

GROUP VEL WINDOW S: Range of group velocities defining time window to be used for S spectrum. Time window is given by (distance/group velocity)

STALTA NREC/REC: There are two STA/LTA algorithms, recursive (0.) and non-recursive (1.).

STATION LINE: One line with processing parameters for phase detection is given for each channel. The parameters are (also see example below):

STAT - station name
COMP - component name
STA - duration of STA
LTA - duration of LTA
RATIO - trigger ratio
MINCOD - minimum coda required for trigger
DTRLE - de-trigger level
FILL - bandpass filter low cut
FILH - bandpass filter high cut

Example of the parameter file `autosig.par`:

```
#
KEYWORD.....Comments.....Par 1.....Par 2

#
# spectral parameters
#
```

```
SPECTRAL S-Q0      Q0                   440.0
SPECTRAL P-Q0      Q0                   85.0
SPECTRAL S-QALPHA  Q = Q0**Qalpha       0.70
SPECTRAL P-QALPHA  Q = Q0**Qalpha       0.70
SPECTRAL KAPPA                          0.02
SPECTRAL P-VELOCITY P velocity          6.2
SPECTRAL S-VELOCITY S velocity          3.6
SPECTRAL DENSITY   Density              2.8

#
# auto signal processing parameters
#
REMOVE MEAN         1. for true         0.
REMOVE TREND        1. for true         1.
CHAR FUNCT          1. for true         1.
K IN CHAR FUNCT     K IN X=Y^2+K*Y'^2.  3.
STALTA NREC/REC     rec 0./ non-rec 1.  1.
STALTA SQUARE/ABS   square 1.           1.
AUTOCODA SQUARE/ABS square 1.           0.
AUTO PHASE          1. for true         0.
                    only if no phase
AUTO SPECTRUM       1. for true         1.
AUTO AMPLITUDE      1. for true         1.
AUTO LOCATE         1. for true         0.

NORM                                    1.
SEARCH ALGORITHM    1.=GA 2.=GRID       2.
#
# window selection
#
SPECTRUM P LENGTH   in seconds          5.
SPECTRUM S LENGTH   in seconds          2.
SPECTRUM PRE LENGTH in seconds          .5
GROUP VEL WINDOW P                      5.0       6.5
GROUP VEL WINDOW S                      2.7       3.7
SPECDURATION CHOICE 0:SPEC. P/S LEN.    1.
                    1:GROUP VEL W. P/S

#
# select phase
#
SELECT PHASE        0:auto  P           2.
                    1:synth P
                    2:synth S
                    3:P from file only
                    4:S from file only
                    5:S or P from file
     --- for 3-5, if no phase and AUTO PHASE is 1. use auto phase pick ---
DIST FREQ SELECT                        1.        10.
```

```
#
# parameters used in the genetic algorithm searching for spectral parameters
#
GA POPULATION SIZE                      50.
GA GENERATIONS                          250.

#
# grid dimension in case of grid search
#
GRID NLOOP                              5.
NGRID FREQUENCY                         100.
NGRID SPECTRAL AMP                      100.

#
# low filter limit to use for auto spectrum
SEPCTRUM F LOW                          .05

#
# parameters controlling output
#
CREATE WAVEOUT                          1.
SPEC OVERWRITE                          0.
PHASE OVERWRITE                         0.

#
# station parameters
#
#       STAT- COMP -sta-- -lta-- -ratio mincod -dtrle fill filh
STATION MOL   S  Z    3.0   20.0   10.0    2.5    1.5  5.0 10.0
STATION MOL   B  Z    3.0   20.0   10.0    2.5    1.5  5.0 10.0
STATION HYA   S  Z    3.0   20.0   10.0    2.5    1.5  5.0 10.0
STATION LSA   L  Z    3.0   20.0   10.0   10.0 9999.0 00.1 10.1
STATION CHTO  L  Z    3.0   20.0   10.0   10.0 9999.0 00.1 10.1
STATION XAN   L  Z    3.0   20.0   10.0   10.0 9999.0 00.1 10.1
```

6.15.3 Detection program for continuous data, CONDET

The CONDET program is a detection program works on data that is organized in a SEISAN continuous database. It performs a detection process similar to real-time processing systems, but of course the data is already there. The program works in two steps, first to run a detector on a single channel, and second to detect events that are on more than a minimum number of stations. Possible applications are processing of data from a temporary deployment (e.g., aftershock monitoring, where continuous were recorded without event detection) and adjustment of detection parameters used in real-time monitoring.

The program has three built-in detection algorithms: 1) standard squared STA/LTA, 2) Carl Johnson's detector (that is for example used in the Earthworm processing system (`http://folkworm.ceri.`

memphis.edu/ew-doc/) and 3) correlation with master event. The program writes out a list of detections (file `condet.out`, which gives station name, component code, trigger time and trigger duration), but also a batch file that can be used to extract the corresponding event data from the continuous data (file extract.batch). Note that wavetool by default takes data from all continuous databases listed in `SEISAN.DEF`.

When started without any command line options, the program works on all stations/databases given by the STATION parameter. The output file has detections from all stations, and the extract.batch file has extract commands for all detections. This is all required if only one station is available. For more than one station, it is possible to search for times at which more than a minimum number of stations have triggered. This is done by starting the program with the command line argument '-net'. In this mode, the output file `condet.out` from the first run is used and the file extract.batch is overwritten. The extract script can now be used to get data for the network detected events.

The input parameters are given in `condet.par`:

STATION: give continuous database name, station and component code

```
STATION                                LICOC       LICO  HH Z
STATION                                LIGLC       LIGL  HH Z
```

START DATE and STOP DATE: give time interval, can be larger than data availability

```
START DATE          yyyymmddhhmmss        200802270000
STOP DATE           yyyymmddhhmmss        200803122359
```

WAVEOUT: Set to 1. to write out waveform files with the original data and trigger channels.
EXTRACT DURATION: Length of extraction window in seconds, used in `extract.batch`
PRE EVENT TIME: Time to start extract before detection time in seconds, used in `extract.batch`
INTERVAL: Length of data segment read at a time. The default is 60 minutes.

DET ALGORITHM: choices for the detection algorithm are STA for squared STA/LTA, COR for correlation and CAR for Carl Johnson's detection algorithm

MIN TRIG DURATION: Minimum duration the trigger level needs to be exceeded for
MIN TRIG INTERVAL: Only allow for one detection within this time, given in seconds
FILTER LOW: Low cut for bandpass filter
FILTER HIGH: High cut for bandpass filter

If DET ALGORITHM is STA:

STA LENGTH: Short term duration in seconds
LTA LENGTH: Long term duration in seconds
TRIGGER RATIO: Ratio of STA/LTA required for trigger
DETRIGGER RATIO: Ratio to detrigger
FREEZE LTA: LTA can be frozen at time STA/LTA goes above TRIGGER RATIO, 1.=to freeze

If DET ALGORITHM is CAR, see Earthworm documentation for details:

CARL RATIO
CARL QUIET

If DET ALGORITHM is COR:

CORRELATION MIN: Minimum correlation between waveforms of master event and the data required for a trigger
MASTER WAVEFORM: Name of waveform file that is used as master event, the master event is cross-correlated against the continuous waveform data

Network detection parameters:

NET MIN DET: Minimum of detections required from different stations with time window given by
NET WINDOW SEC: Time window for network detection in seconds.

6.16 Fault plane solution

SEISAN includes five programs for estimating the fault plane solution, FOCMEC, FPFIT, HASH (called hash_seisan),EBEL and PINV. The programs are described below, they can all be called from EEV. Fault plane solutions can be plotted with EPIMAP (new from version 9.0), W_EMAP (Windows only) and FOC (see below). The program GMTNOR also makes output which can be used with GMT. In EEV fault plane solutions can be added manually to an event with commands inputfps or ifp.

6.16.1 FOCMEC

The program can be used to determine double couple earthquake focal mechanisms using polarities and/or amplitude ratios for both local and global earthquakes. The program also provides an interactive graphical display. The existing solution can be plotted without any station data or location being available, however if existing polarities should be plotted, the event must be locatable in order to calculate angles of incidence. Several solutions can be plotted on the same figure in order to compare solutions.

The SEISAN program FOCMEC provides the interface between the database and the program that determines focal mechanisms, which in SEISAN is the program FOCMEC_EXE. This program is written by Arthur Snoke [Snoke et al., 1984] and distributed as part of the FOCMEC package (`http://www.geol.vt.edu/outreach/vtso/focmec`). FOCMEC_EXE is identical to FOCMEC in Snoke's package and can be easily upgraded (unless formats are changed). Generally the user will use FOCMEC when working with SEISAN data, however, it is also possible to run the original version (see documentation by Snoke: INF/focmec.pdf). Before FOCMEC_EXE is started the user can optionally change the inputfile `focmec.run`.

The program works with polarities and amplitude ratios. See the MULPLT section on how to read polarities and amplitudes. Note that since amplitude ratios are used, there is no need to correct for instrument response provided the response is the same for the different components (within 5-10 %).

Use of amplitudes

Amplitude ratios are computed from amplitude readings given in the S-file. While amplitude ratios can provide additional constraint on the solution, they should be used with caution. Ideally, the solution should be well constrained by polarities only, and then amplitude ratios can provide confirmation of a solution or help to select one of several equally good solutions. The principle behind the amplitude ratio method is that the effect of geometrical spreading will cancel out when forming the amplitude ratios of S and P waves (or SV/SH) of the same phase type, e.g. Pg and Sg. This leaves the following corrections to be made on the amplitudes before the ratios are calculated.

- Calculate angle of incidence at the station and correct for the free surface effect.

- For local earthquakes, use the calculated travel time for a particular phase to correct for Q. Different Q for P and S can be used and the frequency used is the frequency of the maximum amplitude phase.
- For distant earthquakes, correct for t^*. Different t^* for P and S can be used. The frequency used is the frequency of the maximum amplitude phase.

The attenuation parameters have default values of:

$Q = 100 \times f^{1.0}$	for P and S-waves
$t^* = 1.0$	for P-waves
$t^* = 4.0$	for S-waves

Different values can be set in file FOCMEC.DEF, which can be located in DAT or working directory.

The observations to be made are:

- Rotate the seismogram (if three component record) to get R and T components.
- Read maximum amplitude P-phase and corresponding period on Z, phase P.
- Read the maximum amplitude S-phase (same type) and corresponding period on Z, phase (SV).
- Read the maximum amplitude S-phase (same type) and corresponding period on H, phase (SH).

The wave type Pg/Sg or Pn/Sn has to be given when the amplitude is read. When reading on uncorrected seismograms, MULPLT will want a confirmation that the user wants to save uncorrected amplitudes, since, normally, all amplitude observations in an S-file are in nm. It is possible to filter the signals provided the same filter is used for P and S. Ideally, the amplitude observation should be made at a frequency below the earthquake corner frequency and consequently also the filter high cut frequency should be below the corner frequency.

It is also possible to read amplitudes on the radial component. However, SV amplitudes and phases change rapidly around the critical angle and the amplitudes can therefore be unreliable (see INF/focmec.pdf for details). So, although SEISAN will use the amplitudes read on the radial component, it is in general not recommended to use them. Assuming reading on only Z and H, the following amplitude ratios are calculated:

- SV/P
- SH/P
- SV/SH

In reality, the data only provides 2 independent ratios so ideally only 2 should be used. Since it is hard to know which 2 are the most reliable, SEISAN uses all.

Phase names in SEISAN used for amplitudes for FOCMEC have the names AMPG, AMSG, AMPN and AMSN for direct and first arrival (refracted), respectively. For local earthquakes both PG and PN types can be used while for distant earthquakes only PN types can be used.

Polarity selection

Any P-phase (first letter of phase name is P) with a polarity (C or D) is used, like P, Pg, PP etc. For further processing in FOCMEC, C is labeled C if phase onset is ' ' or I and '+' if phase onset is E. Correspondingly, polarity D is labeled D or -. FOCMEC can also use polarities of SV and SH, but this has not been implemented in SEISAN.

Local earthquakes

Any P-phase can be used like Pn and Pg. When few polarities are available, it is an advantage to use both Pg and Pn since these phases have different angles of incidence. Polarities associated with other phases are not used. There is no check if a P-phase has been duplicated.
Amplitude ratios must be determined from the same wave type for example Pg and Sg and the program will only form amplitude ratios from the same wave types. While in principle it should be possible to use ratios determined from refracted waves, generally ratios determined only from direct waves are used since they are easier to identify and have larger amplitudes than refracted arrivals. Particularly the Sn is difficult to identify. This means that the amplitudes readings most often will be made within what is considered the maximum amplitude in the Pg and Sg wave trains. However, the polarity might be read on the first arrival which can be Pn or another refracted arrival.

Distant earthquakes

Polarities of any P-phase can be used (but not pP since first letter is not P). Using amplitudes require events with clear P and S phases and usually this means reading on broad band records. The amplitude phase names AMPN/SN are used to indicate first arrivals.

Program operation

The program makes a grid-search and finds how many polarities and amplitude ratios fit each possible solution. All solutions with less than a given number of wrong polarities and/or amplitude ratios within given error limits, are then written out and can be plotted. With a cursor, the user can then select the preferred solution, which can be stored in the input file or the database. The program is intended to work from within EEV (option F), however it can also work independently (see below). The program uses an input file called `focmec.inp` (automatically generated). This is a Nordic format file. Direct waves have angle > 90 and refracted arrivals angle < 90 degrees. If the angle is > 90, the polarity is plotted at an azimuth+180. If the user wants to use FOCMEC as a freestanding program, the angle of incidence information may have to be put in manually in a standard CAT-file, which is then renamed focmec.inp. This can be done automatically by FOCMEC if a `hyp.out` and corresponding print.out file is available. FOCMEC can also be used to convert angles, like dip, strike and rake to T and P-axis, simply say 'focmec a', where argument a stands for angles and you will be prompted for input.

When the program runs, all amplitude information and corresponding corrections are listed:

```
============ FOCMEC ============
No FOCMEC.DEF file, use defaults

Q: Local: Qp= 100.0**1.00  Qs= 100.0** 1.0   Global: t*(P)=1.00  t*(S)=4.00

STAT  C PH       AMP    PER TRTIME   QCOR ANGINC ANGEMG Fcor   AZ  DIST
SNART Z PG      1582   0.16   12.6    1.2    100     79  0.6  301    77
SNART Z SG      9397   0.19   21.8    1.3    100     79 -0.3  301    77
SNART T SG     10577   0.09   21.8    1.3    100     79  2.0  301    77
MUD   Z PG        53   0.10   26.3    1.4     94     85  0.3  163   179
MUD   Z SG       197   0.15   45.5    1.7     94     85 -0.2  163   179
MUD   T SG       209   0.22   45.5    1.6     94     85  2.0  163   179
```

```
BLS5  Z PG       749   0.28   28.0    1.3     94     85  0.3  326   192
BLS5  T SG      1102   0.10   49.8    1.9     94     85  2.0  326   192
BLS5  Z SG       662   0.10   49.8    1.9     94     85 -0.2  326   192

 STAT  Ratio type  T     Amp 1    Amp 2  Fcor LogRat
 SNART SV(Z)/P(Z)  V      9397     1582   1.0   0.80
 SNART SH(T)/P(Z)  H     10577     1582   0.3   0.34
 SNART SV(Z)/SH(T) S      9397    10577   3.5   0.47
 MUD   SV(Z)/P(Z)  V       197       53   1.3   0.75
 MUD   SH(T)/P(Z)  H       209       53   0.2  -0.12
 MUD   SV(Z)/SH(T) S       197      209   7.5   0.87
 BLS5  SH(T)/P(Z)  H      1102      749   0.2  -0.45
 BLS5  SV(Z)/P(Z)  V       662      749   1.3   0.23
```

The abbreviations are STAT: Station code, C: Component, PH: Phase, AMP: Amplitude in count, PER: Period in sec, TRTIME: Travel time in sec, QCOR: Log Q-correction, ANGINC: Angle of incidence at the source, ANGEMG: Angle of emergence at the station, Fcorr: Free surface correction for this amplitude, Az: Azimuth from the event to the station, DIST: Epicentral distance in km., Ratio type (see text), T: indicator of ratio type, Amp1 and Amp2: The two amplitudes (count) in the ratio, Fcor is the free surface correction in the amplitude ratio (to be multiplied with ratio) and LogRat is the logarithm of the corrected amplitude ratio used.

Note that for station SNART, amplitudes were also read on the radial component so more then 3 amplitude ratios were used.

Following, the user get the choices:

```
Stop                        (0)
Plot saved solution(s)      (1)
Plot new solutions          (2)
Plot selected solution      (3)
Find new solutions          (4)
-1, -2, -3 also plot station
```

1. This is the solution(s) already stored in the data base (S-file). See secetion "Storing and selecting fault plane solutions" below.

2. Plotting new solution after having used option 4

3. Plotting the selected solution after using option 4
   Using e.g. -1 instead of 1, also plots the stations to help identify them on the plot, see Figure 6.23

4. Starting a search for new solutions

   Option 4 gives the following information and questions:

   ```
   There are   10 polarity readings
   Maximum number of allowed polarity errors or -1 to show best solutions only
   ```

   Depending on number of data values, 0-5 is a good answer. To let the program find the minimum number of polarity errors, type '-1', which is particular useful if there is a significant minimum number of polarity errors.

```
There are  8 amp ratio readings
Maximum number of allowed amplitude ratio errors
```

Equivalent for ratios to 'Maximum number of polarity errors', however, error is defined by amplitude ratio error. Number of errors depends on number of observations. For 9 observations 1-2 errors is reasonable.

```
Maximum amplitude ratio error,  return for default of .2
```

Give maximum allowed difference between observed and computed log amplitude ratio, default is 0.2, which often is a good value.

```
Degree increment in search
```

The program will now start the searching and write out on the screen (and in a file) the solutions which fit the requirement of number of misfits. The maximum number of solutions is limited to 100 as a default, or to the value defined by 'FOCMEC MAXSOL' in `SEISAN.DEF`. At the end, the number of acceptable solutions is written out as well as the minimum number of bad fits. This can then be used for the next search. Now option 0 to 4 can be used again.

When plotting the solution with option 2, the cursor comes up. Also, the solutions will be printed in text form to the screen, see Figure 6.23.

The abbreviations are Pol: Number of polarity errors for P, SV(not used) and SH(not used), Rat Err: Number of ratio errors, RMS RErr: The RMS error for the ratios used, RErr (All): The RMS error for all ratios.

The polarities and amplitude ratios can be plotted on the focal sphere using the same convention as the original FOCMEC program, which is:

o =	compression
+ =	emergent compression
Δ =	dilatation
- =	emergent dilatation
V =	amplitude ratio SV/P
S =	amplitude ratio SV/SH
H =	amplitude ratio SH/P

The user can select a preferred solution by moving the cursor near one of the letters T or P (T and P axis). By pressing T, the program will find the nearest T axis (same for P and nearest P-axis) and corresponding fault plane solution, which can be stored in the database and/or plotted with option 3. If no solution is to be selected, press q for quit. If a solution has been selected, the user will be asked if it is to be saved or not after selecting option 0. The saved solution goes into the focmec.out and from there into the S-file (type F-line) in the database if FOCMEC is operated from EEV and the solutiosn will also be written to fps.out.

When working from EEV, the event will always be located before the FOCMEC program starts up. In the Nordic format the solution is stored simply as strike, dip, rake and number of bad polarities (3f10.1,I5). Aki and Richards convention is used. In addition, the name FOCMEC will be written near the end of the line to indicate that the fault plane solution was made by FOCMEC. The other program, which can make a fault plane solution, is INVRAD (see EEV). The line type is F.

The following files are created:

`focmec.dat`: Input parameters to FOCMEC_EXE.
`focmec.log`: Log of the FOCMEC_EXE run.
`focmec.lst`: More details on solutions
`focmec.out`: Gives input parameters and solutions
`focmec.eps`: A Postscript plot file of LATEST plot
`focmec.run`: Run parameters for FOCMEC_EXE, you can re-run FOCMEC by 'focmec_exe ¡ focmec.run'

Running FOCMEC independently of EEV and composite fault plane solution:

Locate event(s) with HYP, then give command focmec. The program then combines the files `print.out` and `hyp.out` to make the focmec.inp file and proceeds as usual. This is actually the way FOCMEC works from within EEV. However, if more than one event is located, FOCMEC assumes that all events shall be used in a composite solution, and focmec.inp will therefore contain the header from the first event and phase lines from all subsequent events. This is the easiest way to make a composite solution.

NOTE, when running FOCMEC outside EEV, the fault plane solution is not put into the database (it does not belong to any particular event !), however it is written out in file focmec.inp and fps.out.

Computer limitations: Total number of polarities must be less than the dimension of array DATA (parameter max_data) for Nordic data (see file `seidim.inc` in INC directory).

Figure 6.23 shows an example of a fault plane solution calculated with FOCMEC.

6.16.2 FPFIT

This well known program, written by Reasenberg and Oppenheimer [1985], uses polarities to find one or several fps's (see manual fpfit.pdf in INF). Quoting the manual "Program FPFIT finds the double couple fault plane solution (source model) that best fits a given set of observed first motion polarities for an earthquake. The inversion is accomplished through a two stage grid search procedure that finds the source model minimizing a normalized, weighted sum of first motion polarity discrepancies". The weighted sum is expressed through the F-factor (0-1) given as output in S-file. A value below 0.5 is a good fit and a value of 1.0 is means a perfect misfit. A station distribution ratio STDR is calculated. Quoting the manual "The station distribution ratio is $0.0 < \text{STDR} < 1.0$. This quantity is sensitive to the distribution of the data on the focal sphere, relative to the radiation pattern. When this ratio has a low value (say, $\text{STDR} < 0.5$), then a relatively large number of the data lie near nodal planes in the solution. Such a solution is less robust than one for which $\text{STDR} > 0.5$, and, consequently, should be scrutinized closely and possibly rejected". This value is also written to the S-file. One advantage with FPFIT compared to FOCMEC is that formal errors are estimated and usually only one solution is given. The software is found at `http://earthquake.usgs.gov/research/software/`

The original program FPFIT is left unchanged except for a minor gfortran adaption. FPFIT is an interactive program with many options for parameters stored in a parameter file and different data input formats can be used. In the SEISAN implementation, this has been simplified and a SEISAN driver program FPFIT_SEISAN is used. This program converts the observations to an input file in hypo71 format, fpfit.dat, makes a parameter file with preset parameters, fpfit.inp and a run file fpfit.run to run the program. After running FPFIT_SEISAN (either free standing or through EEV with command fp), it is possible to run the original program directly with command fpfit and test different FPFIT parameters, using fpfit.inp as a starting parameter file (default). It is then possible to interactively get information about the different parameters. The hardwired parameters essentially use default settings, ensure the use of all data (e.g. no magnitude-distance restrictions) with the same weigh on all data. In addition, the following is set:

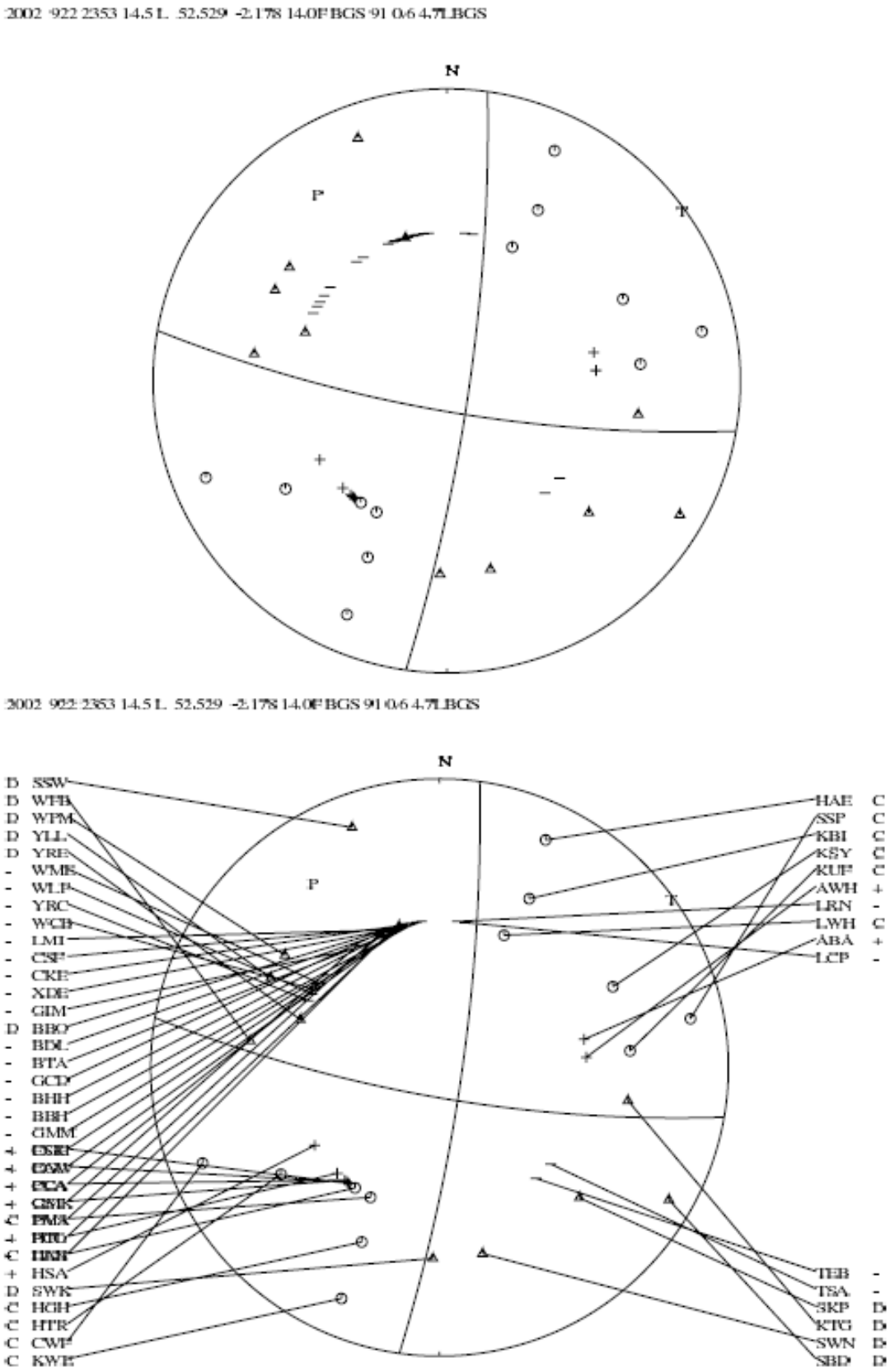


Figure 6.23: Top: An example of a fault plane solution plot. Symbols are explained in the text. Bottom: A fault plane solution also showing the stations with corresponding polarities.

- Search in as fine a grid as the program allows, one deg for fine search.

- Search for multiple solutions, not just the best. Gives an idea of uncertainty.

- Minimum number of polarities to attempt a solution is 6.

Run the program: In EEV, use command fp, first solution is written to S-file. The previous solution of FPFIT will be overwritten. FPFIT in SEISAN implementation can work with both global and local data, while the original FPFIT only works with local data. Outside EEV. See section on composite fault plane solution.

Output files:	
fpfit.out	Details of inversion. In the FPFIT manual, this file is called "Statistical summary file"
fpfit.fps	The fps solution etc. In the FPFIT manual, this file is called "Extended hypocenter summary card file"
fpfit.pol	Station and polarities used, see FPFIT manual
fps.out	The fps in SEISAN format in a cat file

Note: There is no check if polarities are read on Z-channel but it is required that the phase is P.

6.16.3 HASH

This program [Hardebeck and Shearer, 2002, 2003] determines fault plane solutions using P-polarities and amplitude ratios as input, just like the FOCMEC program. The P-amplitude $A_P = \sqrt{(A_r^2 + A_z^2)}$ and the S-amplitude $A_S = \sqrt{(A_{sv}^2 + A_{sh}^2)}$ where A is amplitude, r is radial, z is vertical, sv is SV, and sh is SH. The free surface correction is not built in, but replaced by a fixed factor per station, which has to be determined independently. In order to simplify the input, the free surface corrected amplitude ratios from FOCMEC are used as input for HASH. The program was modified to use only SH and by using the free surface corrected P on the Z-component, the assumption is made that $A_p = A_z(P)$. Thus only one amplitude ratio is used for each station (SH to P). HASH returns solutions with less than a given number of polarity errors and average amplitude errors less than a given limit. If no solutions are found, error limits are increased and normally many solutions are returned. Using this, an estimate of the best solution is made and likely errors calculated. The advantage with HASH is that it finds one or a few best solutions, while for FOCMEC the user must select one among many. Also HASH will not completely change the solution by one wrong amplitude ratio, since the average of the amplitude errors is used as selection criteria and not a single amplitude. FOCMEC does not give any estimate of the errors in the solution. HASH calculates an estimated error; however that requires an input where each event has been located with e.g. 10 different likely input models and all data is used as input in order to get estimate of fault plane solution uncertainties generated from the model. This was not done in the SEISAN implementation so only the error estimated from the spread in solutions is used. This might lead to smaller error estimates as compared to the original HASH implementation. The SEISAN HASH implementation is a simplified implementation compared to the original HASH with

many parameters hardwired, see hash_seisan.for for implementation details and changes. Like FPFIT, the F-fit function is calculated (called weighted fraction of polarity misfits) and similarly the station distribution ratio (see FPFIT). Both values are given in S-file as well as the average amplitude error. For more information, see the HASH manual hash.pdf and FPFIT manual fpfit.pdf in INF. The software is found at `http://earthquake.usgs.gov/research/software/index.php`. HASH does not estimate errors in strike, dip and rake but errors in fault plane and auxiliary plane (degrees).

Running HASH from EEV

Polarities and amplitudes are picked like for FOCMEC. When running the program, the amplitudes are corrected like for FOCMEC (actually done by FOCMEC) so the Q-correction will use the Q-relation given in focmec.def (see FOCMEC description above). The same output, as for FOCMEC, with the available amplitudes, their ratios and corrections will be shown and the control is the handed over to HASH. The questions are:

```
  Grid angle for focal mech. search, enter for def 2           Comment: Smallest is 2

  Max number of polarity errors                                Comment: No default
1
  Max average error in amp rat, log10, def 0.2                 Comment: default 0.2 selected

  Enter angle for computing mechanisms probability, def is 60  Comment. Default 60 deg. Sel.

  Enter probability threshold for multiples, def is 0.1        Comment: Default 0.1 selected
```

Now is following the FOCMEC amplitude information, not shown ...

```
 Number of polarities is                    :    11
 Number of amplitude ratios is              :     5
 Minimum number of polarity misfits overall :     0
 Minimum average amplitude error overall    :  0.13
 New number of pol. misfits inc. extra is   :     1
 New average amp limit inc. extra           :  0.23
 Minimum average amplitude error for pol ok :  0.27
 New average amp limit is                   :  0.37
 Number of solutions found                     92

 Strike,dip,rake              197.3    66.5  -157.4
 Fault+aux plane uncertainty   23.2    10.3
 ================================
```

Explanations on input:

The "mechanism probability" is the probability that the real mechanism is "close" to the preferred mechanism, within "angle for computing mechanisms probability" where angle define "close." If there are clustered outliers, alternative solutions (or "multiples") are found based on those outliers. You can set the minimum probability for the multiples (i.e. ignore multiples with a low probability.)

Explanations on output:

`Minimum number of polarity misfits overall`: Minimum number of wrong polarities for anyone of the grid points disregarding amplitude fit. This is the number of polarity errors to find a solution without

amplitudes.
`Minimum average amplitude error overall`: The minimum average log error for any grid point disregarding errors in polarity.
`New number of pol. misfits inc. extra is`: The new limit for polarity errors.
`New average amp limit inc. extra`: Based on the above, a new amplitude ratio error limit is set.
`Minimum average amplitude error for pol ok`: The new error limit considering polarities within limit.
`New average amp limit is`: In order to get sufficient solutions, the amplitude error limit is increased to this value.

Output files: Hash_seisan.out: A summary of the solutions(s).
Fps.out: The solution(s) in SEISAN format.

Storing and selecting fault plane solutions: Format errors estimates and quality.

The fault plane solutions are stored in the S-file. Different programs give somewhat different parameters and sometimes the same output field is used for different parameters. Some programs give strike of dip instead of strike of fault plane, but values used in SEISAN are converted to strike of fault plane. Each program is indicated with its own name like "`HASH F`" at the end of the F-line. If no characters are written in the blank space, any new solution will overwrite the old one. However if anything is written like "`HASH 1F`", any new solution will create a new line in S-file. This is also the case if a quality indicator is written (see below). An example is:

```
   158.0      53.1    -156.4  7.0  3.8      0.30 0.57 0.76      FCF HASH     F
39.2900   66.3900  -63.6500                          0.13  2 1 FCF FOCMEC   F
  42.0       68.0     -62.0  7.0  5.0  3.0  0.1  0.2           FCF FPFIT  A F
  18.7       67.8     -63.3                                4   FCF PINV     F
```

In this example, there are 4 solutions made by the 4 programs and the solution made by FPFIT has been selected as a prime solution with quality A. The content and format is:

```
Type F Line (Optional): Fault plane solution

Columns Format Description

  1:30 3F10.0 Strike, dip and rake, Aki convention
 31:45 4F5.1  Error in strike dip and rake (HASH), error in fault plane and aux. plane (FPFIT)
 46:50 F5.1   Fit error:  FPFIT and HASH (F-fit)
 51:55 F5.1   Station distribution ratio (FPFIT, HASH)
 56:60 F5.1   Amplitude ratio fit (HASH, FOCMEC)
 61:62 I2     Number of bad polarities (FOCMEC, PINV)
 64.65 I2     Number of bad amplitude  ratios (FOCMEC)
 67:69 A3     Agency code
 71:77 A7     Program used
 78:78 A1     Quality of solution, A (best), B C or D (worst), added manually
 79:79 A1     Blank, can be used by user
 80:80 A1     F
```

`Quality indicator`: The indicator can be any character, but usually A to F is used with A as the best.

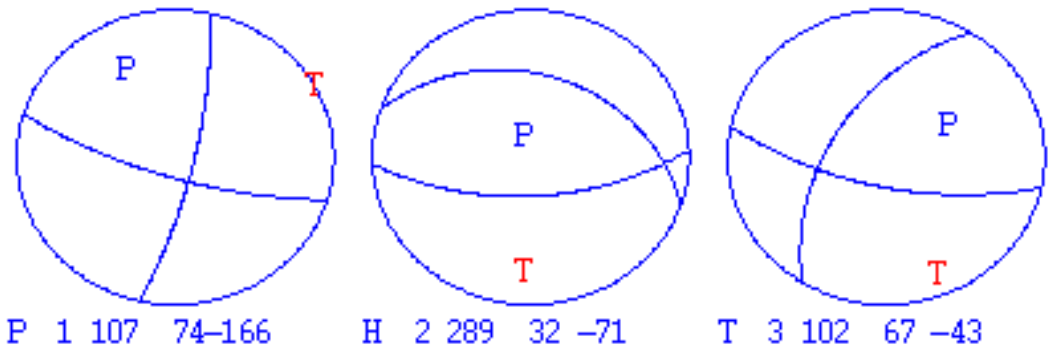


Figure 6.24: Using FOC to plot solutions from fps.out. The program used for each solution is given with a one letter code: P: PINV, F: FOCMEC, H: HASH and T: FPFIT.

It is up to the user to manually assign a quality indicator. Events can later be selected based on quality indicator. Programs SELECT and FOC use quality indicators.

Composite fault plane solutions

A composite solutions means that data from several events, suspected to have the same fps, will be used together. Composite solutions can be useful if little data is available. All programs except INVRAD can be used for a composite solution. Composite solutions must be calculated outside EEV and the solutions therefore do not become part of the data base. Both amplitudes and polarities can be used. Local and global data can be used. The procedure is:

- Select events to be used together in one cat file, e.g. by using SELECT.
- Locate the events with HYP, there will then be a hyp.out and a print.out, which are used as input for the composite solution.
- Start one of the 4 fps programs: FOCMEC, PINV, FPFIT_SEISAN or HASH_SEISAN. The usual questions will come up.
- The solution(s) will be written in the individual program output file. The solution(s) will also be written to the cat-file fps.out in standard SEISAN format. For each run of a program, the solutions accumulate in fps.out. This can be used to compare solutions from different programs, see FOC. An example of the plot is seen in Figure 6.24. To plot the observations, put in solution in hyp.out and plot with FOCMEC.

Which program to use

The different programs all have advantages and disadvantages. FOCMEC uses the most data since it can use more amplitude ratios than HASH but it might be difficult to find the 'correct' solution since a small change in input limits might make a large change in output. If a few of the amplitude ratios are very wrong, an unrealistic high ratio limit must be used and many errors allowed. This problem is avoided by using HASH since the limit is the average amplitude ratio error and not the number of errors. If only working with polarities, all 4 programs can be used. PINV gives a very quick solution which can be used as an indication of a possible best solution, however for final results one of the other 3 programs should be used. It is often a good idea to compare the results from the different programs. Ideally they should

all give the same result, but there will be difference due to different methods used and different data, however if solutions are very different, the solution might not be very stable. It is easy to compare the solutions. Run each program in EEV, then plot using command fo. Each solutions will be plotted in a different color, see Figure 6.25. If doing composite solutions, use program FOC with input from fps.out.

SLICK, inversion of fault plane solutions to get best stress tensor

This program is part of the Slick package doing the following quoting the author Michael [1984] "The slick package uses fault slip data (either field observations or from focal mechanism) to find the stress tensor that best explains the observations. Inputs are the orientation and slip direction of a set of fault planes. Outputs are the orientation and shape of the stress ellipsoid, including confidence regions, and statistics used to judge the success of the inversion. This method uses the linear inversion algorithm and non-parametric bootstrap statistics". The software is available at `http://earthquake.usgs.gov/research/software/index.php`.

In SEISAN, only the inversion part has been implemented so the error analysis is missing. Program SLICK can be run as a separate program, but is normally run as part of FOC which prepares the input for SLICK and plots the output. The method is explained in Michael [1984] where also examples are given with data available at the above web site. Running SLICK: slick "file", where "file" is a file with strike of dip, dip and rake. An example input file is:

```
Strik dp     Dip    Rake
   203.0    51.0   137.0
   280.0    85.0  -161.0
    ...
```

Note that in SEISAN, strike of fault plane is used so the strike of the dip is strike of the fault plane+90 degrees. The output is "file.oput" which gives the found stress tensor and the fit to the data, for details see Michael [1984]. The stress tensor has a corresponding slip angle, (average slip) and for each event the difference in slip angle for the individual event and the average slip is calculated as well as the average difference and standard deviation. When running SLICK with FOC, an input file foc.slick is made for selected events (making the corrections to strike of slip) and the output file is foc.slick.oput. FOC also plots the direction of maximum compressive stress s1, minimum compress stress s3 and null axis s2. In the example below s1 has max value of 0.68 and strike and dip are 19 and 34 respectively. S3 has strike and dip of (113, 5) and s2 (-149, 56) respectively. The average fit angle is 59 with a standard deviation of 51, a bad fit.

```
stress tensor is:
-0.290526  0.236582  -0.146602
0.236582  0.293028  -0.438347
-0.146602  -0.438347  -0.00250278
eigenvalue   vector: E,N,UP,direction,plunge
0.686771  -0.273399  -0.784281  0.556917  19.205739  33.820430
-0.37516  -0.917882  0.385856  0.0927811  112.725966  5.320093
-0.31161  0.287656  0.485818  0.82537  -149.270892  55.589101
variance= 0.283314
phi value= 0.940156

dip direction, dip, rake, fit angle, mag tau
  203.0     51.0    137.0    166.4    0.11
  280.0     85.0   -161.0    167.9    0.19
```

```
   ...
   13.9     68.1    -85.7     11.0    0.46
   14.0     70.0   -130.0     33.3    0.48
fit angle mean= 59.156784 standard deviation= 51.527176
for f=0.8 I= -2.242677 , std. dev.= 1.609795 D norm= 0.248640
avg tau=xx , std. dev.= xx
```

For a complete stress analysis it is recommended to also do the error analysis using the complete slick package or e.g. the program ZMAP (not a SEISAN program, uses MATLAB, found at `http://www.earthquake.ethz.ch/software/zmap/ftp`). FOC writes a file which is formatted for input to ZMAP. However doing stress analysis as implemented in SEISAN gives a good impression of the consistency of the fault plane solutions in a particular area. It is recommended that at least 10 events are used for inversion.

Plotting fault plane solutions

There a 4 ways of plotting fault plane solutions in SEISAN: Through EEV (a single event), program FOC (many events), program EPIMAP (many events) and W_EMAP (many events). The input file is in all cases a CAT-file. In addition, using program SEIGMT, a file to be used with GMT is prepared, however the use must make his own script. Only through EEV is it also possible to plot the observations.

Using EEV
Command fo will plot all events in S-file. This can be a useful ways of comparing solutions obtained by different programs, see Figure 6.25.

Using FOC
See under FOC for how to run the program, see page 228. The plot is seen in Figure 6.24.

W_EMAP (Windows only) plots the solutions as seen in Figure 6.27. In this case the simplest is to give command w_emap file, where file is the CAT file with fault plane solutions. See W_emap manual in INF. NOTE: Some versions of W_EMAP plots some of the fault plane solutions with inverted color e.g. inverse fault becomes a normal fault).

The EPIMAP plot for the same events is shown in Figure 6.27. See EPIMAP for more explanation. EPIMAP can also plot the fault plane solutions in a section, the solutions are still seen in the horizontal plane.

FOC .

FOC is a program doing different things with fault plane solutions given in a CAT-file: Converting data to other formats, plotting many solutions, running the SLICK program and displaying the results, plot P and T axis for many events and make statistics of polarities. The input is:

```
foc
 Give input file
collect.out
 Quality, ABC.., up to 5 chars, enter for all
AB                                          Comment: Different qualities can be selected
 Cumulative(c) or individual misfit(def)    Comment: See later

 Plot all solutions selected (Y=enter/n)    Comment: Analysis can be done without plotting all
N
```

The plot of the many fault plane solutions is seen in Figure 6.28. After plotting the fault plane solutions,

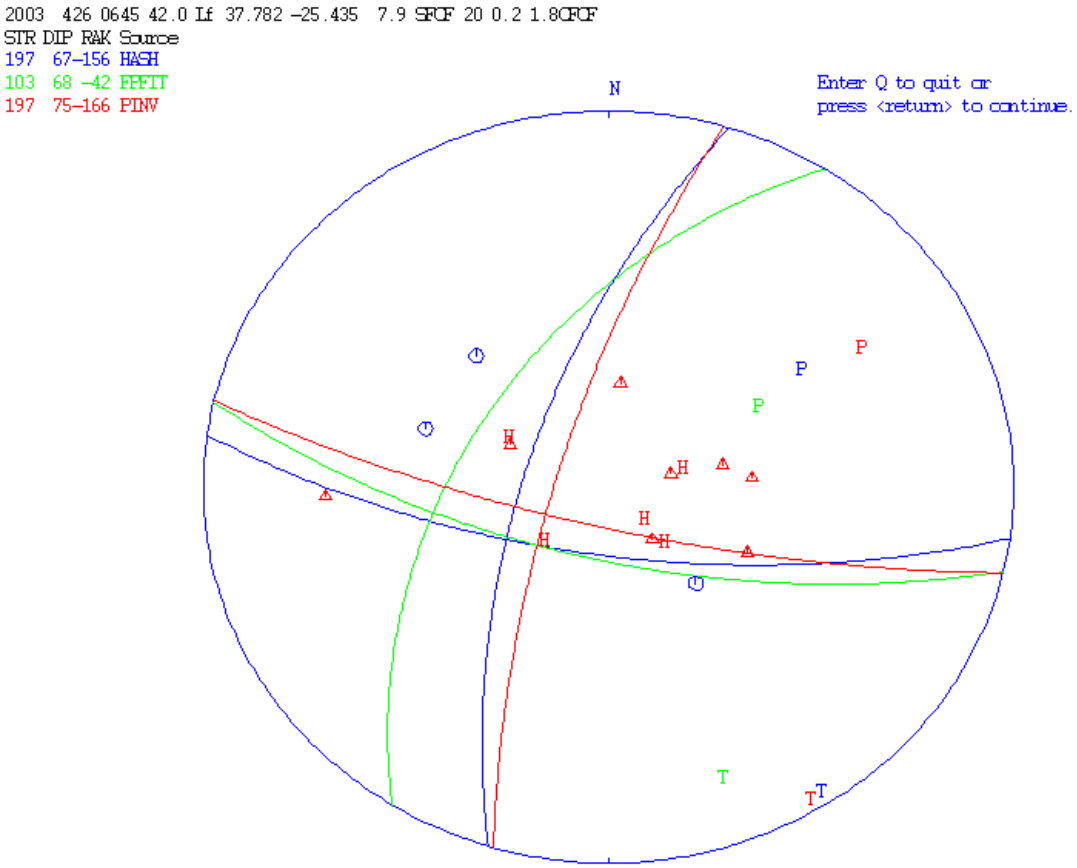


Figure 6.25: Compare fault plane solutions from different programs. For explanation of symbols, see FOCMEC.

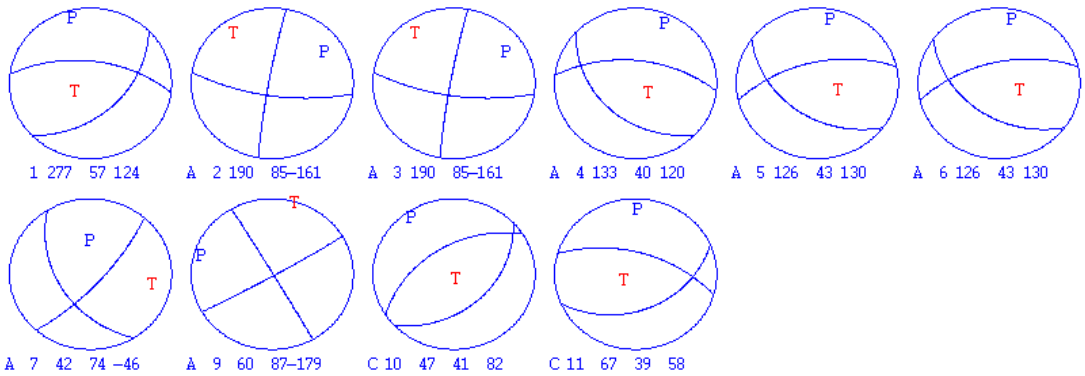


Figure 6.26: Example of plotting many solutions. Each solution is given with number, the fault plane solution and the quality (A-E). Up to 24 solutions can be plotted on one page.

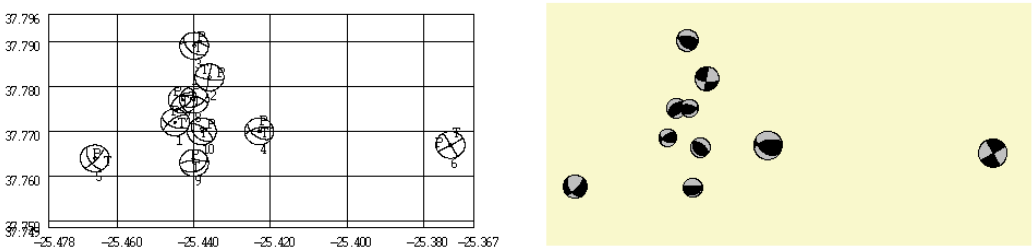


Figure 6.27: Plotting many fault plane solutions. Left: Using W_EMAP. Notice that the colors in the solutions are inverted compared to normal practice. Right: Using EPIMAP. The data for the two plots is the same.

a plot comes up plotting the location of the P and T axis and the results from SLICK, see Figure 6.28.

Output files:
Foc.out: P and T axis for all events, can be used as input to make rose diagrams.
Foc_events.out: The events used based on quality selection
Foc_pol.out: Statistics of polarities:

```
Stat     C    D
AZ05     3    2
MESC    18   60
VIF0     5    2
MIRA    10   32
VIF     44   19
LFA     52   18
PRCH    36    6
PVER    26    3
FRA0     0    2
AZ07     1    1
.
.
SET2     3    0
PSAN     1    3
 Sum of maximum number of polarities         570
 Sum of minimum number of polarities         158
```

For each station the different polarities are counted and a sum of the consistent polarities are given at the end. Foc.zmap: Input file in format used by ZMAP, notice direction of slip is used instead of strike of fault, see SLICK.
Foc.slick and foc.slick.oput: See SLICK.

NOTE: FOC uses the first instance of the fault plane solution found in file for a particular event.

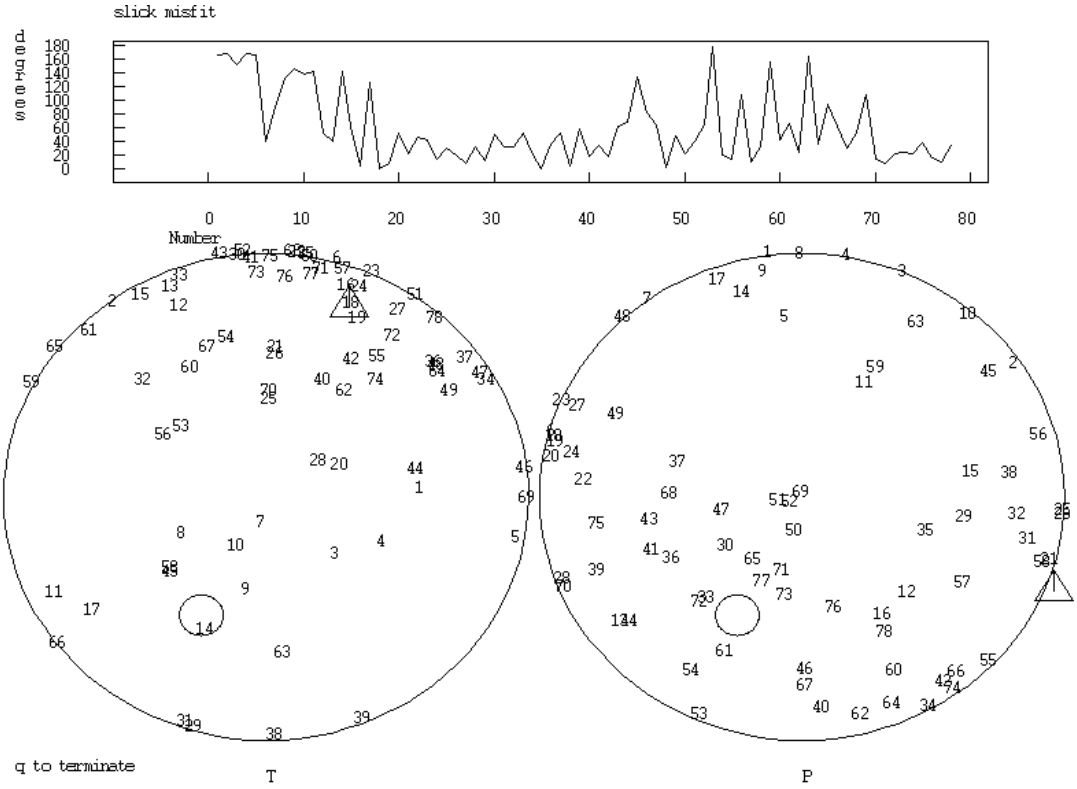


Figure 6.28: Left: The position of the T-axis given by the event numbers. The triangle is the SLICK minimum compressive stress direction and the circle is the null axis. Right: Corresponding for P-axis and the triangle is now the maximum compressive stress direction. Top: The misfit for each event as a function of event number. This figure can also show the cumulative misfit, see example run of FOC.

6.16.4 Moment tensor inversion program, INVRAD

The program is written by John Ebel [Ebel and Bonjer, 1990] for moment tensor inversion for very local events. The program uses instrument-corrected amplitudes of the direct (upgoing) phases of P, SV and SH phases and makes a linear inversion for the moment tensor. The program then finds the largest double couple component of the traceless moment tensor. For more details see file invrad.txt in the INF directory.
The original program has been slightly modified in input and output to be integrated with EEV in SEISAN. The steps to get the fault plane solution are:

Select the event from EEV

1. Plot each trace and select preferably the first clear amplitude of the direct wave. Mark the amplitude as usual and associate the amplitude with amplitude phases AMPG or AMSG (direct phases). This will create a separate line with amplitude readings only. The polarity must also be indicated on a separate phase , which must be Pg or Sg since the inversion program uses the polarity of the amplitude. The amplitudes MUST be picked on instrument corrected traces if all instruments do not have the same response function. At least 5 amplitudes must be selected. S phases picked on vertical or radial components will be considered SV while S-amplitudes picked on transverse components will be considered SH. Phases picked on NS or EW component cannot be used. If these new phases are not to be used for location, they can be weighted out.
2. Update event with command update to make distance and azimuths available.
3. Use command INVRAD to do the inversion. This command does several things hidden for the user:

   - Creates the model input file for INVRAD called invrad.mod. This file is created from the STATION0.HYP file, either from the current directory or DAT.
   - Creates the data input file for INVRAD called invrad.inp. This file is made from the current database file (S-file) by extracting all amplitudes associated with Pg and Sg amplitudes and converts to P, SV or SH amplitudes in microns. The depth of the event is taken from the S-file header and the estimated error is fixed to 0.1 micron.
   - Runs the INVRAD program which produces the invrad.out file
   - Reads the invrad.out file to get the fault plane solutions which overwrite the current fault plane solution in the S-file. If you do not want to get the current solution overwritten, put a character in column 79 on the solution, see also focmec program.

The fault plane solution can then be plotted with FOCMEC.

6.16.5 Moment tensor inversion program, PINV

This program makes a preliminary best fault plane solution based on polarities and is intended as a help to other methods of fault plane solution. The original program was written by Suetsugu and some information is found in [Suetsugu, 1998]. A copy of this report, which also gives general information about fault plane solutions, is available as foc.pdf at `http://iisee.kenken.go.jp/lna/?mod=view\&cid=S0-250-2007`

To run the program from EEV:

Command `pi` will locate the current event and then start PINV (stands for P-inversion). PINV is hardwired to use hyp.out as input file and it will use all polarities from P-phases (capital P as the first letter). The result of the inversion is written out on the screen and in file pinv.out. The strike, dip and rake and number of wrong polarities is also written to the S-file provided at least 5 polarities are available, however PINV will make an inversion with any number of polarities and write the result to the screen. In the S-file, the result is written as an OF-line giving the source of the inversion as PINV. A new inversion will overwrite the previous solution. This means that a PINV solution will be additional to the solution given by the F-line and therefore not considered as prime. It is also possible to directly compare the solution to the solution obtained by FOCMEC.

To run program outside EEV.

The program can run with one or many events (composite solution). First locate event(s) with HYP, then give command pinv and the inversion is made. All polarities in the hyp.out file are used. The result only goes the screen and pinv.out.

Principle of operation:

Moment tensor inversion can be done most simply using amplitudes as observed on the focal sphere. In PINV, polarities are considered to be amplitudes of +1 or -1 corresponding to compression and dilation, respectively. This is a gross oversimplification since there will be large variation of real amplitudes over the focal sphere. This input data of amplitudes is then inverted to get the moment tensor under the restriction of finding a single double couple. Despite the simplification, the advantage of this method is that it very quickly gives a best approximation to the fault plane solution. This best solution, particularly with few data, might be just one of many possible (see FOCMEC), but it serves to give an idea of a possible best solution and it is in general consistent with the observations [Suetsugu, 1998]. Unfortunately PINV does not give an error estimate to judge how reliable the solution is. It is therefore not recommended to use PINV to obtain prime fault plane solution, but rather as a help to select a solution when using FOCMEC unless much well distributed data is available. In some cases, FOCMEC will find a solution where all polarities fit, while PINV will get a similar solution with some polarity errors. This can be explained by PINV using the assumption of +1 and -1 amplitudes and thus an overall fit to amplitudes near nodal planes might be difficult. The original input to the PINV program has an option to give zero amplitude of observations judged nodal, however in our experience it is hard to judge if a first polarity is nodal or just has a small amplitude du to path (e.g. Pn) so this option has not be included.

An example of a run is:

```
Number of data used for inversion=  8
Absolute pseudorank tolerance     0.001210          Pseudorank   5
Strike, dip, rake           72.3      38.9      34.6
Consistent data:     8
Inconsistent data:   0
```

There were 8 observation which all fitted. The pseudorank indicates how many parameters can be determined in the inversion. In this case 5 since there are 8 observations. If less than 5 observations, the pseudorank will be less than 5. The Absolute pseudorank tolerance is a measure of the fit.

6.17 Calculation of coda q, CODAQ

The program will calculate coda Q (hereafter called Q) for a series of events and stations at given frequencies. On completion, the average values are calculated and a Q vs f curve is fitted to the calculated

values. The program will also plot the individual events and filtered coda windows.

The principle for calculation is the standard coda Q method, whereby a coda window is bandpass filtered, an envelope fitted and the coda Q at the corresponding frequency calculated. The envelope is calculated RMS value of the filtered signal using a 5 cycle window. The program used here is the one described in Havskov et al. [1989]. The program can only operate in connection with the SEISAN format S-files. The program can use all waveform file types accepted by SEISAN and there can be more than one waveform file in the S-file. The program will also take advantage of the SEISAN database structure.

Input:
The calculations are controlled by a parameter file called codaq.par and the actual event-station combinations to use are given in `codaq.inp`. Example files are in DAT, and with the test data set and the example files in DAT, a test run can be done. An example of a parameter file is shown below:

```
------------------------------------------------------------------
start in s times and Vp/Vs ratio (optionally)
2.0
absolute start time (sec)
0
window length (sec)
20
spreading parameter
1.0
constant v in q = q0*f**v
1.0
minimum signal to noise ratio
2
Noise window in front of signal and length of RMS noise window
15,5
minimum correlation coefficient
0.50
maximum counts to use
64000
number of frequencies
3
frequencies and bands
8,3
12,4
16,6
default stations(1. line) and components (2. line), 30a5
HYA  ASK  SUE
S  Z S  Z S  N
------------------------------------------------------------------
```

Start in s-times and Vp/Vs ratio (optinal): Normally the coda window starts at twice the S-travel time from the origin, this factor can be varied and might be chosen differently in special cases. Note that the S-time is calculated from the P-time so a P-time must be present. This also means that if a Pn is used, the coda window will start at 2 times the Sn travel time, which might be substantially different from 2 times the Sg travel time. e S-time is calcualted from the P-time using and Vp/Vs = 1.78. Optionally, the

user can select an Vp/Vs ratio to be used. This parmeter is optional so parameter files prior to version 8.3 can be used.

Absolute start time: If 0.0, above parameter is used. However if different from zero, an absolute start time relative to the origin time is used for the start of the coda window. This might be useful since different start times (meaning different lapse times) might produce different q-values. To use this parameter, one must be certain to choose it long enough which can be checked with the plots. If the absolute start time is smaller than (Start in s-times) multiplied by the s travel time, the station will be skipped and a message given.

Window length: This is the coda window length in secs. Use at least 20 secs to get stable results.

Spreading parameter: The geometrical spreading parameter used in q-fit, normally 1.0 is used.

Constant v in $q = q0 * f * *v$: For all $q(f)$ values, $q0$ is calculated using a fixed v, use e.g. 1.0. This parameter has no influence on the individual q calculations.

Minimum signal to noise ratio: In order to accept a q value for the average, the signal to noise ratio must be above this value. The signal to noise ratio is calculated using the last tRMS (see next parameters) secs of the filtered coda window and the first tRMS secs of the data file window. If the data file starts with noise or in the P signal, the s/n ratio will be in error. A reasonable value is 5.0.

Maximum counts to use: If the count value in a coda window is above this value, the window is not used. The intention is to avoid using clipped values. From SEISAN version 7.2, there is also an automatic checking for clipped values in addition to 'maximum counts'.

Noise window in front of signal and length of noise window, tnoise and tRMS: The first number is the number of seconds of noise to plot in front of the signal. In previous versions, 15 secs was hardwired, but sometimes there was not 15 secs of noise before the P. The second number is the length of the noise window used for calculation of the signal to noise ratio. This was earlier hardwired to 5 secs.

Minimum correlation coefficient: In order to use the q value in the average, the correlation coefficient of the coda q fit must be larger than or equal to this value. NOTE. Correlation values are in reality negative, but are always referred to as positive in the following. An acceptable value depends on the data, try to use a value higher than 0.5 (in reality -0.5)

Number of frequencies: Number of frequencies to use, maximum 10, 5 is a good number.

Frequencies and bands: The corresponding center frequencies and frequency bands. The frequency band should increase with increasing frequency to avoid ringing. E.g. 8,3 means that the signal is filtered between 6.5 and 9.5 Hz. It is advisable to use constant relative bandwidth filtering, to get an equal amount of energy into each band. The relative bandwidth is defined as $RBW = (f_u - f_l)/f_o$ where f_u and f_l are upper and lower frequency limit respectively. Such a filter would be e.g. 4 ± 1, 8 ± 2. 16 ± 4. The frequency representing the energy in a particular filter band, is the geometric center frequency calculated as $f_c = sqrtf_u f_l$. Since the user probably wants to calculate coda Q at the given frequency, the normal option (new in SEISAN7.2) is that f_u and f_l are calculated such that the given bandwidth (e.g. 4 Hz) is used, but the actual f_u and f_l will give the specified central frequency. It is still possible to calculate as before, where f_u and f_l will be exactly as specified (but the geometrical center frequency will not correspond to specified center frequency) by giving the bandwidth as a negative number.

Default stations: The stations that will be used if not specified in the `codaq.inp` file. THE LINE MUST CONTAIN AT LEAST SOME BLANK CHARACTERS, if not, stations will not be read from `codaq.inp` file and the program will crash. Note also that the program assumes that you use SP Z channels if there is no components given in the line following. After reading the parameter file, the program will by default use the `codaq.inp` file to get the event station information. However, any other name can be used if

specified interactively, see below. The station codes can have up to 5 characters.

The `codaq.inp` file will consist of a series of lines each giving an event identifier (an INDEX file). An easy way to generate the file is using the SELECT program. The file can also be generated with EEV using the (C)opy option making a file called indexeev.out. An example is shown below:

```
    1  /top/seismo/seismo/REA/BER\_\_/1992/06/16-0343-38L.S199206
    3  /top/seismo/seismo/REA/BER\_\_/1992/06/16-1311-58L.S199206
    7  /top/seismo/seismo/REA/BER\_\_/1992/06/30-1504-30L.S199206
```

The above example only uses the default stations given in `codaq.par`. Below is an example where particular stations and components have been selected with particular events, for this to work the station line in `codaq.par` MUST be blank.

```
    1 /top/seismo/seismo/REA/BER\_\_/1992/06/16-0343-38L.S199206
HYA  KMY  BER  ASK  TRO
S  Z S  E B  E S  Z S  Z
    3 /top/seismo/seismo/REA/BER\_\_/1992/06/16-1311-58L.S199206
HYA
    7 /top/seismo/seismo/REA/BER\_\_/1992/06/30-1504-30L.S199206
HYA  EGD
S  E  S  Z
```

Note that the numbers to the left originate from the index file and do not have any importance. The long name with the directory structure, is the name of the pick file (S-file) in the database, if the S-file is in the local directory, it can have just the event id, in this example starting with 30-....The waveform file name is in the S-file. Following the S-file name is, (like in the parameter file), first a line with station codes followed by a line of component codes. Like in the parameter file, if a component is not given, it will be assumed that the component is S Z. THE COMPONENT LINE MUST BE THERE, EVEN IF BLANK.

Below is an example of a `codaq.inp` file, where it is assumed that the S-files are the current directory. This file can also be generated with DIRF.

```
       16-0343-38L.S199206
HYA  KMY  BER  ASK  TRO
       16-1311-58L.S199206
HYA
S  E
       30-1504-30L.S199206
HYA  EGD
S  N S  E
```

Program operation:
The program first reads the parameter file, default `codaq.par` which must be in your current directory. It then reads the `codaq.inp` file with the events to analyze (also in current directory). The S-file names given here can, as shown in the examples above, be in the database or elsewhere, e.g. in your local directory. In the S-file, the name of the waveform file is given. If more than one waveform file is given, all files will be searched for the specified station and component. The program will first look in the current directory, and then in WAV and thereafter in the WAV database and other directories as given in the `SEISAN.DEF` file in DAT. The program can therefore work without moving the data from the database,

however you can also move both the S- files and waveform files to your local directory. Remember that the S-files must be updated in order to have origin time, since the program uses the origin time and P arrival times from the S-files.

Running the program:

```
Type codaq, the program asks about output:

0: Only q is calculated
1: Q is calculated and a plot on the tek screen is shown
2: Q                                                    ,
   and at the same time hard copy plots are made.
3: Q is calculated and hard copy plots are made, but
   no screen plot.

Parameter file, name codaq.par is default (return)
   Just hit return if default file, otherwise give name.

File with event stations, codaq.inp is default (return)
   Just hit return if default file, otherwise give name.
```

The program will now start to run. Alterantively, the progran can be started with arguments on the promt line:

```
codaq n parameter-file data-file
```

and no questions are asked. n is the choice 0 to 3 above.

If no plot is chosen, one line will appear on the screen for each station used and one for each frequency. The program will start a new page for each new event. If you are plotting on the screen, you will therefore have to hit return to get the next plot. The screen might not have been filled out if there are few data.

All questions will appear in the text window. At the end, a summary is given, which is the same as logged in the output file `codaq.out`.

The abbreviations are:

H:	Focal depth
M:	Magnitude
TP:	P travel time
TC:	Start time of coda window relative to origin time
F:	Frequency
Q:	Corresponding coda q, if 0 value is > 10000 or negative
S/N:	Signal to noise ratio AV
Q:	Average q
SD:	Standard deviation for average
NT:	Total number of q values at all frequencies
N:	Number of q values at given frequency
q:	Average of q values
1/q:	q is calculated as 1/q averages, probably the best to use
f:1/q:	Q values calculated using the relation derived from the 1/q averages q = q0*f**v obtained with the average 1/q-values
cq0:	Constant q0 obtained using the fixed user selected v
v:	Constant v determined
cor:	Correlation coefficient on determining q vs f
corr:	Average correlation coefficients of individual codaq calculations when fitting the envelope, both average and standard deviation is given

If a station is not present or no P is read, a message will be given. The program will search for the first P arrival time in the S-file. If several are present for the same station, it will use the first.

Output:

A file called `codaq.out` is generated. It contains a copy of the parameter file, one line for each event station combination accepted by the program (correlation and s/n ratio) and the average q values. The q values are averaged directly (indicated by q) and $1/q$ are averaged (indicated by $1/q$). At the end are the fits to the $q = q0 * f * *v$ relation. This relation is calulated using the average Q-values for each frequency and each average is weighted by the number of observations used to calculate the average.

A file called codaq.index is created. This index file contains all the events accepted for calculating the codaq values and can therefore e.g. be used for making an epicenter map of events actually used (use collect with the index file) Output file `codaq1.out` contains the same output as `codaq.out` except there is no `print out` for each event.

Example of `codaq.out`:

```
start in s-times                          2.00
absolute start time (sec)                 0.00
window length (sec)                      15.00
spreading parameter                       1.00
constant v in q = q0*f**v                 1.00
minimum signal to noise ratio             5.00
noise window in front of signal and len  15.00      5.00
minimum correlation coefficient           0.50
maximum counts to use                   500000
ASK  SUE  KMY  EGD  HYA
S  Z S  Z S  Z S  Z S  Z
```

```
 \SEISMO\WAV\1996-06-07-1324-51S.TEST__009
 \SEISMO\WAV\1996-06-07-1324-51S.TEST__009
 \SEISMO\WAV\1996-06-07-1324-51S.TEST__009
  1996 6 7132458 KMY   tc  41.3  f 16.0  s/n 37.4  Q 1077  corr -0.55  rms 0.30
 \SEISMO\WAV\1996-06-07-1324-51S.TEST__009
 \SEISMO\WAV\1996-06-07-1324-51S.TEST__009
 \SEISMO\WAV\1996-06-25-0336-34S.TEST__032
  1996 625 33715 ASK   tc  87.2  f  4.0  s/n121.9  Q  340  corr -0.60  rms 0.19
  1996 625 33715 ASK   tc  87.2  f  8.0  s/n 72.5  Q  551  corr -0.56  rms 0.28
 \SEISMO\WAV\1996-06-25-0336-34S.TEST__032
  1996 625 337 5 SUE   tc  57.3  f  4.0  s/n 96.9  Q  193  corr -0.61  rms 0.34
 \SEISMO\WAV\1996-06-25-0336-34S.TEST__032
  1996 625 33730 KMY   tc 143.5  f  8.0  s/n 13.6  Q  506  corr -0.61  rms 0.27
 \SEISMO\WAV\1996-06-25-0336-34S.TEST__032
 \SEISMO\WAV\1996-06-25-0336-34S.TEST__032
  1996 625 33653 HYA   tc  84.4  f  2.0  s/n 51.5  Q  288  corr -0.54  rms 0.12
  1996 625 33653 HYA   tc  84.4  f  4.0  s/n239.5  Q  427  corr -0.56  rms 0.17
  1996 625 33653 HYA   tc  84.4  f  8.0  s/n112.1  Q  504  corr -0.61  rms 0.27

 Freq      2.00     4.00     8.00    16.00
 Band      1.00     2.00     4.00     8.00

        AV Q  SD AV Q  SD AV Q  SD AV Q  SD

 NT=  8    N=  1    N=  3    N=  3    N=  1    N=
    q    288   0  320 118  520  27 1077   0
  1/q    288   0  287 123  519  26 1077   0
f:1/q    206  91  333 147  537 237  867 382

    q   cq0= 84  sd= 37  q0=143  sd= 49  v= 0.65  sd= 0.16  cor= 0.94
  1/q   cq0= 82  sd= 38  q0=128  sd= 57  v= 0.69  sd= 0.20  cor= 0.93

  Corr: 0.540.00 0.590.03 0.590.03 0.550.00

  Average lapse time with sd       83.704498       29.501974
```

Above, the one line per q calculations is showing results from different stations. Only the traces selected (fulfilling selection criteria) are shown. The time indicated, is the start time in the waveform file for that particular station. In general, the start time for each channel of digital data would be different. If some data is missing, it is also show in the `codaq.out` file. Corr is the average correlation coefficient (with standard deviation) for the data selected for that frequency. The average lapse time is the average of the tc - values.

In the DAT directory, there is an example `codaq.par` and `codaq.inp` set up to run on PC assuming that SEISAN has been installed under
seismo. If installed differently, edit the `codaq.inp` file to reflect the installation. For Unix testing, the `codaq.inp` MUST be edited to reflect the installation path or the file is regenerated using EEV as described above.

General recommendations: Coda window should be 15-25 seconds, minimum correlation coefficient larger than 0.5. For comparing coda values in different regions, ALL processing parameters must be identical

and average lapse times should be very similar.

Figure 6.29 gives an example of a codaq plot. There are no options for the codaq plots and the length of the window is always the first 200 secs from the original trace. If origin time or coda window is outside this 200-sec window and data is available, the program continues, but the coda window is not plotted on the figure.

6.18 Merge events near in time ASSOCI

The program will check if two events are close together in time and merge the events if requested. This is partly an alternative to use append in EEV. The program asks for maximum time difference between events to associate. The user will then be asked if events should be physically associated or not. The program is useful when merging a large number of events. The program has two alternatives for merging:

1. Merge events in same data base: One event is compared to the next event in the same data base. If they are close enough in time, the two events are merged and the program moves on to the next event. If 3 events are close in time, only the 2 first are merged. In order to also merge the third, the program has to be run again.

2. Merge events from a file into the data base: This option makes it possible to merge from another data base (use SELECT or COLLECT to create a file) without first completely mixing the two. The event from the file will be merged with as many files from the data base as fit the time difference criteria. So e.g. 2 events from the data base can both get the same event from the file included. At the end of the run, two files are output (associ_rest.out associ_merg.out) with events which were not merged and merged respectively. These can then be put into another data base with split, if desired. This function can also be used to separate the input file in two files.

   Note: When merging within one data base, the first event will get the next one merged into it. If merging from file into a data base, the event in the data base will always be the first and keep the main header. This thus a safe method when you want to keep the main header uncheged in the data base.

6.19 Making synthetic seismograms

BOUCH and BOUSEI, HERRMANN and HERRSEI and WKBJ are all programs which is used for generating synthetic seismograms.

The full wave modeling programs are written by Bouchon and Herrmann, and for WKBJ, Chapman and Valerie Maupin. Valerie Maupin has integrated WKBJ for SEISAN and written the routines that makes it possible to use specific phases. She has also made many improvements in the original installation of BOUCH and HERRMANN and written a large part of this chapter.

Bouchon:
The Bouchon program is somewhat modified for SEISAN. The theory, which is quite straight forward, is given in a series of papers (e.g. Bouchon [1981]). It is based on a discrete wave number representation of the wave fields. Basically, the source is repeated periodically in space, so that integration over the k-domain is replaced by a series. This implies that the periodicity of the source, L (in km), should be large enough so that the information from fictitious sources does not arrive during the time interval of interest. Roughly r < L/2, sqrt((L-r)**2+Z**2) > Vp*t where r is the epicentral distance and Vp is the

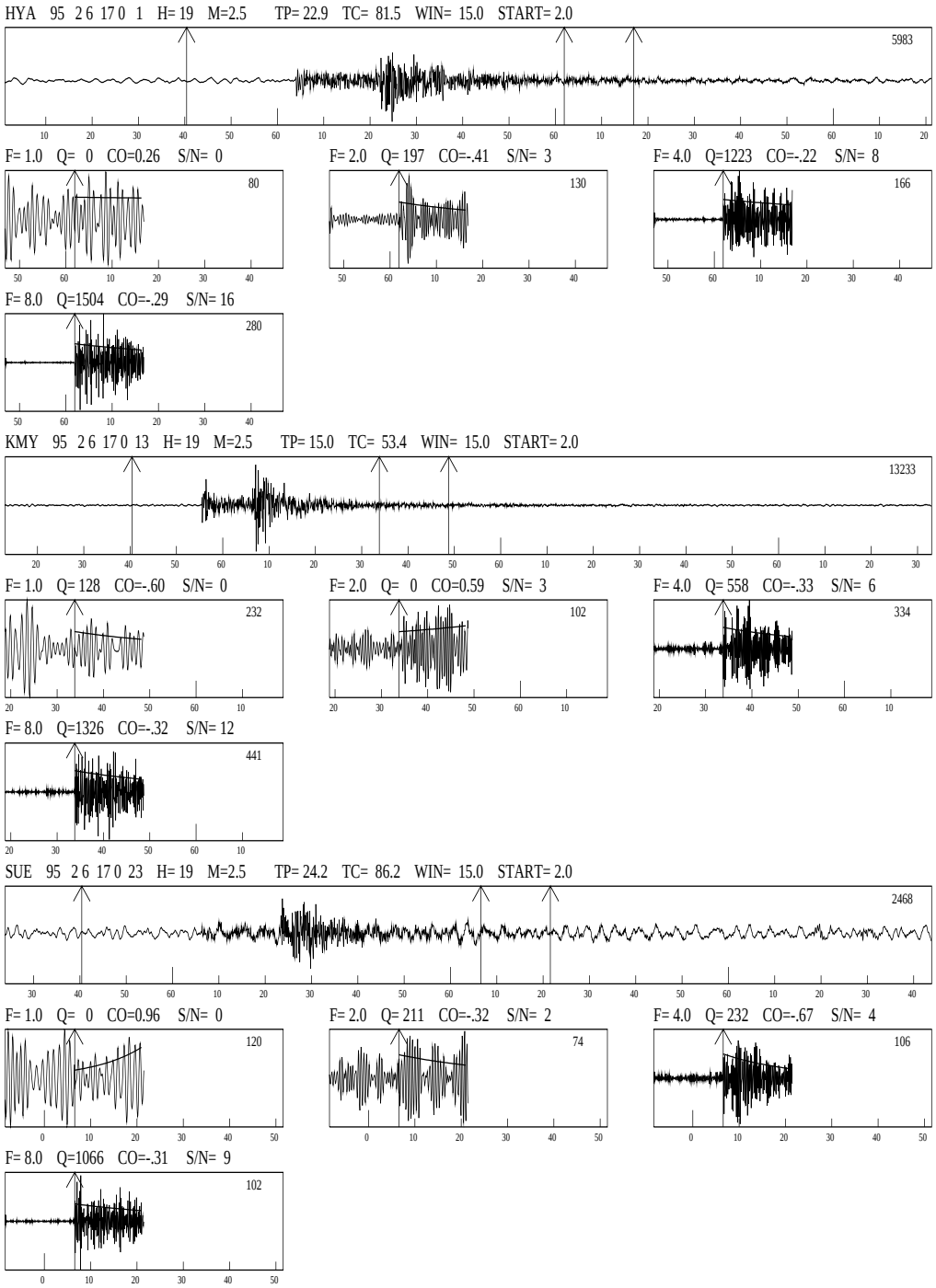


Figure 6.29: An example of a coda Q plot. On top is shown the original trace and below the filtered coda windows. Note that 15 secs of noise are shown in front of the selected filtered coda window. The first 5 secs of the noise shown is used for calculating the S/N ratio. On each filtered plot is given F: Center frequency, Q: Q-value, zero means no Q-value could be calculated, S/N: Signal to noise ratio.

highest P-wave velocity of the model, t is the travel time and Z the hypocentral depth. Only layered (horizontal, parallel) earth model is used. The earthquake source cannot be in the bottom layer or at the surface.

There are 2 programs, BOUCH and BOUSEI. BOUCH computes the frequency response given the model, the source depth, the focal mechanism, the receiver locations and the orientations of the two horizontal components. BOUSEI takes the output file from BOUCH, multiplies it by the source spectrum and uses an FFT to get the synthetic ground motion (displacement, velocity or acceleration). The user must provide the source function (see below) and the original waveform files must be available in WAV or working directory if a file containing both real and synthetic signal is to be generated. Otherwise, only synthetic data will be seen in the output file.

Herrmann:
The Herrmann programs HERRMANN and HERSEI work the same way as BOUCHON and BOUSEI respectively. The major difference is that once HERRMANN has been executed, HERSEI can be executed with different fault plane solutions to obtain the time series, while for the Bouchon programs, both programs must be run again. The Herrmann programs are thus faster for testing many different fault plane solutions.

The description in the following is for the Bouchon programs, but the steps are the same for HERRMANN.

WKBJ:
As opposed to the seismograms calculated with the Bouchon and Herrmann programs, the WKBJ synthetic seismograms contain only the number of phases selected by the user. The execution time for one run of the program is very short. In addition to making the synthetic seismograms, the program calculates the arrival times of these phases, and write them both on the screen and in the iasp.out file for later plotting (see MULPLT). This is intended to be a tool to help identify phases on the data or on the Herrmann or Bouchon synthetic seismograms: it can by no means replace these two programs, which are much better than WKBJ to model the frequency-dependent character of crustal phases at regional distance.

WKBJ seismograms have been introduced in seismology by Chapman [1978]. More details on the method can be found in Dey-Sarkar and Chapman [1978] and in Chapman and Orcutt [1985]. The core of the present program is a code written by Chapman et al. [1988] and is part of the seismological software distributed freely by IASPEI. The synthetic seismograms are given in displacement. Although their spectra contain low frequencies, one should bear in mind that they represent a high-frequency approximation of the wave field. They include a number of non-physical phases due to truncation of the integrals in slowness p. For the most interesting crustal phases, the epicentral distance is usually much larger than the source depth, and these phases interfere with the physical phases and modify their amplitudes.

The head waves on an interface appear automatically as a by-product of the reflected phases, as soon as the epicentral distance is larger than critical. That means for example that the Pn phase appears automatically on the synthetic seismogram as a by-product of the PmP phase. In order to synthesize or calculate the arrival time of a Pn or Sn phase, you must then specify 'PmP' or 'SmS' (see below).

For a receiver at the free surface, the synthetic seismograms must include the free surface reflection coefficient to yield correct amplitude and waveform for the different phases. For S phases, at epicentral distances larger than critical, this includes automatically the SP phase (a P phase which propagates horizontally along the free surface, and which originates from the critical conversion of S to P at the free surface). The critical distance is of the order of the source depth for the Sg phase, and its SP phase usually appears as a large arrival between the P and S wave. The SP phases are physical, but the amplitude of their high frequency part is overestimated with WKBJ. If one wishes to suppress them from the synthetic seismograms, one may optionally do so. With this option, the surface reflection coefficient is omitted and

the synthetic seismograms contain only the upgoing wavefield, that is the wavefield one would get in a borehole, after filtering out the downgoing wavefield. Let us note that this option may strongly modify the amplitudes and waveforms of the different phases compared with those at the free surface.

In addition to the synthetic seismograms, the program calculates the arrival times of the phases you have specified, and write them in the iasp.out file. These times are calculated by interpolation in epicentral distance of the values tabulated in wkbj.tab. For sources close to an interface (in practice for Pg and Sg phases and the source under an interface), there is a limited epicentral distance range in which an arrival time can be calculated. For example, the maximum epicentral distance for Pg is about 250km for a source 0.1 km under Moho in the default SEISAN model. In order to increase the maximal epicentral distance, you may move the source away from the interface, or you may increase the number of ray parameters used in program wkbj_or.for (parameter 'nnpp') called from wkbj.for.

All three programs are hardwired to use triangular sources.

Running the programs

The programs require input about distances, azimuths, depth, crustal model, fault plane solution, time window, number of points and some modeling parameters. Almost all of these parameters are available within SEISAN. The programs have therefore been modified to use an S-file (Nordic format) as input file with additional information about time window, number of points to model and crustal model. A special format has been used to keep the modeling information separate from other information in the file (see below for an example). The steps to model a particular event are as follows:

Problem Bouchon: Use fewer layers, ideally just a halfspace under the deepest ray. The programs seems to become unstable if too many layer are used there.

Step 1
Edit the event in EEV and mark the stations wanted for modeling with a minuscule s in column 1, ONLY mark the station once. Exit from editor and, within EEV, give the command "synt". This will generate all the necessary default input parameters for modeling, which are stored as comment-lines starting with SYNT in the S-file (see below). At the same time, the s's used as markers are removed. Any old modeling information present will remain and override the defaults.However, in case the F-flag is set for the DEPTH parameter, distances and azimuths will be reset according to the current location.

Step 2
Edit event again and check if default parameters are ok (see explanation below).

Step 3
Run one of the programs BOUCH, HERRMANN or WKBJ. These are known commands in EEV. BOUCH: The program will now run for a certain amount of time depending on number of points required. At the standard output, the input parameters used will be printed out and for each frequency, the number of terms in wave number integration is printed out. If the limit of the number of terms is reached, something is wrong, try other parameters. The limit is 2. BOUPAR parameter, currently set at default value of 2000. The speed of this output (NPOINT/2+1 lines) gives a good indication of how long time it will take.
HERRMANN: Takes longer than BOUCH.
WKBJ: Very fast.

Step 4
Generate the seismograms. BOUCH: Use program BOUSEI. The program is interactively asking the seismogram type (displacement, velocity or acceleration). BOUSEI will generate a file bousei.out in SEISAN format containing both original and synthetic traces. The number of traces is determined by the specifications for each station, see below. Output file is `bousei.out`.

HERRMANN: Use program HERSEI, similar to BOUSEI. Output file is `hersei.out`.
WKBJ: The first command WKBJ also makes the seismograms. Output file is `wkbjsei.out`.

In all cases it is possible to shift the original trace relative to the synthetic trace and the program will ask, for each channel, how much it should be shifted. A positive value shifts the real trace up in time (to the left). The default is to shift the trace the amount of the P-travel time residual of the first P found in the S-file for that station in order to line up the P - phases. NOTE: These phases MUST be the same phase types in order to be lined up. If the first modeled phase is Pn and the first observed phase given in the S-file is Pg, there will be a no alignment. The amplitudes for Bouchon are in nm, nm/sec or nm/sec*sec (hopefully !!) assuming a seismic moment of 10 **22 dyne-cm. The output file will normally contain both the original and synthetic traces. However, if no waveform file is available (in local or WAV directory), the output file will contain an empty channel where the original data should have been. The specifications in the `hyp.out` file determine which traces from the modeled stations are included in the output file. If the specification after STATION is only component (e.g. S), then all 3 channels are shown. If a particular channel is given (e.g. S N), then only that channel is shown. Only one or 3 channels can be displayed.
All output traces are given in Z, N and E or Z, R and T depending on the parameter file (see below). The channel names are SH, SB and SW for Herrmann, Bouchon and WKBJ respectively.

Step 5
Plot the traces with mulplt. This can be done within EEV using the command pw, ph or pb for WKBJ, Herrmann or Bouchon respectively. Since there is no instrument correction, it is a good idea to plot both the modeled and observed signals narrow band pass filtered. E.g. for regional events 0.1-1 Hz and for small local events 2- 5 Hz (depending on sample rate).

shows an example of the modeling.

Note: The whole modeling process can be done entirely within EEV and it is intended to be done so. Since the modeling requires updated distances, depths etc when changing model etc, it cannot take its input from the location in the S-file, which only changes when doing an update (see UPDATE program). So when running from within EEV, a location will always be done first to get an updated S-file (in this case the `hyp.out` file) and this is the reason that the modelling programs use the `hyp.out` file instead of the S-file for input. This also means that the modeling program can be run separately from any `hyp.out` file, however it is then up to the user to keep it updated.

The modeling parameters

Below is shown an example of part of an S-file prepared for modeling. The file is one of the events in the test data set and by using EEV to find the event, modeling can start immediately. All parameters have been set automatically.

```
1996  6 7 1325 29.1 L* 59.841   5.127 12.0F BER 12 1.1 2.2CBER 1.9LBER 2.0LNAO1
1996  6 7 1325 29.1 L*                      BER        2.0WBER                1
      8.3      41.0      74.7     1                                          F
1996-06-07-1324-51S.TEST__009                                               6
     535  SOUTHERN NORWAY                                                   3
depth fixed to 12 km, rms lowest with near station (less than 110 km) location3
fault plane solution ok within 10 deg                                       3
SYNT: MODEL--:     THICK        VP        VS      DENS        QP        QS    3
SYNT: MODEL--:    12.000     6.200     3.563     2.600     0.000     0.000    3
SYNT: MODEL--:    11.000     6.600     3.793     2.800     0.000     0.000    3
SYNT: MODEL--:     8.000     7.100     4.080     3.000     0.000     0.000 B  3
SYNT: MODEL--:    19.000     8.050     4.626     3.200     0.000     0.000 N  3
```

```
SYNT: MODEL--:    30.000     8.250     4.741     3.400     0.000     0.000    3
SYNT: MODEL--:    50.000     8.500     4.885     3.600     0.000     0.000    3
SYNT: ST-D-RK:       8.3      41.0      74.7                                   3
SYNT: DEPTH--:      12.0                                                      3
SYNT: NPOINTS:       512                                                      3
SYNT: TIMES--:     TOTAL    60.000   INITIAL     0.000  SY-TRACE    60.000    3
SYNT: BOUPAR-:     800.0      2000     0.010                                  3
SYNT: PHASES-:        Pg        Sg       PmP       SmS       SmP              3
SYNT: DT-Tsou:     0.050      .100                                            3
SYNT: REDVELO:    0.0000                                                     3
SYNT: COMPON-:    RADIAL                                                     3
SYNT: STAT-AT:  no                                                           3
SYNT: NSTAT--:         3-----------------------------------------------------3
SYNT: NEW STAT:------------------------------------------------------------3
SYNT: STATION: EGD  S  Z  DISTANC:      49.0                                 3
SYNT: STATION: EGD        AZIMUTH:       9.0 BAZIMUTH:     189.2             3
SYNT: NEW STAT:------------------------------------------------------------3
SYNT: STATION: KMY  S  Z  DISTANC:      71.0                                 3
SYNT: STATION: KMY        AZIMUTH:     172.0 BAZIMUTH:     352.7             3
SYNT: NEW STAT:------------------------------------------------------------3
SYNT: STATION: ASK  S  Z  DISTANC:      72.0                                 3
SYNT: STATION: ASK        AZIMUTH:       5.0 BAZIMUTH:     185.1             3
STAT SP IPHASW D HRMM SECON CODA AMPLIT PERI AZIMU VELO SNR AR TRES W  DIS CAZ7
EGD  SZ IP     C 1325 35.95   93                                -1.110   48   6
EGD  SZ ES       1325 42.03                                      -1.010   48   6
BER  SZ IP     C 1325 38.12   55                                -1.010   62  11
BER  SZ ES       1325 45.44                                      -1.110   62  11
BER  SZ E        1325 46.71        31.7  0.2                            62  11
ASK  SZ EP     D 1325 39.59   68                                -0.910   70   3
ASK  SZ ES       1325 48.07                                      -0.810   70   3
ASK  SZ E        1325 50.90       111.0  0.3                            70   3
KMY  SZ IP     C 1325 40.26   62                                -0.410   71 175
KMY  SZ ES       1325 48.74                                      -0.410   71 175
KMY  SZ E        1325 48.92        83.6  0.2                            71 175
```

MODEL: The model to be used. THICK is layer thickness, VP is Vp velocity, VS is Vs velocity, DENS is density and QP and QS, are P and S q-values respectively. The model, velocities and Q-values are taken from the `STATION0.HYP` file with first choice from current directory and second choice from DAT directory (like the HYP program). The S-velocities are calculated using the Vp/Vs ratio given there. Moho is indicated with N at the end of the line with the first mantle layer. A Q of zero means infinite Q. The densities are approximate values and should be modified. See below for maximum number of layers.

ST-D-RK: Strike dip and rake is taken from an existing fault plane solution for the given event (F-line) if it exists, otherwise arbitrary values are supplied. (0,0,0) is an explosion. The convention is Aki and Richards.

DEPTH: Focal depth is taken from the current solution. The second field can optionally have the letter F (right justified). If this flag is set, the user can give the synt command to update all distances and azimuths used for modeling which will correspond to the latest location determined as e.g. a result of a changed fixed depth or a changed model. The intention with this flag is that the user should be able to set a fixed depth in the S-file header line, give the synt command to update the parameters for modeling

corresponding to this depth and then model.

NPOINTS: Number of points to model, 512 is set as default, must be 2**N. Used by BOUCH and HERRMAN only.

TIMES–: Three different times:
TOTAL: The total time window for generating data and synthetic seismograms for all channels, see also REDVELO.
INITIAL: The initial time of the earliest trace in the output file, with reference to the source origin time. The synthetics at the station with smallest epicentral distance automatically start also at this initial time.
SY-TRACE: The duration of the synthetic seismogram for each channel, might have different start times, see REDVELO.

DT-Tsou: Sampling interval (used for WKBJ seismograms only), and half-duration of the source used for all three programs. In all programs, the source is triangular, however BOUCH can optionally use several sources, see below.

REDVELO: Reduction velocity to calculate the initial times at subsequent distances (put 0. for no reduction velocity). NOTE: Seems to not be correctly implemented so use 0 always.

PHASES-: The names in format A4 (right justified) of the phases to be synthesized with WKBJ. The phases may be given in any order, with a maximum of 6 phases per line, and there may be several "SYNT: PHASES-" lines.

Possible phases:

Pg (direct P from source to receiver)
Sg (direct S)
PmP (includes automatically Pn at distances larger than critical)
pPmP (includes automatically pPn at distances larger than critical)
sPmP (includes automatically sPn at distances larger than critical)
SmS, pSmS, sSmS (includes automatically Sn, pSn, sSn at distances larger than critical)
SmP, PmS
P1P, P2P, S1S, etc: the same as PmP, SmS etc, but on interface number 1, 2, etc.
(The free surface gets interface number 0 in the convention taken here. Thus in HYP, PN2 is the same as P1N here. There associated head waves are labeled Pn1, Pn2, Sn1, etc.

COMPON-: RADIAL for radial-transverse components, NORTH for North-South, East-West components.

STAT-AT: Is "not free" or "NOT FREE" anywhere within column 16 to 25: Optional line. If this option is chosen, the WKBJ synthetic seismograms are calculated omitting the reflection coefficient at the free surface, at the receiver location.

BOUPAR: Modeling parameters L, Nt and e. L is length of periodicity (should be a few times the hypocentral distance), Nt is maximum number of terms in wave number summation and e is the value used in truncating the summation. Increasing e and decreasing Nt will speed up convergence, but the results might be unreliable. If Nt is reached, the results are unreliable.

NEW STAT: Comment line

STATION: Station to be modeled with component(s) to be displayed. The S means that short period instruments are used. The default is S, so if e.g. BH is used, S must be chaged to BH, else the waveform data is not found. If no component is given, all 3 components are assumed. The other option is to indicate a component (e.g. Z) and only that component will be displayed (see also description of BOUSEI).

DISTANC is epicentral distance used, this distance is taken from the current location, AZIMUTH is azimuth from the source to the station taken from current location, BAZIMUTH is the back azimuth at the station, calculated by EEV, used to rotate if so specified. Each new station isrepresented by the above 3 lines.

The source time function

The time duration of the triangular source time function for Bouchon is given as Tsou above, and is also used in WKBJ and Herrmann.

Hints on modeling

Event 199606071325 in the test data set is set up with modeling parameters and can be tested immediately.

The model

The standard model given in `STATION0.HYP` might be too detailed for most cases and should be simplified to include 3-4 layers by just editing the S-file, this also speeds up modeling. However, if you located the event with one model and model with another, the distances and residuals might not fit. A solution could be to have a `STATION0.HYP` in the local directory with the simplified model.

Alignment of P and S

If the distance calculated by HYP is not correct as indicated by P and S residuals, the synthetic and observed signals will not be aligned. The distance for that station can then be changed manually in the S-file under DIST and/or delays can be applied when generating the seismograms. For line up, it is important that the correct first arrival is included in phase list (WKBJ), see what is identified by HYP. If PN2, then P1N must be given for WKBJ.

Testing different parameters

There is no need to go back to EEV to test for the parameters that do not change the location. Thus to test for different fault plane solutions, time windows, number of points, edit the `hyp.out` directly and rerun. However, if depth or model is changed, relocation must be made. To test for different depths, locate with fixed depths, see HYP.
NOTE: THE SOURCE AND RECEIVER CANNOT BE AT THE SAME DEPTH (BOUCH AND HERRMANN) AND IN NO CASES CAN THE SOURCE BE AT DEPTH ZERO.

Running time

This depends mostly on the number of points and to some degree on number of layers. The number of stations has an insignificant effect on running time.

Program limitations: HERRMANN and WKBJ is set up with max 20 layers and Bouchon with 20 layers. Maximum of 32 stations Change programs and recompile if more layers are needed. Bouchon is compiled for 2048 frequencies (4096) points.

Computer notes:

The original Bouchon program BOUCH is almost unchanged. The only modification is that it uses a subroutine to generate its original input file bouch.inp from the `hyp.out` file. This file still remains after running BOUCH for debugging purposes. The output from BOUCH is bouch.out, which in turn is input to BOUSEI.

Herrmann:
The Herrmann waveform modeling is based on a concept where the synthetic seismograms are computed through a sequence of four distinct processes (programs).

1. The program "hspec8" will calculate the medium response for 10 basic Green's functions, where the response is given in frequency - wavenumber domain F(f,k).
2. The program "rhwvinta" will integrate and take the medium response from F(f,k) → F(f,r)
3. The program "rhfoc10" will convolve the response function with a source time function and with inverse Fourier transform take F(f,r) → F(t,r)
4. The program "mech" will construct a 3 component synthetic seismogram given a focal mechanism.

Herrmann's programs originally had several optional source time functions, however, a triangular source has been hardwired (for all 3 programs) so it is easier to compare the results. The original options can be reactivated by editing the program.
The programs HERRMANN and HERSEI run these 4 programs in an automated sequence.

All References, a detailed manual, source code and parameters as well as other related programs: "Computer Programs in Seismology", Volumes I - VIII. By Robert B. Herrmann, Saint Louis University, Saint Louis, Missouri.

WKBJ:

Input file `hyp.out`, read by "WKBJ.
Output file `iasp.out`, written by "WKBJ". Contains the arrival times of
the different phases at the stations, in SEISAN format.
Output file `wkbjsei.out`, written by WKBJ in the SYNTSEL.FOR subroutines. A waveform file (SEISAN type) containing the data and the synthetics, which can be plotted using "mulplt". Note that there is a code for each synthetic seismogram giving the modeling method (SH: Herrmann, SB: Bouchon, SW: WKBJ), and the component (Z, R, T, N or E).

INTERMEDIATE FILES
`wkbj.inp`, created by WKBJ for input to WKBJ_OR. The same information as in `hyp.out`, in a WKBJ_OR format.
`wkbj.tab`, output from WKBJ_OR, reprocessed by WKBJ. Contains tables as a function of ray parameter.
`wkbj.out`, output from WKBJ_OR, reprocessed by WKBJ. Contains the Green functions.

6.20 Calculation and plotting of travel times

In SEISAN, travel times are generated from a flat crustal model or using the IASP91 global travel time model. It can often be useful to generate travel times for given distances and two programs are supplied to do these calculations. TTIM will calculate travel times for global phases at one given distance and depth and TTLAYER, will calculate a travel time table (layered flat model) for a given depth and a distance range. A special version of TTIM called IASP is used in connection with EEV and MULPLT.

6.20.1 IASPEI travel time software, program TTIM

This program can be used for calculating global travel times, see below for details on phases calculated. The program assumes that you have the travel time tables in the working directory or in DAT, see computer notes below on how to generate these file if not already there. The same files are also used by HYPOCENTER.

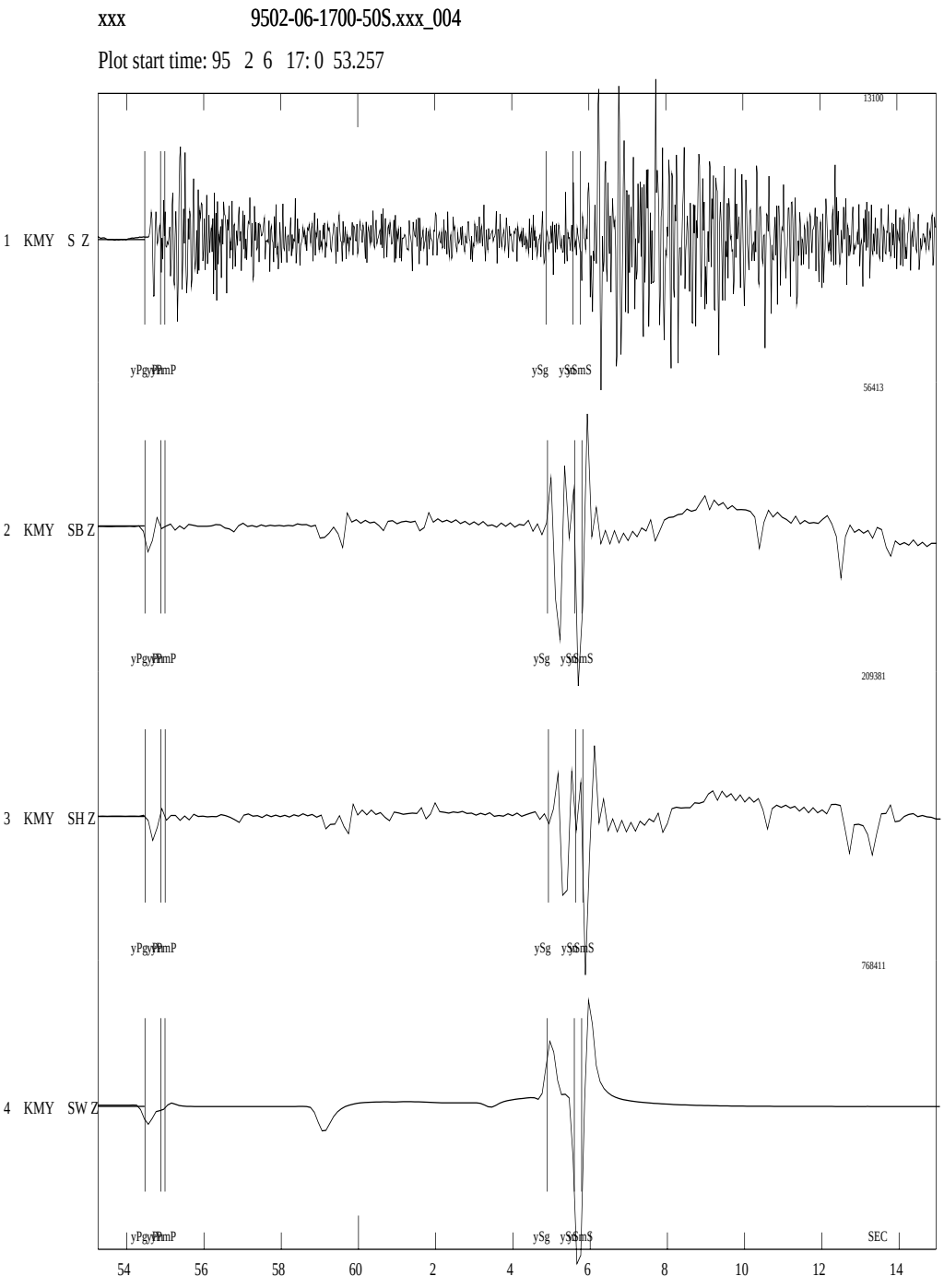


Figure 6.30: An example of synthetic seismograms using Bouchon(2), Herrmann (3) and WKBJ (4). The original seismogram is shown in channel 1. All synthetics are displacement. Also shown are the theoretical travel times calculated by WKBJ.

After starting the program, the first two questions 'do you want xxxxx' relate to range summaries, etc., that are normally not required and can be answered with n(no) followed by ENTER. The program then asks 'Enter phases, one per line...' You can then enter a specific phase, or a keyword defined as follows:

All	gives all phases
P	gives P-up, P, Pdiff, PKP, and Pkikp
P+	gives P-up, P, Pdiff, PKP, Pkikp, PcP, Pp, Ppdiff, PPKP, PPKIKP, Sp, Spdiff, SPKP, and SPKIKP
S+	gives S-up, S, Sdiff, SKS, Ss, Ssdiff, SSKS, Ps, Psdiff, and PSKS
basic	gives P+ and S+ as well as ScP, SKP, PKKP, SKKP, PP, and P'P'

Writing all individual phases, separate by ENTER, terminating the list with an additional ENTER. The program will then enter a loop where phase times are calculated for new distances entered on request. The program is terminated for a particular distance by entering -1, and a new depth can be used, or the program can be terminated by entering -1 again.
A special version of this program used in connection with MULPLT is IASP.
In order to generate the earth model files IASP91.HED and IASP91.TBL, first run program REMODL, then program SETBRN. The program REMODL has the earth model hardwired. Note: These binary files CANNOT be moved between platforms. They are included with SEISAN for each respective distribution. If lost, they must be regenerated on the same platform.
For more information about IASP91 programs, see HYPOCENTER manual by **B. Lienert**.

On at 64 bit computer the IASP files must be regenerated is you have the files from a 32 bit computer, with the programs `REMODL` and `SETBRN`.

6.20.2 Calculation of travel times for layer and gradient model, TTLAYER

The TTLAYER program is written by Barry Lienert to calculate travel times for both layer and gradient model. In this version the program only works for zero depth, and therefore might not be very useful. The program reads a set of velocities and depths from an input file in '`STATION0.HYP`' format and calculates travel times for P and S velocities for a set of uniform-velocity layers, using the HYPOCENTER dtdx2 routine and also for a set of uniform gradient layers, using dtdxg, a new routine written to have the same input arguments as dtdx2.
The routine to calculate travel times for a gradient model uses an adapted version of Fred Klein's TTCAL routine, which he uses in his program TTGEN to generate a table of values from which to interpolate travel times and their derivatives in HYPOINVERSE.
The program is easy to run and the output can be plotted with some standard xy plotting tool.

6.20.3 Plotting of travel times, TTPLOT

Program to plot observed and calculated travel times (Figure 41). The input to the program is an s-file, which has an indicator to a model file (`STATION?.HYP`) and the travel time observations. The program is started by 'ttplot <sfile-name>'. At the start, TTPLOT relocates the event and calculates distances using the HYPOCENTER program. It then plots all observations with a '+' symbol and the theoretical travel times that are calculated by the program for the first P and S arrivals with solid lines. The program can be useful in routine processing to visualize large residuals, which otherwise are seen from the location program output. The program can also be started from EEV using option 'ttplot'. It is possible to click on symbols, which will bring up station code, phase, observed travel time and residual on the rigth. The output files are:

`ttplot.out` - gives station code, phase name, distance, observed travel times and residual.
`ttplot.eps` - Postscript version of plot.

6.20.4 IASP, travel times for MULPLT

This program is a special version of IASP91 to be used in connection with EEV and MULPLT. Giving command iasp from the EEV prompt (or from within MULPLT), the program will read the current active S-file, and for each station, calculate possible IASP91 phases and arrival times relative to the hypocenter and origin time given in S-file. The origin information can be obtained from two places in the S-file: (1) The header lines are searched for hypocenter lines and the first found after the main header will be used, (2) If no secondary header lines, the main header line is used. The intention of this order is that it is possible to put in a PDE solution in a secondary header line (option INPUTONE in EEV) so that theoretical travel times are calculated relative to a fixed solution and not the temporary solution made by the local agency.

The IASP91 tables can be found in the local directory or DAT and have the same names as used in HYP and TTIM. The program generates an output file iasp.out in Nordic format. This file is read by MULPLT and the theoretical phases displayed on the screen. The number of phases calculated can be very large making it hard to see which phase is which. IASP therefore has a definition file, IASP.DEF, where phases to be written out are given. The file can be in the working directory or in DAT. If no definition file is available, all phases will be written to the iasp.out file. Below is an example of a IASP.DEF file.

```
This file contain the definitions of phases to be used when calculating
synthetic phases to be plotted with mulplt. There is one phase pr line
and each phase is preceded with the keyword IASP-PHASE. Only lines with this
keyword will be read. The defined phase then follows in column 13 to 20.
If no phases are defined, all ISPEI91 phases will be used.

Phase ID    Phase---
--------------------
IASP-PHASE  P
IASP-PHASE  PP
IASP-PHASE  PPP
IASP-PHASE  PKP
IASP-PHASE  pP
IASP-PHASE  sP
IASP-PHASE  PcP
IASP-PHASE  S
IASP-PHASE  SS
IASP-PHASE  SSS
IASP-PHASE  SKS
IASP-PHASE  ScS
IASP-PHASE  PS
IASP-PHASE  SP
IASP-PHASE  ScP
```

6.21 Inversion for Q_{Lg}, QLG

The QLG program can be used to determine an average Q_{Lg} or to perform a tomographic inversion. The method is described in Ottemöller et al. [2002]. Here, we use the same names for the damping parameters, and many of the other parameters should be self-explanatory. The program can also produce the input for distance trace plots. Note that using the program is no trivial task. The data set needs to be carefully selected and the instrument calibration has to be known. The input to the program is a Nordic file, which includes several events. The parameter file needs to be carefully set up.

The program can be used in the following way:

1. Determine average Q_{Lg}
2. Perform checker-board test to chose damping parameters
3. Tomographic inversion

Note: The main purpose of including the program is to give an example source code so that the user can make use of it when implementing similar programs. The program uses a linear grid...

Example of the parameter file `qlg.par` :

```
KEYWORD.....Comments.....Par 1.....Par 2.....

FILTER             for distance plot    0.01      15.
DISTANCES          min and max          200.      3000.
GROUP VEL LG       lg group vel window  3.0       3.7
GROUP VEL P        p group vel window   5.0       8.0

INVERSION TYPE     1. for tomography    1.
                   0. for average
ORIENTATION        0.=vert, 1.=horiz.   0.
PHASE ONLY         1.=phase pick requ.  1.
                      in s-file

FREQUENCY          frequency and 1/q    1.        5.2392E-03
FREQUENCY          frequency and 1/q    1.25      4.5246E-03
FREQUENCY          frequency and 1/q    1.60      4.1239E-03
FREQUENCY          frequency and 1/q    2.        3.5312E-03
FREQUENCY          frequency and 1/q    2.5       2.9081E-03
FREQUENCY          frequency and 1/q    3.15      2.2568E-03
FREQUENCY          frequency and 1/q    4.        1.7029E-03
FREQUENCY          frequency and 1/q    5.        1.1228E-03

STATION MIN        min # of stations    4.
VELOCITY LG                             3350.
DAMPING ALPHA      damping parameters   500.
DAMPING SIGMA      ------------------   100.
DAMPING BETA       ------------------   500.
DAMPING LAMBDA     ------------------   0.001
NSMOOTH            smooth spec # times  0.
```

```
CHECKERBOARD       1. for cb-test       0.
CHECKERBOARD DELTA                      0.0004
FIX SITE                                0.
FIX SOURCE                              0.
SOURCE PERTURBATION                     7.        0.2
GAUSSIAN NOISE                          0.1
VERBOSE            0. for quite mode    1.

#
# Grid
#
X START            x start of grid      -92.5
Y START            y start of grid      6.50
X DELTA            x delta grid         1.
Y DELTA            y delta grid         1.
X NPOINTS          x # points           17.
Y NPOINTS          y # points           13.
```

Menke et al. [2006] pointed out the non-uniqueness in attenuation tomography between the source term and Q. They suggest to investigate the non-uniqueness by synthetic tests in which a perturbation is applied to the source term and the inversion for Q is done without inverting for differences in the source term. The solutions obtained are null-solutions and one needs to be careful not to mistake them for real patterns. These tests are possible within QLG by setting the parameter 'SOURCE PERTURBATION, where the first parameter refers to the source that is perturbed and the second parameter gives the amount of perturbation in units of moment magnitude.

It is possible to invert real data without inverting for the site term by setting 'FIX SITE'. This can be a useful test as there is also a trade-off with the site term. Fixing the site term is more problematic, as this is done based on the local magnitude, which may not be the same as the moment magnitude.

Another useful stability test is to add Gaussian noise to the spectra and check the inversion result. This can be done for both real data and the checkerboard test by setting the parameter 'GAUSSIAN NOISE', units are equivalent to change in moment magnitude.

6.22 Wadati

This is a program to make Wadati diagrams and apparent velocity from a Nordic file with one or many events. The apparent velocity is calculated from the arrival times and the calculated epicentral distances as given in the S-file. The apparent velocity is thus approximate and affected by the location.
The purpose of the program is to calculate Vp/Vs values for individual events and calculate the average for a group of events. In addition, the program can calculate the apparent velocity for each event based on P or S-times. Wadati diagrams with plot can also be calculated directly from EEV.

The information can be used to obtain a first impression of crustal parameters. For each calculation, events can be selected based on: Minimum number of stations, maximum rms of the fit (S-P vs P, or arrival times), and minimum correlation coefficient of the fit. For the apparent velocity calculation, the data can also be selected in distance and azimuth ranges.

The output gives:

T0 :	Wadati calculated origin time
N :	Number of stations used for Vp/Vs
VPS :	Vp/Vs ratio
NP :	Number of stations for P- velocity
NS :	Number of stations for S-velocity
AVSP:	Average S-P times with sd
AVDI:	Average distance with sd

The average Vp/Vs is calculated for the whole data set. Individual Vp/Vs values outside the range 1.53 to 1.93 are excluded. An output file wadati.out is generated. A minimum of 3 stations is required for an event to be used. Only same type phases are used (like PG and SG).

Example of a run to calculate Vp/Vs

```
Input file name
collect.out
 Wadati (1), apparent velocity (2) or both (3) ?
1
 Wadati parameters:  Minimum number of stations
3
 Maximum rms
1
 Minimum correlation coefficient
0.9
 1994  616 1841 28.3 No data for Wadati
 1994 10 4 1322 55.8 No data for Wadati
 1995  822 0141  5.3 T0:  141   8.1 N:    8 VPS:  1.84 RMS:   1.19 CORR:  0.997
 1995 1120 0401 58.9 T0:  4 1  59.5 N:    7 VPS:  1.76 RMS:   0.87 CORR:  0.995
 1996  6 3 1955 35.6 No data for Wadati
 1996  6 3 1955 35.6 No data for Wadati
 1996  6 3 1955 40.1 No data for Wadati
 1996  6 6 0648 29.8 T0:  648   1.1 N:   22 VPS:  1.51 RMS:  11.00 CORR:  0.380
 1996  6 6 0648 30.6 T0:  648  38.7 N:    3 VPS:  2.16 RMS:   0.80 CORR:  1.000
 1996  6 6 0648 29.8 No data for Wadati
 1996  6 7 1325 28.5 T0: 1325  28.1 N:    9 VPS:  1.72 RMS:   1.06 CORR:  0.973
 1996  6 7 1325 29.1 T0: 1325  28.1 N:    9 VPS:  1.72 RMS:   1.06 CORR:  0.973
 1996  610  1 4 47.0 No data for Wadati
 1996  610  1 4 47.0 No data for Wadati
 1996  623 0117 57.8 No data for Wadati
 1996  623 0117 58.1 No data for Wadati
 1996  625 0337 31.7 T0:  337  33.1 N:   21 VPS:  1.75 RMS:   1.36 CORR:  0.999
 1996  7 5 0220 46.5 T0:  220  45.8 N:    6 VPS:  1.76 RMS:   0.28 CORR:  0.999
 1996  713 0556 46.0 No data for Wadati
 1996  718 0946 51.4 No data for Wadati
 1996  718 2255  6.0 No data for Wadati
 1996  726 0742 12.0 T0:  742  11.8 N:    6 VPS:  1.74 RMS:   0.47 CORR:  0.993

 Number of events for which vp/vs were calculated            9
 Number of events selected for average                       3
```

```
 Average VP/VS =  1.75  SD=  0.01 N=    3
```

Example of a run to calculate apparent velocity

```
Input file name
collect.out
 Wadati (1), apparent velocity (2) or both (3) ?
2
 Apparent velocity parameters:  Distance range
50 200  Azimuth range
0 180
 Minimum number of stations
2
 Maximum rms
1
AVSP:  30.212.7  AVDI: 152.0  30.7 NP:    4 VP :  6.95 RMS:   0.20 CORR:  1.000
                 AVDI: 143.3  31.1 NS:    3 VS :  3.82 RMS:   0.25 CORR:  1.000
AVSP:  23.0 9.5  AVDI: 158.7  26.6 NP:    3 VP :  6.76 RMS:   0.00 CORR:  1.000
                 AVDI:  90.5  35.8 NS:    6 VS :  3.72 RMS:   1.79 CORR:  0.998
                 AVDI:  89.0  35.6 NS:    6 VS :  3.69 RMS:   2.78 CORR:  0.996
AVSP:  50.536.0  AVDI: 145.8  40.8 NP:    6 VP :  7.60 RMS:   0.42 CORR:  1.000
                 AVDI: 116.3  38.2 NS:    3 VS :  3.97 RMS:   0.04 CORR:  1.000
                 AVDI: 118.8  45.6 NS:    4 VS :  3.61 RMS:   0.66 CORR:  1.000
AVSP:  14.4 7.5  AVDI: 106.0  39.2 NP:    5 VP :  6.75 RMS:   0.94 CORR:  0.999
                 AVDI: 111.7  37.7 NS:    6 VS :  3.82 RMS:   0.97 CORR:  0.999
AVSP:  14.4 7.5  AVDI: 176.7   1.5 NP:    3 VP :  5.13 RMS:   0.07 CORR:  1.000
                 AVDI: 178.0   2.9 NS:    4 VS :  2.80 RMS:   1.06 CORR:  0.859
AVSP:  22.0 5.7  AVDI: 150.0   4.2 NP:    2 VP : 10.17 RMS:   0.00 CORR:  0.000

 Output file is wadati.out
```

6.23 Calculating spectra, the SPEC program

The SPEC program is used for making spectra of many seismic signals in a semiautomatic manner. It can be used for several investigations:

A: Making a large series of signal spectra, which can be corrected for instrument and path.

Average spectra are calculated. There are two options for further processing the calculated spectra:
Option (1): Calculate acceleration density spectra which are plotted compared to the Peterson noise model.
Option(2): Using the slope of the flat part of the displacement spectra to calculate the near surface attenuation kappa (see 6.2.12.

B: Making relative spectra of seismic events or background noise in order to determine the soil response. When using relative spectra of horizontal versus vertical components, this is referred to as the Nakamura method [Nakamura, 1989].

C: Making relative spectra of signals from two stations in order to determine Q. The program makes output files for generating GMT plots in addition to standard SEISAN plots.

Note: Parameter file has changed between SEISAN 7.2 and 8.0 (number of windows and overlap has been added).
The program can technically operate in two ways: (1) Making relative spectra of a series of pairs of stations terminated by the average spectra, (2) Making a series of spectra for a number of stations and events. The spectra can be corrected for distance, Q, and instrument response. In addition, the spectral levels can be expressed in moment or moment magnitude calculated in the same way and with the same units as in MULPLT. All relevant parameters are taken from the CAT files, the CAL files and the input parameter file for SPEC. Window selection for the spectra can be specified to be related to the P, S arrival times or the earthquake origin time and it is thus possible to automatically make e.g. S-wave spectra of a large set of stations and events. Optionally, noise spectra, can be calculated together with the signal spectra. The noise window is selected at the start of the waveform file.
Before the program is started up, the input files must be prepared. The program need two input files. The parameter file (default spec.par) gives the parameters to use and the list of stations to process. The event file (default spec.inp) is a CAT file with events to use or a `filenr.lis` type file with waveform file names (can only be used if no readings are needed, like for Nakamura studies). An example of a spec.par and spec.inp file is found in DAT. These files can be used immediately with the test data set.

The program produces several output files. The main output is in `spec.out` with the parameters used, the station event combinations used and error messages. The other files are giving output of most graphs shown. These ASCII output files can be used in other plotting programs, however they have been specifically formatted for the SEISAN GMTXY plotting script. The number of files depends on number of stations used. Examples of files could be

`spec_all_ASK__S__Z.out`	All spectra from ASK, S Z
`spec_all_BER__S__Z.out`	All spectra from BER, S Z
`spec_all_gmt.out`	All spectra from ASK and BER
`spec_ave_ASK__S__Z.out`	Average spectrum from ASK, S Z
`spec_ave_BER__S__Z.out`	Average spectrum from BER, Z Z

In order to plot these files with GMTXY (only Unix), give e.g. command

```
gmtxy spec_all_ASK__S__Z.out
```

Limitations of amount of data: The program is set up to handle 100 spectra of up to 30000 points each for one run. The dimensions can be increased in spec.for, however the program must then be recompiled. The spectral windows are 10% tapered. The analyzed signals will be checked for clipping and rejected if clipped. A message is then given in spec.out

The `spec.par` file The file contains alternate lines of parameter names and parameter values, and must contain the number of lines shown in the example below.

```
selection criteria 1: P, 2: S, 3: S from P, 4: abs
2
start
1
window length, # of windows,overlap
5.0,1,1
number of times to smooth
5
gain factor of channel 1
```

```
1
noise spectrum 0: n 1: y
1
make relative spectras 1: y, 0; n
1
plot pics
1
frequency band to use
1.0 7.0
response removal: 0: none, 1: displ., 2: vel., 3: accel. 4. noise pow. 5. kappa
0
rotate: 0: no, 1: yes
0
q0, qalpha and kappa
-1.0,0,0
distance correction
1
minimum correlation and minimum sn for kappa
0.5  2.0
velocity and density
0.0 0.0
magnitude spectrum
0
stations and components, format a5,1x,a4,1x,a5,1x,a4
FOO   S  Z SUE   S  Z
```

The parameters:
Selection criteria: Determines how the start of the time window is selected. 1: Start with the P-arrival time, 2: Start with the S-arrival time, 3: Start with the S-arrival time calculated from the P-arrival time assuming a P to S velocity ratio of 1.78, 4: Start with 'start' (see next parameter) seconds after the origin time as given in the CAT file header. This option can be used if no readings are available in the CAT file. When using a P or S-time for start of window, the program uses the first P or S phase found in the CAT file for a given station. Component is of no importance here, so there is only a need for e.g. one P-time for the station being processed if 3 component data is used. This is also the case when rotating the signal, see below. However, on the trace plots, only readings on those components shown will be seen on the plots.

Start: If the selection criterion is 1,2 or 3, this is the number of P or S travel times (from the origin) used to find start time of window. Use 1.0 if the window shall start exactly at the phase time picked. If selection criteria is 4, start is the number of seconds after the origin time.

Window length, #of windows, overlap:

- Window length: Window length in secs for both signal and noise (if selected) .
- # of windows: If more than 1, spectra will be made in several windows following the first window and average spectra will be made. This option can only be used if selection criteria is 4. Used for noise studies or Nakamura studies.
- Overlap: Windows can overlap (factor < 1.0) exactly follow each other (factor=1.0) or have gaps (factor > 1.0). E.g. 0.9 is equal to 10 % overlap.

Number of times to smooth: Number of times to smooth, 0 means no smoothing.

Gain factor of channel 1: Factor that the spectral level for channel 1 is multiplied with. This can be used if the response shape is the same for the two channels and only the levels are different. If the shape is also different, set factor to 1 and use response removal below.

Noise spectrum: If 0, no noise spectrum, if 1, make noise spectrum. The noise window is taken from the beginning of the trace and the window length is the same as given above.

Make relative spectra: If zero, no relative spectra, if 1, make relative spectra. The relative spectra will appear one on each page, and the average relative spectrum on the last plot (see Figure 42). If no relative spectra are chosen, only one trace and one spectrum is shown per page and the average spectrum is shown on the final plot. MUST BE SET to 1 to calculate Q, see below.

Plot pics: If 1, the phase pics in the CAT file spec.inp will be plotted.

Frequency band used: Lower and upper frequency bands for the spectral plots.

Response removal: If 0, no response is removed, else 1: displacement, 2: velocity, 3: acceleration (units is nm, nm/s and nm/s*s), 4: Power spectral density in dB relative to ((1m/s**2)**2)/Hz. This option is used for seismic background noise studies., 5: Determine kappa. The flat part of the spectrum (frequency below corner frequency) is approximated by a straight line and kappa calculated for each event and the average at the end (in spec.out file and on final plot). The spectrum will normally be corrected for Q, BUT NOT kappa. For more details, see [Havskov and Ottemöller, 2010]. Make sure to set appropriate frequency limits and correct distance corrections. Can be used for both P and S-spectra. A cal file for each channel must be available in the CAL directory (see section 4.6). For relative spectra, the response removal has no importance if the response is the same for the channels compared. A simple correction can be made with "Gain factor of channel 1" parameter above. NOTE: If moment or magnitude spectrum is made, response removal MUST be 1.

Rotate components: If 1, the horizontal components are rotated. This means that if the user has specified N or E, radial or transverse respectively will be used instead. The original data remain unchanged. If start time of spectra are chosen by using P or S, there must be a reading from those components if the pics are to be plotted. If the parameter is zero, no rotation is done. See also MULPLT for more details of rotation.

Q0, qalpha and kappa:

Q-correction:
Parameters in Q-relation Q = Q0**qalpha used for spectral correction (see also section on MULPLT for standard attenuation relations). Only used if response is removed. If first 2 parameters are 0,0, no Q-correction. New from SEISAN7.2 is that a kappa correction also can be used (see MULPLT spectral section).

Calculation of Q:
If Q0 and qalpha is set to -1,0, the relative spectra will be used to calculate q as a function of f (see standard relations in MULPLT section) and the plots will show q as a function of f. This can be used for both P-waves and S-waves. The distance correction MUST be set, S-velocity must be given (see below) and it is recommended to assume body wave spreading (amplitude proportional to 1/distance, factor is 1.0 below). If the response of the 2 stations is not identical, correction for response must also be made. There must be an origin time and phase readings for components used must be available in order to calculate Q. Q is calculated as pi * f * (t2-t1) / (ln(A2(f)/A1(f)) + alpha * ln(t2/t2)), where A1 and A2 are spectral levels at frequency f for the two stations, t1 and t2 are travel times and alpha is geometrical spreading exponent (1.0 is body wave spreading). Q values lower than 1 and higher than 5000 are not used, the Q(f) plot might then display a long straight line. The Q=Q0*f**qalpha is calculated from the

'good' values'.

Distance correction alpha: The spectral amplitudes are multiplied by R**(distance correction) if different from zero. This option MUST be set if moment or moment magnitude options (see below) are selected as well as calculation of Q. However, it can be used without instrumental correction. For body waves, use 1. Note that the geometrical spreading use here is simpler than used in MULPLT.

Minimum correlation coefficient and minimum signal to noise ratio for kappa: The minimum correlation coefficient and signal to noise ratio for an event to be included in average kappa. The coefficients are from the linear fit to the flat part of the spectrum.

Velocity and density: Velocity (km/sec) and density (g/cm*cm) used for calculating moment spectra. If set to 0,0, no moment spectra are calculated. See section on MULPLT for details of calculation.

Magnitude spectrum: If 1, the spectral level is converted to moment magnitude, see MULPLT for details of calculation.

Stations and components: Station-component pairs used, one pair per line, format (a5,1x,a4,1x,a5,1x,a4). If no relative spectrum is used, the first station-component on the line is used.

Averaging in spec:
Q: For each frequency, the average linear 1/Q and corresponding sd is calculated. The upper and lower bounds are calculated by subtracting and adding the sd. These values are then converted back to Q and finally the log is taken. Only the 'good' individual values are used. There is a possibility that the lower bound becomes negative. In that case, the log Q is set to zero. Because the average is made in 1/Q, the upper and lower bounds curves will not be symmetric around the average Q-curve.

Power spectrum: For each frequency the dB values are averaged and upper and lower curves should be symmetric.

Kappa: Same as for Power spectrum.

Other spectra: The linear spectra or relative spectra are averaged. The sd used in the log spectra are calculated by subtracting log average spectrum from log(average spectrum + sd).

Running the program:
The program gets the first pair of stations (or one station) from spec.par, calculates the spectra using the list of events in spec.inp and at the end of the station list, calculates the average spectral ratios for all pairs (max 100). All spectra are then shown on one plot together with averages and standard deviation. Then the next pair of stations is processed in the same way and the program continues until the end of file spec.par. Each pair of stations with signals and spectra is plotted on one page. If no relative spectra are made, the plots look similar except that only one station is shown. Hard copy plots are made for each page and sent to the printer if specified (see below). The hard copy postscript file is called `spec.eps` and when the program finishes, a file with the last plot is available on the disk. For each spectrum (relative or single), the average spectrum (or Q) is calculated both as an average of the log spectrum and as an average of the linear spectrum. There is no frequency weighting and since all values shown on the plot are used, the average value will be more representative of the high frequency part of the spectrum since there are more values. This can be regulated by choosing another frequency range. The average spectra shown on the last plot are log-averages. If option to calculate Q is used, the plots show 1/Q as a function of frequency instead of relative spectra (proportional to relative spectra). For each event, Q0 and qalpha are calculated.

When calculating kappa, the average spectrum do not have much physical meaning since the averages are made from absolute spectra of events that might have very different moments. So the kappa calculated from the average spectrum is not to be used.

Interactive output of level and frequency: With a spectral ratio (or Q) plot on the screen, position the cursor at the point of interest on the spectrum and click. The level and frequency will now be displayed on the right side of the plot.

The output file spec.out gives details of the run like averages and missing data. The output file spec_ave.out gives the x and y-values of the average spectrum IF IT HAS BEEN PLOTTED ON THE SCREEN. File spec_rel.out gives the values of the relative spectra.

There are 4 interactive input options:

0: All spectra are calculated but not sent to the plotter or screen except the last plot with the average spectra (sent to both screen and printer). Used for checking the files or making a final run. If no relative spectrum is chosen, no final plot is made. For each station and event combination, check lines are written out on the screen.

1: All plots are shown on the screen, but not sent to the laser printer.

2: All plots are shown on the screen and at the same time sent to the laser printer.

3: No plots are shown on the screen, all are sent to the laser printer. For each station event combination, check lines are written to the screen.

How to run the program with only waveform files available: Two options:

(A) Using S-files
Step 1: Generate S-files in your local directory with AUTOREG,
Step 2: Make the spec.inp file with COLLECT.

(B) Using `filenr.lis`
Step 1 : Make a dirf of waveform files to use
Then use `filenr.lis` as input file name

With only waveform files and no readings in the spec.inp file, it is only possible to use option 4 (absolute time) for start criteria. Since the events have not been located, the "origin time" read from the S-files will be identical to the waveform file start time, so the parameter "start" can then be set to number of seconds after waveform file start time. Figure 42 shows an example.

Figure 42. An example of using the SPEC program. On top the original traces are shown with windows chosen, in the middle the spectra of each channel and at the bottom, the relative spectrum. Lower right shows the input parameters used. In some cases (kappa and Q) the values calculated for this case are also shown.

6.24 Seismic risk related programs

This section is written by **K. Atakan**. Extensive testing of the programs was done over the years by many users. **A.Ojeda**, performed testing and prepared input files for the CRISIS99 and CRIATT programs.

Introduction

Currently, the SEISAN package includes a series of stand-alone programs that can be used in a number of tasks that are needed to perform seismic hazard analysis. The basic requirements for performing a probabilistic seismic hazard analysis may be summarized as follows:

Homogenize the earthquake catalogue and assess the completeness
Define the seismic source zones.

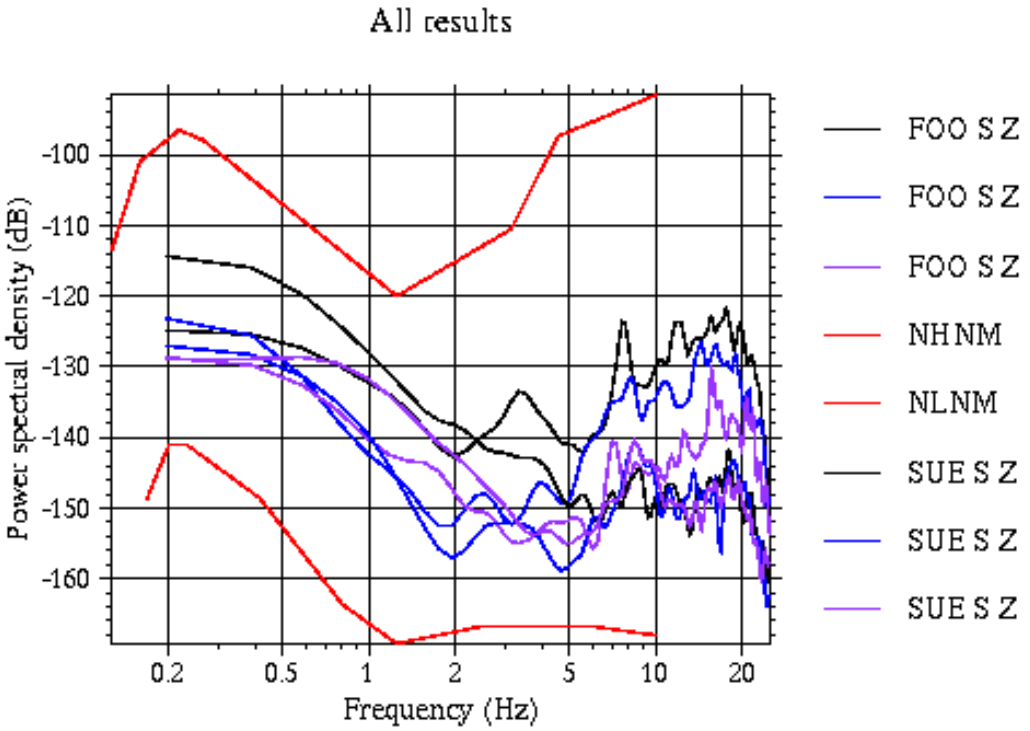


Figure 6.31: An example of a GMT plot. The figure shows an example of making noise spectra of several traces.

Prepare input parameters from the earthquake catalogue for each source zone.
Prepare attenuation relations for the region.
Compute hazard in terms of peak ground acceleration (PGA).
Assess site effects.
Prepare response spectra.

Following is a list of programs that constitutes the part of the SEISAN analysis package, which deals with seismic hazard and related problems. Most of these programs are described in more detail in different sections of the SEISAN manual.

SELECT: Select a subset of earthquake data according to given criteria.
STATIS: Statistical information about the database is computed and can be used in the analysis.
CATSTAT: Program to compute and plot the yearly, monthly and daily number of events from a given catalogue.
CAT_AGA: Program to reorder the hypocenter lines in a CAT-file according to hypocenter agency in order to put the prime estimate in the beginning.
CLUSTER: Program that searches for the dependant events in time and distance in a given earthquake catalogue.
EXFILTER: Identifies the probable explosions, based on the user-defined parameters involving time-of-day distribution and the mining locations. It can be used for catalogue clean up and discrimination between the earthquakes and man-made explosions.
MAG: Magnitude regression and conversion program. Prepares also a plot showing the scatter data and the best-fitted line. Magnitude conversions are then performed after a user defined priority list.
EPIMAP: Plots coastlines, national boundaries and earthquake epicenters. It can also contour the produced output map file from hazard programs such as EQRISK, and overlay on the epicenter map. It is also possible to select a subset of earthquakes from a chosen polygon on the epicenter map.
BVALUE: Prepares magnitude-frequency of occurrence diagrams and computes a- and b-values with maximum likelihood and least square approximation. In addition, the threshold magnitude and the maximum observed magnitude can be obtained.
CODAQ: Computes the Q value from a given set of seismograms. This can be used later in the CRIATT program to create the attenuation table.
CRIATT: Computes attenuation tables for a given set of parameters using the random vibration theory.
CRISIS99: Computes seismic hazard in terms of the probability of exceedance vs earthquake intensity measures such as peak ground acceleration (PGA) or any other spectral ordinate. It can also compute hazard for a given grid of map co-ordinates corresponding to user-defined different return periods. (**SUN and PC**). The Windows program must be installed separately, look for ZIP file in SUP.
EQRISK: Program to compute seismic hazard in terms of probabilities of exceedances vs earthquake intensity measures such as peak ground acceleration (PGA), for a given site or a grid of sites for up to eight different return periods. Currently 1975 version is used.
EQRSEI: Converts the output file from the EQRISK program "eqrisk.out", to individual contour files corresponding to each return period specified. These files can later be used directly as an input to EPIMAP to plot the PGA contour maps.
SPEC: Computes amplitude spectra for a given set of earthquake records and plots spectral ratios. It can be used to assess local site effects.

Probabilistic earthquake hazard computations can be done, using the two alternative programs CRISIS99 or EQRISK. In addition, the programs listed above and a number of other programs that manipulates earthquake data within the SEISAN package, are useful tools to assess the parameters that are needed to perform a seismic hazard analysis for an area of interest. The two main programs, CRIATT for computing the attenuation tables and CRISIS99 (modified version 1999) to compute seismic hazard are explained in more detail in the following. Both programs are written by **Mario Ordaz** of the Institute of Engineering,

UNAM [Ordaz, 1991, 1999]. The well-known hazard program EQRISK, on the other hand, is written by **Robin K. McGuire** and the original manual is distributed through United States Department of the Interior, Geological Survey [McGuire, 1976].

The two alternative hazard programs CRISIS99 and EQRISK have a number of features that are present in both. However, there are some advantages and disadvantages with both programs. In terms of the computing time and parameter input both programs require the same time. In the case of EQRISK, earthquake source zones are defined as arbitrary polygons (quadrilaterals). CRISIS99, on the other hand, operate with completely arbitrary polygons for the definition of the source zones and dipping planes may also be defined. In the MS-Windows 95 version, the source zones and the input parameters can be checked interactively through a user-friendly interface. In terms of the attenuation relations, CRISIS99 uses a table created by a separate program (CRIATT) and is therefore flexible (it also allows different attenuation relations for different source zones), whereas the attenuation relation, in the case of EQRISK, is given through a pre-determined mathematical formulation. Finally, CRISIS99 is superior to EQRISK, as it takes into account the uncertainties through the standard deviations introduced on several input parameters.

Step by step procedure for seismic hazard analysis

Following is a summary of the steps that have to be completed in order to produce a seismic hazard map.

1. Compile a catalogue for the area of interest from local, regional and global sources.

2. Evaluate the preliminary catalogue completeness by plotting histograms showing the distribution of events in time for different magnitude intervals. It may be necessary to divide your catalogue into two; (i) pre-instrumental and (ii) instrumental. Programs SELECT and CATSTAT can be used for this purpose.

3. Convert magnitudes into one uniform magnitude, preferably to moment magnitude MW. To do this, regression curves must be prepared for different magnitude scales. Program MAG can be used for this purpose.

4. Clean up the catalogue for dependant events (i.e. induced seismicity, non-earthquakes, foreshocks, aftershocks, earthquake swarms). Here a search has to be made for clusters of events both in time and space. Plots of histograms for specific sequences of time and space will reveal this. Program CLUSTER can be used for this purpose. The probable explosions may be removed by using the program EXFILTER.

5. The evaluation of the catalogue completeness is dependent upon the clean-up process and the magnitude unification. It is therefore necessary that steps 2-4 be repeated until a reliable catalogue is prepared.

6. Select the set of earthquakes from your catalogue from the part, which is complete for the chosen threshold magnitude and uniform in magnitude scale. Program SELECT can be used with different criteria for this purpose. Note the catalogue time span.

7. Prepare a seismicity map for the area of interest with the selected data, using EPIMAP. Delineate the earthquake source zones. Here, zooming and the area selection procedures of EPIMAP may be used.

8. Use additional information from geology, geophysics, seismotectonics, paleoseismology etc. to improve the source zonation.

9. For each earthquake source zone select the subset of events that fall in the chosen area. This can be done by using the EPIMAP program, which enables to draw polygons interactively on the screen and put the subset of events within this polygon into a file. Alternatively SELECT program can be used to extract the subsets of data corresponding to the defined source zones.

10. If the hazard is to be computed using CRISIS99 or by EQRISK, note the x, y (longitude, latitude), co-ordinates for each corner of the polygon.

11. The seismicity within each source zone is assumed to be uniform following a Poissonian occurrence. In order to define this, a set of critical parameters has to be assessed for each source. These are: Number of earthquakes above a threshold magnitude: This is the a-value for the lower bound magnitude. Catalogue time span: This is the time span of your catalogue where it is complete. Beta (bvalue * ln (10)) and its standard deviation: The b-value is the slope of the best-fitted line to the cumulative curve for the magnitude frequency of occurrence distribution (Gutenberg-Richter relation). Maximum expected magnitude with its standard deviation: This is usually inferred through other available information, such as geology, palaeoseismicity, or subjective judgement of the scientist. It is usually set to half a magnitude higher than the maximum observed when no information is available. Maximum observed magnitude: This is the largest magnitude observed within the catalogue time span. Threshold magnitude: The so-called lower bound magnitude, which is chosen, based on the engineering considerations. Usually magnitudes less than 4.0 are not considered engineering significant. In order to obtain each of the above critical parameters, a thorough evaluation of the earthquake catalogue is needed. BVALUE program can be used to obtain some of these parameters. However, while running the program, choosing the magnitude interval and the magnitude increment has to be done critically, taking into account the catalogue completeness and the detection threshold. These parameters will later be used in the input for the seismic hazard analysis program CRISIS99. Alternatively, the same input parameters are also needed for the EQRISK program. For each source zone, plot the magnitude- frequency of occurrence curves.

12. Try to assess whether there are characteristic earthquakes in your region. This can be done with a careful examination of your catalogue and the active faults in the area. Studying the magnitude-frequency of occurrence through the BVALUE program will help assessing this.

13. Try to establish an acceptable attenuation relation for your area. This can be done through empirical estimations or theoretically based on the random vibration theory (RVT). CRIATT program can be used to create the attenuation table. Alternatively, if you have an already established attenuation relation, this can be directly used in the EQRISK program. In this case, you can skip the steps 13-16, and continue from step 17 and onwards.

14. Establish a reliable Q factor by using the CODAQ program. This will be used in the attenuation program CRIATT to create the attenuation tables necessary for the hazard analysis.

15. Create the necessary input file for the CRIATT by modifying the sample-input file ‘criatt.inp’. or use program CRIPAR.

16. Run CRIATT to create the attenuation table necessary for the CRISIS99.

17. Create the input file for the CRISIS99 program by modifying the example-input file ‘crisis99.inp’. Make sure that the critical parameters are reliable and the geometry of the source zones are correct (see the program description).

18. Run the CRISIS99 program with the input file you have created and the output attenuation table from CRIATT. The program will generate the output files with the probability of exceedance rate

vs earthquake intensity (e.g. PGA), for the required return periods. Alternatively, if you have prepared the input for the EQRISK program, hazard can be computed by running the EQRISK program, for a given set of return periods (up to eight), for selected sites or for a grid of sites.

19. Repeat stages 6 to 17 to refine your model and the corresponding results.

20. Convert the output hazard "map" file from CRISIS99 for the computed return periods to individual contour files. Alternatively, if you have used EQRISK to compute hazard, the output file "eqrisk.out" can be converted using EQRSEI program, into individual contour files for previously defined return periods.

21. Plot the hazard maps for the desired return periods. Contouring option from EPIMAP can be used for this purpose (only for the EQRISK). Plot also the graphs for probability of exceedance rates vs PGA for selected critical sites.

22. Try to assess the local site effects for the critical sites. SPEC program can be used to obtain the amplification factors due to unconsolidated sediments. These factors can be used later to adjust the response spectra.

Many of the programs mentioned above are described individually throughout this manual at different sections. In the following the programs that are directly relevant to hazard computations and not described in other sections of the manual are explained in detail.

CRISIS99:

CRISIS99 is a computer program to compute seismic hazard in extended regions. It was developed at the Institute of Engineering, UNAM, Mexico, by **Mario Ordaz** (mors@pumas.iingen.unam.mx), **Armando Aguilar** and **Jorge Arboleda**.

Basic input data are: geometry of the sources, seismicity of the sources, and attenuation relations. Source geometry can be modeled as: 1) area sources, using a polygon with at least three vertex; longitude, latitude and depth must be given for each vertex, so this type of source can be used to model, for instance, dipping plates or vertical strike-slip faults; 2) fault sources, using polylines; and 3) point sources, included essentially for academic purposes.

Seismicity of the sources can be modeled either as Poisson or characteristic earthquake process. In the first, magnitude frequency relations are smoothly truncated Gutenberg-Richter curves, whereas for the second, the program assumes a Gaussian distribution of the magnitudes. Hazard computations can be performed simultaneously for several intensity measures, for instance, PGA, PGV, and several spectral ordinates. Required attenuation laws are given in the form of tables containing the median values of the intensity measures as a function of magnitude (the rows of the table) and focal distance (the columns of the table). Several attenuation models can be used in the same run, assigning an attenuation pattern to each source. Using a recursive triangularization algorithm, spatial integrations are performed optimizing the number of calculations, so CRISIS99 will integrate with more points for the nearest sources and less (or none) for distant sources.

CRISIS99 considers two different kinds of earthquake occurrence processes: Poisson process and characteristic earthquake process. CRISIS99 is oriented to computing hazard in extended regions. Hazard estimations are made for points in a grid that is not necessarily rectangular. The program can run under SunSOLARIS, SunOS and on PC (Windows95 or higher). Sun versions are to be used as a stand-alone program. The Windows version, on the other hand, also contains a windows interface for visual inspection of the input data as well as the results. Data validation options are available (only for the Windows version) and parameters can be given in a user-friendly graphic environment. CRISIS99 contains also a post-processing module that can be used to visualize the results, given in terms of maps of intensity

measures for an arbitrary return period or exceedance rate curves for a selected site, not necessarily a point in the original grid of sites. Also, if several intensity measures are included in the computations, uniform-hazard spectra can be produced. The main results of a run are also written to ASCII files, so the user can use his/her own post-processing techniques/software.

For the Windows version, a separate compressed file 'crisis99.zip' is included with sample-input data in SUP. Instructions on how to install the Windows version are included in the file 'crisis99.txt' in the INF directory. The Sun UNIX versions, are part of the standard SEISAN distribution and need not be installed specifically.

Detailed description of the input and output files is given in the pages below.

Input files for the CRISIS99

There are basically two input files that are required. First is an attenuation table (or several tables), and second is the major input parameter file where the file name for the attenuation table is also given. The input file can be prepared based on the format descriptions given below or modifying the example input file. An example-input file is included in the DAT directory with the file name "crisis99.inp".

There are some limitations in the input parameters. Following is a summary of the maximum values set in the program:

```
Attenuation Models :                    5
Intensity levels*:                      20
Structural periods:                     15
Number of regions:                      200
Magnitudes in attenuation model:        10
Distances in attenuation model:         21
Number of sub-sources per region:       4000
```

(* the term 'intensity' here should not be mixed with macroseismic intensity. In this context 'intensity' is meant as any chosen ground motion measure, such as PGA, PGV or any other spectral ordinate).

In the following the input file is described in more detail (by **Mario Ordaz**).

I. GENERAL DATA FILE

Format is free unless indicated otherwise.

1. General title of the run. 1 line
   TITGEN (A80)

2. Global parameters of the run. 1 line.
   NREG, NMOD, NT, NA
   NREG: Total number of regions (sources) in which the seismogenic area is divided.
   NMOD: Number of different attenuation models.
   NT: Number of spectral ordinates (or, in general, measures of intensity) for which seismic hazard is to be computed.
   NA: Number of levels of intensity for which seismic hazard will be computed.

3. Parameters for each spectral ordinate. NT lines. Free format
   T(I), AO(I), AU(I)
   T(I): Structural period of i-th spectral ordinate. It is used only for identification purposes, so in the cases in which structural period has no meaning, it can be just a sequential number.

AO(I): Lower limit of intensity level for i-th spectral ordinate.
AU(I): Upper limit of intensity level for i-th spectral ordinate. Exceedance rates for the i-th intensity will be computed at NA values, logarithmically spaced between AO(I) and AU(I)

4. More Global parameters
   RMAX, TR1, TR2, TR3, TR4, TR5
   RMAX: Parameter controlling the spatial integration process. Sources at distances greater than RMAX kilometers from a site will be ignored.
   TR1,...,TR5: CRISIS-99 will generate a file containing intensity levels for fixed return periods TR1,...,TR5. See below for the description of this output file. Five values must be always given.

5. Parameters defining the basic grid of points in which hazard is to be computed. 1 line
   LOI, LAI, DLO, DLA, NLO, NLA LOI,
   LAI: Longitude and latitude, respectively, of the origin of the grid.
   DLO, DLA: Longitude and latitude increments
   NLO, NLA: Number of lines of the grid in the longitude and latitude directions, respectively.

   Results will be given for points (LO(I),LA(I)), where

   LO(I) = LOI + (J-I)*DLO , J=1, NLO
   LA(I) = LAI + (I-1)*DLA, I=1, NLA

6. Number of polygons to be used to reduce the initial rectangular grid. 1 line.
   NPOLGRID

   Introducing one or more boundary polygons can reduce the initial rectangular grid of points. If polygons are given (NPOLGRID¿0) the computation of hazard will be performed only for those points of the grid, which are inside one of the polygons. If NPOLGRID=0 computations will be made for all points in the rectangular grid. NPOLGRID¡=10. If NPOLGRID¿0 then the following lines must be given for each polygon:

7. Definition of the k-th boundary polygon.
   NVERGRID (K)
   LONG (K, 1), LAT (K, 1)
   ⎫
   ⎬ NVERGRID(K) lines
   LONG (K, 1), LAT (K, 1)
   ⎭

   NVERGIRD(K): Number of vertex of polygon k. NVERGRID(K)<= 30.
   LONG (K, I), LAT (K,I), I=1,...,NVERGRID(K): Co-ordinates of the polygon's vertex. The polygon must be described counter clockwise.

8. Files of attenuation tables. NMOD lines MODELO (I) (A20)

   MODELO (I): Name of the file containing the i-th attenuation table (including path). The format of attenuation tables is explained below.

9. Data defining seismicity in each region. NREG blocks.
   TITULO (N) (A80)
   IC(N), IE(N), IMO(N)
   NV(N)
   LONG(1),LAT(1),PROF(I)
   ⎫
   ⎬ NV lines
   ...
   ⎭
   LONG(NV), LAT(NV), PROF(NV)
   Poisson model: (IC(N)=1)
   LAMBDA0(N), EB(N), CB(N), EMU(N), SMU(N), MMAX(N),M0(N)

Characteristic model: (IC(N)=2)
EMT(N), T00(N), D(N), F(N), SMT(N), M0(N), MU(N)

TITULO(N): Identification name for source N
IC(N): Flag defining the type of occurrence model assumed for N-th source. IC(N)=1 for Poisson model, IC(N)=2 for characteristic-earthquake model.
IE(N): Defines type of source. IE(N)=0 for area source, IE(N)=1 for line source and IE(N)=2 for point source.
IMO(N): Number of the attenuation model that will be used with this source. Must be between 1 and NMOD.
NV(N): Number of vertex defining source N.

LONG(I), LAT(I), PROF(I), I=1,...,NV(N): Co-ordinates of vertex I of source N. LONG(I) and LAT(I) are geographical coordinates of point i, whereas PROF(I) is the depth of the point, in km, which must be positive.Sources can be of three types: areas (polygons), polylines or points. Polylines and points can be given in any order. In general, in the case of an area source, CRISIS99 will divide the polygon into triangles. It first checks if triangulation can be made in the XY plane. Numbering of the vertex of the polygon must be done counter-clockwise in this plane when looked from above the surface of the Earth. If there are vertical planes, CRISIS99 will try to triangulate the area in the XZ plane, so numbering of vertex must be done counter-clockwise in this plane. Finally, CRISIS99 will try to triangulate in the YZ plane. There are some bizarre source geometries that cannot be well resolved by CRISIS-99, for instance, an L-shaped vertical plane. In these cases, an error will be reported.

Poisson model:
LAMBDA0(N): Exceedance rate of magnitude M0(N). The units are earthquakes/year.
EB(N), CB(N): Expectation and coefficient of variation, respectively, of the "b-value" for the source, given in terms of the natural logarithm.
EMU(N), SMU(N): Expected value and standard deviation, respectively, of the maximum magnitude for the source.
MMAX(N): Maximum observed magnitude in this source.
M0(N): Threshold magnitude for source N. The catalogue of earthquakes is assumed to be complete for M>M0. Earthquakes with M<M0 are absolutely ignored.

Characteristic model:
EMT(N): Median value of the times between characteristic earthquakes with M>M0. This is the inverse of the exceedance rate for M>M0.
T00(N): Time elapsed since the last occurrence of a characteristic earthquake.
D(N), F(N): Parameters defining the expected magnitude as a function of time, as in the slippredictable model. It is assumed that

E(M|t)=max(M0(N),D(N)+F(N)*LN(t))

Of course, if F(N) is set to zero, then D(N) becomes the expected time-independent magnitude of the characteristic earthquake.
SMT(N): Standard deviation of the magnitude of the characteristic earthquake. It is assumed independent of time.
M0(N): Minimum possible magnitude of a characteristic earthquake. Earthquakes with M¡M0 are absolutely ignored
MU(N): Maximum magnitude of the characteristic earthquake to be used in the integration process.

10. Name of the map file. 1 line
File name (including path) containing the base map to be used in post-processing with CRISIS99 for

windows. This name does not have any influence in the hazard computations. However, CRISIS99 expects a line here.

11. Name of the file of cities. 1 line
    File name (including path) containing the co-ordinates of cities, to be used in post-processing with CRISIS99 for windows. This name does not have any influence in the hazard computations. However, CRISIS99 expects a line here.

12. ATTENUATION TABLES
    NMOD attenuation tables must be given each one in a different file.

The tables give to CRISIS99 the relations between magnitude, focal distance and median intensities. CRISIS99 expects the following parameters in the i-th attenuation file, I=1,...,NMOD:

1. Parameters defining the magnitude limits. 1 line
   MINF(I), MSUP(I), NMAG(I)
   MINF(I): Lower limit of magnitude given in the table.
   MINF(I): Upper limit of magnitude given in the table.
   NMAG(I): Number of magnitudes for which intensity is given.

   CRISIS99 assumes than intensities are given for magnitudes M(K), where M(K)=MINF(I)+(K-1)*DMAG, where DMAG=(MSUP(I)-MINF(I))/(NMAG(I)-1).

2. Parameters defining the distance limits. 1 line
   RINF(I): Lower limit of distance given in the table.
   RINF(I): Upper limit of distance given in the table.
   NRAD(I): Number of distances for which intensity is given.

   CRISIS-99 assumes than intensities are given for distances R(K), where log(R(K))=log(RINF(I))+(K-1)*DLRAD, where DLRAD=(log(RSUP(I))-log(RINF(I)))/(NRAD(I)-1). That is, distances are supposed to be logarithmically spaced.

3. For each of the NT different intensity measures, the following block of lines:
   T(I,J), SLA(I,J), AMAX(I,J)
   SA(I,1,1,1), SA(I,1,1,2),..., SA(I,J,K,L),....,SA(I,NT,NMAG(I),NRAD(I)
   T(I,J): Structural period of j-th spectral ordinate. It is used only for identification purposes, so in the cases in which structural period has no meaning, it can be just a sequential number. SLA(I,J): Standard deviation of the natural logarithm of the j-th measure of intensity in the i-th model.
   AMAX(I,J): Maximum possible value of the j-th intensity in model I. The integration process will be truncated, regarding as impossible (zero probability) values larger than AMAX(I,J). If AMAX(I,J) is set to zero, then integration with respect to possible values of intensity will be performed from 0 to ∞.
   SA(I,J,K,M): Median value of the intensity in model I, for the J-th spectral ordinate, the K-th magnitude and the L-th distance.

   For each attenuation model, given in a separate file, CRISIS99 reads the above mentioned parameters in the following form:

```
DO J=1,NT
  READ(8,*) T(I,J),SLA(I,J),AMAX(I,J)
  DO K=1,NMAG(I)
    READ(8,*) (SA(I,J,K,L),L=1,NRAD(I))
  ENDDO
ENDDO
```

Output files from CRISIS99

CRISIS99 generates several output files, whose names begin with the base name requested at the beginning of the run. The output files are:

1. Main results file. This file with - .res -extension contains a printout of the name of the run, the values assigned to the variables, characteristics of the attenuation models, geometrical and seismicity description of the sources, the data defining the computation grid, etc. It also gives the final results, that is, exceedance rates for each site and type of intensity. It also gives a brief

   summary of the computations for each site, indicating which sources are of interest to the site and which sources were skipped.

2. Graphics file. The principal graphics file with - .gra - extension contains a brief identification header, and the exceedance rates for the type and levels of intensity requested. This file can be used as input file to plot intensity versus exceedance rate curves. CRISIS99 generates also a binary file with the exceedance rates for each structural period, so CRISIS99 will generate NT binary files. These binary files will be used only in the Windows System version of CRISIS99 to make hazard maps. The names of these files are base_name.b1, base_name.b2,..., base_name.bNT.

3. Map file. This file with - .map - extension contains intensity levels for fixed return periods (TR1,...,TR5) for each type of intensity and site. It also gives the co-ordinates of each site. This file can be used to generate contour or 3d maps of intensity levels associated to constant exceedance rates.

Example output files are included in the DAT directory (crisis99.res, crisis99.gra, crisis99.map).

CRIATT:

In this program, an earthquake source model and results from Random Vibration Theory (RVT) [e.g., Boore, 1983, 1989], are used to estimate attenuation of ground motion parameters as a function of moment magnitude, MW, and hypocentral distance, R. Ground motion is assumed to be band-limited, stationary and of finite duration.

For estimating the Fourier acceleration spectra, a(f), it is assumed an omega square constant stress drop source model given by Brune [1970]. The expression for a(f) is:

$$a(f) = CG(R)S(f)D(f) \tag{6.1}$$

where

$$C = (4\pi 2R_{V\phi}FV)/(4\pi\rho\beta^3) \tag{6.2}$$

$$S(f) = M_0 f^2/(1 + f^2/f_0^2) \tag{6.3}$$

and

$$D(f) = P(f)e^{-\pi Rf/\beta Q(f)} \tag{6.4}$$

Thus the spectrum a(f) is the multiplication of a constant C (independent of frequency), geometrical spreading term G(R), source function S(f), and diminution function D(f). In C, Rv is equal to average

radiation pattern (0.55), F is free surface effect (2.0), V is partition of a vector into two horizontal components (0.707), . is density in gm/cm3, and is shear wave velocity in km/sec.

In S(f), M0 is the seismic moment and f0 is the corner frequency, given by Brune [1970]

$$f_0 = 4.9x10^6\beta(\Delta\sigma/M_0)^{1/3} \tag{6.5}$$

where β is in km/sec, $\Delta\sigma$ is the stress drop in bars, and M_0 is in dyne-cm. The diminution factor $D(f)$, accounts for loss of energy due to internal friction and scattering.

At distances less than a certain critical value of R_c, the strong motion records are dominated by S-waves. Thus for $R < R_c$, $G(R) = 1/R$ is the geometrical spreading. For $R > Rc$, $G(R) = 1/(RR_c)^{1/2}$.

The diminution function $D(f)$ in equation 6.4 requires $Q(f)$ and $P(f)$, where the quality factor defined by the regional attenuation is expressed by $Q(f) = Q_0 f^{\epsilon}$ (f is frequency and $\epsilon \leq 1.0$), and whereas

$$P(f) = e^{-\pi\kappa f}$$

(23)

$P(f)$ reconcile an additional attenuation term which may be related to near-surface loss of energy where *kappa* is a high frequency decay factor [Singh et al., 1982].

Input file for CRIATT

The standard input file for the CRIATT program can be created by modifying the example input file. A total of 23 parameters provide the necessary input for calculating the attenuation tables, which is based on equation 6.1, described earlier. The user should define the magnitude and the distance limits. It is important to note here that some combinations of parameters may result in 0 values for large distances in the table, which creates problems for the CRISIS99 program. In order to avoid this, the distance ranges are set to $R_{min} < 10km$ (CRISIS99 requires one-digit only) and $R_{max} < 500km$. Usually the regional attenuation term and the site factor are the most critical factors in the definition of a(f). The effect of the high-frequency decay factor can only be seen when the combination of the kappa parameters (*kappa*0 and *kappa*1) are chosen correctly (e.g. increasing kappa1 with *kappa*0 kept constant, would result in low ground motion values). An example input file is included in the DAT directory with the file name `criatt.inp`.

Output file from CRIATT

The output of the CRIATT program is a file containing the attenuation tables for the selected spectral ordinates (i.e. as a default only PGA corresponding to a period of $0.005sec$ is computed). For each spectral ordinate, the file will contain a set of values (e.g. PGA) for different distances. This file is then used as one of the inputs to the CRISIS99 program. The file name is user defined. An example output file is included in the DAT directory with the file name `criatt.tab`.

CRIPAR

The program was used earlier to generate input for both crisis and criatt but now it is only used with criatt due a format change for crisis99.

EQRISK:

This popular program for computing seismic hazard is written by McGuire [1976], and the complete manual is published as an open file report. The following is a short summary of the program operation and a full description of the input parameters as well as format of the input file. These descriptions are as they are given in the original manual [McGuire, 1976]. The program EQRISK evaluates risk (hazard)

for each site-source combination and intensity level and calculates the total annual expected number of occurrences of intensity greater than those levels of interest at a site by summing the expected numbers from all sources. Seismic source areas are specified as a set of arbitrarily shaped quadrilaterals. For ease of use, gross sources may be divided into sub-sources, which are a string of quadrilaterals, each two adjacent subsources having two common corners. A Cartesian co-ordinate system is used and the location of the origin is arbitrary.

Input file for EQRISK

The default input file is named "DATA" and is hardwired into the program (for the SUN version file name should be uppercase). An example input file is included in the DAT directory with the name "eqrisk.inp", which should be renamed to "DATA" before running. Following is the description of the individual parameters and their format as described in the original manual [McGuire, 1976].

Card 1 (Format 20A4): Title. Any 80 characters can be used to describe the problem.

Card 2 (Format 3I10): NSTEP, JCALC, JPRINT.
NSTEP is the number of integration steps used in integrating over distance for each site-source combination.
JCALC is the flag indicating how integration on magnitude is to be performed (JCALC=0 is used for analytical integration, and the form of the attenuation function is described in the original manual. JCALC=1 is used for numerical integration on magnitude. The user must supply own attenuation function in subroutine RISK2.)
JPRNT is the flag indicating the desired output (JPRNT=0 is used to print only total expected numbers and risks at a site which is normally used when a grid of sites being examined. JPRNT=1 is used to print expected numbers from each site-source combination, normally used when examining a single site).

Card 3 (Format I5, 12F5.3): NLEI, TI(1), TI(2), ... TI(NLEI).
NLEI is the number of intensities to be examined. TI(1), TI(2) and so on, are intensities for which expected numbers and risks are calculated at each site. Note, that the values for TI(i) may be Modified Mercalli Intensity or the natural logarithm of ground acceleration, velocity, displacement or spectral velocity. In printing results, the program prints both TI(i) and its antilogarithm. Values for array TI must be specified in increasing order.

Card 4 (Format 8F10.2): RISKS(1), RISKS(2), ... RISKS(8).
RISKS(1), RISKS(2), and so on are risks (probabilities of exceedance) for which the corresponding intensities are desired. These intensities are calculated by interpolation on a logarithmic scale, between intensities (in the list of examined intensities, TI) having larger and smaller risks. Both the corresponding intensity and its antilogarithm are printed. Values for array RISKS must be specified in order of decreasing risk. If fewer than eight values are desired, leave succeeding spaces on the card blank. To avoid large errors and subsequent misinterpretation, the program will not extrapolate to calculate intensity values corresponding to risk levels specified; it is the user's obligation to choose values for array TI which will result in risks which bound those specified in array RISKS. This is of course, a matter of judgement and experience. The user must be cautioned that in a grid site system appropriate values for array TI may vary considerably for the different sites examined. The intensities interpolated for levels specified in RISKS will be most accurate for closely spaced values of TI.

Card 5 (Format 8F10.2): C1, C2, C3, SIG, RZERO, RONE,
AAA, BBB. C1, C2, C3 and RZERO are parameters in the attenuation equation for mean intensity

discussed in the original manual [McGuire, 1976]:

$$m_I(S,R) = C1 + C2 * S + C3 * ALOG(R + RZERO)$$

SIG is the standard deviation of residuals about the mean. If no dispersion of residuals are desired, insert a very small value for SIG (rather than exactly 0.0). RONE is the limiting radius inside of which no attenuation of motion is desired, for values of focal distance closer than RONE, the mean intensity is calculated using RONE in place of R in the attenuation equation above. If this feature is not desired, insert zero for RONE. AAA and BBB are parameters in the equation limiting the mean intensity:

$$max\, m_I(s) = AAA + BBB * S$$

The value specified for BBB must be between zero and C2 for this limiting equation to make sense. If it is not, an error message will result and program operation will terminate.

Card 6 (Format I10, 6F10.2): NGS, NRS(1), NRS(2), ... NRS(NGS).
NGS is the number of gross sources to be specified.
NRS(1), NRS(2), and so on are the number of subsources in gross source 1,2, etc. See the original manual [McGuire, 1976], for a general description of the source specification.

Card (set) 7 (Format I10, 6F10.2): LORS(I), COEF(I), AM0(I), AM1(I), BETA(I), RATE(I), FDEPTH(I).
There must be NGS+1 of these cards, one for each gross source and one for background seismicity.
LORS(I) is a flag indicating whether the source area has a loose or strict lower bound (LORS=0 implies a loose lower bound and LORS=1 implies a strict lower bound).
COEF(I) is a coefficient modifying the expected number of exceedances from gross source I. Its most common value is +1.0. AM0(I) is the loose or strict bound lower magnitude or intensity for gross source I.
AM1(I) is the upper bound magnitude or intensity for gross source I.
BETA(I) is the value of β for gross source I. It is equal to the natural logarithm of 10, times the Richter b-value for the source.
RATE(I) is the rate of occurrence of events having magnitudes of intensities greater than AM0(I). If a discrete distribution on intensities has been used to calculate the rate, the user may wish to specify AM0(I) as one-half intensity unit lower than the lowest intensity used to establish the rate. Note that for gross sources RATE(I) is in units of number per year; for background seismicity it is in units of number per year per 10,000 km.
FDEPTH(I) is the focal depth of events in gross source I, in km. If epicentral distances are required for all sources and for background seismicity for the attenuation function, insert zero for FDEPTH(I).
If no background seismicity is desired, leave the last card in this set completely blank.

Card 8 (Format 4F10.2): X1, Y1, X2, Y2.
There must be NRS(1)+NRS(2)+ .... +NRS(NGS)+NGS of these cards. The first NRS(1)+1 cards specify co-ordinates of subsources in gross source 1, the next NRS(2)+1 cards specify coordinates of subsources for gross source 2, and so on. Internally, the point X1, Y1 is connected to X2, Y2, as well as both to the previous and the subsequent points designated as X1, Y1, as long as these are both in the same gross source. Point X2, Y2 is connected similarly. An example is elucidating. The following points define two gross sources having two subsources.

```
 0.0  0.0 10.0  0.0
 0.0  5.0  8.0  8.0
-5.0 10.0  6.0 15.0
10.0 20.0 11.0 20.0
```

```
15.0 15.0 16.0 15.0
15.0  0.0 16.0  0.0
```

Card 9 (Format 2I5, 4F10.2): NX, NY, XZERO, YZERO, XDELTA, YDELTA.
There can be any number of these cards, one for each site or grid of sites to be examined. NX and NY are the number of grid points in the X (East-West) and Y (North-South) directions; that is, they are the number of columns and rows in a grid of sites to be examined. For specification of a single site, NX and NY must have values of unit. Zero or negative values for NX and NY are meaningless and will cause program to terminate.
XZERO and YZERO are the co-ordinates of the site to be examined, or are the lower left corner of the grid if NX and/or NY are greater than one.
XDELTA and YDELTA are the grid spacing in the X and Y directions. When the grid option is not used, these variables may be left blank or set equal to zero.

Final card: Insert one blank card at the end of the input deck.

Output file from EQRISK

There is only one standard output file generated by EQRISK which has a default file name of "eqrisk.out". This file contains the results of the hazard computations for each site for the specified exceedance probabilities. This output file can easily be converted to individual intensity (e.g. PGA) contour files (one for each level of exceedance probability), using the program EQRSEI. The resulting contour maps from these output files may then be plotted by EPIMAP.

The detailed format of this output file is described in the original manual [McGuire, 1976], and is not repeated here. A test set of input and output files are given in DAT.

EQRSEI:

The program EQRSEI converts the output file `eqrisk.out` from the EQRISK program into individual intensity contour files for the previously defined return periods. There may be up to eight such files (eqrsei1.out, eqrsei2.out ...., eqrsei8.out). These files can then be used as input to the EPIMAP program to plot the contours of PGA values on the epicenter maps. Each file contains also some header information, where the individual contours and the contour intervals are given. In addition, the color codes are also given. The individual contours and the contour intervals can be modified by editing the header lines of these files.

CLUSTER:

This is a program that searches for the dependant events in a given catalogue (compact file) with respect to time and distance. It is written by Juan Pablo Ligorria and Conrad Lindholm. The input is a standard Nordic file with header lines only (compact file). The user has to give the number of days to be searched before and after the main event, and the distance limits in km. The magnitude of the main event over which the search will be performed is also user defined. The output is a repetition of the input catalogue with "?" placed at the end of each dependant event which falls within the limits defined in the interactive input in time and distance. The default file name is `cluster.out`. The user should then work systematically through these events and decide whether they should be cleaned or not. This process, we feel, should be done manually, because deleting events from the catalogue (especially the historical part), may have serious implications later in the hazard computations. Clusters of foreshocks, aftershocks or other dependant events such as earthquake swarms can be delineated by this program.

6.25 Magnitude relations, MAG

The MAG program calculates simple magnitude relations. The program has three functions: (1) Calculate parameters for a magnitude scale (Ml or Mc), (2) Calculate relation between two different magnitudes and/or spectral parameters and (3) Calculate a new magnitude as a function of an existing magnitude, a natural step following function (2). All three functions can be done at the same time. Function (3) can also be used for moving a particular magnitude type and/or agency to the first magnitude position in line 1 to be plotted with EPIMAP.

ALL HEADER LINES ARE SEARCHED FOR MAGNITUDE INFORMATION

Input: The data input is a CAT-file like one made with SELECT or COLLECT or it can be a compact file if only magnitude comparison is made. Optionally there can be a parameter file, which MUST be, called mag.par and MUST reside in the working directory. An example of the parameter file is found in DAT and also shown below. The parameter file is not needed for all operations, see details below.

1: Magnitude scales

Coda magnitude Mc: The coda magnitude scale used is

$$Mc = A * log(coda) + B * dist + C$$

where Mc is the coda magnitude, coda is the coda length in secs, dist is the hypocentral distance in km (calculated from epicentral distance and depth in CAT file) and A, B and C are constants to be determined. This is done in two ways

3d regression

$$m = A * log(coda) + B * dist + C$$

2d regression

$$m = A * (log(coda) + dist_coff * dist) + C$$

with B = A * dist_coff where dist_coff is given in the parameter file and m is the reference magnitude. SO B AND dist_coff ARE DIFFERENT. The CAT-file must contain coda readings, epicentral distances and a magnitude in the header line. A linear regression is then made between the known magnitude from a given agency and the observed coda lengths following the relations above. The user has the option to choose the type of magnitude to use in the regression. Usually Ml or Mb are used. All station-event combinations are used to determine simultaneously the 3 constants A, B and C. Since the data often is too bad to determine all 3 parameters at the same time, the program will also calculate just A and C using a fixed user supplied value for the distance correction to the coda. The constant dist_coff is given in the mag.par file as the second parameter under MAG_TYP_COF (see below). IN ORDER FOR THE CODA SCALE OPTION TO WORK, THE DISTANCE COEFFICIENT MUST BE DIFFERENT FROM ZERO.

Output: On the screen the constants will be printed out and a file mag_coda.out will contain pairs of values m and (log(coda) + dist_coff*dist), which can be used to plot the distance corrected coda relation. If results from the 3D is to be plotted, dist_coff must be calculated as dist_coff=B/A, put into mag.par and mag run again. On the other hand, if a best dist_cof has been found, B is calculated as B=A*dist_cod

A typical coda magnitude relation is :

$$Mc = 2.0 * log(coda) + 0.0035 * dist - 0.87$$

[Lee et al., 1972]

Local magnitude Ml:
The local magnitude scale is calculated by determining an amplitude attenuation scale using amplitudes and distances in CAT file. The parameters in the Ml magnitude scale are computed for every event individually, parameters are determined as averages of all events.

For each event (only type L and R are used) a,b,c are calculated if at least 3 stations are available using least squares regression as follows:

$$log(amp) = a * log(dist) + b * dist + c$$

The relation above can be derived from the standard geometrical spreading and attenuation relations:

$$amp = (dist * *a) * exp(pi * f * dist/(v * q))$$

where f is the frequency, v is the velocity and $q = q0 * f * *qalpha$. The relation can be rewritten

$$log(amp) = a * log(dist) + (pi * f * dist)/(v * q0 * f * *qalpha * 2.3)$$

Since *qalpha* often is close to 1.0, the relation can be simplified to the frequency independent relation:

$$log(amp) = a * log(dist) + (pi * dist)/(v * q0 * 2.3)$$

If body wave spreading is assumed ($a = 1$), $q0 = 100$ and $v = 3.5km/sec$, the relation is

$$log(amp) = 1.0 * log(dist) + 0.004 * dist$$

which is comparable to the relation shown below for California.

Similarly to the coda relation, a 2D relation is also calculated

$$log(amp) - b * dist = a * log(dist)$$

where $b = dist_coff$ is fixed to the value given in the mag.par file (same parameter as used for coda). This gives a more stable solution, however b = dist_coff must be determined by trial and error or fixed using known values from e.g. q-studies.

The amplitudes are assumed to be ground displacements (in SEISAN they are ground displacements highpass filtered at 1.25 Hz to resemble Wood Anderson seismograms, see MULPLT). The distance ratio between stations with the maximum distance and minimum distance must be more than 3 for the event to be selected for analysis. It is assumed that a and b will be the same for all events, while c is different (magnitude dependent). At the end, the average constants a and b are calculated of all values a and b which are not deviating too much (a must be in the range 0 to -5, hardwired). Distance attenuation coefficients a and b are supposed to be negative since amplitude decrease with distance. To get the local magnitude scale

$$Ml = log(amp) - a * log(dist) - b * dist - C$$

the constant C must be determined by fixing the magnitude at some reference distance like the original Wood Anderson definition with $Ml = 3$ at $dist = 100km$ and $amp = 1/2200mm = 454nm$ (assuming gain of the Wood Anderson seismograph to be 2080, [Hutton and Boore, 1987]. The determination of a and b does not work well unless the observations are very good. The relation for California is [Hutton and Boore, 1987]

$$Ml = log(amp) + 1.1 * log(dist) + 0.00189 * dist - 2.09$$

Output: On the screen the constants will be printed out and a file mag_amp.out will contain the values of a, b and c.

2: Magnitude relations and/or spectral parameter relations

Linear regression (maximum likelihood) can be made between any two magnitudes and/or spectral parameters on any of the header lines of an event in a CAT-file or a compact file. The user is interactively prompted for the magnitude type and/or spectral parameters and agencies to compare. If none is given, no magnitude comparison will be made. If several magnitudes/spectral parameters fit the requirement, the last one is used. If e.g. the first header line has a BER Ml and the last header line also has a BER Ml, the last one will be used. Maximum likelihood linear fitting is used. It is assumed that both variables have normal and correlated errors. See subroutine maxlik.for in LIB for more info.

The following parameter can be selected:

Any magnitude and agency
Seismic moment(log)
Stress drop (log)
Corner frequency (log)
Source radius(log)
Spectral decay
Omega zero level (log)

If any of the spectral parameters are selected, or moment magnitude is without agency, there will be an additional question about which station and component. A blank return means the average will be used. With these parameter selections, it is possible to compare spectral parameters from any two channels, compare the average spectral parameter with the parameter from one channel etc.

Output: A plot will be shown on the screen with the observations and the least squares fit and the values are also printed out on the screen. A file mag_mag.out contains the pairs of magnitudes used.

3: Magnitude conversions

If a relation between two magnitude scales is known, e.g. by using option 2 above, an output file can be made with the converted magnitudes. The relation to use is specified in the mag.par file. Several different input magnitude types and agencies can be used and the relation-agency used is given in a priority list in the mag.par file, see example below. It is here shown that if a BER Mc is available, this will be the first choice. If no BER Mc then BER Mb will be the next choice etc. The new magnitude will have type X and agency NEW. Output: The output file is mag_new.out and has the same format as the input file. On the header line, the old magnitudes are removed and in the first magnitude position will be the converted magnitude (NEW) while in the second magnitude position, the magnitude selected for conversion will be given. The third magnitude position is blanked out. The conversion option can also be used to move magnitudes around by using a 1 to 1 relation as shown in mag.par example.

Summary of output files:

`mag_amp.out` :	Details each event for amplitude regression.
`mag_coda.out` :	Magnitude vs coda, see text.
`mag_mag.out` :	Pairs of magnitudes used for regression.
`mag_new.out` :	Events with converted magnitudes only.
`mag_newa.out` :	All events, both converted and non converted (due to no correct input magnitude available).
`mag_spec.out` :	Summary of normal header line, all associated magnitudes and spectral parameters.
`mag_ml_inv.out` :	From Ml inversion

In DAT there is an example `mag.par` file.

An example of the `mag.par` parameter file:

```
This file is for parameters for MAG and called MAG.PAR. The name must
be in lower case on Sun. The following shows the parameters which can be set.
The file can contain any lines in any order, only the lines with
recognized keywords and a non blank field under Par 1 will be read. The
comments have no importance. The text fields are left justified, the
real numbers can be anywhere within the 10 columns of the parameter.
 MAGAGA is the magnitude type and agency to use for the converted magnitude.
 MAGREL gives the magnitude and agency to use for conversion e.g. LBER and
the parameters 2 and 3 gives the relationship:

magnitude out = magnitude in * par2 + par3

The magnitude conversion uses one of the MAGREL relationships, where the
priority is in the same order as found in the list.

SCREENOUT can be (Y)ES or (N)O, indicates if a line is printed on the screen
for each event.

When doing a magnitude regression on coda or amplitude, BAD STATION indicates
stations not to be used.  MAG_TYP_COF is the magnitude type and agency to correlate
coda readings with and the second parameter is the distance correction
term used when calculating a coda magnitude relation with a fixed distance
term. The same parameter is also used for amplitude regression.

The input file can be either Nordic or compact Nordic, however if coda or
amplitudes are to be used, it must be NORDIC.


KEYWORD........Comments.........................Par.1.....Par.2.....Par.3.....
BAD STATION    Station not used for mag relation  BER
BAD STATION    Station not used for mag relation  XXX
BAD STATION    Station not used for mag relation  BER
MAGAGA         Magnitude type and agency, new mag XNEW
MAGREL         Mag type, agency, a,b for new mag  CBER        1.0       2.0
```

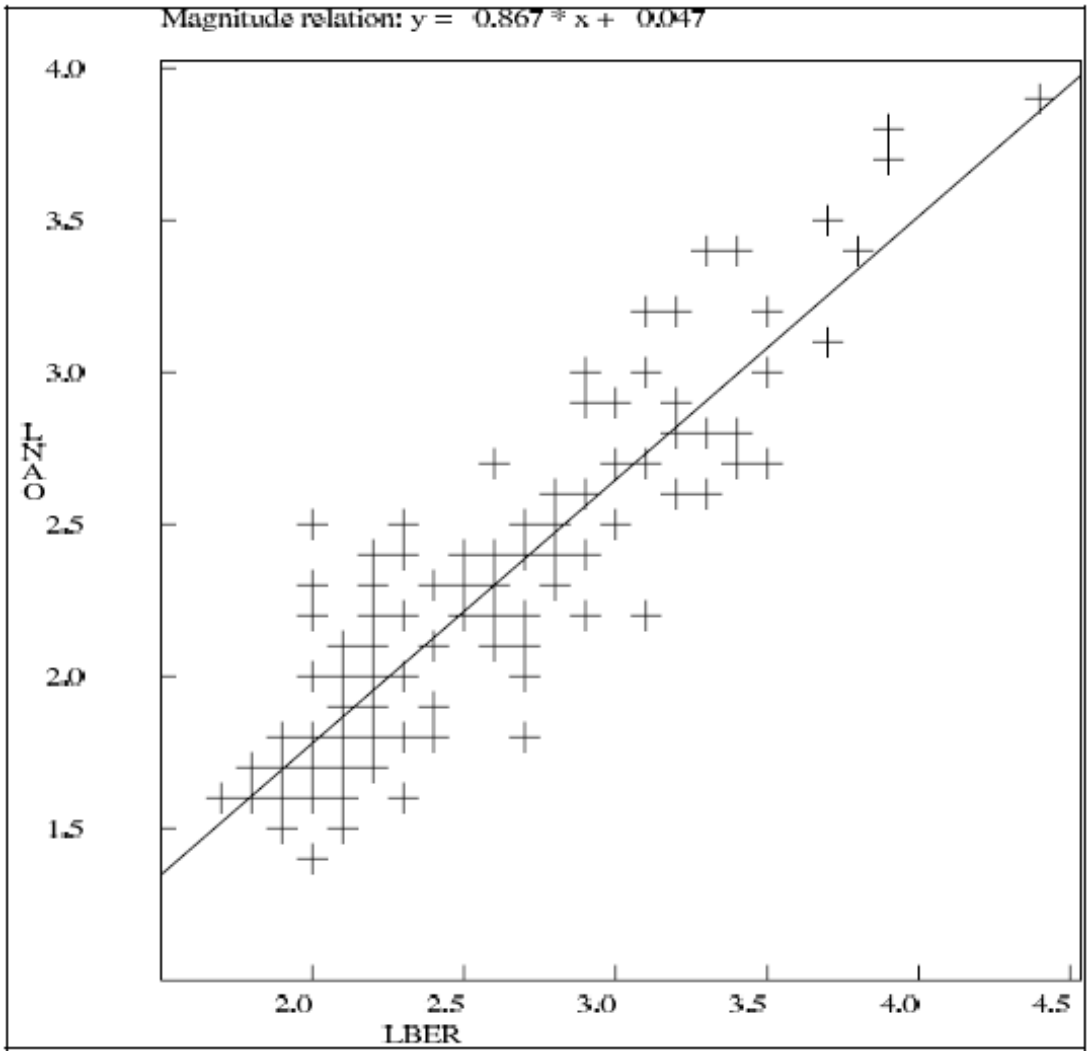


Figure 6.32: Example of using the MAG program. Relation between NORSAR and Bergen local magnitudes.

```
MAGREL          Mag type, agency, a,b for new mag  BBER        1.0       2.0
MAGREL          Mag type, agency, a,b for new mag  SBER        1.0       2.0
MAGREL          Mag type, agency, a,b for new mag  WBER        1.0       2.0
SCREENOUT       Header line printed each event     N
MAG_TYP_COF     Mag. type for corr., dist coff.    LNAO        0.002
```

6.26 ML inversion, MAG2

MAG2 is a program to invert for the local magnitude scale ML. The difference to the inversion done in MAG is that MAG2 inverts amplitudes from all events simultaneously for the scale and station corrections. The program can invert for different scale parameters depending on selected distance ranges. The reason for this is that it is known that the geometrical spreading is not the same for example between Pg and Pn. Some authors have suggested distance dependent scales, but most commonly a single scale is used for all distances for simplicity.

The general ML scale is given by

$$ML = log_{10}A + alog_{10}(R) + bR + S + c$$

where we measure the displacement amplitude A in nm, R is the hypocentral distance in km, S is the station correction of the individual stations, and c is a constant added to make the scale comparable to other places at a reference distance. The station corrections add up to 0. The region dependant parameters in the scale are a, accounting for geometrical spreading, and b, accounting for attenuation. The part $(alog_{10}(R) + bR + c)$ is commonly written as $(-log_{10}A0)$.

The program applies singular value decomposition using the Numerical Recipe [Press et al., 2003] routines to invert the observations for a, b and S. It then computes the parameter c based on the reference given through distance, amplitude and magnitude. This allows to calibrate scales between different regions so that they are the same at the reference distance. Commonly c is set such that 480 nm amplitude at 17 km gives ML=2 (this is equivalent to 10 mm on a Wood-Anderson seismograph giving ML=3 at 17 km [Hutton and Boore, 1987]. The inversion can be setup to invert for the geometrical spreading term a in the scale for up to three distance ranges. However, a single attenuation term b is used.

As input the program requires a parameter file mag2.par in the working directory,and a standard station file e.g. STATION0.HYP. Then the user only has to enter the input file of events in Nordic format. An example file, mag2.par, and an input file, mag2nor.cat, with events from Norway are in given in DAT.,

The parameter file has the following settings given by keywords (any order):

INVERSION TYPE (f10.1) - 1. = singular value decomposition (no other choice yet)
DISTANCES (2f10.1) - distance range in km for observations to use
MINIMUM NUMBER OF OBS/EVEN (f10.1) - only events with more or equal number of observations are used
MIN DISTANCERANGE RATIO (f10.1) - minimum range required computed as ratio of distances defined by DISTANCES
ORIENTATION (f10.1) - use of components: 0. = horizontal and vertical, 1. = horizontal only, 2. = vertical only
SYNTHETIC (f10.1) - set to 1. for synthetic test, scale defined by FIX SCALE A and FIX SCALE B; 0. for inversion of data

NOISE (f10.1) - ratio of amplitude to be added as noise to synthetic test
FIX SCALE A (3f10.1) - set the fixed parameter a in scale, possible for the three distance ranges given by SCALE DISTANCE
FIX SCALE B (3f10.1) - set the fixed b parameter in scale, possible for the three distance ranges given by SCALE DISTANCE
FIX SITE (f10.1) - set to 1. to not invert for station corrections; 0. for default inversion for station corrections
REFERENCE DISTANCE, REFERENCE AMPLITUDE, REFERENCE MAGNITUDE (all f10.1) - setup of the reference, used to calculate parameter c, give amplitude as Wood Anderson amplitude in mm
SCALE DISTANCE (2f10.1) - give up to two distances which give the transition between the possibly three distance dependent scales; blank or numbers larger than maximum distance will give only one scale
IGNORE COMP (a4) - give component not to be used
IGNORE STAT (a5) - give station not to be used

The program produces a number of output files:

mag2_amp_dis.out - amplitude versus distance for each event
mag2_amp_obs.out - list of observed amplitudes
mag2_events_read.out - listing of events that were read in
mag2_events_used.out - events used in Nordic format
mag2_evxy.out - coordinates of events used
mag2.out - general output file, lists data used and computed scale
mag2_paths.out - event-station path coordinates
mag2_station_hyp.out - hyp station file with scale and station corrections
mag2_stat_list.out - simple output of stations used
mag2_statxy.out - coordinates of stations used

The output file mag2.out will give some details on the input data, as number of stations, events and observations. It reports the reference used to fix the scale at the reference distance. Next it gives the scale, consisting of three parts if inversion is done for all possible segments. This will be given by a1, a2, a3, while b will be assumed to be the same. First the scale is presented to include the reference distance, second it is shown without the reference distance included with the scale. Then comes a section with the stations and the respective site terms, and finally the list of events with the source term inverted for.

The output file mag2.out for the example in the DAT directory should look like this:

```
ML inversion output

 SVD inversion

 Total number of events:           69
 Total number of stations:           23
 Total number of observations:          600

 Reference distance  =    100.0000
 Reference amplitude =    1.000000
 Reference magnitude =    3.000000

 Ml = log A + a log(dist/refdist) + b (dist-refdist) + c + S
```

```
     a1=  0.84717 +/-  0.39844
     a2=  0.00000 +/-  0.00100
     a3=  0.00000 +/-  0.00100
     b =  0.00061 +/-  0.00136
     c =  0.31807 +/-  0.00000

 Ml = log A + a log(dist) + b (dist) + c + S
    a1 =  0.84717 +/-  0.39844
    a2 =  0.00000 +/-  0.00100
    a3 =  0.00000 +/-  0.00100
     b =  0.00061 +/-  0.00136
    c1 = -1.43679
    c2 =  0.25756
    c3 =  0.25756

Station #    1 STAV  -0.120  +/- 0.2612   58.935    5.702
Station #    2 BLS5  -0.044  +/- 0.2204   59.423    6.456
Station #    3 ODD1  -0.085  +/- 0.2125   59.911    6.627
Station #    4 EGD    0.043  +/- 0.2221   60.270    5.223
...
Station #   21 MOR8  -0.018  +/- 1.3061   66.285   14.732
Station #   22 ESK    0.851  +/- 1.1257   55.317   -3.205
Station #   23 AKN    0.171  +/- 0.3797   62.178    6.997
 Average site term:   0.00
  Event #     1 2002051922484590  ML =  1.95 +/- 0.438
  Event #     2 2002052614481700  ML =  1.68 +/- 0.577
...
```

6.26.1 MAGSTAT

The program MAGSTAT can be used with MAG2 to produce statistics that allow to evaluate the magnitude scale. As input it takes a STATION file, as produced by MAG2, and a Nordic input file. Here, one can use any data set, or the events that were used in the MAG2 inversion.

This program is still under construction.

6.27 Explosion filtering, EXFILTER

The program EXFILTER is used to identify probable explosions in a catalog of seismic events. Man-made seismic events like quarry blasts, mining explosions and other explosions show a certain distribution in time and space. Therefore the method of explosion identification here is based on normalizing the time of day distribution of seismic event occurrence as a function of area. The program works on the following principle: Areas where explosions occur are defined. If an event is located in one of these areas, with a magnitude below a given maximum magnitude, with a depth less than a given maximum depth, within a given time of day interval and within a given year interval, it is identified and marked as probable explosion. The areas are defined by polygons of any shape. For definition of the filter areas, a list of mine locations (with consideration of location accuracy), locations of explosions and locations of event clusters (they might be clearly related to mine locations, but others might indicate unknown explosion

sites) can be used. The next step is to define the parameters for each area to get a normal time of day distribution. They can be determined following the steps:

1) - get the time of day distribution of events (program CATSTAT)
   - select a time window of probable explosions
   - select events within time window of probable explosions

2) - get the distribution of magnitudes of events within time window of probable explosions (program BVALUE)
   - select the maximum magnitude

3) - test parameters defined with program EXFILTER for the defined area and adjust the parameters if the time of day distribution is not normal.

For more details, see Ottemöller [1995].

The program uses a parameter file, `EXFILTER.PAR` which MUST be located in the DAT directory.

An example of the parameter file `EXFILTER.PAR`

```
--------------Parameter file for program EXFILTER-----------------
-------------------------------------------------------------
        This file must follow the following format rules: __________
1. Any number of comment lines
2. Any line with first character # defines the parameters of that area
3. Any line with first character * defines the parameters of exceptions within
   an already defined area. I.e. you can define an area around a volcano,
   and make an exception for that very small  area.
4. In the first line of each parameter set the parameters are:
    Maxdepth: Events with depth above this value are not explosions
    Maxmag  : Events with magnitude above this value are not explosions
    Lhour   : Lower limit of time interval for explosion time
    Hhour   : Upper limit of time interval for explosion time
    Btime   : Lower limit of yearly interval
    Etime   : Upper limit of yearly interval
    N       : Number of latitude longitude pairs in polygon
5. From the second line the lat long pairs are given
-------------------------------------------------------------
 Area------------Maxdepth Maxmag   Lhour   Hhour   Btime   Etime       N

# area 1           100.0     3.6      11      15  197001  200012      12
                   64.0    23.8    66.70   23.80   68.80   20.20   69.80    18.30
                   70.50   20.0    71.20   26.00   70.80   29.50   70.00    27.50
                   68.50   28.50   67.00   29.50   64.30   29.50   64.00    34.00

# area 2           100.0     3.8      15      20  197001  200012       4
                   67.90   17.80   68.80   20.20   66.70   23.80   66.50    21.00

# area 3           100.0     3.5       0       1  197001  200012       4
```

```
                  67.90   17.80   68.80   20.20   66.70   23.80   66.50    21.00

# area 4          100.0     4.0       2      17  197001  200012       4
                  68.50   28.50   68.50   36.00   67.00   36.00   67.00    29.50
```

The EXFILTER program searches for probable explosions using a catalog-file as input and marks events that might be explosions with 'P' as Event ID in the output file `exfilter.out`. Example of program run

```
<exfilter>

NUMBER OF AREAS: 55

 FILENAME... ?
june.cat
 ***********************************
 Number of probable explosions found: 90
 Output written in file: exfilter.out
 ***********************************
```

The top figure shows the filter areas used for Scandinavia. The bottom right figure shows the time of day distribution for a 10 year Scandinavian catalog before filtering (made with CATSTAT) and the figure bottom left shows the distribution after filtering.

6.28 Inversion of travel time data and joint hypocenter determination

6.28.1 VELEST

Introduction

The program VELEST is used to solve the coupled hypocenter velocity model problem for local earthquakes. It performs a simultaneous inversion for hypocenters and velocity model. The inversion is limited to first arriving phases. A detailed program description is given in the 'VELEST USER'S GUIDE' [Kissling, 1995]. A recipe for preparing data and use of the inversion routine is presented in 'Initial reference models in local earthquake tomography' [Kissling et al., 1994]. The two documents are available in one Postscript file in the INF directory, the filename is 'velest.ps'. The derived model can be used as an improved model for earthquake location or as a starting model for 3-D inversion. For a fixed velocity model and constant station corrections, VELEST in simultaneous mode performs the Joint-Hypocenter-Determination (JHD).

Before you start please see the two articles Kissling-1988.pdf and Kisslig-1994.pdf in the INF directory.

The original version of VELEST by Kissling is included in the Sun and Linux versions. A version modified to compile VELEST under Windows has been provided by Freddy Aldersons (e-mail: faldersons@earthlink.net). This Windows package is included in the file velest_pc_3.3.zip, which is located in the SUP directory. The files have to be extracted to the PRO directory.

The implementation of VELEST to SEISAN is given by the program VELMENU.

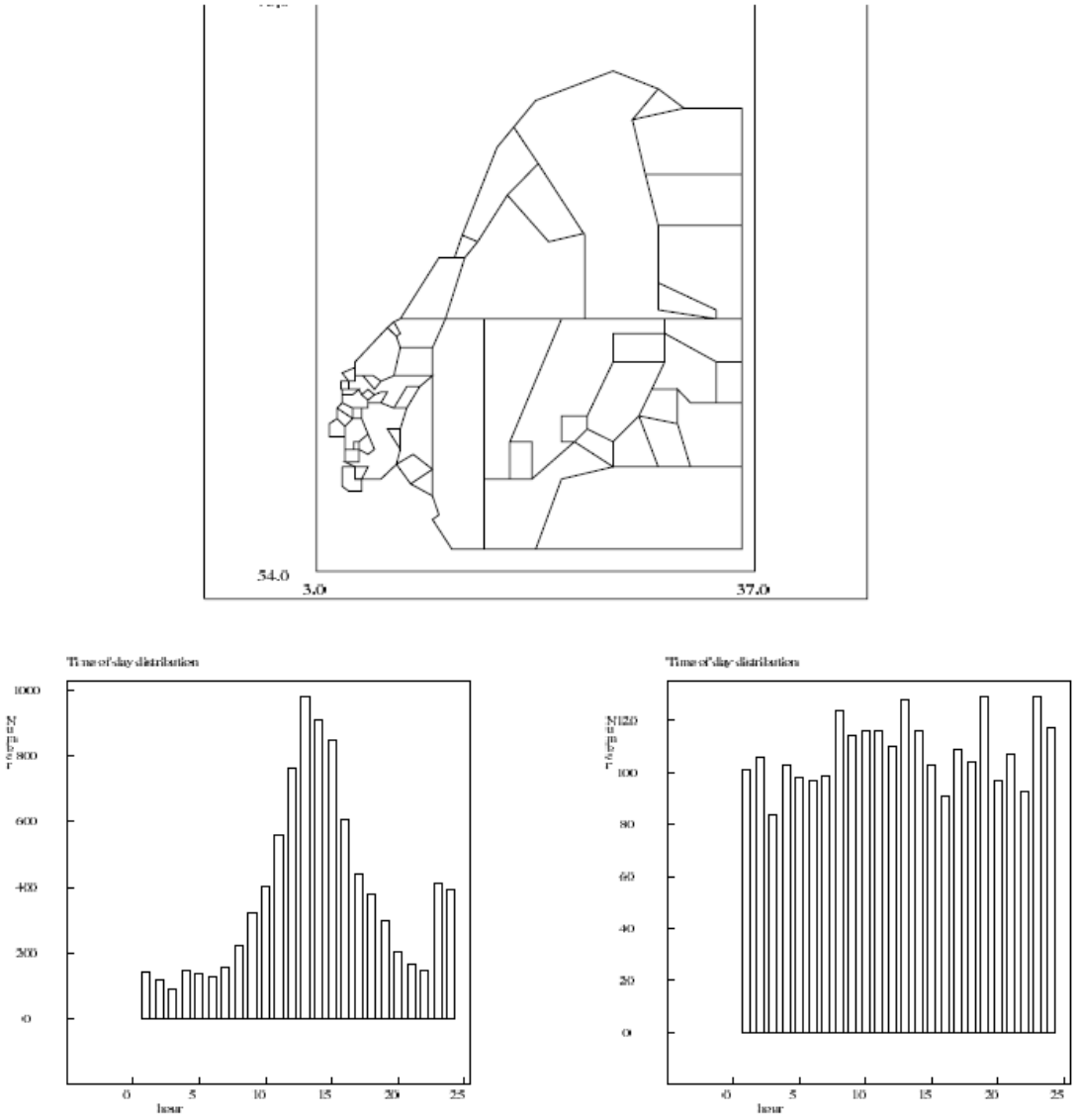


Figure 6.33: show how the filter works for events in Scandinavia.

VELMENU provides:

- automatic format conversion to VELEST
- generation of parameter files using the SEISAN system
- execution of VELEST
- conversion back to SEISAN format

After preparing a dataset of local earthquake data, VELMENU can be used to work with the VELEST inversion routine. The first time VELMENU is used, all input files for the inversion with default parameters can be generated. These parameter files then can be changed interactively and the inversion started with VELMENU.

Running VELMENU

The program is started with 'velmenu'. After entering the filename of the earthquake data the menu of VELMENU appears.

Example of program run :

```
velmenu

File name of earthquake data in Nordic Format :
select.out

VELEST MENU
-----------

1. Create VELEST command file (vele\index{Velest.cmn}st.cmn)
2. Edit/change VELEST command file (velest.cmn)
3. Create station select file (selstat.lis)
4. Edit/change station select file (\index{Selstat.lis}selstat.lis)
5. Create model file
6. Edit/change model file
A. RUN VELEST
B. Edit inversion output file
C. Convert VELEST output to Nordic format and make diff-file
Q. End

Choice ?
```

The complete inversion-process of earthquake data in SEISAN format, including all conversions and preparation of parameter files, can be done with VELMENU. The steps are as follows:

1: Create VELEST command file (`velest.cmn`)
The user is asked for inversion or JHD and the appropriate parameters are set. The file `velest.cmn` is the central VELEST parameter file. To create it, the file of earthquake data is read to determine the parameters that depend on the data. These are the number of events and the center of Cartesian coordinate system, which is simply determined as the average of latitude and longitude of epicenter locations. The remaining parameters are set to default values.

3: Create station select file (`selstat.lis`)
For the inversion, VELEST will use phases from stations with an epicentral distance below a maximum distance only. In addition in VELMENU a selection of stations has to be used, only phases from stations given in the file `selstat.lis` will be used for inversion. When generating the file, the maximum distance between station and hypocenter (parameter 'dmax') is read from `velest.cmn` and the input data are scanned to get a list of stations, which are within the limit to any epicenter. Editing the file, stations can be added or removed. If all stations should be used for inversion, the parameter 'dmax' in the file `velest.cmn` has to be increased.

Example of `selstat.lis`:

```
#
# STATION SELECT FILE FOR PROGRAM VELEST
#
# STATIONS WILL BE USED IN THE VELEST
#      INVERSION PROGRAM
#
# COMMENT LINES START WITH
#
#
KONO
BER
NRAO
...
```

NOTE: The order of the stations is as given by the input data file. VELEST uses the last station as reference station, so you may want to change the order.

5: Create model file
The input model file 'model.inp' is created using the model as given in the '`STATION0.HYP`' file. The '`STATION0.HYP`' file, if available, will be read from the local directory, otherwise from the DAT directory. This might be a reasonable starting model, but of course the model file has to be changed.

A. RUN VELEST

Once the parameter files are created the inversion program can be started. The inversion study requires interactive changing of parameters, which is supported by VELMENU. All input parameter files can be changed from VELMENU. NOTE:, '... please accept the warning: To calculate a Minimum 1-D model a single or even a few VELEST runs are useless, as they normally do not provide any information on the model space!' [Kissling, 1995]. The conversions and the inversion programs are started as one process.

Before the inversion routine is started the station locations will be converted from the `STATION0.HYP` file and the earthquake data in Nordic format will be converted to CNV (hypocenters and travel times) format. NOTE: VELEST does not support 5 character station codes, therefore in the conversion to VELEST, only the first 4 characters are used if the station code has 5 characters. In the conversion of the earthquake data only phase readings from stations included in the station selection file will be used. Arrivals with a time residual, given in the Nordic input file, above five seconds are omitted. Only the first arriving phase of P and S respectively are used. The hypocenter location given by the inversion will be determined by first arrivals only. The original data might include more phases like Pg, Sg or Lg. Therefore, to get a comparison of hypocenter locations between the HYP location program and VELEST, a Nordic file including the same data as the CNV file is created and the HYP program run on this file before VELEST is started. The HYP program can be skipped by pressing 'CTRL+C', while it is running.

The results of the inversion will be given in a text file that can be viewed within VELMENU. VELMENU provides an option to convert the VELEST output file with final hypocenter locations in CNV format back to Nordic format and to write a file that shows differences (`velout.dif`) in location and time between the two location routines, HYP and VELEST, based on the same input data.

Example of `velout.dif` :

```
differences: inversion output - SEISAN input
first line input, second line output, third line difference

 1984  8 5 0235 22.7 L  59.449   4.968  0.0  BER  9 1.3 1.9CBER 3.7BBER        1
 1984  8 5  235 22.1 L  59.458   5.140  0.1  BER  9                            1
time:   -0.6 latitude:   0.009 longitude:    0.172 depth:    0.1

 1984  8 6 0406 26.6 L  59.538   5.678  1.9  BER  9 1.8 1.7CBER 3.6BBER        1
 1984  8 6  4 6 26.6 L  59.505   5.655  6.2  BER  9                            1
time:    0.0 latitude:  -0.033 longitude:   -0.023 depth:    4.3
...
```

Files will be overwritten, when VELMENU is started again. To work with different datasets or parameter files it is recommended to work on different directories or to change the filenames, but note that the default filenames (see below) will be used in VELMENU.

Problems: VELEST skips events without phase readings and therefore the number of events read by VELEST will be different from the number given in the `velest.cmn` file. If this is the case VELEST stops with the message STOP: ...end...(VELEST was running with the SINGLEEVENT-OPTION). Events without phase readings will not be listed in the invers.out file, and should be deleted from the input file.

Joint-Hypocenter-Determination (JHD)

VELEST for fixed velocities and station corrections can be used as a JHD routine. For JHD, VELMENU is used in the same way as described above for inversion. The only difference is that when generating the `velest.cmn` you have to choose JHD. The appropriate file for JHD is then generated. Some parameters in the '`velest.cmn` file are different, compared to the inversion. These are dmax, nsinv and invertratio, see 'VELEST USER'S GUIDE' for details. The output of final hypocenter locations as described above can be converted to Nordic format, but note that the JHD will be based on first arriving phases only.

Example of JHD:

Potential problem: We have seen cases where in JHD mode the depth parameter in the inversion is sensitive to invertratio, which when set to 1. in JHD means that VELEST inverts for station correction in every iteration. VELEST in this case worked better with an invertratio of larger than 1. See VELEST manual for details.

List of files generated by VELMENU / VELEST

`data.cnv`	earthquake data in CNV format, VELEST input, generated by VELMENU
`data.nor`	earthquake data in Nordic format, HYP input, generated by VELMENU
`fin_hyp.cnv`	final hypocenter locations in CNV format, VELEST output
`hyp.out`	earthquake data in Nordic format, HYP output
`hypsum.out`	HYP output file
`input.mod`	input model, VELEST input, generated by VELMENU
`invers.out`	documentation of inversion, VELEST output
`nor1.date`	earthquake data in Nordic format, VELMENU input
`print.out`	HYP output file
`selstat.lis`	selection of stations, generated by VELMENU
`sta_cor.out`	station corrections, VELEST output
`station.sta`	station locations, VELEST input, generated by VELMENU
`velout.dif`	difference file between HYP and VELEST location routine, VELMENU output
`velout.nor`	final hypocentre locations, same as `fin_hyp.cnv`, in Nordic format,

VELMENU output
velest.cmn VELEST control file, VELEST input, generated by VELMENU

6.28.2 NOR2DD

By **Brian Baptie**, BGS

This program produces input for the Double Difference Program HYPODD [Waldhauser, 2001; Waldhauser and Ellsworth, 2002] from Nordic and `STATION0.HYP` files. The Nordic file has to be given as argument when running the program, example:

```
nor2dd select.out
```

The files created are:

`phase.dat` : phase input data
`station.dat` : station coordinates

HYPODD is available from: `http://geopubs.wr.usgs.gov/open-file/of01-113/`

6.28.3 NOR2JHD_PUJOL

By **Brian Baptie**, BGS

This program produces input for Pujol's JHD program [Pujol, 2003] from Nordic and `STATION0.HYP` files. The Nordic file has to be given as argument when running the program, example:

```
nor2jhd_pujol select.out
```

The files created are: `syn.times` : phase input data
`syn.vel` : velocity model
`stalist.elev` : station list

The inversion tool is available from:
ftp://beagle.ceri.memphis.edu/pub/pujol/JHD
`http://www.orfeus-eu.org/links/softwarelib.htm`

6.29 Analysis of volcanic earthquakes

SEISAN is often used for volcanic monitoring. Many of the standard tools used in SEISAN can also be used for volcanic earthquakes, like epicenter location and magnitude. However, more special tool are also needed and below there is a description how this is done at the British Geological Survey (BGS).

Another description, made by Andrew Lockhart at the USGS, is given in a separate manual (`seisan_volcano.pdf` in INF or at `http://seis.geus.net/software/seisan/seisan_volcano.pdf`). This manual is a mini SEISAN manual detailing the steps under Windows.

Volcano monitoring at BGS

By **Brian Baptie**

Background

An important part of volcanic seismology and the seismic monitoring of active volcanoes is the correct recognition of the different types of seismic event generated by the volcanic activity. The principal event types include, volcano-tectonic events, caused by shear or tensile failure of rocks; long period events, generated by a volumetric source in a liquid; hybrid events; and volcanic tremor.

To be of value for volcanic monitoring, any database of seismic events should include the type or sub-class of individual events. This should allow users to then extract phase and location information over a selected time period for individual event types and calculate hourly and daily rates of event. Simple histogram plots showing the distribution of subclasses over time can be generated with the program VOLCSTAT (Unix only).

Initialization

The user should create a text file in the DAT directory called `VOLCANO.DEF` (an example is already in the directory). The format of this file will be one line of text (80A) followed by successive lines with the format "i2,1x,6A,1X,40a" for number, sub-class code and description. An example of the file is shown below. Comments are preceded with '!'.

```
Current volcano sub-classes:   ! Comment line 80 characters
1  vt     volcano-tectonic     ! Individual sub-class line
2  hybrid hybrid
3  lp     long-period
4  tremor volcanic tremor
5  rf     rockfall
6  un     unknown
7         QUIT                 ! The last line should contain
                                  this entry
```

Registering volcanic sub-classes

Registration should be carried out as normal in MULPLT. From multi-trace mode enter 'p' to create a new s-file for the event in the database. Answering 'LV' to the prompt for event type marks the event as a local volcanic in the headers. If the `VOLCANO.DEF` file has been set up correctly in the DAT directory, the information on the different sub-classes will be printed to the terminal. Choosing an appropriate number selects the volcanic sub-class. The sub-class code is then entered in the s-file.

Modification of the s-file to incorporate volcanic sub-classes

The volcanic sub-class information is stored in a type 3 line within the s-file, e.g.

```
 VOLC MAIN tremor 3

Columns 2:10  'VOLC MAIN'    : Header identifier
Columns 12:17    a6          : Sub-class flag
Column  80       '3'         : line type identifier
```

This allows the use of a maximum 6-character sub-class identifier, e.g. 'hybrid', which can then be searched for and selected.

VOLCSTAT: Creating histogram plots

The program reads S-files directly from the database, and creates input files as well as a GMT script to produce histogram plots of the distribution of subclasses over time. The user needs to enter database name, start and end time, and the subclasses that are to be plotted. An example of a plot is shown in Figure 46. The program supports 1-char subclass names only.

The following output files are created:

`volcstat.batch` - c-shell script to generate Postscript output using GMT
`volcstat_counts.ps` - Postscript output file
`volcstat_counts_<type>.out` - for each event, the Julian date is written out, one file per subclass
`volcstat_daily_<type>.out` - number of events per day, files written for each subclass
`volcstat_counts_total.out` - total event counts for each subclass

RSAM

1-minute RSAM data can be created with WAVETOOL.

Future Extensions:
It is intended that additional parameters can be included in the above structure to included routine measurements of the volcanic earthquakes. For example, signal duration, peak amplitude and mean frequency can be calculated for individual stations and included on additional type 3 lines with a volcanic identifier. Parameters on each channel can then be averaged an inserted on the volcanic header line.

The proposed format for these lines is as follows

```
column      format         description
2:5          a4              'VOLC'    Volcanic identifier
7:10         a4              station
12:15        a4              component
19:20        a2              'PA' field identifier
22:29        g8.3            peak amplitude
31:32        a2              'DU' field identifier
34:41        g8.3            signal duration
43:44        a2              'MF' field identifier
46:53        g8.3            mean frequency
55:56        a2              'SB' field identifier
58:65        g8.3            signal bandwidth
66:79                        blank
80           a1              '3' line type identifier.

For example
```

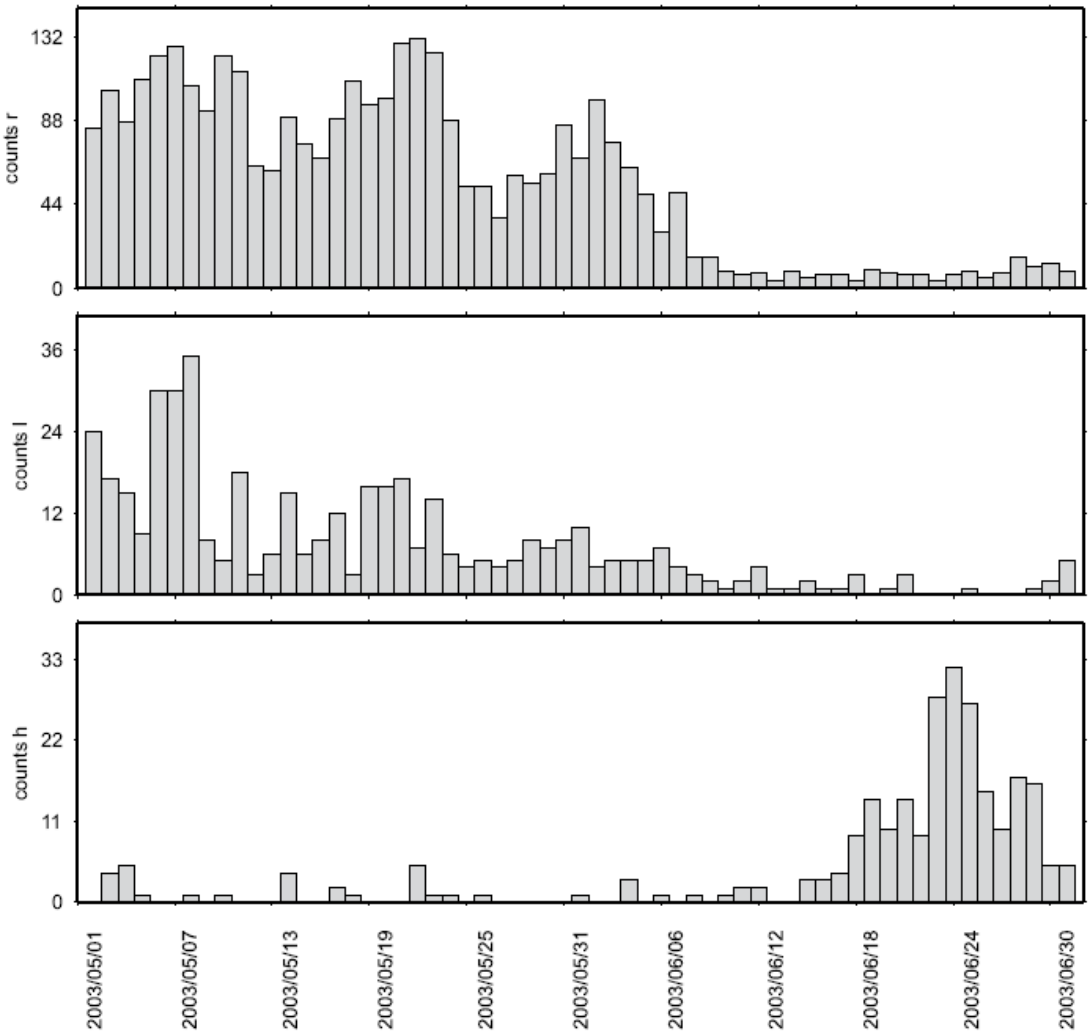


Figure 6.34: Bar diagrams showing distribution of events of different subclasses over time.

```
VOLC MAIN tremor                                                          3
VOLC KTK1 S  Z   PA .152E 06  DU    1.325 MF    2.472                     3
VOLC KTK1 S  N   PA .167E 06  DU    1.997 MF    2.067                     3
VOLC KTK1 S  E   PA .141E 06  DU    1.543 MF    1.998                     3
```

This method of inclusion of volcanic parameters should allow for future flexibility such as incorporation of an additional parameter fields in columns 66 to 79. Also the use of type 3 lines means that existing software, such as the update program, are unaffected by these lines.

6.30 FK Analysis

The FK routines were provided by **Tormod Kværna** from NORSAR and implemented into SEISAN by **Andrius Pacesa**.

Some basics

The FK-analysis, more strictly slowness analysis, is a standard tool in seismic array processing. It is used to find the apparent velocity and back azimuth of an incoming wavefront. Apparent velocity can be used to identify the type of wave (P, S, Lg and etc.) and the approximate distance to the source can be determined for teleseimic events. Utilizing azimuth and distance to the source, one can define the approximate location of the signal source.

A description of frequency-wavenumber analysis - f̈-k analysis¨ may be found in Capon (1969). This method has been further developed to include wide-band analysis and maximum-likelihood estimation techniques - see Kvaerna and Doornbos [1986].

The principle of slowness analysis is beamforming in the frequency domain for a number of different slowness values and calculating the power for each beam. The beam power will be a maximum in case the slowness of the beam coincides with the slowness of the wavefront crossing an array. So the beam having the maximum power will indicate the slowness of the incoming signal.

Running the program

The FK program can be started directly with command 'fk' or from MULPLT. The program expects that the file 'waveform.out' with the seismic traces as input data, is available in the current directory. If the program is invoked directly, this file has to be created before using mulplt, selecting a window and creating the 'waveform.out' file.

In general it is more useful, to start the FK program from MULPLT since the input file needs to be created by mulplt. The result of the fk analysis can be saved to the S-file.

The steps are:

- start MULPLT
- select channels and a time window
- use option fk to start FK program (this option creates file 'waveform.out' and starts FK program), accept maximum or pick value with mouse

  The options in FK are:

  R-Redo: Repeat fk analysis with different parameters

M-Mouse: 'm' or mouse click to pick values different from maximum
S-Save and quit: save picked value to file and quit
Q-Quit: quit

- use option 'save and quit' to save your result, so that it can be used by MULPLT
- back in MULPLT: pick phase on the first trace used, to store back azimuth and apparent velocity in the S-file
- in case of teleseismic events, the apparent velocity can be used for location, the fk analysis has to be done on the P phase

Note: The FK program only works by default with station file '`STATION0.HYP`'. If coordinates are in e.g. `STATIONt.HYP`, the user will be asked to specify another station file letter, in this case 't'.

Example

Input:

```
Enter lower frequency,<ENTER> for default value 1 Hz

Enter higher frequency,<ENTER> for default value 5 Hz

Enter max slowness, <ENTER> for default value 0.4 s/km

Enter No of grid points, <ENTER> for default value 51

Do you want to plot level numbers, N/Y, <ENTER>=Y

It will take some time ...
APPARENT VELOCITY  10.26   AZIMUTH          140
NORM. POWER MAX     0.99   POWER MAX IN dB  72.44
MAX X SLOWNESS      0.06   MAX Y SLOWNESS   -0.07
LOW FREQUENCY       0.94   HIGH FREQUENCY    5.00
QUALITY (1=best, 4=worst)  1
 Plot file is fk.eps
```

Example of output file fk.out:

```
DATE, TIME AND WINDOW LENGTH     99 256  9 13 12  0 33.843     7.62
APPARENT VELOCITY  10.26   AZIMUTH          140
NORM. POWER MAX     0.99   POWER MAX IN dB  72.44
MAX X SLOWNESS      0.06   MAX Y SLOWNESS   -0.07
LOW FREQUENCY       0.94   HIGH FREQUENCY    5.00
QUALITY (1=best, 4=worst)  1
VALUES TO SAVE
APPARENT VELOCITY  10.26   AZIMUTH          140
Station     Long      Lat         Elev   Xcoord    Ycoord Zcoord
NRA0 S  Z   11.54150  60.73533    300         0         0    300
NRA1 S  Z   11.54233  60.73650    291        45       129    291
```

```
 NRA2 S  Z   11.54333  60.73433    311        99      -111    311

 NRD4 S  Z   11.56333  60.72717    379      1186      -907    379
 NRD5 S  Z   11.54750  60.72217    348       326     -1463    348
 NRD6 S  Z   11.52883  60.72334    352      -688     -1334    352
 NRD7 S  Z   11.51617  60.73017    337     -1377      -574    337
 NRD8 S  Z   11.51667  60.73900    301     -1349       407    301
 XSLOW   YSLOW  POWER
  0.40    0.40   0.05
  0.38    0.40   0.05
  0.37    0.40   0.05
  0.35    0.40   0.05
...
```

6.31 Surface wave analysis (SUN)

The programs by **Robert Herrmann** (Herrmann, 1996) to estimate the shear wave velocity of the earth by inversion of surface wave group velocities are distributed with Seisan. The programs are part of Herrmann's package "*Computer Programs in Seismology*". For more information check:

`http://www.eas.slu.edu/People/RBHerrmann/ComputerPrograms.html`

The programs have been implemented with SEISAN by Vunganai Midzi, who has written a guide on how to use the programs. This document is given as the Word file 'surface.doc' in the INF directory. Also see section 6.2.12 for details on output files that can be generated with MULPLT. The programs are included in the tar file 'surface.tar' in the PRO directory, but are not installed as part of the standard installation. If you wish to use the programs, you need to extract the programs:

```
cd <SEISAN_TOP>/PRO
tar xvf surface.tar
```

You can recompile the programs with the commands:

```
make -f Makefile.sur clean
make -f Makefile.sur
```

6.32 Instrument response

In SEISAN the instrument response can be stored as pairs of frequency, amplitude and phase or as poles and zeros. The formats that can be used include GSE2, SEISAN and SEED. The SEISAN and GSE response formats are described in Appendix C. For a detailed description of the GSE format, the reader is referred to GSETT-3 [1997]. The program RESP creates response files in SEISAN and GSE format. SEED format response files can be extracted from a SEED volume.

6.32.1 Create instrument response files, RESP

Introduction

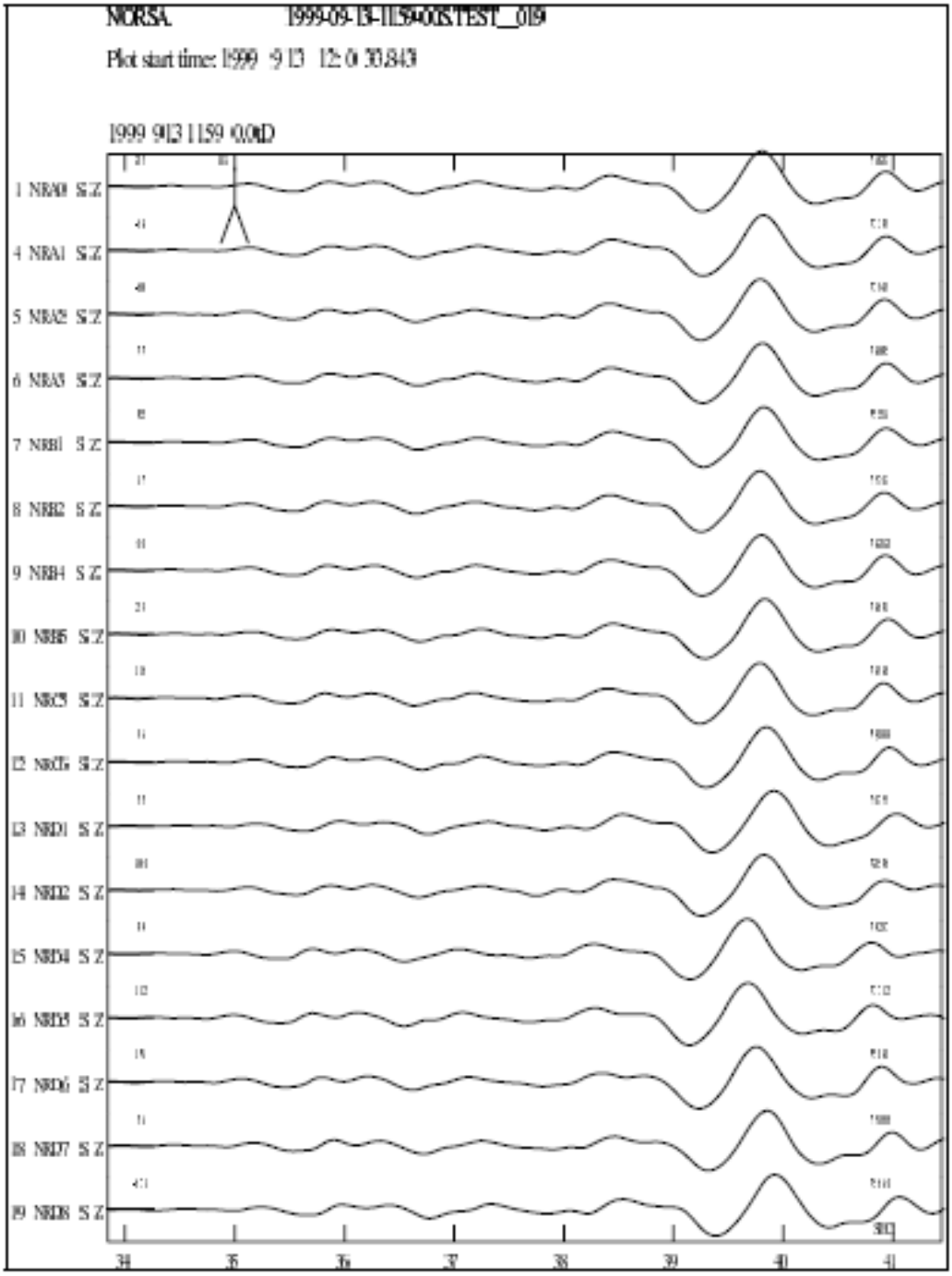


Figure 6.35: The FK program can be started from MULPLT. The traces shown were selected and used as input to the FK program. The result of the FK analysis is shown in Figure 6.36. The event shown here is part of the testdata set.

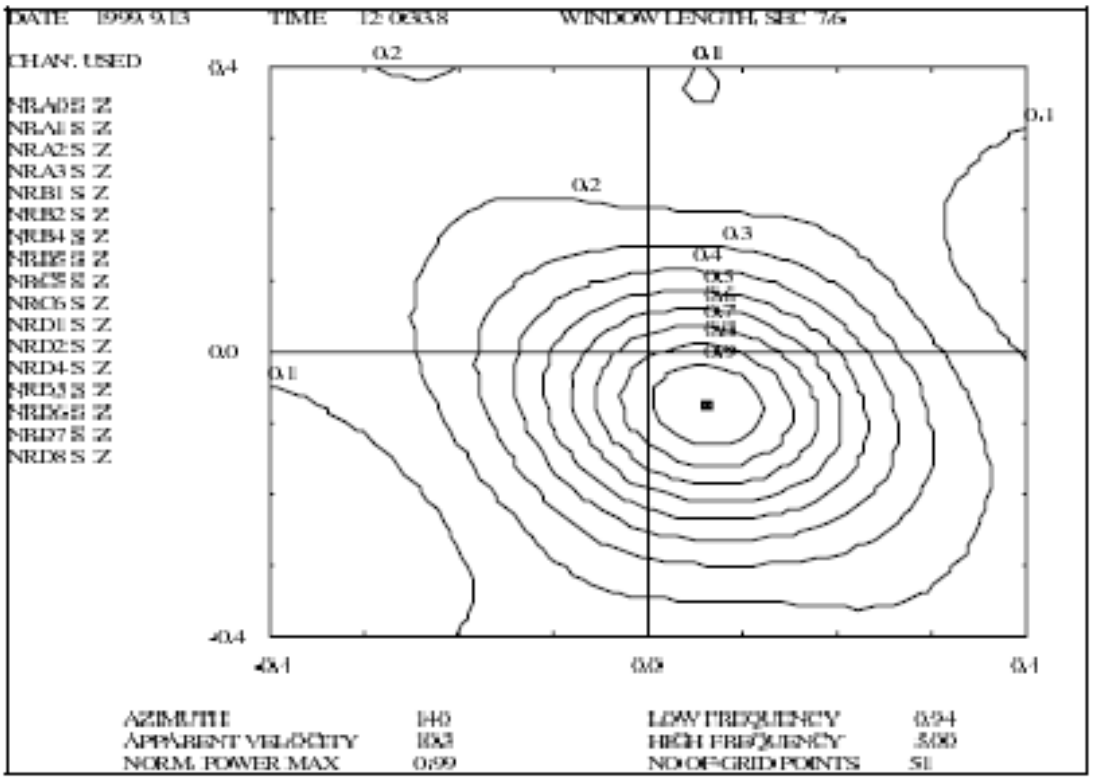


Figure 6.36: Output from the FK program. Contours and values are the normalized maximum power.

The purpose of this program is to (1) Make Seisan or GSE2 response files, (2) Provide the engineer maintaining seismic instrumentation with a practical tool for calculating and checking response functions of the most common elements of a seismic system. The program can calculate response functions of velocity transducers, accelerometers, filters and amplifiers, input poles and zeros or tabulated values and multiply the combinations together to get complete system response functions. The program produces a table with the response function and a simple graphical expression of the response curve. For the purpose of checking measured values, a file with these values can be used as input and will be plotted together with the theoretical values. The program can calculate, acceleration, velocity or displacement response. Program PR_ RESP can make a table of many response files.

The instrument response

The seismic recording system can consist of seismic sensor, analog-digital converter, amplifier and filters. For a detailed discussion the user is referred to Scherbaum [1996]. The combined response can be given in the frequency domain as frequency response function or in the Laplace domain as transfer function. The frequency response is given in pairs of frequency amplitude phase (FAP), while the transfer function is given as poles and zeros (PAZ). The combined frequency response is obtained through multiplication of the response from the individual components, while the transfer function is obtained by combing the PAZ from the components. Amplifiers and accelerometers are specified simply by a constant gain. Filters are assumed to be Butterworth. RESP can be used to write finite impulse response (FIR) coefficients [Scherbaum, 1996] that are used as anti-alias filters in most modern digitizers if GSE is used as output format. SEISAN has no capability to read the FIR filters or to correct for them. However, the FIR filters are part of a full description of the instrument response and should be at least included for information if possible.

The seismometer is assumed to have the following velocity frequency response:

$$T(\omega) = \frac{\omega^2}{\omega_0^2 - \omega^2 + i2\omega\omega_0 h}$$

Which corresponds to the transfer function:

$$T(s) = \frac{-s^2}{\omega_0^2 + s^2 + 2s\omega_0 h}$$

where $s = i\omega$, ω is the angular frequency $2\pi f$ in Hz, ω_0 the resonance frequency of the seismometer, $i = \sqrt{-1}$ and h the damping (normally around 0.7).

NOTE: In the equation for the frequency response, the sign "+ 2*i*h.." was "-" before March 2000, so old parameter files may have to be regenerated. The sign depends on the definition of the signs in the Fourier transform and therefore may be different in different text books. It may even be wrong although it looks right, if a wrong Ansatz is done. Due to the wrong sign, the FAP values in the SEISAN response files were wrong, however the programs use the constants given in the files and the correct response is generated. If you have the instrument constants in your old response files and not just FAP, the old response files can be used.

The transformation from displacement to velocity or back is done by multiplying with i*..

In addition to or instead of using the equation above, values can also be entered as discrete values or as poles and zeros.

The SEISAN response function is calculated for 60 frequencies between 0.01 and 100 Hz and the steps between the frequencies are approximately logarithmic. The response function is normalised at 1.0 Hz (

see Table 1) and the gain at 1.0 Hz is given separately.

NOTE: ALL UNITS ARE IN METERS, SECONDS OR G (9.8ms-2)

NOTE: It seems that although the GSE format is clearly defined, there has been different interpretations. This has also led to changes in SEISAN since the GSE response was introduced with SEISAN. For more details, see Appendix C.

Which format to use

SEISAN, since version 7.1, supports the GSE2 calibration format in addition to the SEISAN response file format. We recommend that you use the GSE2 format, since it presents one of the most widely used calibration formats. Storage of the response in terms of PAZ is recommended over FAP, since the PAZ representation describes the continuous transfer function. You may continue using existing SEISAN response files and add new files in GSE2 format, or replace the old SEISAN response files with new GSE2 files.

How to run the program

The program has quite a few options, which easily may lead to confusion. Before you start you should know which format you want to use (GSE2 or SEISAN) and whether you want to describe the response in terms of FAP or PAZ. The recommended choice is to use GSE2 and PAZ.

Type RESP to start the program. You will then get a series of questions as indicated below in upper case letters. All input is format free. A sample run is shown below.

CHOSE OUTPUT FORMAT:
0: NO OUTPUT FILE
1: SEISAN FAP
2: SEISAN PAZ
3: GSE2 FAP
4: GSE2 PAZ

Answer with 0-4, options 1-4 will create respective response files in selected format, option 0 will only calculate and show the response on the screen. SEISAN PAZ can only be used if number of poles and number of zeros are less than 38. If more are input, a table will be generated automatically in FAP format.

TYPE OF SENSOR:
1: NONE
2: SEISMOMETER
3: ACCELEROMETER

Answer with 1, 2 or 3. Number 1 is used when only calculation of filters or amplifiers are desired, 2 is a standard velocity transducer and 3 a standard accelerometer. If a seismic sensor is used, you will get additional questions on the constants of the sensor. If a seismometer is chosen, the following questions must be answered:

SEISMOMETER NATURAL PERIOD ?

This is measured in seconds. For most short period systems the value would be 1.0 second.

SEISMOMETER DAMPING RATIO ?

The damping ratio should ideally be 0.7. This depends on the damping resistance.

For both the seismometer and accelerometer, the following question is given:

SENSOR LOADED GENERATOR CONSTANT (V/M/S OR V/G) ?

This is the generator constant of the sensor in terms of volt per unit of of ground motion (meter/second

or g). It is important to note that this is the loaded constant, which means the effective output of the sensor taking into account amplifier input and damping resistances.

Now comes questions about amplifier, filter and recording unit.

RECORDING MEDIA GAIN (COUNT/V OR M/V) ?

If you have a recording media, the gain can be given here, otherwise just enter 1.0

If the output format is GSE, the response is always calculated in displacement units, while for SEISAN output and seismometer or accelerometer, the following options appear:

TYPE OF RESPONSE: 1: DISPLACEMENT
2: VELOCITY
3: ACCELERATION

Normally for a seismometer, one wants to calculate the displacement response and for an accelerometer, the acceleration response. However it might sometimes be interesting to look at e.g. the velocity response for a seismometer (after all, the seismometer is normally a velocity transducer !!). Enter the appropriate number.

AMPLIFIER GAIN (DB) ?

This is the amplifier gain in dB. Since this question is only asked once, this gain must include gain of all units except the recorder (asked below). This could e.g. include gain of the VCO system.

NUMBER OF FILTERS (0-10), RETURN FOR NONE ?

Up to 10 filters can be specified. If you answer 0, no filters are used and no more questions on filters will appear. Otherwise one line of input must be given for each filter as follows:

FREQUENCY AND NUMBER OF POLES FOR EACH FILTER,
POLES NEGATIVE FOR HIGH PASS

Each line requires two numbers, the corner frequency of the filter and the number of poles. A high pass filter is given by letting number of poles be negative. It is not always easy to know whether a filter is e.g. one 2 pole or two 1 pole filters, the user needs to experiment with this.

FILE NAME FOR FILE WITH POLES AND ZEROS, RETURN FOR NO FILE

Here a file with poles and zeros can be entered. If seismometer constants have been chosen above, the values calculated with poles and zeros are multiplied with the values previously calculated. The free format file contains:

1. line: NP: Number of poles, NZ: Number of zeros, Norm: Normalization constant
Following NP lines contain one pair each of real and imaginary poles
Following NZ lines contain one pair each of real and imaginary zeros

NOTE: The unit of frequency is radian/s so if in Hz, multiply with 2π and normalization constant in $radian = (normalizationconstantinHz)2\pi^{(numberofpoles-numberofzeroes)}$.

The next 2 options are only shown if the output file is selected to be FAP:

FILE NAME FOR TABULATED VALUES, RETURN FOR NO FILE

Here a file with tabulated values are entered. If seismometer constants or poles and zeros have been chosen above, the tabulated values will be interpolated and multiplied with the values previously calculated for from above. The free format file contains:

1. line: N: Number of tabulated values, Norm: Normalization constant

Following N lines contain one each frequancy, amplitude and phase(deg)

GIVE FILE NAME FOR MEASURED VALUES, RETURN FOR NONE

Give file name for measured values. In most cases you have none so just make a return. The format of the input file is as follows:

frequency, amplitude, phase
frequency, amplitude, phase
etc.

e.g.

0.2,0.7,200
0.7,0.8,100
10.0,0.1,33

The file has no blank lines and can contain up to 60 data sets. It is important to note that the amplitude values should be NORMALIZED at 1.0 Hz.

Now there is no more input to the response parameters, and the output is:

GAIN FACTOR AT 1.0 HZ: 12345.6

This is the gain of the system at 1.0 Hz and is also the value for normalizing the response curve, that is, all calculated values are divided by this number. There is no unit for gain of an amplifier and for displacement response using a seismometer and drum recording. If the recording is digital, the unit would be counts/meter and for a velocity response counts/meter/second etc. If a file with poles and zeros is used without any other information, the normalization constant must have the unit of count/m, similar for the tabulated input.

Further output is given in a file called resp.out, see Table 1 for an example.

The response curves (amplitude and phase) are now printed/plotted on the screen. First comes the amplitude response (amplitude in db versus log frequency). By pushing return, the phase response is shown (phase shift (deg) versus log frequency). After the plots, the SEISAN calibration file can optionally be made, follow instructions, see example below. The response file MUST be calculated for the displacement response, and all calculation in SEISAN assume that response is calculated in counts/m.

After the SEISAN response file is made, the current parameters will be displayed and one or several can be changed without entering all again. Like if the gain has changed at a certain date, only change date and gain. This feature (new in SEISAN7.2) has been put in to be able to quickly make many similar response files, like when all files have to be put in for a network.

Comments to data for response files

Station and channel codes
It is important that the station and channel codes are made exactly as they appear in the waveform files. If not, SEISAN is not able to identify the channel.

Date
The date given here corresponds to the date from which the calibration information is valid. The SEISAN system will always look for the most recent calibration file relative to the date of the earthquake.

Latitude, longitude and elevation
These data are for information only, it is not used anywhere in SEISAN, so it does not have to be entered, however there is room for it in the SEISAN waveform file headers.

Comment
No information used by the system.

Plot
After the response file has been written out, a plot is made with PRESP of the file. There will also be a plotfile, `presp.eps`, which can be sent to the printer. The response file can store the response in different ways:

1. Parameters used for calculating the response: Generator constant, filters etc. In addition, the response (amplitude and phase) at 30 frequencies are listed. In this case the response is calculated from the parameters.

2. Incomplete set of parameters or no parameters and the response at 30 frequencies. In this case the response is calculated by interpolation of the 30 values.

3. Poles and zeros: No discrete values are given and the response is calculated directly from the poles and zeros.

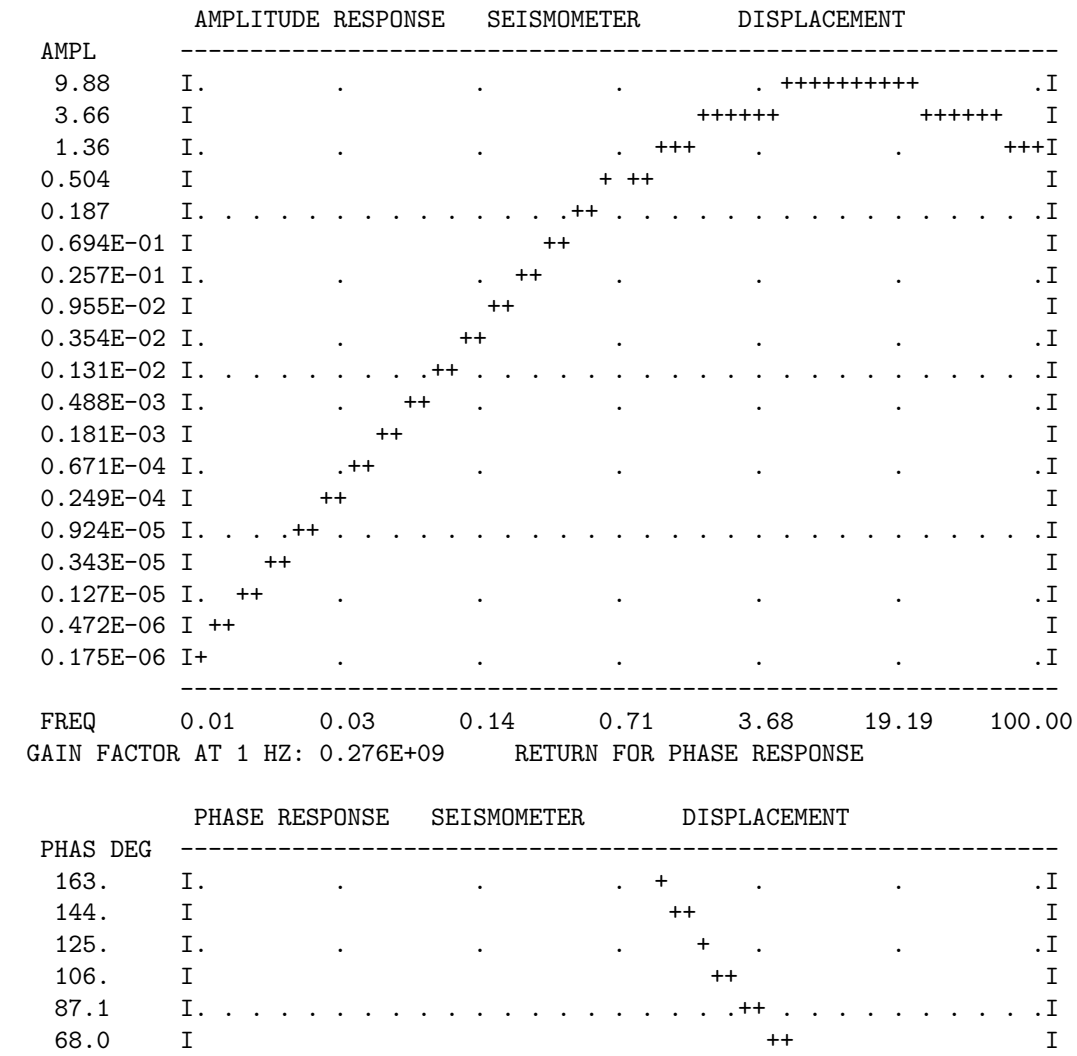
See also Appendix B for the SEISAN waveform file format and section 4.6.

IMPORTANT: PUT RESPONSE FILE IN CAL DIRECTORY OR ONE OF ITS STATION SUBDIRECTORIES. Response files can also be in working directory but this is not advisable except for testing.

Example of running the program:

```
oxygen-(larso)23: resp

  RESP - PROGRAM TO CREATE RESPONSE FILES IN SEISAN
         OR GSE FORMAT. THE RESPONSE CAN BE CREATED
         AS POLES AND ZEROS (PAZ) OR FREQUENCY
         AMPLITUDE AND PHASE (FAP). THE SAME
         TRANSFER FUNCTION AND FILTERS ARE USED
         IN BOTH CASES.

  CHOSE OUTPUT FORMAT: 0: NO OUTPUT FILE
                       1: SEISAN FAP
                       2: SEISAN PAZ
                       3: GSE2 FAP
                       4: GSE2 PAZ
4
  TYPE OF SENSOR:       1: NONE
                        2: SEISMOMETER
                        3: ACCELEROMETER
2
  SEISMOMETER NATURAL PERIOD ?
1
  SEISMOMETER DAMPING RATIO ?
.7
  SENSOR LOADED GENERATOR CONSTANT (V/M/S OR V/G) ?
300
```

```
  RECORDING MEDIA GAIN (COUNT/V OR M/V) ?
2048
  AMPLIFIER GAIN (DB) ?
40
  NUMBER OF FILTERS (0-10), RETURN FOR NONE ?
1
  FREQUENCY AND NUMBER OF POLES FOR EACH FILTER,
  POLES NEGATIVE FOR HIGH PASS
10 2
  FILE NAME FOR FILE WITH POLES AND ZEROS, RETURN FOR NO FILE

  FILE NAME FOR MEASURED VALUES, RETURN FOR NO FILE
```




```
GAIN FACTOR AT 1 HZ: 0.276E+09     RETURN FOR PHASE RESPONSE
```

```
   49.0     I.         .         .         .         .  ++     .         .I
   29.9     I                                            +                 I
   10.8     I.         .         .         .         .     +   .         .I
  -8.21     I. . . . . . . . . . . . . . . . . . . . . . .++ . . . . . .I
  -27.3     I.         .         .         .         .        +.         .I
  -46.3     I                                                 +++        I
  -65.4     I.         .         .         .         .         .  ++++   .I
  -84.4     I+++++++++++++                                             ++++I
  -103.     I. . . . . . . ++++++++++. . . . . . . . . . . . . . . . . .I
  -123.     I                        +++                                 I
  -142.     I.         .         .      ++ .         .         .         .I
  -161.     I                             + +                            I
  -180.     I.         .         .         . +       .         .         .I
            ------------------------------------------------------------
   FREQ     0.01      0.03      0.14      0.71      3.68     19.19    100.00
 GSE RESPONSE FILE (Y/N=default)?y
 Enter station code. e.g. BERGE, max 5 chars
TEST
  Enter component (4 chars) e.g. SH Z
  First character is type, should be one of the following:
  E, S, H, B, M, L, U
  Second character is instrument code, should be one of the following:
  H, L, G, M, N
  Third character is not used, leave blank
  Last character is orientation, must be Z,N or E
S  Z
 Enter date as YYYYMMDDHHMMSS, at least up to the day  (e.g. 19880123):20000101
 Latitude (Up to 4 decimal places and - for south), return for none:
 Longitude (Up to 4 decimal places and - for west), return for none:
 Enter elevation in meters (integer), return for none:
 Enter comments, one line. e.g. amp type, sensor type return for none

 Response file name is: TEST_S__Z.2000-01-01-0000_GSE


  RESPONSE CURVE IS IN FILE resp.out
```

Problem: The response file naming has not changed according to the SEED convention so location and network code cannot be entered using RESP, however they can be entered manually in response file (see response file format). This menas that e.g. SHZ must be entered as SH Z since only 3 of the 4 character are used. For older data in SEISAN format, all 4 characters can be used.

Examples of response files is given in Appendix C.

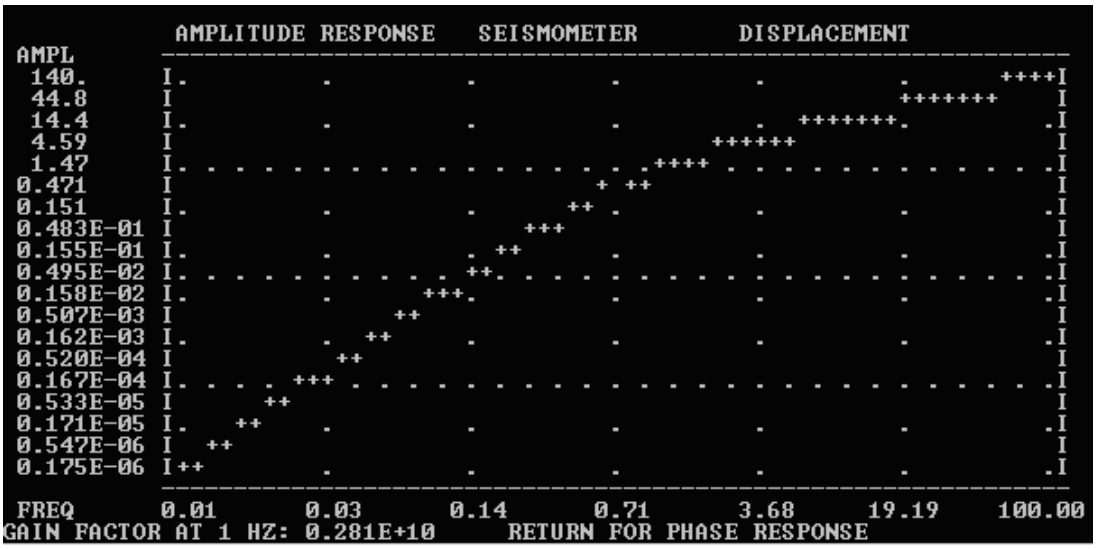


Figure 6.37: Making response file with parameters.

6.32.2 Examples of main response files from seismometers and accelerometer

Example of a Güralp DM 24 digitizer with CMG-40-1 (1 Hz)

Digitizer:

The sensitivity of the digitizer is given to $3.197 \mu V/count$. The SEISAN gain is in counts/V so

SEISAN recording media gain =1000000/3.197 = 312793 count/V

Sensor:

Sensitivity is 2 X 1001 V/m/s =2002 V/m/s

Making response file with parameters

For calculating with parameters, it is assumed that the free period is 1.0 s and damping is 0.7. Using the resp program answering as follows

Output format: 0	Only testing
Type of sensor: 1	It is a seismometer
Seismometer period: 1.0	
Seismometer damping: 0.7	
Generator constant: 2002	
Recording media gain: 312793	
Amplifier gain: 0	No amplifier
Number of filters: enter	No filter
File with poles and zeroes: enter	We use parameters now
File with tabulated values: enter	
File with measured values enter	

Then the plot below comes up

Making response file with poles and zeros

The poles and zeroes velocity response in units of Hz is given as

Poles
-0.707 0.707
-0.707 -0.707
-62.4 135.4
-62.4 -135.4
-350.0 0.0
-75.0 0.0

Zeros
0.0 0.0
0.0 0.0

SEISAN units are radians/sec so poles and zero values are multiplied by 2π.
The normalization constant is given as 585.810^6. To convert to radian is done as follows

Normalization constant in radian $= 585.810^6 2\pi^{(numberofpoles-numberofzeroes)} = 585.810^6(2\pi)^4 = 9.1210^{11}$.

SEISAN also uses displacement so one zero is added. The values are then

Poles
-4.442 4.442
-4.442 -4.442
-392.0 850.7
-392.0 -850.7
-2199.0 0.0
-475.0 0.0

Zeros
0.0 0.0
0.0 0.0
0.0 0.0

To get total constant (gain and normalization constant), we multiply by sensor gain and digitizer gain

Total normalization constant $= 9.1210^{11} x 2002 x 312793 = 5.7110^{20}$

A SEISAN input file is then made

6 3 5.71e20 6 poles, 3 zeros and total gain constant
-4.442 4.442
-4.442 -4.442
-392.0 850.7
-392.0 -850.7
-2199.0 0.0
-475.0 0.0
0.0 0.0
0.0 0.0
0.0 0.0

The resp program now makes the SEISAN response file with this input as follows

Output format: 0 Only testing
Type of sensor: 0 Sensor response is in poles and zero file
Recording media gain: 1 Gain has been put into total gain constant

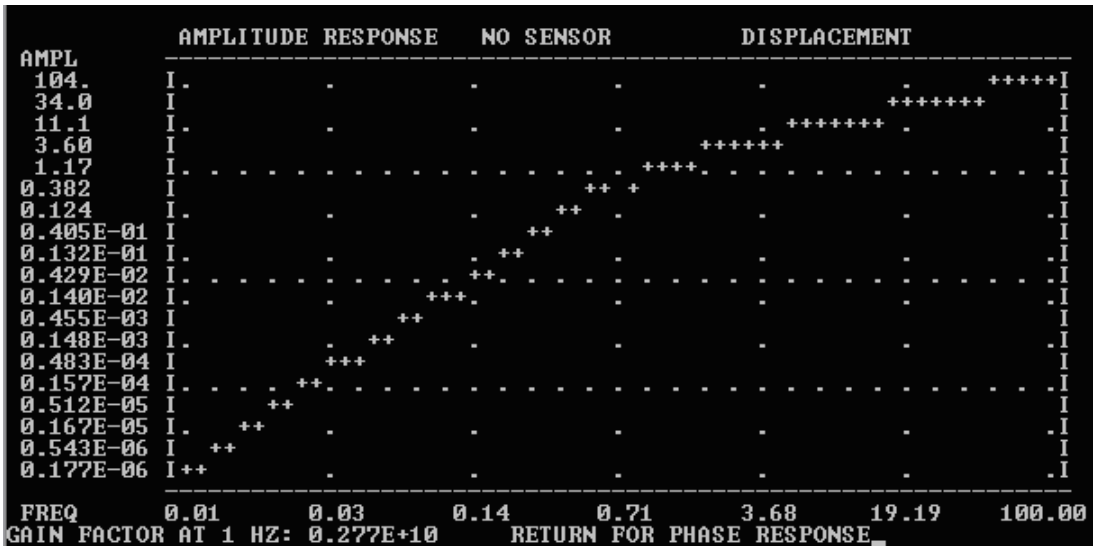


Figure 6.38: Making response file with poles and zeros.

Amplifier gain: 0 No amplifier
Number of filters: enter No filter
File with poles and zeroes: `resp.inp` File with poles and zeros, can be any name
File with tabulated values: enter
File with measured values enter

Then the plot below comes up

It is seen that the two ways of making the response file gives almost the same result, however using poles and zeroes is the most accurate, particularly for active sensors. In both cases no consideration was made for antialias filters which normally can be disregarded if a modern sharp filter.

Example of a Gurlp DM 24 digitizer with CMG-5T accelerometer

The digitizer is the same as before

Using parameter format, SEISAN currently requires the component name to start with A. According to international standards, the component code for an accelerometer should be something like ENZ so a parameter format cannot be used and poles and zeroes must be used. For the CMG-5T, the only information about the sensor is the sensitivity of 1V is equivalent to 0.970 m/s2 1.03 V/ms-1. In SEISAN parameter format this should be converted to V/g so sensitivity is then

9.81 (ms-2/g)/0.97(ms-2/V) = 10.1 V/g

Parameter format

The input is:

Output format: 0 Only testing
Type of sensor: 3 It is an accelerometer
Generator constant: 10.1
Recording media gain: 312793
Amplifier gain: 0 No amplifier

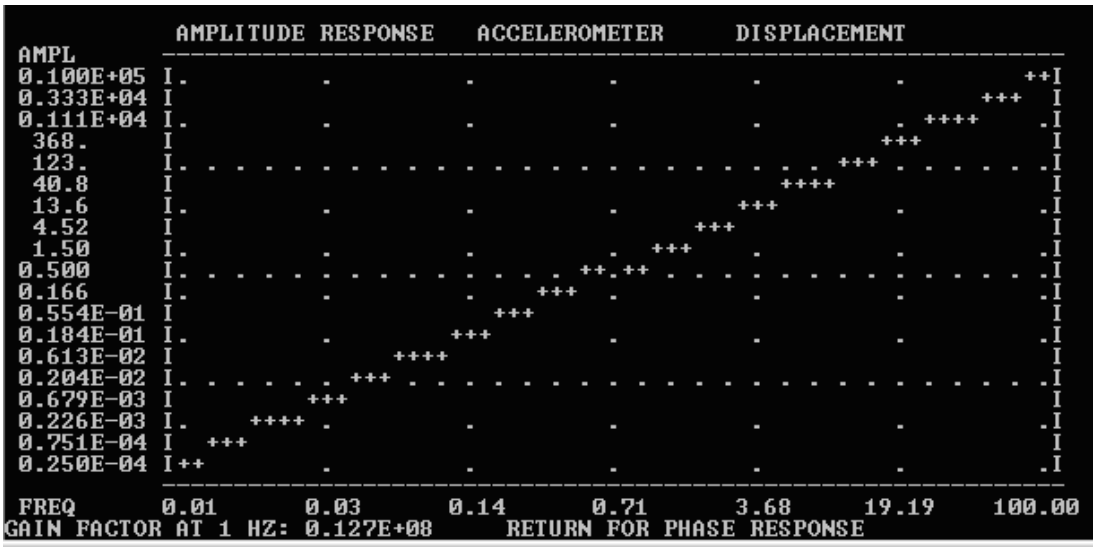


Figure 6.39: Making response file for an accelerometer woth parameters.

Number of filters: enter No filter
File with poles and zeroes: enter We use parameters now
File with tabulated values: enter
File with measured values enter

The plot below comes up

Poles and zeros

The displacement response for an accelerometer consists of 2 zeros and normalizarion constant of 1. The total gain constant is then

312793 x 1.03 = 322000

So the input file for resp is

0 2 322000
0 0
0 0

The manual input is exactly as above in the other example of using a poles and zero input file and the output is exactly as for the example of using parameter input.

Making a response file for a particular station

For a particular station, chose output format SEISAN PAZ or GSE2 PAZ and later answering yes to question of making the SEISAN response file (see SEISAN manual ??????????????). If e.g. the station has station code TEST and component name S Z, the a response file valid from January 1, 2007 will have the name `TEST_S__Z.2007-01-00-0000_SEI`. In case of a SEISAN poles and zero file, the content is:

```
TEST S  Z107   0  1  0  0  0  0.000                                          P

     6    3 0.5710E+21 -4.442      4.442     -4.442     -4.442     -392.0
  850.7     -392.0     -850.7     -2199.      0.000     -475.0      0.000
```

```
0.000       0.000       0.000       0.000       0.000       0.000
```

So the file could have been made without using resp.

Table 1 Example of `resp.out`:

```
SENSOR TYPE: SEISMOMETER       RESPONSE: DISPLACEMENT
SEISMOMETER PERIOD=    1.00000
GENERATOR CONSTANT=    300.000
DAMPING RATIO      =   0.700000
AMPLIFIER GAIN(DB)=    40.0000
RECORDING GAIN=        2048.00
FILTER CONSTANTS
F= 10.00   POLES=  2
GAIN AT 1 HZ=          2.75728E+08

F=  0.0050   T=  200.00    AMP=      0.000000    AMPDB=-135.1    PHAS=   -90.4
F=  0.0059   T=  169.49    AMP=      0.000000    AMPDB=-130.8    PHAS=   -90.5
F=  0.0070   T=  142.86    AMP=      0.000000    AMPDB=-126.4    PHAS=   -90.6
F=  0.0083   T=  120.48    AMP=      0.000001    AMPDB=-121.9    PHAS=   -90.7
F=  0.0098   T=  102.04    AMP=      0.000001    AMPDB=-117.6    PHAS=   -90.9
F=  0.0120   T=   83.33    AMP=      0.000002    AMPDB=-112.3    PHAS=   -91.1
F=  0.0140   T=   71.43    AMP=      0.000004    AMPDB=-108.3    PHAS=   -91.2
F=  0.3900   T=    2.56    AMP=      0.082352    AMPDB= -21.7    PHAS=  -125.9
F=  0.4600   T=    2.17    AMP=      0.133868    AMPDB= -17.5    PHAS=  -133.0
F=  0.5500   T=    1.82    AMP=      0.224204    AMPDB= -13.0    PHAS=  -142.3
F=  0.6500   T=    1.54    AMP=      0.356744    AMPDB=  -9.0    PHAS=  -152.9
F=  0.7700   T=    1.30    AMP=      0.554684    AMPDB=  -5.1    PHAS=  -165.6
F=  0.9100   T=    1.10    AMP=      0.820676    AMPDB=  -1.7    PHAS=  -179.7
F=  1.1000   T=    0.91    AMP=      1.198877    AMPDB=   1.6    PHAS=   163.3
F=  1.3000   T=    0.77    AMP=      1.580098    AMPDB=   4.0    PHAS=   148.6
F=  1.5000   T=    0.67    AMP=      1.933016    AMPDB=   5.7    PHAS=   137.0
F=  1.8000   T=    0.56    AMP=      2.420457    AMPDB=   7.7    PHAS=   123.6
F=  2.1000   T=    0.48    AMP=      2.877005    AMPDB=   9.2    PHAS=   113.5
F=  2.5000   T=    0.40    AMP=      3.460298    AMPDB=  10.8    PHAS=   103.0
F=  2.9000   T=    0.34    AMP=      4.027073    AMPDB=  12.1    PHAS=    94.6
F=  3.5000   T=    0.29    AMP=      4.855642    AMPDB=  13.7    PHAS=    84.1
```

FOR MORE DETAILS ON HOW TO UNDERSTAND GSE AND SEED RESPONSE PARAMETERS, SEE [Havskov and Alguacil, 2004], chapter 6.

6.32.3 SEED response

SEISAN can directly read SEED responses, which is poles and zeros, given as velocity response and transfer function types A (Laplace Transform in Rad/sec) and B (Analogue in 1/sec). Storage of response in one of these is the most common. The resp files can be created with rdseed from a full or

dataless SEED volume (`rdseed -R -f seed_volume`). RDSEED creates files with the pattern texttt-tRESP.NC.STAT.LC.CHC, where `NC`=network code, `STAT`=station code, `LC`=location code (not used by SEISAN) and `CHC`=channel code. The resp files need to be stored in the CAL directory and SEISAN will find the correct file. The resp file can contain response information from several time intervals. SEISAN uses the date and time of the waveform data to find the corresponding instrument response.

SEED response files are given in stages, for example seismometer, digitizer and FIR filters are stored as individual stages. The overall response is made by combining all the stages. SEISAN uses the following blockets from the SEED resp file (for more details see IRIS Consortium [1993]):

B052F22 - start date
B052F23 - end date

B053F03 - transfer function type, A=Laplace Transform (Rad/sec), B=Analog (1/sec)

B053F07 - A0 normalization factor (A0 is checked against poles and zeros at normalization frequency and changed if not correct). The product of poles and zeros at the normalization frequency and A0 gives 1.

B053F08 - Normalization frequency

B053F10-13 - zeros, if transfer function type is B, normalization factor A0 is changed to (A0)/(2 pi) for each zero

B053F15-18 - poles, if transfer function type is B, normalization factor A0 is changed to (A0)*(2 pi) for each pole

B058F04 - gain

The overall gain factor is given by the product of normalization factors and gain factors from all stages. One zero is added to convert to displacement response. It is assumed that input units are V/m and output units are counts, no checks are done on input and output units.

6.32.4 SEED response to GSE, SEEDRESP2GSE

SEEDRESP2GSE converts SEED resp files as written out by rdseed to GSE format. The program only supports poles and zeroes and transfer function type Laplace Transform. The program asks for station and component names and a time. This is because the resp file could have data from several channels and cover several time intervals with different instrument configuration.

6.32.5 GSE response to SEED, GSERESP2SEED

GSERESP2SEED can be used to build dataless SEED volumes from a set of GSE calibration files. The conversion is based on the GSE2SEED program by **Reinould Sleeman** (email sleeman@knmi.nl). Input can be single filenames or a list of files given in `filenr.lis`. The program produces a single channel SEED volume for each channel given by a GSE response file. At the end of the output filename, GSE is replaced by SEED. Other tools have to be used to merge several channels into one SEED volume. If there are several GSE files for a channel from different time periods, a stop date has to be given in the CAL2 line of the respective GSE file. Station coordinates are taken from the `STATION0.HYP` file. The program can use the site name, if it is part of the GSE response file through a comment line as in the following example:

```
(GSE2SEED_SITENAME Charnwood Forest, Leicestershire, England, UK)
```

6.33 Macroseismic data in SEISAN

Macrocosmic information in SEISAN are of 2 kinds. The summary information with maximum intensity, macroseismic epicenter etc has a special line in the S-file (see Appendix1) and the SELECT program can search for felt events. In addition, from SEISAN, version 8.1, a format has been defined to store macroseismic observations used to create e.g. maps with isoseismals. The observations files are stored in a local ISO format. For a format description and suggestion for file names, see Appendix A. There are currently no SEISAN programs that generates these files so they have to be made manually from the observations. The files in ISO are linked to the events as given by the S-file data base structure in the same way as waveform files are linked to the events. The line type is MACRO3 and an example is

```
 2007-01-21-1345-00.MACRO                                          MACRO3
```

Thus information about event source parameters and felt information is available together. An example of a file is

```
Brattvg, More og Romsdal     2007  121 1345  0 GMT 2007  121 1445  0 Local time
 Comment
    62.200     6.110  5.0 EMS 6150        RSTA
    62.200     6.110  4.0 EMS 6150        RSTA
    62.890     7.680  4.0 EMS 6480        BATNFJORDSRA
    62.500     6.680  6.0 EMS 6260        SKODJE
    62.460     6.130  6.0 EMS 6013        lesund
    62.200     6.110  6.0 EMS 6150        RSTA
    62.460     6.290  6.0 EMS 6017        SESTRANDA
    62.460     6.010  4.0 EMS 6055        GODY
    62.490     6.176  5.0 EMS 6057        ELLINGSY
    62.430     6.180  4.0 EMS 6030        LANGEVG
```

The file format is given in Appendix A. Program EPIMAP can plot the new files (use macroseismic file instead of a hypocenter file). The requirement is that the the first 3 letters after the '.' is mac or MAC (as example above). The intensities will be plotted as number on the map. A new Unix program can also be used with the data. Program MACROMAP can use the macroseismic observation file as input to create a map of the observations using GMT. The program generates a GMT script file, macromap.gmt, which then is executed from within the program to create a PostScript output file, macromap.ps. This file is then displayed, from within the program, with Unix command gv (GhostView). The program also runs under Windows, but does not plot.

The input can also be from a file made with macroquest (web based interactive program for input from the public, to be distributed with SEISAN CD). In addition to making the plot, a conversion from the web format to SEISAN format is made (output file macromap.out). This option requires an input file with postal codes in order to get location of the observations. MACROMAP can also be executed directly or from eev. When executed directly from the prompt line, the options are: -macroinput file with macroseismic observations, SEISAN format, abs path or in ISO -placename optional additional file with place names, to be shown on map, abs path or in DAT, epimap format is used -postfile optional file with postal code, abs path, used with web option If used with eev, the place name file must have name place_names.macro

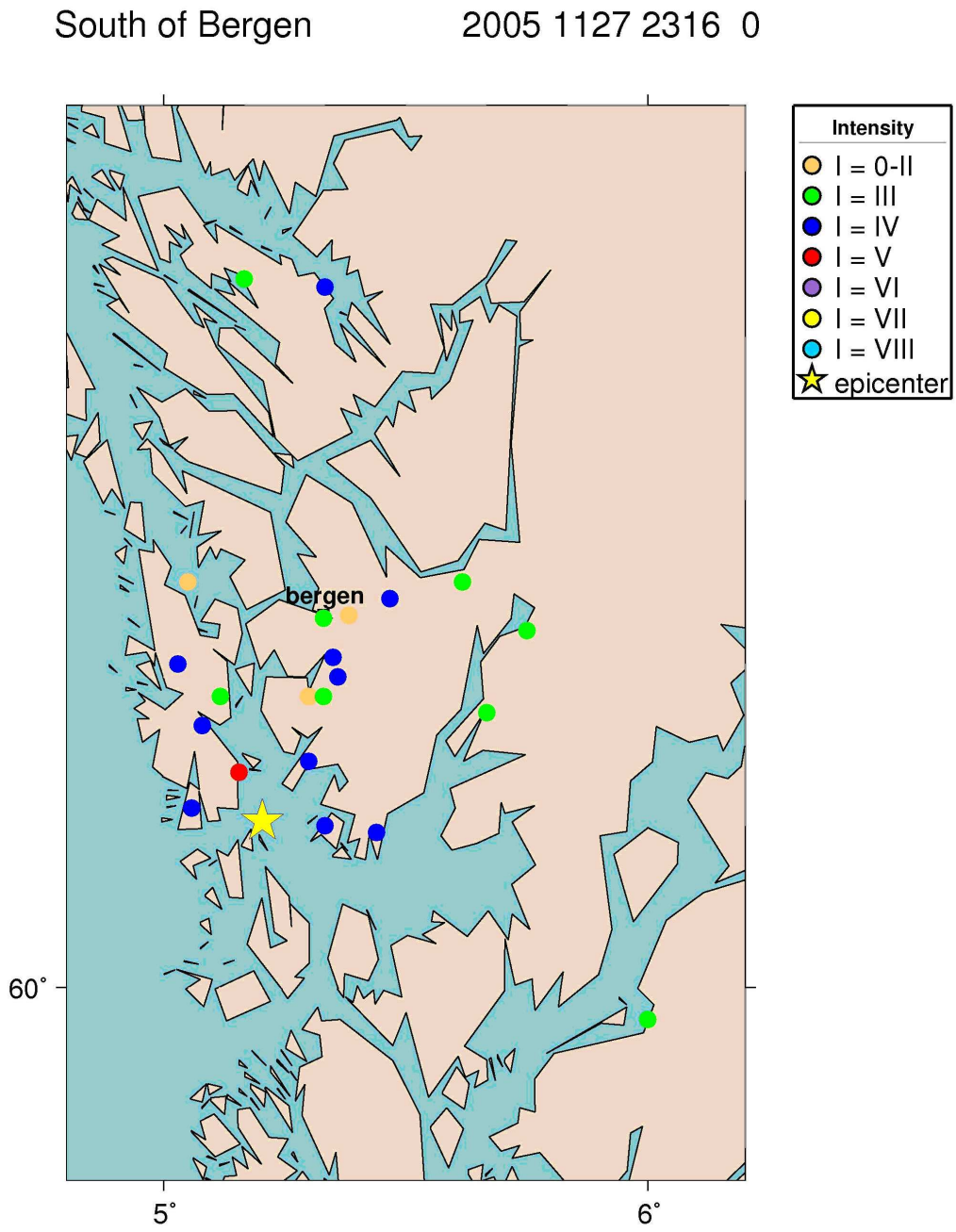


Figure 6.40: Macroseismic map made with MACROMAP using EEV. The epicenter, taken from the S-file, is shown with the star.

An example of the postal code file is

```
Brattvg, More og Romsdal     2007  121 1345  0 GMT 2007  121 1445  0 Local time
 Comment
    62.200     6.110  5.0 EMS 6150        RSTA
    62.200     6.110  4.0 EMS 6150        RSTA
    62.890     7.680  4.0 EMS 6480        BATNFJORDSRA
    62.500     6.680  6.0 EMS 6260        SKODJE
    62.460     6.130  6.0 EMS 6013        lesund
    62.200     6.110  6.0 EMS 6150        RSTA
    62.460     6.290  6.0 EMS 6017        SESTRANDA
    62.460     6.010  4.0 EMS 6055        GODY
    62.490     6.176  5.0 EMS 6057        ELLINGSY
    62.430     6.180  4.0 EMS 6030        LANGEVG
```

The content is postal code, latitude, longitude and location. The format is a10,ff10.3,2x,a30. the postal code does not have to be a number, but can be any string.

6.34 Correlation of waveform signals, CORR and detection of event clusters XCLUST

The cross-correlation function provides a measure of similarity between signals. In seismology cross-correlation can be used to measure the similarity of waveforms between seismic events and in case of similarity to determine relative arrival times of a seismic wave between two events. Waveform similarity is caused by proximity in hypocenter location and similarity in focal mechanism between two events. Cross correlation can be computed with the program CORR and the output be processed with program XCLUST to detect groups of similar events.

CORR

The program CORR computes correlation and can be used to measure relative arrival times. It also can be used to determine correlation of a master event with continuous data and extract event based data. The output of maximum correlation for a station recording different events can be used to identify event clusters within the data set, this can be done using the program XCLUST. The cross-correlation function of signals x and y is computed in the time domain as:

$$r_{xy}(i) = \frac{\sum_{j=1}^{n} x_j y_{(j+i-1)}}{\sqrt{\sum_{j=1}^{n} x_j^2}\sqrt{\sum_{j=1}^{n} y_{j+i-1}^2}}$$

Phase arrivals

Phase arrivals of similar events can be determined accurately through cross-correlation. P and S arrivals can be determined independently. The procedure starts by selecting and picking phases for a master event that is representative for the group of events. Analysis of this event needs to be done accurately as it is the basis for the subsequent analysis. Phases of the other events in the group can be determined through cross-correlation of either the complete trace or a selected phase window with the master event.

To pre-select a time window, manual identification of the phase for subsequent events is required prior to running CORR. This may be necessary for example if the waveform file contains several events. The phase arrival time is given by the maximum of the cross-correlation function that needs to exceed a minimum threshold. The arrival time is written out as absolute time. Filtering is applied to the signals if selected for both master and subsequent events, this may be necessary especially when dealing with events of different size. The filtering introduces a phase shift, which is applied to both signals. However, the absolute phase arrival for the subsequent event is consistent with the master event picked time.

The calculation of the relative phase time (dt) is done by taking the travel time for the master event (AT1 -OT1) (where AT is arrival time and OT origin time) minus the travel time of the other event (CAT -OT2), where CAT is the time corresponding to the maximum amplitude in the cross correlation function. The output file dt.cc can be used with the double difference location program HYPODD.

Cross correlation matrix

In this mode the cross-correlation is computed between the same stations for all pairs of events (that fulfill the criteria for maximum distance between the events, and event and station). The resulting cross-correlation matrix (for each station containing $\sum_i^n (i-1)$ values, where n is the number of events) can be used to identify groups of similar events using the program XCLUST.

Continuous mode

The main objective of running CORR in this mode is to identify a master waveform signal in a continuous data stream, given by waveform data files. The times when correlation is higher than the selected threshold level are written out, and can be visualized by splitting the corr.out file and using EEV and MULPLT. In addition, it is possible to cut out individual event files (see CONTINUOUS EXTRACT parameter).

Input file

Input to the program is given through the file corr.inp. A sample file is given in the DAT directory; the data used in the example are part of the test data set (TEST database 2003/06, see training document). The program is run by command corr in the same directory as corr.inp and the s-files. The waveform files can be in any SEISAN standard place. All standard waveform formats can be used.

The parameters in `corr.inp` are as follows:

Event file names:

SFILE MASTER:	sfile name of master event, remove or comment out this parameter to run program in group identification mode to determine cross-correlation matrix between all events and identify group of similar events
SFILE EVENT:	sfile name of events that will be either cross-correlated among themselves, or compared to the master event, there can be several of these.
SFILE INDEXFILE:	textttfilenr.lis file can be used to give S-file names instead of listing them with 'SFILE EVENT'

General parameters:

INTERACTIVE:	set to 1. for interactive use where graphics are displayed on the screen, which is useful for testing; or set to 0. for non-interactive run
CC MATRIX WINDOW:	time interval in seconds to use in computation of cross-correlation matrix instead of the duration given for each station given on STATION line, if different from 0.; full trace is used if EVENT SELCRIT is set to 0.
CONTINUOUS MODE:	write out all detection times for correlation above threshold if set to 1.; otherwise only phase for maximum correlation CONTINUOUS EXTRACT: extract event waveform files for correlation above threshold, 0. for no extract, 1. to extract single channel used, 2. extract all channels
EVENT SELCRIT:	cross-correlation with the master signal can be computed either for the complete trace of the subsequent event (0.) or the same part of the signal (either P or S) as for the master event (1.) including the pre signal part and of the same duration as defined in STATION line
FILTER:	this parameter allows to enable (1.) or disable (0.) filtering as defined for each channel with parameter line
STATION FIX DEPTH:	allows to fix depth to given value in corr.out, which can be useful if data is input to location program; set to 999. to disable
MAX EVENT DISTANCE:	maximum distance between event pair to compute correlation
MAX STAT DISTANCE:	maximum distance between event and station
MIN CORR:	minimum correlation required either for grouping or phase identification
MIN CORR CHAN:	number of stations required for event pair to be correlated.
N DOUBLE SRATE:	if sampling rate is to be increased give factor n to double sampling rate n times, 0. for none, this makes it possible to get phase reading with resolution greater than sampling interval
PRE SIGNAL:	duration of signal to include before phase arrival if used
SINGLE BIT:	the data can be reduced to 1 bit; reduce to 0 (-) and 1 (+) if set to 1, full data if set to 0.
START LATITUDE and START LONGITUDE:	these can be set to write values to corr.out, which is in Nordic format and can be used as input for location programs; this can be used if all events analyzed belong to one cluster and the same starting location is to be used for all of them; set to 999. to disable
TRACE OUTPUT:	flag to write corr.trace output file (1. for true)
WAVE CORR OUT:	CORR can write out cross-correlation function and input traces to waveform output files of the selected duration, to disable set this parameter to 0., 1. for full data or 2. for reduced data, where 1 for data ¿ MIN CORRELATION, otherwise 0
WAVENAME OUT:	CORR writes out waveform filenames to corr.out, it is possible to either keep original waveform names (0.) or put corr output file names (1.), which after SPLIT of corr.,out allows inspection of the results using eev and mulplt

Station parameters:

STATION:	one line for configuration of each channel	
	STAT, COMP:	station and component codes
	SELCRIT:	1=P, 2=S, 4=full trace
	DURATION:	signal duration in seconds if (selcrit¡4) starting from either P or S
	FLOW, FHIGH:	filter limits for bandpass filter, can be; can be disabled by FILTER (see above)

Example of STATION line:

```
KEYWORD...STAT......COMP......SELCRIT...DURATION..FLOW......FHIGH.....
--- p ---
STATION   PCA       S  Z      1.        6.        3.        8.
--- s ---
STATION   EDI       S  E      2.        5.        3.        8.
```

Output files:

`corr.out`: This is the main output file. The file is in Nordic format and contains the phase readings if run in phase detection mode and can be used with the SEISAN location programs directly. In continuous mode, the file can contain more than one phase reading per channel. In group identification mode the file contains the event list, cross-correlation matrix and suggested groups of similar events.
`corr.trace`: This files gives details of program run and can provide information on cause of errors. dt.cc: Input file for hypodd giving relative phase times and correlation (see hypodd manual for details), e.g.

```
#    1    2 0.0
GMK   -0.136 0.940 P
GMK   -0.136 0.977 S
PCA   -0.142 0.963 P
PCA   -0.142 0.967 S
PCO   -0.152 0.968 P
PCO   -0.152 0.952 S
```

`cc_pairs.out`: List of event pairs giving, index and s-file of first event, index and s-file of second event, number of stations and average correlation of all stations. This file is used as input to XCLUST.

```
   1 05-0028-32L.S200405    2 05-0413-55L.S200405   4 0.934
   1 05-0028-32L.S200405    3 05-1644-21L.S200405   2 0.838
   1 05-0028-32L.S200405    4 05-2102-06L.S200405   1 0.844
   1 05-0028-32L.S200405    5 05-2143-52L.S200405   3 0.905
   1 05-0028-32L.S200405    6 06-2136-12L.S200405   3 0.880
   1 05-0028-32L.S200405    7 06-2301-54L.S200405   4 0.901
```

XCLUST

XCLUST is a simple program for cluster analysis of output from program CORR (`cc_pairs.out`) to identify groups of similar events. This is done in a rather simple approach:

- sort event pairs with descending correlation
- find group
  - o start with highest correlation
  - o add events that are linked into group this group in several loops over all pairs until no more events can be added to group; link into group is given by one of the events in the pair being correlated with any of the events in the cluster
- continue to find next group

Visual inspection of the waveforms is highly recommended to confirm the clustering results.

Input file: `xclust.par`

This is the file for the main parameters, which are:
MINIMUM CORRELATION: minimum correlation required for pair to be used
MINIMUM STATIONS: minimum number of correlated stations required for pair to be used
MINIMUM PERGROUP: minimum number of events required to make a group
TRACE OUTPUT; flag to write trace output file (1. for true)

Output files:

`xclust.trace`: gives some details of what the program does, useful for debugging
`xclust.out`: gives list of events for each cluster and for each event the number of links with other events in that cluster

```
=============
 group:     1 number of events:      20
-------------
  event links
-------------
      5     7
     11     8
      6     5
      2     7
      1     6
      7     9
...
```

`Index.xxx`: Index file where xxx refers to number of cluster. This file can be used with eev (e.g. `eev index.001`) to work on a specific cluster.

Chapter 7

Programming in SEISAN and list of files in SEISAN distribution

This chapter gives a bit more technical details of SEISAN starting with a short programmers guide with description of sample and test programs.

7.1 Programmers guide and some test programs

SEISAN is conglomerate of programs and subroutines and it can be difficult to find out which routines to use and how to start a new SEISAN program. The most common method is to use an existing program and modify it. The intention with this section is to make it easier by providing a few sample programs which then can be modified to do specific tasks. The compilation of existing SEISAN programs has been described in section 3.8 and details of the commands are found in the Makefiles. In this distribution, 3 sample programs have been included, which each illustrate the used of some SEISAN features. All 3 programs are included in the Makefiles and can therefore be compiled directly, modified and recompiled.

Reading and writing S-files

A basic operation is to be able to read and write S-files, since all parameters are contained in the S-files. Starting with version 7.2, a new library (`rea.for`) and include block (`rea.inc` for definition of variables) has been included to make it easier to read and write data into S-files. Earlier, S-files were only read and written as text strings and individual parameters were then read/written to the text strings. Now the new routines do it all. These routines are now used in a few programs, but will be included whenever a program is substantially modified. The sample program is called `sample_read_write_s.for`. The program illustrates how to read all parameters in an S-file, make modifications and write out the file again. The program can be useful, if the user needs a program where special parameters are needed for a particular analysis or for output in another format.

Reading and writing waveform files

In SEISAN, waveform files can be in SEISAN, SAC (Unix only) or GSE format. SEISAN format is slightly different depending on which compute platform it is written and byte swapping has to be done in some cases. In order to automatically handle the reading of waveform files, irrespective of format and computer platform, a set of standard routines are used (waveform.for) and an include block where all parameters

and data end up (`waveform.inc`). The sample reading program is called `sample_read_wav.for`.

The program illustrates how to read many waveform files belonging to one event as if it was one file, irrespective of format. It also demonstrates how to read just one waveform file. There is no detail on how to write a SEISAN binary file in this program, but some info is given under the format description in Appendix B and the program `tsig.for` described below illustrates a simple write. The sample program sample_read_cont illustrates how to extract out a time segment of the continuous data base. The program also shows how to write a Seisan file with all headers. The program is started from the command prompt:

sample_read_cont start_time interval

where start time is yyyymmddhhmmss and interval is interval in minutes.

There is a similar program for reading data from archives, sample_read_arc. There is also a routine in Java available to read all SEISAN binary formats. The program is called SFORMAT (written by T. Utheim). Similarly there is a sample program to read all SEISAN binary formats in Perl (written by Angel Rodriguez). The program is called `seibinasc.pl` and you need a Perl interpreter to run it. Before starting the program, a DIRF must first be made of waveform input files. The output is identical to a SEISAN ASCII file as made by SEIASC.

The sample write program is called `sample_write_wav.for`. It is a simple example of writing a straight line. The output formay is SEISAN.

Graphics in SEISAN

SEISAN uses a set of graphics routines, which are identical in call on all 3 platforms . These routines then call low level routines which are platform dependent (X on Unix and Windows calls on PC). The programmer only has to use the high level routines. The routines also generate a PostScript output if a given parameter is set. The program is called `sample_grapichs.for`. The program illustrates how to initiate graphics, make a few simple calls, get up and use the mouse and make a hard copy file. Most of the general graphics routines are located in file `seiplot.for` and common variables in `seiplot.inc`. The program can be useful for testing functionality of the mouse.

Program LSQ is a simple example of how to make xy-graphics. It also shows how to make the output files for gmtxy. In order to find more info (apart from manual in INF) on gmtxy, see file `gmt.for` in LIB and `gmt_xy_par` in INF.

Program to make test signals, TSIG:

It is often useful to be able to work with controlled waveform data so a program making test signals is included. The program makes several traces, all with same length and sample rate and trace 1 is the sum of all traces. For each trace selected, the parameters selected are: Frequency, amplitude (remember this is integer numbers in file so use at least 1000), phase, delay (delay time when the signal appears on trace relative to start of trace, the data before is zero) and damping. The damping is used to simulate seismometer damping or simple a damped signal and has a similar physical meaning as the seismometer damping constant, but period is not recalculated to simulate changing period with damping. Zero damping is no damping.
An additional trace can be made with a Brune displacement pulse generated with parameters corner frequency (f0), Q and kappa (see MULPLT) and travel time. Travel time is used for Q-correction and also places the pulse at travel time distance from the origin (start of trace), so length of trace must be longer than travel time. If zero q and kappa, no attenuation is used. The program also write an S-file with relevant parameters. The program illustrates a simple writing of a SEISAN waveform file.

Java programs in SEISAN

The Java programs are each given as a Jar-file like `jseisan.jar`, the jar-files are located in the PRO

directory. The jar-file contains all the Java source code, the Java classes and the project file so a user can decompress the jar file, change the script and make a new version of the program. The programs are started using a script file in the COM directory and no classpath has to be set, when SEISAN has been correctly installed.

7.2 CONTENTS OF PRO, LIB, INC, INF, COM, DAT, SUP, ISO and PIC DIRECTORIES

The PRO, LIB, INC and COM directories contain software, the DAT directory parameter files for operating the SEISAN system and INF contains documentation and manuals. All files are listed and explained in the file `seisan.all` in the INF directory. The ISO directory contains macroseismic information.

The program CHECK can check if a distribution is complete. Run CHECK and use option 'basic'. The content of the distribution is compared to the `seisan.all` file in the INF directory.

The `seisan.all` file also list programs no longer compiled, but with source code included in case there should be a future need for these programs

Chapter 8

Acknowledgments

A large number of people have contributed to the SEISAN system. From the British Geological Survey, Jim Bolton has spent several months cleaning up the software, putting in error checking and put in the new version of EPIMAP. Jane Exton has also been involved in several of the SEISAN database problems. A main contributor has been Barry Lienert who has spent several months at our institute to modify and adopt the HYP program, he has also written the complete HYP manual. Bent Ruud has contributed with the core of the AUTOPIC software and has helped out in practical aspects of installing it. Kuvvet Atakan has written the seismic hazard part. Ronald Arvidson has tested large parts of the system and done work on several programs and contributed with the modified version of FOCMEC. The help of Arne Sjursen has been essential for implementing X. Vunganai Midzi did the complete testing of version 6.0. Mario Villagran with programs, many suggestions and bug reports has interacted with the development of SEISAN. Ezra Twesigomwe, Berit Storheim, K. Atakan and Alice Walker have debugged the manuscript. Version 7: Bladimir Moreno has made the Windows graphics, made SEISAN run under Linux, written several other programs and has thus enormously contributed to this version. The hazard part has been updated by Kuvvet Atakan and Anibal Ojeda. Andrius Pacesa has implemented the FK routine. The programs have been tested and the manual been checked by Margaret Grandison, Waldo Taylor, Vunganai Midzi, Berit Storheim, Anne Lise Kjærgaard, Anibal Ojeda, Ileana Boschini and Cecilie Langeland. Version 7.1: This version was tested by Anne Lise Kjærgaard, Margaret Grandison and Vunganai Midzi. Version 7.2: Several contributions, including changes to MULPLT and implementing HYPO71, were made by Brian Baptie from the BGS. W_EMAP is a new program that was provided by Fernando Carrilho. The CPLOT program was written by Susanne Lund Jensen from KMS. Susanne has also checked this version of the manual. Version 8.0: The graphics part on Unix has been improved thanks to Freya Cromarty and Frederik Tillmann. The on-site-inspection group at the CTBTO has financially supported the development of SEISCONF and JSEISAN, as well as the modification of other programs. JSEISAN and SEISCONF were written by Bladimir Moreno. A new version of W_EMAP was provided by Fernando Carrilho. Mario Ordaz contributed the DEGTRA A4 software. Angel Rodriguez contributed the Perl reading routine and Terje Utheim the sfile Java program. Mathilde Bøttger Sørensen has revised the manual and tested the distribution. Dieter Stoll has provided information on how to compile on MacOSX and tested the software on Mac. RefTek has provided the rt_seis program. Version 8.1: Rodrigo Luciano Pereira Canabrava made a major contribution to SEISAN by writing and implementing routines to read and write SEED data. Richard Luckett implemented the ISC location program. Mathilde Bøttger Sørensen wrote scripts for the new macroseismic part. Brian Baptie contributed tools to convert event data for use with travel time conversion programs. Version 8.3: Wayne Crawford has contributed to a number of programs and made valuable suggestions. Version

9.0: Ruben Soares Luís has made the interface routines for reading archives and contributed several Java programs. Finally we will thank all the patient users who have suffered from the "bugs" and have given useful feedback.

Bibliography

Anderson, D. (1982). Robust earthquake location using M-estimates. *Phys. Eartg Plan. Int.*, 30:119–130.

Banfill, R. (1996). PC-SUDS Utilities. A collection of tools for routine processing of seismic data stored in the seismic unified data system for DOS (PC-SUDS), Version 2.5, Small Systems Support. Technical report.

Boore, D. M. (1983). Stochastic simulation of high-frequency ground motion based on seismo logical models of the radiated spectra. *Bull. Seismol. Soc. Am.*, 73:1865–1894.

Boore, D. M. (1989). Quantitative ground motion estimates. In Jacob, K. H. and C.J.Turkstra, editors, *Earthquake Hazards and the Design of Constructed Facilities in the Eastern United States*, volume 558, pages 81–94. Annals of the New York Academy of Sciences.

Bouchon, M. (1981). A simple method for calculating Green's functions for elastic layered media. *Bull. Seismol. Soc. Am.*, 71:959–972.

Brune, J. N. (1970). Tectonic stress and the spectra of seismic shear waves from earthquakes. *J. Geophys. Res.*, 75:4997–5009.

Chapman, C. H. (1978). A new method for computing synthetic seismograms. *Geophys. J.R. astr. Soc.*, pages 481–518.

Chapman, C. H., Jen-Yi, C., and Lyness, D. G. (1988). The WKBJ seismogram algorithm. In Doornbos, D. J., editor, *Seismological algorithms*, pages 47–74. Academic Press, London.

Chapman, C. H. and Orcutt, J. A. (1985). The computation of body wave synthetic seismograms in laterally homogeneous media. *Reviews of Geophysics*, 23:105–163.

Chiu, J., Isacs, B. L., and Cardwell, R. K. (1986). Studies of crustal converted waves using short-period seismograms recorded in the Vanatu Island arc. *Bull. Seismol. Soc. Am.*, 76:177–190.

Dey-Sarkar, S. K. and Chapman, C. H. (1978). A simple method for computation of body-wave seismograms. *Bull. Seismol. Soc. Am.*, 68:1577–1593.

Draper, N. R. and Smith, H. (1966). *Applied regression analysis*. John Wiley and Sons, New York.

Ebel, J. E. and Bonjer, K. P. (1990). Moment tensor inveriosn of small earthquakes in southwestern Germany for fault plane solution. *Geophys. J. Int.*, 101:133–146.

Goldstein, P. (1999). SAC user's manual. Technical report, Lawrence Livermore Laboratory, University of California.

GSETT-3 (1997). Provisional GSE 2.1, Message Formats & Protocols, Operations Annex 3. Technical report.

Hardebeck, J. L. and Shearer, P. M. (2002). A new method for determining firstmotion focal mechanisms. *Bull. Seismol. Soc. Am.*, 92:2264–2276.

Hardebeck, J. L. and Shearer, P. M. (2003). Using S/P Amplitude Ratios to Constrain the Focal Mechanisms of Small Earthquakes. *Bull. Seismol. Soc. Am.*, 93:2434–2444.

Havskov, J. and Alguacil, G. (2004). *Instrumentation in earthquake seismology*. Springer.

Havskov, J., Malone, S., McCloug, D., and Crosson, R. (1989). Coda Q for the state of Washington. *Bull. Seismol. Soc. Am.*, 79:1024–1038.

Havskov, J. and Ottemöller, L. (2010). *Processing earthquake data*. Book in preparation, preliminary version at SEISAN web site fall 2008.

Herrmann, R. B. (1985). An extension of random vibration theory estimates of strong ground motion to large distances. *Bull. Seismol. Soc. Am.*, 75:1447–1453.

Herrmann, R. B. and Kijko, A. (1983). Modelling some empirical vertical component Lg relations. *Bull. Seismol. Soc. Am.*, 73:157–171.

Hutton, L. K. and Boore, D. (1987). The Ml scale in Southern California. *Bull. Seismol. Soc. Am.*, 77:2074–2094.

IRIS Consortium (1993). Standard for the Exchange of Earthquake Data - Reference Manual, 2nd Edition. Technical report.

Kanamori, H. (1977). The energy release in great earthquakes. *J. Geophys. Res.*, 82:2981–2987.

Kissling, E. (1995). Program VELEST USER'S GUIDE - Short introduction. Technical report, Institute of Geophysics, ETH Zurich, Zurich, Switzerland.

Kissling, E., Ellsworth, W. L., Eberhart-Phillips, D., and Kradolfer, U. (1994). Initial reference model in local earthquake tomography. *J. Geophys. Res.*, 99:19635–19646.

Klein, F. W. (1984). Users guide to HYPOINVERSE, a program for Vax and PC350 computers to solve for earthquake locations. open file report 84-000, USGS.

Kvaerna, T. and Doornbos, D. (1986). An integrated approach to slowness analysis with arrays and tree component stations. Norsar semiannual technical summary, 1 october 1985 - 31 march 1986, NORSAR, Kjeller, Norway. Scientific Report No. 2-85/86.

Lee, W. H. K., Bennett, R. E., and Meagher, L. (1972). A method for estimating magnitude of local earthquakes from signal duration. Open file report, USGS.

Lienert, B. R. E. (1991). Report on modifications made to Hypocenter. Technical report, Institute of Solid Earth Physics, University of Bergen, Bergen, Norway.

Lienert, B. R. E. (1994). HYPOCENTER 3.2: A computer program for locating earthquakes locally, regionally and globally. Technical report, Hawaii Institute of Geophysics & Planetology.

Lienert, B. R. E., Berg, E., and Frazer, L. N. (1986). Hypocenter: An earthquake location method using centered, scaled, and adaptively least squares. *Bull. Seismol. Soc. Am.*, 76:771–783.

Lienert, B. R. E. and Havskov, J. (1995). A computer program for locating earthquakes both locally and globally. *Seis. Res. Lett.*, 66:26–36.

McGuire, R. K. (1976). EQRISK. Evaluation of earthquake risk to site. Open File Report 76-67, United States Department of the Interior, Geological Survey. 90pp.

McNamara, D. E. and Buland, R. P. (2004). Ambient noise levels in the continental united states. *Bull. Seismol. Soc. Am.*, 94:1517–1527.

Menke, W., Holmes, R. C., and Xie, J. (2006). On the nonuniqueness of the coupled origin time-velocity tomography problem. *Bull. Seismol. Soc. Am.*, 96:1131–1139.

Michael, A. J. (1984). Determination of stress from slip data: faults and folds. *J. Geophys. Res.*, 89:11,517–11,526.

Nakamura, Y. (1989). A method for dynamic characteristics estimation of subsurface using microtremor on the ground surface. *QR of RTRI*, 30:25–33.

Ordaz, M. (1991). CRISIS. Brief description of program CRISIS. Internal report, Institute of Solid Earth Physics, University of Bergen, Norway. 16 pp.

Ordaz, M. (1999). User's manual for program CRISIS-99. Technical report, Universidad Nacional Autonoma de Mexico, Mexico City.

Ottemöller, L. (1995). Explosion filtering for Scandinavia. Norwegian national seismic network technical report, IFJF, University of Bergen, Norway. 209 pp.

Ottemöller, L. and Havskov, J. (1999). SeisNet: A General Purpose Virtual Seismi Network. *Seis. Res. Lett.*, 70:522–528.

Ottemöller, L. and Havskov, J. (2003). Moment magnitude determination for local and regional earthquakes based on source spectra. *Bull. Seismol. Soc. Am.*, 93:203–214.

Ottemöller, L., Shapiro, N. M., Singh, S. K., and Pacheco, J. F. (2002). Lateral variaion of Lg wave propagation in southern Mexico. *J. Geophys. Res.*, 107.

Peterson, J. (1993). Observation and modeling of seismic background noise. Open-File report 93-322, USGS. 95pp.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (2003). *Numerical Recipees in Fortrran 77, 2nd edition*. Cambridge University Press, Cambridge, U.S.A.

Pujol, J. (2003). Software for joint hypocentral determination. In Lee, W. H. K., Kanamori, H., Jennings, P. C., and Kisslinger, C., editors, *International Handbook of Earthquake and Engineering Seismology*, volume 81B. IASPEI, Academic Press.

Reasenberg, P. and Oppenheimer, D. (1985). Fpfit, fpplot, and fppage: Fortran computer programs for calculating and displaying earthquake fault plane solutions. Technical report, U.S. Geol. Survey.

Roberts, R. G., Christoffersson, A., and Cassidy, F. (1989). Real time events detection, phase identification and source location estimation using single station component seismic data and a small PC. *Geophys. J. Int.*, 97:471–480.

Ruud, B. O. and Husebye, E. S. (1992). A new three-component detector and automatic single station bulletin production. *Bull. Seismol. Soc. Am.*, 82:221–237.

Ruud, B. O., Husebye, E. S., Ingate, S. F., and Christoffersen, A. (1988). Event location at any distance using seismic data from a single, three-component station. *Bull. Seismol. Soc. Am.*, 78:308–325.

Scherbaum, F. (1996). *Of Poles and Zeros - Fundamentals of Digital Seismology*. Kluwer Academic Publishers.

Singh, S. K., Apsel, R. J., Fried, J., and Brune, J. N. (1982). Spectral attenuation of SH-waves along the Imperial fault. *Bull. Seismol. Soc. Am.*, 72:2003–2016.

Snoke, J. A., Munsey, J. W., Teague, A. G., and Bollinger, G. A. (1984). A program for focal mechanism determination by combined use of polarity and SV-P amplitude ratio data. *Earthquake notes*, 55.

Suetsugu, D. (1998). Practice on source mechanism, iisee lecture note. Technical report, Tsukuba, Japan.

Veith, K. F. and Clawson, G. E. (1972). Magnitude from short period P-wave data. *Bull. Seismol. Soc. Am.*, 62:435–452.

Waldhauser, F. (2001). hypoDD – A program to compute double-difference hypocenter locat ions. Technical report, U.S. Geol. Survey, Menlo Park, CA.

Waldhauser, F. and Ellsworth, W. L. (2002). Fault structure and mechanics of the Northern Hayward Fault, California, from double-difference earthquake locations. *J. Geophys. Res.*, 107. doi 10.1029/2000JB000084.

Appendix A

The Nordic format

Free columns are included for two purposes:

1. To obtain a readable format
2. To have some space for possible future extensions

Here are examples, top 3 lines for positioning only.

```
         1         2         3         4         5         6         7
12345678901234567890123456789012345678901234567890123456789012345678901234567890
.         .         .         .         .         .         .         .
--------------------------------------------------------------------------------

 1984 1022 2102 23.2 LE 69.330 27.440 11.0F NAO 34 5.2 3.8LNAO 4.0bPDE 3.2sISC1
 NORTHERN FINLAND                                                              3
 NRSA SZ IPN 1 D 2244 13.44 0345 1234.6 1.33 245.2 08.6 841022 120.2 3 5 12345
 NRSA SZ ILG 1 D 2244 13.44 0345 1234.6 1.33 265.0 03.6 841022 120.2 3 5 12345

 1985 510 21 5 16.1 LE 60.240   6.170 30.0F BER  6 2.3 3.8LNAO 4.0bPDE 3.2sISC
                1.5     0.5     0.9    5.0             0.4                    5
 8505210425.WNN                                                                6
 NORTHERN HORDALAND       F 3.5 61.22 0.5   5.33 0.8  23456 2   456 2 99 11BER1
 STAT SP IPHASW D HRMM SECON CODA AMPLIT PERI AZIMU VELO SNR                    7
 BER  SZ IPG  2 U 2105 25.41  200
 HYA  SZ ISG  1   2105 33.1
 ODD  SZ IP   3   2105 20.1   250
 ODD  SZ EPG      2105 22.9
 ODD  SZ  LG      2105 55.8

--------------------------------------------------------------------------------
```

Below are examples of how the last free columns of type 4 lines are used in the Nordic Databank in Helsinki and in Bergen:

```
 1985 510 21 5 16.1 LE 60.240 6.170 30.0F BER  6 2.3 3.8LNAO 4.0bPDE 3.2sISC  1
                1.5     0.5   0.9    5.0             0.4                      5
 8505210425.WNN                                                                6
 ACTION:UPD 93-07-09 09:40 OP:jens STATUS:               ID:19920101080359    I
 STAT SP IPHASW D HRMM SECON CODA AMPLIT PERI AZIMU VELO SNR AR TRES W DIS CAZ7
 NRSA SZ IPN  1 D 2105 13.44 0345 1234.6 1.33 245.2 08.6 5.5  2 -0.7 9 555 235
 BER  SZ IPG  2 U 2105 25.41  200
 HYA  SZ ISG  1   2105 33.1
 ODD  SZ IP   3   2105 20.1   250
 ODD  SZ EPG      2105 22.9
 ODD  SZ  LG      2105 55.8
```

Note in this example the fault plane solution line(F) and the HYP error line(E)

```
 1993 1028 0800 26.4 L 57.518 7.119 18.8  BER  6  .6 2.6CBER                  1
 GAP=201        1.20    6.4   7.0    6.8   .3359E+01  -.2719E+00      .3054E+02E
 93.2 74.8 -48.2 2 F
 ACTION:SPL 95-01-08 09:40 OP:jh   STATUS:               ID:19931028080019     I
 9310-28-0800-19S.NSN\_17                                                      6
 STAT SP IPHASW D HRMM SECON CODA AMPLIT PERI AZIMU VELO SNR AR TRES W DIS CAZ7
 BLS5 SZ EP     D 8  0 56.80  129                                 -.110 216 349
 BLS5 SZ ESG      8  1 23.59                                      -.910 216 349
 BLS5 SZ EP       8  0 56.80  129                                 -.110 216 349
 BLS5 SZ ESG      8  1 23.59                                      -.910 216 349

Location parameters:
AR  : Azimuth residual when using azimuth information in locations
TRES: Travel time residual

W   : Actual weight used for location ( inc. e.g. distance weight), i2
DIS : Epicentral distance in km
CAZ : Azimuth from event to station

--------------------------------------------------------------------------------
Note: Type 1 line must be the first, all type 4 lines should be together and
the last line must be blank
--------------------------------------------------------------------------------
```

FORMAT DESCRIPTION:

```
Type 1 Line:

Columns Format Description                  Comments
  1            Free
  2- 5   I4    Year
  6            Free
  7- 8   I2    Month
  9-10   I2    Day of Month
 11            Fix o. time                  Normally blank, an F fixes origin time
```

```
12-13   I2    Hour
14-15   I2    Minutes
16            Free
17-20   F4.1  Seconds
21            Location model indicator       Any character
22      A1    Distance Indicator             L = Local, R = Regional, etc.
23      A1    Event ID                       E = Explosion, etc.
                                             P = Probable explosion
                                             V = Volcanic
                                             Q = Probable volcanic
24-30   F7.3  Latitude                       Degrees (+ N)
31-38   F8.3  Longitude                      Degrees (+ E)
39-43   F5.1  Depth                          Km
44      A1    Depth Indicator                F = Fixed, S = Starting value
45      A1    Locating indicator             ---------------------------, * do not locate
46-48   A3    Hypocenter Reporting Agency
49-51         Number of Stations Used
52-55         RMS of Time Residuals
56-59  F4.1   Magnitude No. 1
60 A1         Type of Magnitude L=ML, b=mb, B=mB, s=Ms, S=MS, W=MW,
                                 G=MbLg (not used by SEISAN), C=Mc
61-63  A3     Magnitude Reporting Agency
64-67  F4.1   Magnitude No. 2
68 A1         Type of Magnitude
69-71  A3     Magnitude Reporting Agency
72-75  F4.1   Magnitude No. 3
76 A1         Type of Magnitude
77-79  A3     Magnitude Reporting Agency
80 A1         Type of this line ("1"), can be blank if first
              line of event
```

If more than 3 magnitudes need to be associated with the hypocenter in the first line, a subsequent additional type one line can be written with the same year, month, day until event ID and hypocenter agency. The magnitudes on this line will then be associated with the main header line and there is then room for 6 magnitudes.

Type 2 line (Macroseismic information)

```
 1-5          Blank
 6-20         a Any descriptive text
21            Free
22    a1      Diastrophism code (PDE type)
                     F = Surface faulting
                     U = Uplift or subsidence
                     D = Faulting and Uplift/Subsidence
23    a1      Tsunami code (PDE type)
                     T = Tsunami generated
                     Q = Possible tsunami
24    a1      Seiche code (PDE type)
```

```
                    S = Seiche
                    Q = Possible seiche
 25     a1      Cultural effects (PDE type)
                    C = Casualties reported
                    D = Damage reported
                    F = Earthquake was felt
                    H = Earthquake was heard
 26     a1      Unusual events (PDE type)
                    L = Liquefaction
                    G = Geysir/fumerol
                    S = Landslides/Avalanches
                    B = Sand blows
                    C = Cracking in the ground (not normal faulting).
                    V = Visual phenomena
                    O = Olfactory phenomena
                    M = More than one of the above observed.
 27             Free
 28-29 i2       Max Intensity
 30     a1      Max Intensity qualifier
                    (+ or - indicating more precicely the intensity)
 31-32 a2       Intensity scale (ISC type defintions)
                    MM = Modified Mercalli
                    RF = Rossi Forel
                    CS = Mercalli - Cancani - Seberg
                    SK = Medevev - Sponheur - Karnik33 Free
 34-39 f6.2     Macroseismic latitude (Decimal)
 40             Free
 41-47 f7.2     Macroseismic longitude (Decimal)
 48             Free
 49-51 f3.1     Macroseismic magnitude
 52     a1      Type of magnitudeI = Magnitude based on maximum Intensity.
                    A = Magnitude based on felt area.
                    R = Magnitude based on radius of felt area.
                    * = Magnitude calculated by use of special formulas
                        developed by some person for a certain area.
                        Further info should be given on line 3.
 53-56 f4.2     Logarithm (base 10) of radius of felt area.
 57-61 f5.2     Logarithm (base 10) of area (km**2) number 1 where
                    earthquake was felt exceeding a given intensity.
 62-63 i2       Intensity boardering the area number 1.
 64-68 f5.2     Logarithm (base 10) of area (km**2) number 2 where
                    earthquake was felt exceeding a given intensity.
 69-70 i2       Intensity boardering the area number 2.71 Free
 72     a1      Quality rank of the report (A, B, C, D) 73-75 a3 Reporting agency
 76-79          Free
 80     a1      Type of this line ("2")

Type 3 Line (Optional):
```

```
Columns Format Description  Comments

  1            Free
  2-79 A       Text       Anything
 80    A1      Type of this line ("3")

 This type of line can be used to specify xnear, xfar and the starting depth for use with
HYPOCENTER. For example

 XNEAR  200.0 XFAR  400.0 SDEP   15.0                                              3

   8-13  f6.1  Xnear
  20-25  f6.1  Xfar
  32-36  f5.1  Starting depth

Type 4 line:

Columns Format Description  Comments

  1 Free
  2- 6 A5 Station Name  Blank = End of readings = end of
event
  7 A1 Instrument Type S = SP, I = IP, L = LP etc
  8 A1 Component  Z, N, E ,T, R, 1, 2
  9 Free or weight, see note below
 10 A1 Quality Indicator  I, E, etc.
 11-14 A2 Phase ID  PN, PG, LG, P, S, etc. **
 15 I1 Weighting Indicator (1-4) 0 or blank= full weight, 1=75%, 2=50%, 3=25%,
              4=0%, 9: no weight, use difference
             time (e.g. P-S).
 16 Free or flag A to indicate automartic pick, removed when picking
 17 A1 First Motion  C, D
 18 Note: Currently 15 to 18 can also be used for phase assuming
           column 11-14 is not blank. See note ** below.
 19-20 I2 Hour  Hour can be up to 48 to
             indicate next day

 21-22 I2 Minutes
 23-28 F6.0 Seconds
 29 Free
 30-33 I4 Duration (to noise)  Seconds
 34-40 g7.1 Amplitude (Zero-Peak) in units of nm, nm/s, nm/s^2 or counts.
 41 Free
 42-45 F4.0 Period  Seconds
 46 Free
 47-51 F5.0 Direction of Approach  Degrees
 52 Free
 53-56 F4.0 Phase Velocity  Km/second
 57-60 F4.0 Angle of incidence (was Signal to noise ratio before version 8.0)
```

```
 61-63 I3 Azimuth residual
 64-68 F5.1 Travel time residual
 69-70 I2 Weight
 71-75 F5.0 Epicentral distance(km)
 76 Free
 77-79 I3 Azimuth at source
 80 A1  Type of this line ("4"), can be blank, which it is
               most often

NB: Epicentral distance: Had format I5 before version 7.2. All old lines can be read with
format F5.0 with same results, but now distance can also be e.g. 1.23 km which cannot be read
by earlier versions. However, an UPDATE would fix that.


  ** Long phase names: An 8 character phase can be used in column 11-18. There is then not
room for polarity information. The weight is then put into column 9. This format is recognized
by HYP and MULPLT.

Type 4 cards should be followed by a Blank Card (Type 0)

Type 5 line (optional): Error estimates of previous line, currently not used
                        by any SEISAN programs.

Columns Format Description  Comments
  1 Free
  2-79 Error estimates in same format as previous line, normallytype 4
 80 A1 Type of this line ("5")

Type 6 Line (Optional):

Columns Format Description  Comments
  1 Free 2-79 A Name(s) of tracedata files80 A1 Type of this line ("6")

Type 7 Line (Optional):

Columns Format Description  Comments

  1 Free
  2-79 A Help lines to place the numbers in right positions
 80 A1 Type of this line ("7")

Type F Line (Optional): Fault plane solution

Columns Format Description

  1:30 3F10.0 Strike, dip and rake, Aki convention
 31:45 4F5.1  Error in strike dip and rake (HASH), error in fault plane and aux. plane (FPFIT)
 46:50 F5.1   Fit error:  FPFIT and HASH (F-fit)
 51:55 F5.1   Station distribution ratio (FPFIT, HASH)
 56:60 F5.1   Amplitude ratio fit (HASH, FOCMEC)
```

```
 61:65 I2     Number of bad polarities (FOCMEC, PINV)
 64.65 I2     Number of bad amplitude  ratios (FOCMEC)
 67:69 A3     Agency code
 71:77 A7     Program used
 78:78 A1     Quality of solution, A (best), B C or D (worst), added manually
 79:79 A1     Blank, can be used by user
 80:80 A1     F


Type E Line (Optional): Hyp error estimates

Columns Format Description

  1 Free
  2 - 5 A4 The text GAP=
  6 - 8 I3 Gap
 15-20 F6.2 Origin time error
 25-30 F6.1 Latitude (y) error
 31-32 Free
 33-38 F6.1 Longitude (x) error (km)
 39-43 F5.1 Depth (z) error (km)
 44-55 E12.4 Covariance (x,y) km*km
 56-67 E12.4 Covarience (x,z) km*km
 68-79 E14.4 Covariance (y,z) km*km

Type I Line, ID line

Columns Format description1 Free

  2:8 Help text for the action indicator
  9:11  Last action done, so far defined SPL: Split
           REG: Register
           ARG: AUTO Register, AUTOREG
           UPD: Update
           UP : Update only from EEV
           REE: Register from EEV
           DUB: Duplicated event
           NEW: New event
 12 Free
 13:26 Date and time of last action
 27 Free
 28:30 Help text for operator
 36:42 Help text for status
 43:56 Status flags, not yet defined
 57 Free
 58:60 Help text for ID
 61:74 ID, year to second
 75 If d, this indicate that a new file id had to be created which was
        one or more seconds different from an existing ID to avoid overwrite.
```

```
 76 Indicate if ID is locked. Blank means not locked, L means locked.

Type H line, High accuracy hypoenter line

Columns

  1:55 As type 1 line
 16 Free
 17 Seconds, f6.3
 23 Free
 24:32 Latitude, f9.5
 33 Free
 34:43 Longitude, f10.5
 44 Free
 45:52 Depth, f8.3
 53 Free
 54:59 RMS, f6.3
 60:79 Free
 80 H

Type E13 and EC3 line, explosion information

Example

1980 0124 0927 CHARGE(T): 0.5 E13  LE Haakonsvern,  HAA underwater explosion  E13 EC3

Information on explsion site, time and agency, same format as a type 1 line, no magnitudesused, la
EC3 Information on charge and site

Columns
  2:11 Info text
 11:12 Blank
 13:22 Charge in tons, f10.3
 23:77 Any information, a
 78:80 EC3

Type MACRO3 line: File name of macroseismic observations in ISO directory

Example:

1980-03-14-0456-05.MACRO MACRO3

An example of the file is:

Sunnfjord 1980 314 456 5 GMT 1980 314 556 5 Local time
 Comment
60.500 5.270 1.0 EMS 5088 MJOELKERAAEN
60.560 5.260 1.0 EMS 5100 ISDALSTOE
60.570 5.050 1.0 EMS 5112 ROSSLAND
```

```
1.  Line
Location, GMT time, Local time. Format a30,i4,1x,2i2,1x,2i2,1x,i2,'
GMT',1x,i4,1x,2i2,1x,2i2,1x,i2,1x,'Local time'
2.  Line Comments
3.  Line Observations: Latitude, Longitude,intensity, code for scale, postal code or similar,
location,Format 2f10.4,f5.1,1x,a3,1x,a10,2x,a. Note the postal code is an ascii string and
left justified (a10).

Type 3 line giving xnear/xfar

Definition of xnear and xfar to be used with HYPOCENTER.

Example
XNEAR 1000.0 XFAR 2000.0 3
Columns
  8-13: xnear value
 20-25: xfar value
```

Appendix B

The Seisan waveform file format

The file is written from Fortran as an unformatted file. This means that the file contains additional characters (not described below, see end of this Appendix) between each block, which must be taken into account if the file is read as a binary file. If read as Fortran unformatted, the content will appear as described below. However, the internal structure is different on Sun, Linux and PC. SEISAN automatically corrects for these differences. The SEISAN ASCII format has identical headers to the binary files, however the binary samples are written as formatted integers, one channel at the time just like the in the binary format.

```
 EVENT FILE HEADER

 CONTAINS MINIMUM 12 ASCII STRINGS OF 80 BYTES.
 ALL FORMATS I OR A UNLESS OTHERWISE SPECIFIED.

line 1
   1     1: FREE
   2    30: NETWORK NAME,
             COULD E.G. BE WESTERN NORWAY NETWORK
  31    33: NUMBER OF CHANNELS, MAX 999
  34    36: YEAR-1900, e.g. 101 for 2001 (I3)
  37
  38    40: DOY
  41
  42    43: MONTH
  44
  45    46: DAY
  47
  48    49: HR
  50
  51    52: MIN
  53
  54    59: SEC, FORMAT F6.3
  60
  61    69: TOTAL TIME WINDOW (SECS), FORMAT F9.3
```

```
 70    80: FREE
 71
 72
 73    80: FREE

                                                                |
line 2
  1    80: FREE
                                                                |
line 3
  1
  2     5: STATION CODE (A4), first 4 characters
  6     7: FIRST two COMPONENT CODES (A2), SEED style
  8      : NOT USED
  9      : LAST COMPONENT CODE (A1), SEED style
 10      : STATION CODE (A1), LAST CHARACTER IF 5 CHARACTER STATION CODE
 11    17: START TIME RELATIVE TO EVENT FILE TIME (SECS) F7.2
 18      : BLANK
 19    26: STATION DATA INTERVAL LENGTH (SECS)  F8.2
 27    52: SECOND CHANNEL
 53    78: THIRD  CHANNEL
 79    80: BLANK                                                |
                                                                |

line 4-XX, where XX depends on number of channels, however, XX
           is at least 12 so there might be some blank lines.
  1    80: THREE MORE CHANNELS (SAME FORMAT AS line 3)
                                                                |

EVENT FILE CHANNEL HEADER

HEADER IS 1040 BYTES LONG, WRITTEN AS ONE VARIABLE DEFINED AS
CHARACTER*1040
THE PARAMETERS ARE WRITTEN FORMATTED WITH INTERNAL WRITE INTO
1040 BYTE TEXT STRING.
FORMAT IS ALWAY I FORMAT UNLESS OTHERWISE SPECIFIED

  1     5: STATION CODE (A5)
  6     7: FIRST TWO COMPONENT CODES (A2), SEED style
  8      : FIRST LOCATION CODE (A1), SEED style
  9      : LAST COMPONENT CODE (A1), SEED style

 10    12: YEAR - 1900, e.g. 101 for 2001, (I3)
 13      : SECOND LOCATION CODE (A1), SEED style
 14    16: DOY
 17      : FIRST NETWORK CODE (A1), SEED style
 18    19: MONTH
 20      : SECOND NETWORK CODE (A1), SEED style
 21    22: DAY
```

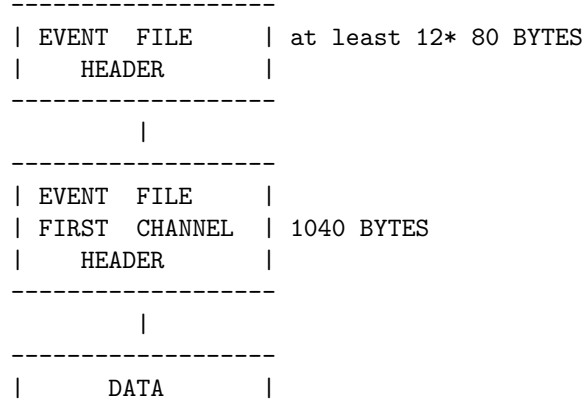
```
 23
 24    25: HR
 26
 27    28: MIN
 29      : TIMING INDICATOR, BLANK: TIME IS OK, E: UNCERATIAN TIME
 30    35: SECOND (F6.3)
 36
 37    43: SAMPLE RATE  (F7.2 or any f-format)
 44    50: NUMBER OF SAMPLES (I7)
 51
 52    59: LATITUDE (F8.4), optional
 60
 61    69: LONGITUDE (F9.4), optional
 70
 71    75: ELEVATION (METERS), optional
 76      : Indicate gain factor: Blank: No gain factor, G: Gain factor in
         column 148 to 159
 77      : 2 OR 4 FOR 2 OR 4 BYTE INTEGER, BLANK IS 2 BYTE
 78      : P: Poles and zeros used for response info, blank: Seismometer
              period etc used for response info. See below for details.
           T: Use up to 30 tabulated values irrespective of what is given
              below. If less than 30, blank characters must be given.
 79      : C: a combination of table, poles and zeros or instrument
              constants have been used, for information only. Value in 78
              must then be T.
           F: Force use of header response, e.g. generated by MULPLT. Only
              gain at 1 hz is correct and 78 must be set to T.
 80 -  80: FREE
 148- 159: Normally comment, if 76 set to G, this is a gain factor,
           format G12.7. All samples read from channel are multipled
           by this factor when read by routine seisinc. Used when data
           is stored in units of e.g. nm where values can be less than 1.
           Currently only generated by MULPLT when option OUT is used to
           extract part of a waveform file.
 81 - 160: COMMENT LINE DESCRIBING THE SYSTEM RESPONSE (A80)

If character 78 is blank, option 1:

161 - 240: (10G8.3) 1. SEISMOMETER PERIOD
                    2. FRACTION OF CRITICAL DAMPING
                    3. SEISMOMETER GENERATOR CONSTANT (V/m/s) or
                       ACCELEROMETER SENSITIVITY (V/G)
                    4. AMPLIFIER GAIN
                    5. RECORDING MEDIA GAIN (I.E. 2048 COUNTS/VOLT)
                    6. GAIN AT 1.0 HZ, UNITS: COUNTS/METER
                    7. CUTOFF FREQUENCY FOR FILTER1 (HZ)
                    8. # OF POLES FOR FILTER1 (NEGATIVE FOR HIGHPASS)
                    9. CUTOFF FREQUENCY FOR FILTER2 (HZ)
                   10. # OF POLES FOR FILTER2 (NEGATIVE FOR HIGHPASS)
```

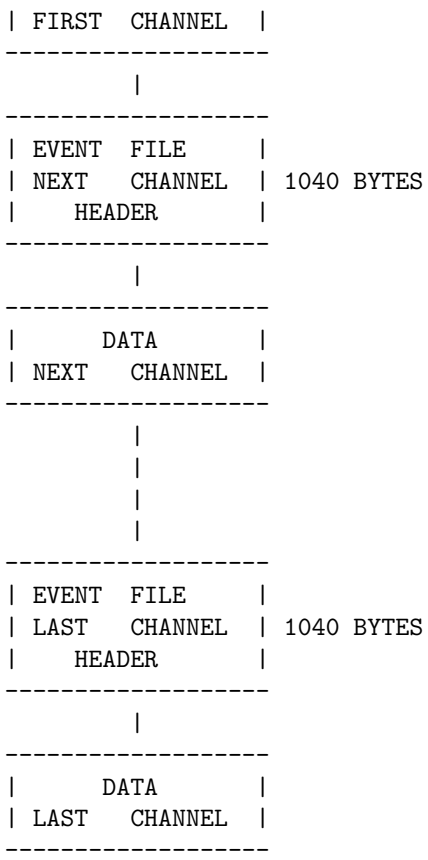
```
 241 - 320: (10G8.3) FREQUENCIES AND #'S OF POLES FOR FIVE MORE FILTERS
 321 -1040: RESPONSE CURVES (9(10G8.3) FREQ., AMPL. (REL. 1.0 HZ) AND PHASE,
            WRITTEN IN GROUPS OF 10 FREQUENCIES, 10 AMPLITUDES AND 10 PHASES

  If character 78 is P, option 2:

  161 - 182 (1X,2I5,G11.4) 1. NUMBER OF POLES
                           2. NUMBER OF ZEROS
                           3. NORMALIZATION CONSTANT, COUNTS/M
  183 - 240 (5G11.4)       2 Poles in pairs of real and imaginary parts
  241 -1040 (G11.4)        Remaining poles and zeros. 7 values are written
                           and then 3 spaces are left blank, see example
                           below.
```

For each pole or zero, there are two real numbers representing the real and imaginary part of the pole or zero, thus the number of poles is half the number of values written. First all the poles are written in pairs of real and imaginary parts, then follow the zeros. There is room for a total of 37 poles and zeros (74 pairs). The poles and zeros are written in a simulated line mode to make it easier to read, thus the 3 blanks after writing 7 values. It is assumed that the response is in displacment with units of counts/m.

```
SLR  L  E 86 199  7 18 15  6 35.960   1.000   1320                          4P

    11    5  .2760E+11  .3770      .1830      .3770      .1830      .6540
  .0000      .2320      .0000      .2320      .0000      .2320      .0000
  .3280      .0000      .3280      .0000      .3280      .0000      .2140E 01
  .0000      .2140E 01  .0000      .0000      .0000      .0000      .0000
  .0000      .0000      .0000      .0000      .0000      .0000
```

NOTE: The component information in character 6 IS VERY IMPORTANT. It MUST be A if an accelerometer is used, any other character assumes a velocity transducer. This is only relevant however if option 1 is used where response values will be calculated from the free period etc. If option 1 with discrete values or poles and zeros are used, the first component character can be anything.





To write a SEISAN file: If main headers are called mhead, channel header chead, data is data (integer), there is nchan channels and each has nsamp samples, then the file is written as

```
Do i=1,12
  Write(1) mhead(i)
Enddo
Do k=1,nchan
  Write(1) chead
  Write(1) (data(i),i=1,nsmap)
Enddo
```

This example only works up to 30 channels when writing main header. For more channels, see e.g. program SEISEI how to do it.

Details of binary file structure

When Fortran writes a files opened with "form=unformatted", additional data is added to the file to serve as record separators which have to be taken into account if the file is read from a C-program or if read binary from a Fortran program. Unfortunately, the number of and meaning of these additional characters are compiler dependent. On Sun, Linux, MaxOSX and PC from version 7.0 (using Digital Fortran), every write is preceded and terminated with 4 additional bytes giving the number of bytes in

the write. On the PC, Seisan version 6.0 and earlier using Microsoft Fortran, the first 2 bytes in the file are the ASCII character "KP". Every write is preceded and terminated with one byte giving the number of bytes in the write. If the write contains more than 128 bytes, it is blocked in records of 128 bytes, each with the start and end byte which in this case is the number 128. Each record is thus 130 bytes long. All of these additional bytes are transparent to the user if the file is read as an unformatted file. However, since the structure is different on Sun, Linux, MacOSX and PC, a file written as unformatted on Sun, Linux or MacOSX cannot be read as unformatted on PC or vice versa. . The files are very easy to write and read on the same computer but difficult to read if written on a different computer. To further complicate matters, the byte order is different on Sun and PC. With 64 bit systems, 8 bytes is used to define number of bytes written. This type of file can also be read with SEISAN, but so far only data written on Linux have been tested for reading on all systems. This means that version 7.0 can read all earlier waveform files on all platforms from all platforms. However, files written on version 7.0 PC cannot be read by any earlier versions of Seisan without modifying the earlier seisan version. In SEISAN, all files are written as unformatted files. In order to read the files independently of where they were written, the reading routine (buf_read in seisinc, in LIB) reads the file from Fortran as a direct access file with a record length of 2048 bytes. The additional bytes are thrown away, the relevant bytes fished out and swapped if the file is written on a different computer than where it is read. Since there is no information stored in the header of the file giving the byte address of each channel, the routine must read the first file-header, calculate how many bytes there are down to where the next channel starts, jump down and repeat the process until the desired channel is reached (this is also how SUDS files are read). However, compared to reading the file as unformatted, only a fraction of the file is read to fish out a particular channel. Once the channel header has been read, the start address is stored in the subroutine so any subsequent access to that channel is very fast. Overall, random access to SEISAN waveform files is much faster with the binary read than the previous (version 5.0 and earlier) unformatted read. Only in the case where the whole file is read is the unformatted read faster.

PC file structure Up to and inluding version 6.0	Sun and Linux file structure PC structure from version 7.0
one byte: K indicates start of file	4 bytes: # of bytes following
one byte: # of bytes following	one block of data
128 bytes or less of data	4 bytes: # bytes in prev. write
one byte: # of bytes in previous record	4 bytes: # of bytes following
one byte: # of bytes in following record	one block of data
128 bytes or less of data	.....
.....	.....
.....	.....

For 64 bit systems, the above 4 byte numbers are 8 byte numbers.

From version 7.0,the Linux and PC file structures are exactly the same. On Sun the structure is the same except that the bytes are swapped. This is used by SEISAN to find out where the file was written. Since there is always 80 characters in the first write, character one in the Linux and PC file will be the character P (which is represented by 80) while on Sun character 4 is P.

Appendix C

Response file formats used in SEISAN

SEISAN can use either SEISAN response file format or GSE response file format. The detailed SEISAN format is given here as well as the parts of the GSE format used in SEISAN.

SEISAN response file format

There are three SEISAN response formats. The instrument response can either be presented as (1) instruments constants, (2) pairs of frequency amplitude and phase or as (3) poles and zeros.

```
Line 1
   1     5: STATION CODE (A5)
   6     9: COMPONENT (A4), E.G. SZ L : SZ LOW GAIN, AN: ACCELERATION NS
  10      : CENTURY, 0: 1900, 1:2000
  11    12: YEAR
  14    16: DOY
  18    19: MONTH
  21    22: DAY
  24    25: HR
  27    28: MIN
  30    35: SECOND (F6.3)
  52    59: LATITUDE (F8.4), optional
  61    69: LONGITUDE (F9.4), optional
  71    75: ELEVATION (METERS), optional
  78      : P: Poles and zeros
            Blank: Seismometer period etc used for response info. See below for
                   details.
            T: Use up to 30 tabulated values irrespective of what is given
               below. If less than 30, blank characters must be given.
  79      : C: a combination of table, poles and zeros or instrument
               constants have been used, for information only. Value in 78
               must then be T.
            F: Force use of header response. Only
               gain at 1 hz is correct and 78 must be set to T.
```

```
Line 2
   1   80: COMMENT LINE DESCRIBING THE SYSTEM RESPONSE (A80)


OPTION 1

If character 78 is blank, option 1:

Line 3
  1 - 80: (10G8.3)   1. SEISMOMETER PERIOD
                     2. FRACTION OF CRITICAL DAMPING
                     3. SEISMOMETER GENERATOR CONSTANT (V/m/s) or
                        ACCELEROMETER SENSITIVITY (V/G)
                     4. AMPLIFIER GAIN
                     5. RECORDING MEDIA GAIN (I.E. 2048 COUNTS/VOLT)
                     6. GAIN AT 1.0 HZ, UNITS: COUNTS/METER
                     7. CUTOFF FREQUENCY FOR FILTER1 (HZ)
                     8. # OF POLES FOR FILTER1 (NEGATIVE FOR HIGHPASS)
                     9. CUTOFF FREQUENCY FOR FILTER2 (HZ)
                    10. # OF POLES FOR FILTER2 (NEGATIVE FOR HIGHPASS)

Line 4
  1 - 80: (10G8.3) FREQUENCIES AND #'S OF POLES FOR FIVE MORE FILTERS


Lines 5-13: RESPONSE CURVE in 3 blocks of 3 lines
  1 - 80: (10G8.3) FREQ.
  1 - 80: (10G8.3) AMPL. (REL. 1.0 HZ)
  1 - 80: (10G8.3) PHASE,

The are only for info in option 1


OPTION 2
If character 78 is T, use tablated values, format as in option 1


OPTION 3
If character 78 is P, option 2:

Line 3
  1 - 22:   (1X,2I5,G11.4) 1. NUMBER OF POLES
                           2. NUMBER OF ZEROS
                            3. NORMALIZATION CONSTANT, COUNTS/M
  23 - 80 (5G11.4)          2 Poles in pairs of real and imaginary parts

Lines 4-13
  1 - 80: (G11.4) Remaining poles and zeros. 7 values are written
```

```
                          and then 3 spaces are left blank.
```

Example of SEISAN FAP response file

```
KBS  B  Z100   1  1  1  0  0  0.000

360.    .700    .260E+04  0.    .419E+06.684E+10  0.       0.000  0.       0.000
  0.        0.00  0.        0.00  0.        0.00  0.        0.00  0.        0.00
.500E-02.700E-02.980E-02.140E-01.190E-01.270E-01.370E-01.520E-01.730E-01.100
.480E-02.694E-02.978E-02.140E-01.190E-01.270E-01.370E-01.520E-01.730E-01.100
 138.366 123.400 113.340 106.128 101.813  98.283  96.034  94.289  93.054  92.229
.140    .200    .280    .390    .550    .770    1.10    1.50    2.10    2.90
.140    .200    .280    .390    .550    .770    1.10    1.50    2.10    2.90
  91.592  91.114  90.796  90.571  90.405  90.289  90.203  90.149  90.106  90.077
4.10    5.80    8.10    11.0    16.0    22.0    31.0    43.0    60.0    85.0
4.10    5.80    8.10    11.0    16.0    22.0    31.0    43.0    60.0    85.0
  90.054  90.038  90.028  90.020  90.014  90.010  90.007  90.005  90.004  90.003
```

Example of SEISAN PAZ response file using the same constants as above

```
KBS  B  Z100   1  1  1  0  0  0.000                                          P

     2    3 0.1089E+10-0.1222E-01 0.1246E-01-0.1222E-01-0.1246E-01     0.
     0.          0.          0.          0.          0.          0.          0.
     0.          0.          0.          0.          0.          0.          0.
     0.
```

GSE response file format

Below is an example of a GSE response file generated by RESP for the same parameters as used for the SEISAN file above. The GSE response format is rather complex, and can contain parts that will not be understood by SEISAN.

Below follows an example of the GSE response using FAP. The first line gives station and sensor type. The numbers following are gain in nm/c (0.15) at reference period (1. seconds), sample rate (20) and date. The following line (FAP2) gives a gain factor (1.) and the output units (V for Volts). Finally follows the frequency, gain and phase triplets.

```
CAL2 KBS   B Z      STS-1    0.15E+00      1.   20.00000 2003/ 1/ 1  0: 0
FAP2  1 V                60
    0.00500  0.48047733E-02  138
    0.00590  0.57846815E-02  130
    0.00700  0.69361408E-02  123
    0.00830  0.82667302E-02  117
    0.00980  0.97841555E-02  113
    0.01200  0.11995634E-01  108
    0.01400  0.14000172E-01  106
```

```
...
   60.00000  0.59999992E+02   90
   71.00000  0.70999992E+02   90
   85.00000  0.84999992E+02   90
  100.00000  0.99999985E+02   90
```

In the poles and zero representation, the file looks like:

```
CAL2 KBS   B Z      STS-1    0.15E+00      1.   20.00000 2003/ 1/ 1  0: 0
PAZ2  1 V  0.26000000E-05                 2   3 Laplace transform
 -0.12217305E-01  0.12464144E-01
 -0.12217305E-01 -0.12464144E-01
  0.00000000E+00  0.00000000E+00
  0.00000000E+00  0.00000000E+00
  0.00000000E+00  0.00000000E+00
DIG2  2  0.41900000E+06    20.00000 quanterra
```

The first line is the same as before. The PAZ2 lines has the following meaning: Normalization constant of seismometer and filter (0.26e-5), number of poles and zeroes (2 and 3), and the type of response (Laplace transform). The DIG2 line has the gain of amplifier and AD converter combined (0.419e6 c/V) and sample rate.

In the simplest case, the response is given by the PAZ and a scaling factor. It is common (like in SEED) to have two scaling constants, one that normalizes the PAZ to amplitude 1 at a calibration period, and another constant that gives the amplitudes in the physical units. This is NOT the case with the GSE2 format. The GSE2 response for PAZ normally contains at least two parts, the CAL2 line and a PAZ2 line. The scaling factor should scale the PAZ to output/input units, NOT normalize. In the CAL2 line, the system sensitivity at a calibration period is given in units input/output, but is generally not needed. The total response is given by the PAZ, multiplied with the PAZ2 scaling factor, or the product of several stages.

This is how SEISAN reads the response, however, if it finds that the PAZ2 gives normalized values at the calibration period, the response is multiplied with the sensitivity given in the CAL2 line (this is done because such GSE files have been seen).

FIR filters can be specified in GSE as an additional stage and can be written out by the RESP program. An example is given below. The FIR filter coefficients are required to completely describe the instrumentation. However, they are not used in SEISAN.

FIR2 3 0.10E+01 3 0.030 A 180
0.18067658E-06 0.88848173E-06 0.24195760E-05 0.37699938E-05 0.32037496E-06
-0.18581748E-04 -0.69605186E-04 -0.16990304E-03 -0.32499805E-03 -0.51408820E-03
-0.68242848E-03 -0.75194239E-03 -0.65436028E-03 -0.37627667E-03 0.94138085E-05
0.35409257E-03 0.49653835E-03 0.35531260E-03 -0.29224902E-05 -0.37382543E-03
...

Index

Q_{Lg}, 252
3-component stations, 77, 117, 206
64 bit computer, 21, 250
64 bit computer , 344

A-value, 264
Acceleration, 115
Add waveform format, 10
Adding BUD archive to nordic file, 113
Aftershock processing, 214
AGA, 9
Agency, 19, 86, 184, 277
Agency, select for, 159
AHSEI, 192, 195
Align traces, 180
All channel mode, 111
AMP, 113
Amplitude, 114
Amplitude attenuation, 276
Amplitude, automatic, 114
Andalucia, 183, 191, 198
Angle of incidence, 218
Apparent velocity, 49, 88, 114, 118, 253
APPEND, 171
Append another event to current event, 47
ARCFETCH, 192
ARCSEI, 192
Arcsei.def, 192
Array processing, 293
Array stations, 77
ASCII files, 179, 199
ASCII waveform format, 189
ASCII, convert to, 199
ASCII, convert to SEISAN, 195
ASCII, SEISAN to, 178
ASCSEI, 195
ASSOCI, 240
Associ_rest.out, 240
Associate events in time, 52
Attenuation equation, 272
Attenuation relations, 262
AUTO, 206
AUTO_LOCATE, 136
AUTO_PROCESS, 135
Automate EPIMAP, 142
Automatic amplitude, 114
Automatic coda length, 114, 136
Automatic phase picking, 53, 206
AUTOPIC, 25
AUTOPIC.INP, 208
Autopic.out, 206
AUTOREG, 171
Autoscale depth profile, 140
Autoscaling, 119
Autosig, 210
Autosig.par, 212
Average station residual , 88
Azimuth, 114, 206
Azimuth error, 88

B-value, 204, 264, 273
Back azimuth, 49
Backazimuth, 118
Background noise, 128, 258
Background seismicity, 273
Bandpass filter, default in mulplt, 136
BASE, 43
Bash shell, 19
BGISEI, 195
Binary SEISAN waveform file structure, 343
Bisquare weighting, 88
Body wave magnitude, 79
BOUCH, 240
Bouch.out, 247
BOUSEI, 40, 240
Brune spectrum, 320
BUD archive data, 12
BUL, 165
BUL.INP, 25, 165
Bul.ps, 166
Bullen table, 96
Bulletin, 165

Bulletin program, 25
BVALUE, 204
Bvalue.eps, 206
Bvalue.out, 206

CAL directory, 70, 197
Calibration file, 301
CAT directory, 9
CAT-file, 9, 14, 37, 90
CAT_AGA, 183, 184
Cat_aga.par, 184
Catalog work, 10, 40, 71, 72, 163, 169, 264, 282
Catday.out, 170
Cathour.out, 170
Catmonth.out, 170
CATSTAT, 169
Catyear.out, 170
Change filters, 106
Change of day, 90
Channel order, 111
Channel selection, 98, 106
Check, 321
Check data base, 43
CHECK_BASE, 43
CHECKRE, 43
China, 195
Chronological order, 163, 164
CITSEI, 195
CityShark, 195
CLUSTER, 274
CNVSSR and CNVSSA, 196
Coda length, 87, 113, 114, 136, 275
Coda magnitude, 78, 87, 157
CODAQ, 40, 233
Codaq.inp, 234
Codaq.par, 234
COLLECT, 161
COLOR.DEF, 25
Command to the operating system, 49
Commands in EEV, 45
Common tasks in SEISAN, 38
Compact file, 14, 163
Compile programs, 27
Componen.for, 114
Component codes, 11
Component names in S-file, 114
Compression, 11
CONDET, 214
CONGAP, 172
connoi, 173
Conrad interface, 87, 88
Consistency check, 88
Console window, 69
Cont, 111
Continuous data, 13, 100, 111, 178
Continuous data base, default selection, 30
Continuous plotting, 99
Contour file, 141
Contour maps, 274
Conversion of component codes, 189
Conversion of station codes, 189
Conversion programs, 183
Convert angles, 218
Convert to ASCII, 199
Converted magnitude, 277
Copy events, 47
COPY_WAV_DIR, 172
Corner frequency, 126, 271
CORR, 313
Correctpolarity, 182
Correlation, 313
Crash, check base, 43
CRIATT, 270
CRIPAR, 264, 271
CRISEI, 141
CRISIS99, 262, 265
Crustal model, 25
Crustal parameters, 253
Crustal structure determination, 83
CSS format, 195
Cut MiniSEED files, 177

Daily number of events, 169
Damped signal, 320
DAT directory, 25
Data base content, 43
Data base error, 48
Data base, check, 43
Database generation, 24
Database name, 24
Database security, 42
Database structure, 55, 72
DATABASE2MSEED, 174
Dataseis, 196
DC level, 107
Default channel, 134
Default continuous data base, 30
Default filter, 136
Defaults, 134
DEL directory, 7

DELET database, 48, 55
Delete automatic picks, 112
Delete event, 42, 47
Delete ID-line, 163
Delete many S-files, 165
Delete S-file, 112
Delete waveform file, 112, 118
DELF, 165
Depth as function of rms, 52
Depth profile, 140
Depth, fixing, 82
Depth, select for, 158
Depth, start value, 82
Different databases, 55
Digital data, 41
Dimension, 20, 115
Diminution function, 123, 270
Dip, 220
DIRF, 165
Displacement, 115
Distance id, 158
Distance indicator, 112
Distance order, 111
Distance range, select for, 159
Distance weighting, 82
Double difference location, 289
DR3024 and DR3016, 195
DRSEI, 195
Drum recording, 99
Duplicate event, 48
Duplicate ID, 42, 163
Duration magnitude, 87

Earthquake catalogue, 262
Earthquake statistics, 168, 169
EDBPICK picking program, 186
Editor, 19
EDRNOR, 184
EEV, 37, 44
EEV over several months, 48
EEV print command, 52
EEV, jump to another month and database, 49
EEV.OUT, 47
End time, 25
Environmental variable, 19
Epicenter area selection, 142
EPIMAP, 25, 59, 137, 168
Epimap.cor, 142
Epimap.inp, 142
Epimap.num, 142
EQRISK, 262, 271
EQRSEI, 141, 262, 274
Error estimate, HYP, 89
Error in S-file, 48
Errors in hypocenter, 158
EUROPE.MAP, 25
evanoi, 173
Event detection, 215
Event felt, select for, 157
Event ID, 112
Event type, 9, 112
Event, duplicate ID, 163
EXFILTER, 282
Explosion, 48, 71, 112, 158
Explosion filtering, 282
Extract data in JSEISAN, 63
Extract data in MULPLT, 119
Extract from waveform file, 119
Extract waveform files, 174, 177
Extracting events from the database, 161

Fault plane solution, 48, 232
Fault plane solution, composite , 221
Fault plane solution, plot, 143
Fault plane solution, plotting several, 219
FDSN, 199
Federation of Digital Seismographic Networks, 199
Felt information, 311
File conversion and modification programs, 183
File number, 97, 98
File types, 14
FILENR.LIS, 14, 71, 165
Filter and spectral analysis, 106
Filter in WAVETOOL, 176
Filter, high pass, 100
Filter, instability, 100
Filter, low pass, 100
Filters in MULPLT, 106
Fixed scaling, 119
Fixing depth, 82, 88
Fixing location, 83, 88
Fixing origin time, 83, 88
FK analysis, 118, 293
Focmec.eps, 221
Focmec.inp, 218
focmec.lst, 221
Focmec.out, 220
Format, 11
Format, mulplt, 99
Format, Nordic, 329

Format, volcanic information, 290
Format, waveform, 339
Fourier acceleration spectra, 270
FPFIT, 221
Free surface effect, 271

Güralp, 196
Gain factor, 342
Gap range, select for, 159
GBV recorder, 196
GCF format, 196
Generic Mapping Tools, 188
Geometrical spreading, 123, 270, 276
GeoSig, 196
GeoSys, 196
Geotech Instruments, 197
Get_wav.out, 177
GETPDE, 183, 184
GhostView, 23
GIINOR, 184
GIISEI, 196
Global event location, 81
Global phases, 114
Global station coordinates, 188
GMAP, 48
Gmap, 147
GMAP SEISAN.DEF parameters, 31
GMT, 73, 188, 291
GMTMAP, 48
GMTXY, 73, 170, 320
Go to event by number or date, 47
Googel map, 147
Google Earth, 143, 147
Graphics, 73, 320
Graphics problem on PC, 24
Graphics problem on UNIX, 21
Graphics window, size, 26
Ground displacement, 276
Ground motion seismogram, 115
GSE, 10, 70
GSE format, 187
GSE response file format, 350
GSERESP, 196
GSN, 196
GSRSEI, 196
GURSEI, 196

H_model.par, 83
H_model_out, 83
Hard copy files, 20
Hardcopies, 99
Hardcopy in single trace mode, 106
Hardwired constants, 129
HARNOR, 184
Harvard moment tensor solution , 184
HASH, 223
Header times, 180
Help in SEISAN, 65
Herkij_distance, 30, 135
Herrmann, 40, 240
Herrmann's package, 295
HERSEI, 242
High accuracy, 30, 43
High frequency decay factor, 271
High pass filter, 106
HINNOR, 184
Hints on modelling, 247
Homepage, 5
Horizontal component, 118
HYP output in S-file, 85
HYP problem, azimuth, 78
HYP, do not locate, 81
HYP, multiple model mode, 83
Hyp.cat, 163
Hyp.out, 221
HYP_ISC, 96
HYPINV, 96
HYPNOR, 184
HYPO style weight, 114
HYPO71, 48, 86, 91
HYPO71 files to Nordic files, 184
Hypocenter program, 77
Hypocenter, add to S-file, 49
Hypocenters, more than one, 71
HYPODD, 289, 314
Hypoinverse, 48, 95
Hypoinverse format, 187
Hypoinverse to Nordic, 184
Hypsum.out, 90

IASP, 251
Iasp.out, 251
IASP91.HED, 15, 250
IASPEI phases, 77, 89
IASPEI Seismic Format, 186
IASPEI software library, 197
IASPEI91, 15, 81, 88, 114
Iceland, 201
ID, lock, 164
ID-line, 9, 112, 163

Identifying seismic phases, 114
IL, 49
IMS1.0, 186
Index file, 14, 44, 47, 55, 72, 159
index.out, 159
Indexeev.out, 47, 236
Input file for CRIATT, 271
Input file for EQRISK, 272
Input files for the CRISIS99, 266
INPUTEPI, 49, 114
INPUTONE, 49
INPUTONE., 114
Inserting events in the database, 162
Instrument response, 69, 295
Interactive work, 44
International Monitoring System, 186
Inversion, 284
inversion of surface wave group velocities, 295
Invertratio, 288
INVRAD, 49, 232
IRIS, 196, 199, 201
IRISEI, 196
ISC, 81, 184, 186
ISC location, 49, 96
ISC to Nordic, 184
Iscloc.def, 96
ISCNOR, 184
ISCSTA, 186
ISF format, 186
ISFNOR, 186
ISMSEI, 196
Israel, 183, 196

Java, 320, 321
Java, start command, 19
JHD, 284, 288
Joint hypocenter determination, 284, 288
JSEISAN, 55
jseisan, 19
JSEISAN, configuration parameters, 65
JSEISAN, configure, 55

K2SEI, 197
KACSEI, 196
Kappa, determine, 255
Kinemetrics, 186, 196
KINSEI, 196
KW2ASC, 197

LD_LIBRARY_PATH, 18, 19
Least squares, 170
Lennartz, 197, 199
LENPCQ, 197
LENSEI, 197
Lg phase, 77
Lg phase velocity, 88
Lg waves, 252
Linux file structure, 345
Local database, update, 163
Local index file, 55, 72
Local magnitude, 78, 157, 276
Local magnitude parameters, 89
LOCAL_CAL, 20, 70
Locate event, 49
Locate event in MULPLT, 115
Locate two events together, 49
Locate with Bullen table, 96
Locate with ISC program, 49, 96
Location code, 12, 189
Location programs, 77
Location with one station, 78
Location, max distance, 88
Location, min # of stations, 89
Lock ID, 164
LOG directory, 9
Log file, 163, 164
Long phase names, 81, 115, 334
Low pass filter, 106
Lower bound magnitude, 264
LSQ, 170, 320
Lsq.eps, 170
Lsq_gmt.out, 170

M88SEI, 197
MacOSX, 21
MACROMAP, 311
Macromap.ps, 311
Macroseismic information, 49, 311
MAG, 127, 168, 275
MAG2, 280
Mag_amp.out, 277
Mag_coda.out, 277
Mag_mag.out, 277
Mag_new.out, 277
Mag_newa.out, 277
Mag_spec.out, 277
Magnitude, 78, 87, 89, 275
Magnitude conversion, 277
Magnitude correction, 90
Magnitude in epimap, 138

Magnitude order, 168
Magnitude relation, 275
Magnitude residuals, 86
Magnitude weight, 87
Magnitude, fixed, 81
Magnitude, move from 2 header to first, 168
Magnitude, select for, 159
Magnitudes without type, 157
Magnitudes, more than 3, 71, 81
MAGSTAT, 282
Mailing lists, 5
Makefile, 27
MAKEREA, 24, 171
MAP, 57
Map files, 25, 188
MARS88, 197
Maximum expected magnitude, 264
Maximum likelihood b-value, 206
Maximum no of iterations, 88
Maximum number of traces, 20
Maximum observed magnitude, 264
mB, 157
mb, 79, 116, 157
Mc, 78, 275
Memory for waveform data, 97
Merge catalogs, 71
Merge events near in time, 240
Merge waveform files, 119, 179
MERGE_WAVEFORM, 119
Merging files from different networks, 179
Ml, 78, 89, 275, 276
MODEL.DEF, 83
Modelling parameters, Q, 87
Modes of operation, 37
Moho, 87, 88
Moment, 126, 256, 271
Moment magnitude, 80
Moment tensor inversion, 232
Monthly epicenter files, 163
Monthly number of events, 169
Mount Unix disk on PC, 24
Mouse, test, 320
Moving files between Sun and PC, 15
MS, 157
Ms, 79, 116, 157
MSCUT, 177
MSFIX, 197
MULPLT, 25, 41, 50, 97
Mulplt, format, 99
Mulplt, option cont, 111
MULPLT.DEF, 25, 111, 129
Mulplt.ext, 100
MULPLT.OUT, 128
Multiple event S-file, 162
Multiplexed files, 41
Multitrace mode, 99
Mw, 80, 126

Nakamura, 255
Name of the local Index file, 72
Nanometer, 78, 115
Nanometrics, 197
NANSEI, 19, 197
NDC, 186
Nearest station, 82
NEIC, 197
NEISEI, 197
Network code, 12, 189
Network code, put in waveform file, 189
New events into SEISAN, 112
NEWEVE, 42
Noise spectrum, 125, 128, 256, 258
Noise study, 128, 258
Nordic format, 9, 14, 329
Nordic to HYPO71, 187
NORGSE, 187
NORHIN, 187
Norhin, 96
Norhtrhine-Westphalia, 191
NORHYP, 187
NORIMS, 186
Normalization constant, 301
NRWSEI, 191

Omega square, 270
Operator ID, 163
Order of magnitudes, 168
ORFEUS, 192
OS9SEI, 197
Other databases, 9
Output file from CRIATT, 271
Output file from EQRISK, 274
Output files from CRISIS99, 270
Output, corrected trace, 119

P phase, 77
P-phase picking, 210
P_ALIGN, 180
Parameter files, 25

PCEQ format, 197
PCK files, 186
PCQSEI, 197
PCSUDS, 201
PDAS files to SEISAN, 197
PDASEI, 197
PDE, 187
PDE bulletin file, 183
PDE e-mail, 187
PDENOR, 187
Peak ground acceleration, 262
Peak ground velocity, 262
Period, 114
Perl, 320
Peterson noise model, 128, 255
PFIT, 49
pfit.out, 50
Pg phase, 77
PGA, 262
Phase AMP, 113
Phase mouse key, 134
Phase name, 115, 334
Phase name key, 134
Phase picking, 38, 62, 97, 113
Phase picking mode, 106
Phase reading indicator, 98
Phase weight key, 134
Phase, compare theoretical, 49
Phase, select in data base, 159
Picking phases, use of filters, 106
PINV, 232
PITSA, 199, 201
Pitsa program, 52
Place names, 141
Plot contours, 137
Plotting continuous data, 111
Plotting epicenters, 143
Plotting place names, 141
Plotting response files, 71
Plotting traces, 50
Pn phase, 77
Polarity, 106, 114, 216
Polarity change, 182
Polarity, correct, 182
Polarity, select for, 158
Poles and zeros, 197, 300, 342
Polygon, 282
Polygon, select, 159
Polygon, select in, 57, 140
Postal code , 311
Postscript, 20, 99
Power spectral density, 258
Power spectrum, 125, 127
PR_RESP, 71, 298
Preprocessing of data, 112, 135
Presp, 71
Presp.eps, 302
Print S-file, 52
Print.out file, 90
Printer, 20
Printing, 72
Printing on PC, 23
Probabilistic seismic hazard analysis, 260
Probable explosion, 158
Problem HYP, location, 91
Problem HYP, Moho not found, 87
Problem MULPLT extract, 176
Problem UPD, 165
Problem WAVETOOL, zero output, 176
Problem, backup files in EEV, 48
Problem, Bouchon, 243
Problem, EPIMAP, 143
Problem, file name length, 14
Problem, filenr.lis, 165
Problem, filter, 100
Problem, HYP, 82, 83, 85
Problem, instrument correction, 117
Problem, MULPLT area, 140
Problem, MULPLT spectra, 128
Problem, no location, 81
Problem, PC, 24
Problem, picking amplitude, 114
Problem, q-correction, 127
Problem, reading synthetic phase, 115
Problem, RESP, 298
Problem, rotation, 129
Problem, Rotation and removing response, 118
Problem, select, 161
Problem, UPD, 165
Problem, UPDATE, 164
Problem, VELEST, 288
Problem: Do not find waveform file, 11
Problem: Reading Steim2, 197
Problem: Response file naming, 304
Profile, hypocenters, 140
Profile.num, 142, 143
PSD, 173
PSNSEI, 198

Public Seismic Network, 198

Q, determine by spectral fitting, 127
QNX, 197
QNXSEI, 198
Quality factor, 233, 271, 276
Quality factor, correct for, 255
Quality factor, determine by spectra, 255
Quality factor, determine with coda, 234
Quanterra, 196

Radiation pattern, 271
Rake, 220
Raster file, 148
RDSEED, 199
Rdseed, 178
rdseed, 70
REA directory, 9
REA.LOG, 43
Read S-file, 319
Recover files from the DELET database, 47
Reference station, 50
REFTEK, 192, 201
REG_KEEP_AUTO, 112
Regional attenuation, 271
Register event, 171
Register event, keep phases, 31
register run process, 30
Registration, 98
Registration and preprocessing, 113
Relative spectra, 258
Rename event type, 52
Reorder Hypocenters, 183
REPORT, 127, 166
Report.inp, 167
Report.out, 168
Report_n.out, 168
Reporting agency, 87
RESAMP, 178
Resampling, 178
RESET TEST, 87
Residual weight, 89
Resolution, 135
RESP, 70, 295
Response file, 14, 69, 115, 342
Response in header, 342
Response removal, 62, 258
Response, from where, 116
Response, GSE, 196
Response, show curves, 116
Rg phase velocity, 88
RMS as a function of depth, 52
RMSDEP, 52
Rotated seismograms, 118
RSAM, 176
RSASEI, 198
RT_SEIS, 192
RTU.INI, 198

S phase, 77
S-file, 9, 14, 37, 41, 161
S-file, error, 48
S-file, parameters, 319
S-file, print, 52
S-file, update, 52
S-files, collecting, 162
S-P and L-S differences, 81
SAC, 10, 201
SAC response, 71
SACAUX, 20
SACLIB, 20
SAMBA, 24
Sample rate, 178
SAMPLE_GRAPHICS, 320
SAMPLE_READ_ARC, 320
SAMPLE_READ_CONT, 320
SAMPLE_READ_WAV, 320
Search, 52
Search by day and hour, 47
Searching for text string in nordic files, 161
Searching headers in waveform files, 180
Searching in the data base, 57, 155
SEED, 10, 199
SEED data, time gap, 99
SEED response, 70
SEED, extract data, 178
SEI2PSXY, 198
SEIASC, 178
SEIASC format, 339
SEIBINASC, 320
SEICUT, 179
SEIDEL, 179, 180
Seidim.inc, 20
SEIGMT, 188
SEIM88A, 199
SEIPITSA, 199
SEIS2VIEWER, 153
SEISAF, 199
SEISAN ASCII format, 339
SEISAN continous data, 13

Seisan extension, 20
SEISAN shortcut, 23
SEISAN waveform file, 14
SEISAN waveform files to or from ascii, 178
SEISAN.bash, 19
SEISAN.csh, 19
SEISAN.DEF, 28, 57, 65, 129
Seisan.ini, 24
SEISAN2MSEED, 199
SEISAN_EDITOR., 20
SEISAN_TOP, 19
SEISANPR, 23
SEISARCH, 20
SeiscomP archive data, 12
seisconf, 19
SEISEI, 179
SeisGram, 199
SEISLOG, 197, 198
Seismic design spectra, 262
Seismic hazard analysis, 263
Seismic moment, average, 91
Seismic noise, 128
Seismic risk related programs, 260
Seismic source zones, 260
Seismogram, 99, 100
Seismometer constants, 300
SEISNET, 39, 112
SELECT, 10, 55, 57, 155
Select epicenters in an area, 142
Select events from database, 47
Select for phase, 159
select.inp, 159
Select.out, 155
SELECTC, 161
Selection in polygon, 140
Selection on errors, 158
SELMAP, 188
SELSEI, 180
Serial number, instrument, 189
SESAME, 199
SFORMAT, 320
sformat, 19
Sg phase, 77
SGRSEI, 199
Short user guide, 38
Signal duration, 210
Signal, damped, 320
Signal.out, 119
SILSEI, 201
Simulate drum recording, 99
Single channel files, 179
Single files, 9
Single station location, 78
Single trace mode, 106
Sismalp, 201
SISSEI, 201
Site effects, 262
Size of graphics window, 26
Sn phase, 77
Soil amplification, 255
Sonic boom, 158
Source displacement spectrum, 123
Source function, 270
Source parameters, 123
Source radius, 124
Source time function, 247
Spain, 188, 198
SPEC, 255
spec.out, 124
Spec.par, 256
Spectral analysis, 255
Spectral fitting, 127
Spectral parameters, 135, 168, 211, 255, 277
Spectral parameters, update, 164
Spectral ratio, 255
Spherical harmonic station corrections, 87
SPLIT, 42, 161, 162
Split an event, 48
Sprengnether, 195
SSA, Kinemtrics, 195
SSR, Kinemetrics, 195
STA/LTA, 210
Stand-alone programs, 72
Start and end date, 72
Start depth, 82
Start time, 25
Starting location, 82, 88, 89
STASEI, 188
Station and channel codes, 301
Station azimuth, 118
Station coordinates, 25, 186, 188
Station listing in print.out, 90
Station selection, 186
Station, only 4 characters, 185
STATION0.HYP, 25, 86
STATIS, 168
Statistics, 168, 169
STATSTAT, 170

Steim2, 197
Stress drop, 124, 126, 270
Strike, 220
Subset of database, 72
SUD2ASC, 201
SUDS, 201
SUDS format, 344
SUDSEI, 201
Sun and PC differences, 11, 344
Surface wave, 79
Surface wave analysis, 124, 295
Surface wave dispersion, 123
Swarm, identify, 170
Swarm,seismic, 170
Swarm.out, 170
SYNT, 243
Synthetic seismograms, 240
System response, 69

T and P-axis, 218
T-phase, 88
Terra Technology, 201
TERSEI, 201
Test graphics, 320
Test mouse, 320
Test programs, 319
Test signals, make, 320
Theoretical phases, 49, 115
Three component plotting, 111
Threshold magnitude, 264
Time correction, 180
Time delay, 180
Time gap, 99
Time, uncertain flag in waveform file, 198
Timing error, 100, 106
Timing indicator, 342
TMP, 11
Top directory, 19
Trace plotting, 38, 97, 136, 198
TRACE_PLOT, 136
Travel time error, 88
Trigger, 214
Tsd.out, 180
TSIG, 320
TTLAYER, 250
TTPLOT, 52, 250
ttplot.eps, 251
ttplot.out, 251

Uncertain time, 106
Uncertain timing flag, 189
Unformatted read and write, 343
Units, 125, 299
Unknown type, 158
UPD, 164
UPDATE, 44, 52, 55, 72, 126, 163, 164
Update header line, 80
Update spectral parameters, 164
Update without relocation, 164
Updated S-files, 52, 163
Upper and lower case, 15
Upper bound magnitude, 273
UPPER case, 114
USERCOM, 52
USGS, 188
USGSNOR, 188
Using azimuth phases, 88

VELEST, 284
Velest for Windows, 284
VELMENU, 284
Velocity, 115
Velout.dif, 288
Version, 28
Volcanic, 158
Volcanic event, 112
Volcanic tremor, 290
VOLCANO.DEF, 290
VOLCSTAT, 290, 291
Vp/Vs ratio, 87
Vp/Vs, calculate, 253

W_EMAP, 147
WADATI, 253
Wadati, 53
WAV data base, 11
WAV directory, 10
Waveform conversion programs, 189
Waveform file format, 339
Waveform file name, 180
Waveform file structure, 343
Waveform file, cut, 179
Waveform file, delete, 118
Waveform file, extract from, 119
Waveform file, fix headers, 182
Waveform file, split, 179
Waveform files, 10, 159
Waveform files, join, 171
Waveform files, merge, 119
Waveform.for, 320

Waveform.inc, 320
Waveform_names.out, 159
WAVETOOL, 174
WAVFIX, 171
Wavfix.tim, 180
WAVFULLNAME, 183
WEB options, 161
Weight, 82, 88, 114
WGSSEI, 201
Wildcard, 165
Windows, 67
Windows configuration, 24
Windows NT, 23
Windows95, 23, 37
WKBJ, 240, 242
Wood Anderson, 115, 276
Working directory, 67
WORLD.MAP, 25
Write S-file, 319

X Windows, 19
XCLUST, 313
Xnear and xfar, 87

Y5DUMP, 197
Yearly number of events, 169

Zoom in JSEISAN, 62
Zoom in map, 57
Zoom in MULPLT, 106