



InSite seismic processing software: User manual- Technical Appendices

version 3.20

February 2024



Applied Seismology Consulting Ltd.
34-35 Butcher Row, Shrewsbury SY1 1UW, UK
Tel: +44 (0) 1743 384171
Email: asc-info@appliedseismology.co.uk • Web: appliedseismology.co.uk



VAT No 729 0817 21 • Registered in England and Wales No 3457130

ISO 9001:2015

Applied Seismology Consulting Ltd. is an ISO 9001:2015 certified company and operates a Quality Management System that provides quality assurance of our products and management procedures.

Acknowledgments

The authors would like to acknowledge the individual contributions made by the staff of Applied Seismology Consulting Ltd. to the production of the InSite software and this documentation.

Table of Contents

Acknowledgments.....	ii
Chapter 1 Introduction.....	1
Chapter 2 Import and Export File Formats.....	3
2.1 SEN File.....	4
2.2 CSV Sensor File.....	5
2.3 CSV Time Dependent Sensor Coordinate File.....	6
2.4 The LOG File.....	8
2.5 The RDB File.....	9
2.6 The SRC File.....	11
2.7 The PUN File.....	13
2.8 DOF Format.....	14
2.9 The Very Versatile Format VVF file.....	18
2.7.1 Header Lines.....	19
2.7.2 Calculating the Bit Flag.....	21
2.7.3 Data Lines.....	22
2.7.4 The VVFs Versatility.....	23
2.7.5 Implementation for InSite.....	31
2.10 The ASCII Trace File ATF.....	31
2.11 Reading the Nicolet WFT File.....	33
2.12 The FFT Format File.....	35
2.13 The Report RPT Format File.....	37
2.14 The Trigger Report File TRF.....	37
2.15 Event Parameters File EPF.....	38
2.16 Waveform Pick Results PED Format File.....	41
2.17 File for Mechanism Axes DIP format.....	44
2.18 Column Format Survey File DCF.....	45
2.19 Stereonet Format Survey File SSF.....	47
2.20 Filter File FLT.....	50
2.21 CSV Event Data File.....	53
2.22 CSV Instrument Data File.....	57
2.23 Well Visualiser Object File.....	60
2.24 PFC File Format.....	61
2.25 Waveform Parameters File.....	63
Chapter 3 Source Location Algorithms.....	64
3.1 Velocity Models.....	64
3.1.1 Homogeneous Isotropic.....	64
3.1.2 Time-Dependent Homogeneous Isotropic.....	65
3.1.3 Transversely Isotropic.....	66
3.1.4 Time-Dependent Transversely Isotropic.....	66
3.2 Simplex Routine.....	66
3.2.1 Methodology.....	67
3.2.2 Specific Configuration Parameters.....	67
3.3 Geiger Routine.....	68
3.3.1 Methodology.....	68

3.3.2	Error Ellipsoids	69
3.3.3	Specific Configuration Parameters.....	70
3.4	Collapsing Grid Search Routine	71
3.5	Analysis of Source Location Uncertainties.....	73
3.5.1	Residuals and the RMS Residual	73
3.5.2	The RMS Error	73
3.6	Post Analysis of Source Locations.....	73
3.7	Clustering Analysis Techniques	74
3.8	Relative Location Routine.....	74
3.8.1	Methodology.....	75
3.8.2	Specific Configuration Parameters.....	77
3.9	Advanced Source Location Algorithms.....	78
3.9.1	Travel Time Modelling Wavefront Construction.....	78
3.9.2	Tian Raytrace.....	80
3.9.3	VTI Tian Raytrace.....	83
3.9.4	Fast Sweeping Method.....	85
3.9.5	Source Scan Method	88
Chapter 4	Source Parameter Algorithms	91
4.1	Data Initialisation for Automatic Source Parameters	91
4.2	Instrument Magnitude	93
4.3	Frequency-Source-Model Parameters Calculated in the Time Domain	93
4.4	Calculation of Source Parameters	94
Chapter 5	Source Mechanism Algorithms.....	97
5.1	Inversion	97
5.2	Decomposition	99
5.3	Error Analysis.....	99
5.4	Fault Plane Solution from First Motion Polarities	101
Chapter 6	Cross-Correlation Algorithms	103
6.1	Introduction to the CCR Technique	103
6.2	Description of the CCR procedure	104
6.3	Data Input.....	106
6.3.1	Trace Data	106
6.3.2	Source and Receiver Parameters	107
6.3.3	Configuration Parameters.....	107
6.3.4	Configuration Flags.....	107
6.4	Construction of the CCR Data Arrays.....	108
6.4.1	Window construction.....	110
6.4.2	Waveform signal processing	112
6.4.3	Filling the CCR Data Array	114
6.5	CCR Processing	116
6.6	Velocity Calculations.....	117
6.7	Data Output.....	118
6.8	Testing of the CCR Routines	119
Chapter 7	Advanced Waveform Processing	121
7.1	Waveform Rotation	121
7.1.1	Coordinate System Geometry	121
7.1.2	The Montalbetti-Kanasewich Method	125
7.1.3	Single station Location	125
7.2	Spectral Processing Algorithms	126

7.2.1	Taper Types	126
7.2.2	Power Spectrum Estimation Methods	127
7.2.3	Polarisogram Algorithm	129
7.3	Spectral Autopicking.....	131
7.4	Waveform Parameters	132
7.5	Confidence Number	133
7.6	Akaike Information Criterion AIC.....	135
7.6.1	AIC Simple	135
References	136

Chapter 1 Introduction

This document provides background information associated with file formats and algorithms provided in the InSite Seismic Processor. It should be used in conjunction with the InSite Operations Manual. The information given is intended as a guide only. Applied Seismology Consulting Ltd. (ASC) reserves the right to amend the functions at any time. Information on function updates can be obtained from asc-support@appliedseismology.co.uk.

To further understand the algorithms the user should refer to the references provided or contact ASC. We welcome any comments and feedback on the algorithms and procedures used and will assess any suggestions provided by users to improve the functionality.

Chapter 2 Import and Export File Formats

[SEN File](#)

[The LOG File](#)

[The RDB File](#)

[The SRC File](#)

[The PUN File](#)

[DOF Format](#)

[The Very Versatile Format VVF file](#)

[The ASCII Trace File ATF](#)

[Reading the Nicolet WFT File](#)

[The FFT Format File](#)

[The Report RPT Format File](#)

[The Trigger Report File TRF](#)

[Event Parameters File EPF](#)

[Waveform Pick Results PED Format File](#)

[File for Mechanism Axes DIP format](#)

[Column Format Survey File DCF](#)

[Stereonet Format Survey File SSF](#)

[Filter File FLT](#)

[CSV Event Data File](#)

[CSV Instrument Data File](#)

[CSV Sensor File](#)

[Well Visualiser Object File](#)

[PFC File Format](#)

[Waveform Parameters File](#)

10	Full-scale voltage on acquisition system	Float
11-12	Low and High frequency bandwidth of the sensors for filtering	Float
13-15	Sensor orientation in a NED co-ordinate system	Float
16	Sensor polarisation - +ve (1) or -ve (-1)	Integer
17	P-wave station correction in seconds	Float
18	S-wave station correction in seconds	Float
19	Monitoring array ID	String
20	Local Instrument Number	Integer
21	Local Channel Number	Integer

Note that for InSite the first two columns must read as e.g. Channel 01.

This can differ from SEN file formats used by ESG with their acquisition systems.

2.2 CSV Sensor File

Sensor array coordinates and properties can be exported and imported into InSite using a Comma Separated Values (CSV) File. This format offers greater flexibility over the SEN file ([SEN File](#)). An example of the CSV sensor file is shown in Figure 2.2.

```

1 Instrument_Number,Instrument_Label,Channel_Number,Channel_Label,North,East,Down,On,Type,Axis_Number,Gain,Sensitivity,Vm
ax,LowFreq,HighFreq,Orientation_N,Orientation_E,Orientation_D,Motion,P_Station_Correction,S_Station_Correction,Owner_Ar
ray,Array_Instrument_Number,Array_Channel_Number
2 1,S001,1,S001,0.000000,0.000000,0.000000,1,5,0,1.000000,1.000000,1.000000,0.000000,0.000000,0.000000,0.000000,
1,0.000000,0.000000,,0,0
3 2,S002,2,S002,0.000000,0.000000,0.000000,1,5,0,1.000000,1.000000,1.000000,0.000000,0.000000,0.000000,0.000000,
1,0.000000,0.000000,,0,0
4 3,S003,3,S003,0.000000,0.000000,0.000000,1,5,0,1.000000,1.000000,1.000000,0.000000,0.000000,0.000000,0.000000,
1,0.000000,0.000000,,0,0
5 4,S004,4,S004,0.000000,0.000000,0.000000,1,5,0,1.000000,1.000000,1.000000,0.000000,0.000000,0.000000,0.000000,
1,0.000000,0.000000,,0,0
6

```

Figure 2.2: CSV sensor file format

The format consists of 24 columns, which can be arranged in any order. The first line is the file header, and each subsequent line contains sensor data for a single channel. The columns are as follows:

Table 2.2: Description of data columns in the CSV sensor file

Column	Data Type	Format
1	Instrument Number	Integer
2	Instrument Label	String
3	Channel Number	Integer
4	Channel Label	String
5-7	North, East, Down components of the sensor location	Float
8	Status – on (1) or off (0)	Integer

9	Sensor type,where: -0, 3, 4 Velocity -1, 2 Acceleration -5 Volts	Integer
10	Axis number -0 X Axis -1 Y Axis -2 Z Axis	Integer
11	Gain Multiplier	Float
12	Receiver Sensitivity. Units as follows: Acceleration - volts/g Velocity – volts/m.s ⁻¹	Float
13	Full-scale voltage on acquisition system	Float
14-15	Low and High frequency bandwidth of the sensors for filtering	Float
16-18	Sensor orientation in a NED co-ordinate system	Float
19	Sensor polarisation - +ve (1) or –ve (-1)	Integer
20	P-wave station correction in seconds	Float
21	S-wave station correction in seconds	Float
22	Monitoring array ID	String
23	Local Instrument Number	Integer
24	Local Channel Number	Integer

2.3 CSV Time Dependent Sensor Coordinate File

InSite can incorporate changes in position of the different sensors in an array such as those induced due to the deformation a monitored rock sample. The time history of the displacement of the different sensors is input into InSite through an ASCII Comma Separated Values (CSV) file. Sensor positions for events are calculated according to their time stamp using the starting array and the correction in the time dependent sensor coordinate file. An example of the CSV time dependent sensor coordinate file is shown in Figure 2.3.

```

1 Date,Time,second (s),S001 North,S001 East,S001 Down,S002 North,S002 East,S002 Down,S003 N
2 29/03/2019,09:27:49,0,0,0,-0.00626,0,0,-0.00626,0,0,-0.00626,0,0,-0.00626
3 29/03/2019,09:28:00,0,0,0,-0.01872,0,0,-0.01872,0,0,-0.01872,0,0,-0.01872
4 29/03/2019,09:28:11,0,0,0,-0.03064,0,0,-0.03064,0,0,-0.03064,0,0,-0.03064
5 29/03/2019,09:28:22,0,0,0,-0.03948,0,0,-0.03948,0,0,-0.03948,0,0,-0.03948
6 29/03/2019,09:28:33,0,0,0,-0.04607,0,0,-0.04607,0,0,-0.04607,0,0,-0.04607
7 29/03/2019,09:28:44,0,0,0,-0.05028,0,0,-0.05028,0,0,-0.05028,0,0,-0.05028
8 29/03/2019,09:28:55,0,0,0,-0.05197,0,0,-0.05197,0,0,-0.05197,0,0,-0.05197
9 29/03/2019,09:29:06,0,0,0,-0.05418,0,0,-0.05418,0,0,-0.05418,0,0,-0.05418
10 29/03/2019,09:29:17,0,0,0,-0.05319,0,0,-0.05319,0,0,-0.05319,0,0,-0.05319
11 29/03/2019,09:29:27,0,0,0,-0.04843,0,0,-0.04843,0,0,-0.04843,0,0,-0.04843
12 29/03/2019,09:29:38,0,0,0,-0.0416,0,0,-0.0416,0,0,-0.0416,0,0,-0.0416
13 29/03/2019,09:29:49,0,0,0,-0.03681,0,0,-0.03681,0,0,-0.03681,0,0,-0.03681
14 29/03/2019,09:30:00,0,0,0,-0.03276,0,0,-0.03276,0,0,-0.03276,0,0,-0.03276
15 29/03/2019,09:30:11,0,0,0,-0.03677,0,0,-0.03677,0,0,-0.03677,0,0,-0.03677
16 29/03/2019,09:30:22,0,0,0,-0.03923,0,0,-0.03923,0,0,-0.03923,0,0,-0.03923
17 29/03/2019,09:30:33,0,0,0,-0.0394,0,0,-0.0394,0,0,-0.0394,0,0,-0.0394
18 29/03/2019,09:30:44,0,0,0,-0.03963,0,0,-0.03963,0,0,-0.03963,0,0,-0.03963
19 29/03/2019,09:30:55,0,0,0,-0.03903,0,0,-0.03903,0,0,-0.03903,0,0,-0.03903
20 29/03/2019,09:31:06,0,0,0,-0.03727,0,0,-0.03727,0,0,-0.03727,0,0,-0.03727
21 29/03/2019,09:31:17,0,0,0,-0.03845,0,0,-0.03845,0,0,-0.03845,0,0,-0.03845
22 29/03/2019,09:31:28,0,0,0,-0.0379,0,0,-0.0379,0,0,-0.0379,0,0,-0.0379
23 29/03/2019,09:31:39,0,0,0,-0.03702,0,0,-0.03702,0,0,-0.03702,0,0,-0.03702

```

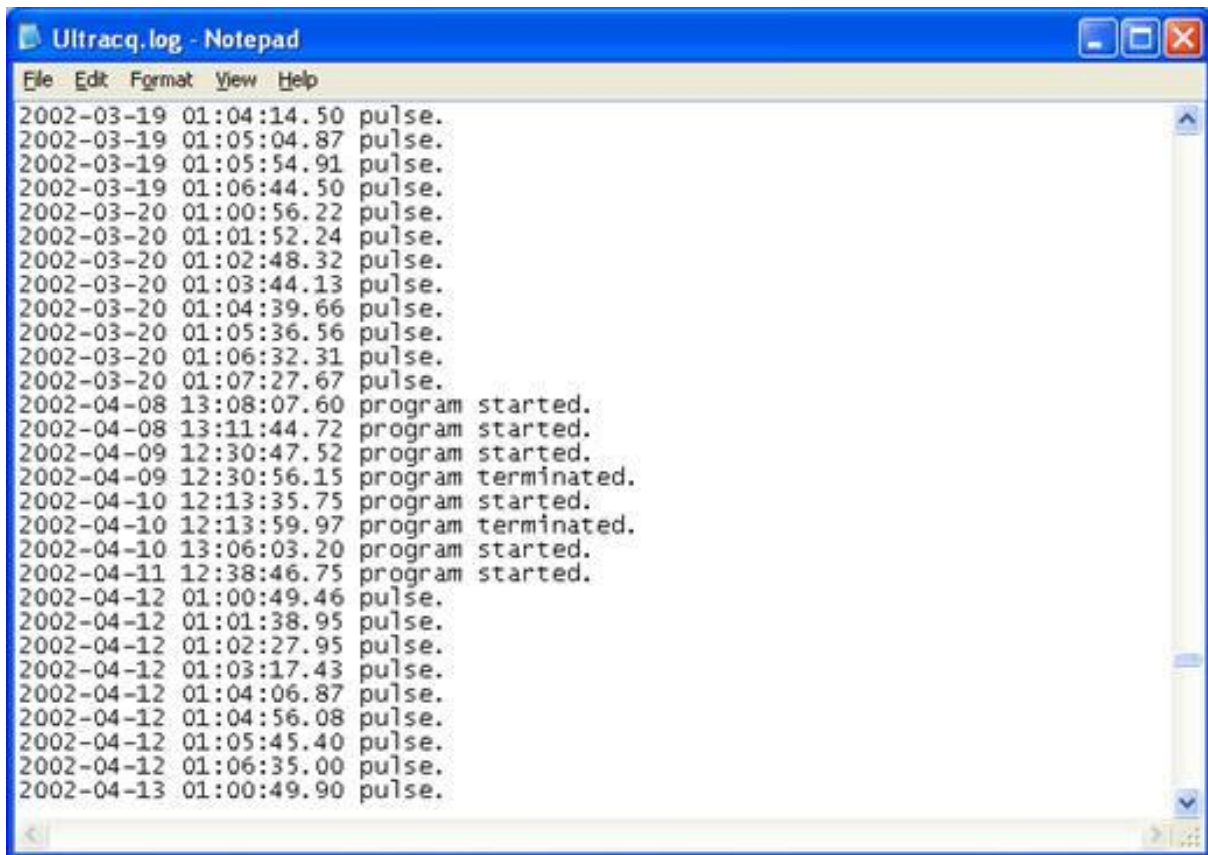
Figure 2.3: CSV time dependent sensor coordinate file example

The format consists of one header line followed by lines each one corresponding to a time step where sensor displacements have been measured. The number of columns is dependent on the number of sensors ($3 + 3 \times \text{Number of sensors}$). Displacements are not cumulative and should be entered as discrete displacements in the relevant time interval.

Column Header	Description	Format
Date	Date in day/month/year format	DD/MM/YYYY
Time	Time in hour:minute:second format	hh:mm:ss
Second	Decimal part of the second	Float
S001 North	Sensor 1 displacement in the North direction	Float
S001 East	Sensor 1 displacement in the East direction	Float
S001 Down	Sensor 1 displacement in the Down direction	Float
.....		
S00N North	Sensor N displacement in the North direction	Float
S00N East	Sensor N displacement in the East direction	Float
S00N Down	Sensor N displacement in the Down direction	Float

2.4 The LOG File

A LOG file is a text file that contains additional information regarding data acquisition (Figure 2.4). Each line logs a command performed by the acquisition software and contains the date and time of the command followed by a brief description.



```
Ultracq.log - Notepad
File Edit Format View Help
2002-03-19 01:04:14.50 pulse.
2002-03-19 01:05:04.87 pulse.
2002-03-19 01:05:54.91 pulse.
2002-03-19 01:06:44.50 pulse.
2002-03-20 01:00:56.22 pulse.
2002-03-20 01:01:52.24 pulse.
2002-03-20 01:02:48.32 pulse.
2002-03-20 01:03:44.13 pulse.
2002-03-20 01:04:39.66 pulse.
2002-03-20 01:05:36.56 pulse.
2002-03-20 01:06:32.31 pulse.
2002-03-20 01:07:27.67 pulse.
2002-04-08 13:08:07.60 program started.
2002-04-08 13:11:44.72 program started.
2002-04-09 12:30:47.52 program started.
2002-04-09 12:30:56.15 program terminated.
2002-04-10 12:13:35.75 program started.
2002-04-10 12:13:59.97 program terminated.
2002-04-10 13:06:03.20 program started.
2002-04-11 12:38:46.75 program started.
2002-04-12 01:00:49.46 pulse.
2002-04-12 01:01:38.95 pulse.
2002-04-12 01:02:27.95 pulse.
2002-04-12 01:03:17.43 pulse.
2002-04-12 01:04:06.87 pulse.
2002-04-12 01:04:56.08 pulse.
2002-04-12 01:05:45.40 pulse.
2002-04-12 01:06:35.00 pulse.
2002-04-13 01:00:49.90 pulse.
```

Figure 2.4: Example LOG File

InSite recognises some commands contained in the LOG file and adds information to events that correspond in time (within 1 second before the command). Table 2.3 contains valid commands. Commands that are not recognised are ignored.

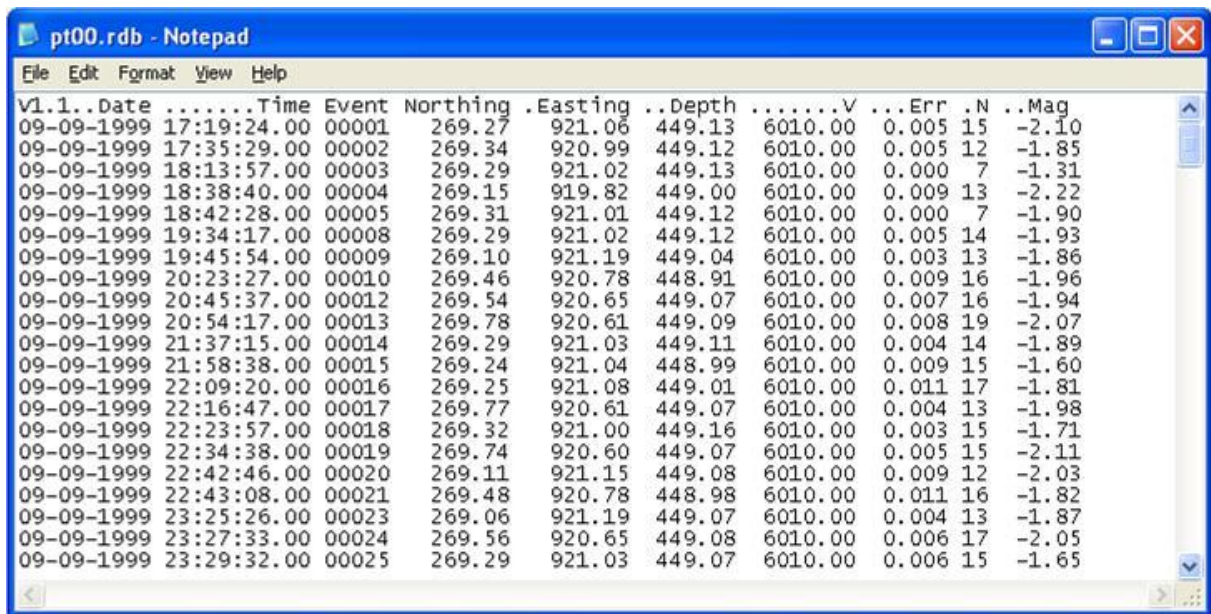
Table 2.3: Valid commands contained in a LOG file that can be read for InSite

Log flag	Description
pulse	Corresponding event is a shot
shot	"
blast	Corresponding event is noise

2.5 The RDB File

An RDB file is an ASCII text file that contains event location data in space-delimited column format. Each line is a single event. The user will be prompted for the location distance units used in the file. The choice includes kilometres, metres, centimetres or millimetres. The columns are imported into InSite as follows (Figure 2.5).

- Date (MM-DD-YYYY)
- Time
- Event Number (the first two characters can contain a mine stamp e.g. TS123)
- Location Northing
- Location Easting
- Location Depth
- Material Velocity (not imported)
- RMS Location Error
- Number of arrival times used in location
- Location Magnitude



The screenshot shows a Notepad window with the following data:

V1.1..Date	Time	Event	Northing	Easting	Depth	V	Err	N	Mag
09-09-1999	17:19:24.00	00001	269.27	921.06	449.13	6010.00	0.005	15	-2.10
09-09-1999	17:35:29.00	00002	269.34	920.99	449.12	6010.00	0.005	12	-1.85
09-09-1999	18:13:57.00	00003	269.29	921.02	449.13	6010.00	0.000	7	-1.31
09-09-1999	18:38:40.00	00004	269.15	919.82	449.00	6010.00	0.009	13	-2.22
09-09-1999	18:42:28.00	00005	269.31	921.01	449.12	6010.00	0.000	7	-1.90
09-09-1999	19:34:17.00	00008	269.29	921.02	449.12	6010.00	0.005	14	-1.93
09-09-1999	19:45:54.00	00009	269.10	921.19	449.04	6010.00	0.003	13	-1.86
09-09-1999	20:23:27.00	00010	269.46	920.78	448.91	6010.00	0.009	16	-1.96
09-09-1999	20:45:37.00	00012	269.54	920.65	449.07	6010.00	0.007	16	-1.94
09-09-1999	20:54:17.00	00013	269.78	920.61	449.09	6010.00	0.008	19	-2.07
09-09-1999	21:37:15.00	00014	269.29	921.03	449.11	6010.00	0.004	14	-1.89
09-09-1999	21:58:38.00	00015	269.24	921.04	448.99	6010.00	0.009	15	-1.60
09-09-1999	22:09:20.00	00016	269.25	921.08	449.01	6010.00	0.011	17	-1.81
09-09-1999	22:16:47.00	00017	269.77	920.61	449.07	6010.00	0.004	13	-1.98
09-09-1999	22:23:57.00	00018	269.32	921.00	449.16	6010.00	0.003	15	-1.71
09-09-1999	22:34:38.00	00019	269.74	920.60	449.07	6010.00	0.005	15	-2.11
09-09-1999	22:42:46.00	00020	269.11	921.15	449.08	6010.00	0.009	12	-2.03
09-09-1999	22:43:08.00	00021	269.48	920.78	448.98	6010.00	0.011	16	-1.82
09-09-1999	23:25:26.00	00023	269.06	921.19	449.07	6010.00	0.004	13	-1.87
09-09-1999	23:27:33.00	00024	269.56	920.65	449.08	6010.00	0.006	17	-2.05
09-09-1999	23:29:32.00	00025	269.29	921.03	449.07	6010.00	0.006	15	-1.65

Figure 2.5: Example RDB File

2.6 The SRC File

A SRC file is an ASCII text file that contains event location and source parameter data in space-delimited column format. Each line is a single event. The user will be prompted for the location distance units used in the file. The choice includes kilometres, metres, centimetres or millimetres. The columns are imported into InSite as follows (Figure 2.6).

- Date (MM-DD-YYYY)
- Time
- Floating-point time (not imported)
- Mine-stamp Flag (imported to event label)
- Event Number
- Automatic Event Type Flag (not imported)
- Percentage confidence in Event Type Flag (not imported)
- Location Northing
- Location Easting
- Location Depth
- Material Velocity (not imported)
- RMS Location Error
- Number of arrival times used in location
- Number of uniaxials used (not imported)
- Flag (not imported)
- Location Magnitude
- Number of triaxials used (not imported)
- Flag (not imported)
- Instrument Magnitude
- Moment Magnitude

- Seismic Moment
- Energy Release
- S:P wave energy release ratio
- Source Radius
- Asperity Radius (not imported)
- Static Stress Drop
- Apparent Stress
- Dynamic Stress Drop (not imported)
- Maximum displacement (not imported)
- Peak velocity (not imported)
- Peak acceleration (not imported)

Time	SerialTime	Id	Ev#	T	Conf	Northing	Easting	Depth	Velocity	Error	Ns	Nu	uSt	uMag	Nt	tSt	tMag	Mc
09-25-1991 15:21:45	33504.640105	MB 148	e	0.00	451.53	776.37	125.12	5970.00	.25	16	0	num	-9.99	12	utm	2.01	-1.6	
09-25-1991 15:24:05	33504.641733	MB 149	e	0.00	451.38	776.45	125.00	5970.00	.22	16	0	num	-9.99	14	utm	1.85	-2.1	
09-25-1991 15:24:14	33504.641833	MB 150	e	0.00	451.29	776.17	125.18	5970.00	.27	16	0	num	-9.99	12	utm	1.87	-1.9	
09-25-1991 15:24:16	33504.641856	MB 151	e	0.00	451.28	776.27	125.28	5970.00	.31	15	0	num	-9.99	14	utm	1.56	-2.3	
09-25-1991 15:24:29	33504.642003	MB 152	e	0.00	451.30	776.00	124.97	5970.00	.24	16	0	num	-9.99	14	utm	2.03	-1.6	
09-25-1991 15:24:32	33504.642048	MB 153	e	0.00	451.32	776.41	125.07	5970.00	.27	16	0	num	-9.99	15	utm	1.61	-2.3	
09-25-1991 15:24:58	33504.642341	MB 154	e	0.00	451.22	776.29	125.06	5970.00	.28	16	0	num	-9.99	15	utm	1.56	-2.3	
09-25-1991 15:25:30	33504.642720	MB 157	e	0.00	451.51	776.72	125.06	5970.00	.59	14	0	num	-9.99	13	utm	1.53	-2.4	
09-25-1991 15:28:37	33504.644873	MB 158	e	0.00	451.23	776.12	125.10	5970.00	.30	16	0	num	-9.99	13	utm	1.89	-2.0	
09-25-1991 15:37:08	33504.650788	MB 159	e	0.00	451.13	775.99	125.35	5970.00	.25	14	0	num	-9.99	12	utm	1.68	-2.3	
09-25-1991 15:49:35	33504.659440	MB 160	e	0.00	451.15	776.01	125.21	5970.00	.16	16	0	num	-9.99	13	utm	1.97	-1.7	
09-25-1991 15:51:56	33504.661075	MB 2	e	0.00	451.24	776.11	124.96	5970.00	.24	15	0	num	-9.99	15	utm	1.53	-2.3	
09-25-1991 15:54:50	33504.663082	MB 3	e	0.00	451.23	776.11	124.96	5970.00	.27	15	0	num	-9.99	14	utm	1.52	-2.3	
09-25-1991 15:58:40	33504.665746	MB 4	e	0.00	452.02	775.61	125.10	5970.00	.66	15	0	num	-9.99	15	utm	1.35	-2.4	
09-25-1991 15:58:59	33504.665971	MB 5	e	0.00	451.05	775.98	125.20	5970.00	.30	15	0	num	-9.99	14	utm	1.77	-2.0	
09-25-1991 16:00:52	33504.667277	MB 6	e	0.00	451.47	776.15	124.93	5970.00	.31	16	0	num	-9.99	13	utm	2.03	-2.0	
09-25-1991 16:01:28	33504.667692	MB 7	e	0.00	451.51	776.20	124.99	5970.00	.25	16	0	num	-9.99	13	utm	2.02	-2.1	
09-25-1991 16:01:50	33504.667950	MB 8	e	0.00	451.06	776.01	125.13	5970.00	.28	16	0	num	-9.99	14	utm	1.59	-2.0	

Figure 2.6: Example SRC File

2.7 The PUN File

A PUN file is an ASCII text file that contains event location data in space-delimited column format and is output by the HYPO71 program. Each line is a single event.

The user will be prompted for a conversion between planetary spherical coordinates and Cartesian coordinates in kilometres.

The columns are imported into InSite as follows:

- Date (YYMMDD)
- Time
- Latitude North
- Longitude West
- Depth (kilometres)
- Location Magnitude
- Number of arrivals used in location
- Event Data (not imported)
- Event Data (not imported)
- RMS Location Error
- Event Data (not imported)
- Event Data (not imported)
- Event Data (not imported)

2.8 DOF Format

This basic file format is specific to InSite. Drawing objects (i.e. additional objects that the user wishes to appear on the screen) can be imported into and exported from the Location Visualiser by means of a Drawing Object File (DOF). The following shows an example of the DOF format. It is a standard ASCII text file that can be created in any suitable editor (e.g. MS Notepad).

```
InSite Drawing Object File v2.07
[Borehole]
Name=B1
BoreholeType=INST
DistUnits=10.0000
EnableDateTime=05-09-1985 00:00:00
DisableDateTime=15-05-2002 00:00:00
[Data]
200.0 100.0 -100.0
200.0 -50.0 -75.0
175.0 -100.0 -50.0
[Borehole]
[Data]
100.0 100.0 100.0
100.0 -50.0 75.0
75.0 -100.0 50.0
[HotPlane]
Name=HotPlane
DistUnits=1.0000
Azimuth=120.094
Dip= 80.017
ColourRed=0
ColourGreen=255
ColourBlue=0
EnableDateTime=01-01-0100 00:00:00
DisableDateTime=01-01-3000 00:00:00
[Data]
12.783 5.2827 -77.883
-6.0282 37.743 -77.883
-10.828 34.961 -46.367
7.9837 2.5012 -46.367
[BoreholeLog]
Name=A2
BoreholeType=WINJ
DistUnits=1.000
EnableDateTime=02-10-1985 15:00:00
DisableDateTime=09-10-1985 08:00:00
[Data]
400.0 190.0 -100.0 4000
320.0 -250.0 -75.0 4100
```

75.0 100.0 -50.0 4150

[Complex]

Name=MyComplex

DistUnits=1.0000

ColourLinesRed=255

ColourLinesGreen=0

ColourLinesBlue=0

EnableDateTime=01-01-0100 00:00:00

DisableDateTime=01-01-3000 00:00:00

[Line]

12.0 12.0 12.0

23.0 32.0 11.0

78.0 64.0 45.0

26.0 12.0 32.0

19.0 28.0 77.0

[Line]

100.0 100.0 100.0

222.0 38.0 116.0

[3DFace]

6.2 8.7 11.0

3.5 6.3 9.1

5.7 8.1 4.9

[3DSurface]

0 0 0

0 1 0

1 0 0

1 1 0

0.5 0.5 1

[Sensor]

Mode=Receiver

Label=S01

On=1

DistUnits=1

ChannelNumber=1

CasingNumber=1

Axis=0

DistUnits=1.000

```
ColourRed=255
ColourGreen=255
ColourBlue=0
SymbolType=Cross
Northing=100.0
Easting=100.0
Depth=100.0
EnableDateTime=01-01-0100 00:00:00
DisableDateTime=01-01-3000 00:00:00
VMax=100
Gain=1
Sensitivity=100
LowFreq=10
HighFreq=10000
Diameter=0
OrientationN=0
OrientationE=0
OrientationD=-1
Motion=-1
```

The first line is a header description line that can contain anything. The rest of the file then consists of a set of Object Data Blocks. Each block begins with a line describing the type of object. Drawing objects that can be imported include the following.

```
[Borehole]
[HotPlane]
[BoreholeLog]
[Complex]
[Sensor]
```

Following this is a set of configuration parameters distinguished using a WINDOWS INI format.

```
Name=B1
```

The top [Borehole] object block shown above contains all of the configuration parameters valid for a Borehole object. These can be in any order and some, or all, of the parameters can be omitted, as shown in the second [Borehole] object block. In this case, default values are used when creating the object.

Following the configuration parameters there must then be a set of North, East, Down coordinates. For a [Borehole] or [HotPlane] object these are identified by a [Data] statement. The three numbers can be separated by either a space or a tab. For a Borehole object there can be as many coordinates as required. For a HotPlane there must only be four coordinates identifying the four corners of the plane.

The [BoreholeLog] object block is effectively the same as the Borehole object but contains a fourth column.

These values plot as a log line down the length of the borehole.

A [Complex] object can consist of as many polylines and triangles as the user wishes. For a [Complex] object the coordinates are identified by a [Line] or [3DFace] statement (there is no single [Data] identifier). The lines

containing the coordinates then follow each identifier. A

[Line] can contain as many coordinates as the user wishes. A [3DFace] must only contain 3 coordinates being the apexes of a triangle.

A [3DSurface] object is used to represent 3D terrains within the Location Visualiser. It is used for surface interpolation, i.e. estimating the height at a point given nearby sample heights. InSite uses the 2D Delaunay Triangulation algorithm to perform this interpolation. This involves creating, from the sample points, a set of non-overlapping triangularly bounded facets, where the vertices of the triangles are the input sample points. The triangulation is unique (independent of the order in which the sample points are ordered) for all but trivial cases. One limitation of the 2D Delaunay triangulation algorithm is that it will only work when there are unique x y pairs. This means that only non-overlapping surfaces can be rendered within InSite using this technique. The coordinates are entered in exactly the same way as for the [3DFace] object, with one x y z coordinate per line. The coordinates in the example above generate the surface of a pyramid.

A [Sensor] object is used to add an instrument within the Location Visualiser that is not included in the default receiver array. The fields that can be given for a sensor are shown in the example above, though a valid location is the only required field.

Each data line in the file corresponds to an arbitrary object. This object consists of a number of data blocks (one or more). The possible data blocks are described in

Table 2.4 with the combination of blocks in the object defining what that object is. Table 2.6 contains the possible object combinations for the PFC 3D VISUALISER.

Each data line is split into a series of data columns. The columns appearing in a section depend on which data blocks are present and whether that data is available for those objects.

[Header Lines](#)

[Calculating the Bit Flag](#)

[Data Lines](#)

[The VVFs Versatility](#)

[Implementation for InSite](#)

2.7.1 Header Lines

The first header line has the following example format:

7,5

- The first integer is a bit flag identifying the data blocks that exist in each object in the section.
- The second integer is the number of objects in the section.

The second header line has the following example format:

```
I6,S12,S10,F8.2,F8.2,F8.2,F8.4,F8.4
```

This is a format statement with the comma-delimited character fields corresponding to the format of the data columns in the section.

- I6 denotes an integer of 6 characters long
- S12 denotes a string of 12 characters long
- F8.2 denotes a floating-point number; 8 characters in length with 2 decimal places (note that the decimal point is included in the character count)

The third header line has the following example format:

```
EVENT_LABEL,          EVENT_DATE,EVENT_TIME,EVENT_LOC_NORTH,EVENT_LOC_EAST,EVENT_LOC  
_DOWN,EVENT_LOC_ERROR,EVENT_PARAM_LOCMAG
```

The comma-delimited character strings (e.g. EVENT_LOC_NORTH) denote what parameter resides in each column. The available parameters are described in Table 2.7. This format can also be replaced by a list of integer codes, such as the following (also see Table 2.7); 0,1,2,101,102,103,104,201

Table 2.4: Bit identifiers for available block types included in an object

Bit	Block Type	enum declaration
0	Date/Time stamp	DATE_AND_TIME
1	Location Data	EVENT_LOCATION
2	Source Parameters	EVENT_SOURCE_PARAMS
3	Error Ellipsoid Data	EVENT_ERROR_ELLIPSE
4	Source Mechanism Data	EVENT_SOURCE_MECH
5	PFC Ball	PFC_BALL
6	PFC Velocity	PFC_VELOCITY
7	PFC Displacement	PFC_DISPLACEMENT
8	PFC Crack	PFC_CRACK
9	Reserved for expansion?	

2.7.2 Calculating the Bit Flag

The bit flag is calculated in the following manner, where A, B, C, D are example combinations of data blocks.

Table 2.5: Using the bit identifiers to calculate the bit flag value

BIT ->	12	11	10	9	8	7	6	5	4	3	2	1	0	-
Integer value ->	-	-	-	-	256	128	64	32	16	8	4	2	1	Flag Value
A	0	0	0	0	1	0	0	0	0	0	0	0	0	256
B	0	0	0	0	0	1	0	1	0	0	0	0	0	160 (128+32)
C	0	0	0	0	1	0	0	0	1	0	0	0	0	272 (256+16)
D	0	0	0	0	0	0	0	0	0	1	0	1	1	11 (8+2+1)

A – PFC Crack Only

B – PFC Ball + PFC Displacement

C – PFC Crack + Source mechanism

D – Event time, location and error ellipsoid

2.7.3 Data Lines

The data line has the following format:

```
1 08-19-1997 04:00:23 401.49 715.57 128.07 0.0424 -1.2496
```

The characters in each column must be right justified and the column formatting of the data lines must exactly match the information contained in the three header lines.

Table 2.6: Object combinations for the PFC 3D Visualiser

Object Combination	Bit flag value
DATE_AND_TIME & EVENT_LOCATION & EVENT_SOURCE_PARAMS	7
DATE_AND_TIME & EVENT_LOCATION & EVENT_SOURCE_PARAMS & EVENT_ERR OR_ELLIPSE	15

Object Combination	Bit flag value
DATE_AND_TIME & EVENT_LOCATION & EVENT_SOURCE_PARAMS & EVENT_SOURCE_MECH	23
DATE_AND_TIME & EVENT_LOCATION & EVENT_SOURCE_PARAMS & EVENT_ERROR_ELLIPSE & EVENT_SOURCE_MECH	31
PFC_BALL	32
PFC_BALL & PFC_VELOCITY	96
PFC_BALL & PFC_DISPLACEMENT	160
PFC_BALL & PFC_DISPLACEMENT & PFC_VELOCITY	224
PFC_CRACK	256
PFC_CRACK & EVENT_SOURCE_MECH	272

2.7.4 The VVFs Versatility

The file format is very versatile for the following reasons.

1. The column formatting is contained in the section header allowing widely varying scales of values to be used.
2. The columns contained in the file are indexed in the section header allowing different sections to contain different types of data. The indexing allows the safe omission and inclusion of data columns.
3. The bit flag indicates the presence of data blocks in the file allowing easier manipulation of the data in the software.
4. The combination of bit flag and column indexing allows software backward-forward compatibility as data parameters can be safely added or removed from the export/import process.

Table 2.7: Valid data parameters for a VVF

Block	Parameter Name	enum declaration	Data Code	Format
Date/Time Stamp	Optional Integer Label	EVENT_LABEL	0	Integer
	Date	EVENT_DATE	1	String: MM-DD-YYYY
	Time	EVENT_TIME	2	String: HH:MM:SS
	Number	EVENT_NUMBER	3	Integer
	Identification String	EVENT_IDSTRING	4	String

Event Location	North	EVENT_LOC_NORTH	101	Float
	East	EVENT_LOC_EAST	102	Float
	Down	EVENT_LOC_DOWN	103	Float
	Error	EVENT_LOC_ERROR	104	Float
	Distance Units	EVENT_LOC_UNITS	105	Float
		EVENT_LOC_PICKS	106	Int
	T0 Time	EVENT_LOC_T0TIME	107	Float
	Independent Sensors	EVENT_LOC_INDEPENDANTS	108	Int
Source Parameters	Location Magnitude	EVENT_PARAM_LOCATIONMAG	201	Float
	Instrument Magnitude	EVENT_PARAM_INSTRMAG	202	Float
	Seismic Moment	EVENT_PARAM_MOMENT	203	Float

Block	Parameter Name	enum declaration	Data Code	Format
	Moment Magnitude	EVENT_PARAM_MOMENTMAG	204	Float
	Radiated Energy	EVENT_PARAM_RADENERGY	205	Float
	S to P Energy Ratio	EVENT_PARAM_STOPENERGY	206	Float
	Source Radius	EVENT_PARAM_SOURCERADIUS	207	Float
	Apparent Stress	EVENT_PARAM_APPSTRESS	208	Float
	Static Stress Drop	EVENT_PARAM_STATICSTRESSDROP	209	Float
	PWave Corner Freq	EVENT_PARAM_PCORNERFREQ	210	Float
	SWave Corner Freq	EVENT_PARAM_SCORNERFREQ	211	Float
Error Ellipsoid	A Axis Length	EVENT_ELLIPSE_ALENGTH	301	Float
	Ai	EVENT_ELLIPSE_AI	302	Float
	Aj	EVENT_ELLIPSE_AJ	303	Float
	Ak	EVENT_ELLIPSE_AK	304	Float
	B Axis Length	EVENT_ELLIPSE_BLENGTH	305	Float
	Bi	EVENT_ELLIPSE_BI	306	Float
	Bj	EVENT_ELLIPSE_BJ	307	Float
	Bk	EVENT_ELLIPSE_BK	308	Float
	C Axis Length	EVENT_ELLIPSE_CLENGTH	309	Float
	Ci	EVENT_ELLIPSE_CI	310	Float
	Cj	EVENT_ELLIPSE_CJ	311	Float
	Ck	EVENT_ELLIPSE_CK	312	Float
Source Mechanism	P EigenValue	EVENT_MECH_PEIG	401	Float
	Pi	EVENT_MECH_PI	402	Float
	Pj	EVENT_MECH_PJ	403	Float

	Pk	EVENT_MECH_PK	404	Float
	B EigenValue	EVENT_MECH_BEIG	405	Float
	Bi	EVENT_MECH_BI	406	Float
	Bj	EVENT_MECH_BJ	407	Float
	Bk	EVENT_MECH_BK	408	Float
	T EigenValue	EVENT_MECH_TEIG	409	Float
	Ti	EVENT_MECH_TI	410	Float
	Tj	EVENT_MECH_TJ	411	Float
	Tk	EVENT_MECH_TK	412	Float
	Mech T Value	EVENT_MECH_TV ALUE	413	Float
	Mech K Value	EVENT_MECH_KV ALUE	414	Float
	Mech Moment	EVENT_MECH_M OMENT	415	Float
	Amplitudes used	EVENT_MECH_AM PSUSED	416	Integer

Block	Parameter Name	enum declaration	Data Code	Format
	Condition Number	EVENT_MECH_CO NDNO	417	Float
	Quality Index	EVENT_MECH_QI NDEX	418	Float
	M1	EVENT_MECH_M1	419	Float
	M2	EVENT_MECH_M2	420	Float
	M3	EVENT_MECH_M3	421	Float
	M4	EVENT_MECH_M4	422	Float
	M5	EVENT_MECH_M5	423	Float
	M6	EVENT_MECH_M6	424	Float
	M Uncertainty 1	EVENT_MECH_MU NC1	425	Float
	M Uncertainty 2	EVENT_MECH_MU NC2	426	Float
	M Uncertainty 3	EVENT_MECH_MU NC3	427	Float
	M Uncertainty 4	EVENT_MECH_MU NC4	428	Float
	M Uncertainty 5	EVENT_MECH_MU NC5	429	Float
	M Uncertainty 6	EVENT_MECH_MU NC6	430	Float
	RMS Error	EVENT_MECH_ME ANERROR	431	Float
	Amplitude residual	EVENT_MECH_ME ANRESIDUAL	432	Float
	Type	EVENT_MECH_FL AGSMTYPE	433	Float
	ISO %	EVENT_MECH_ISO	434	Float
	DC %	EVENT_MECH_DC	435	Float
	CLVD %	EVENT_MECH_CL VD	436	Float
	T Val Error	EVENT_MECH_TVAL	437	Float
	K Val Error	EVENT_MECH_KVAL	438	Float
PFC Ball	Ball Location X	PFC_BALL_LOCX	501	Float

	Ball Location Y	PFC_BALL_LOCY	502	Float
	Ball Location Z	PFC_BALL_LOCZ	503	Float
	Ball Radius	PFC_BALL_RADIUS	504	Float
PFC Velocity	Velocity i	PFC_VELOCITY_D IRI	601	Float
	Velocity j	PFC_VELOCITY_D IRJ	602	Float
	Velocity k	PFC_VELOCITY_D IRK	603	Float
	Velocity Magnitude	PFC_VELOCITY_M AG	604	Float
PFC Displacement	Disp i	PFC_DISP_DIRI	701	Float
	Disp j	PFC_DISP_DIRJ	702	Float
	Disp k	PFC_DISP_DIRK	703	Float
	Disp Magnitude	PFC_DISP_MAG	704	Float
PFC Crack	Crack Location X	PFC_CRACK_LOCX	801	Float
	Crack Location Y	PFC_CRACK_LOCY	802	Float
	Crack Location Z	PFC_CRACK_LOCZ	803	Float

Block	Parameter Name	enum declaration	Data Code	Format
	Crack Radius	PFC_CRACK_RAD IUS	804	Float
	Crack Norm i	PFC_CRACK_NORMI	805	Float
	Crack Norm j	PFC_CRACK_NOR MJ	806	Float
	Crack Norm k	PFC_CRACK_NOR MK	807	Float
	Crack Type	PFC_CRACK_TYPE	808	Integer

Table 2.8: Valid instrument data parameters for when a VVF is exported from a component

Block	Parameter Name	enum declaration	Data Code	Format
Date/Time Stamp	Optional Integer Label	EVENT_LABEL	0	Integer
	Identification String	EVENT_IDSTRING	4	String
Event Location	Instrument Number	INST_NUMBER	101	Integer
		INST_FLAG_IN_LOC	102	Integer
		INST_FLAG_IN_SOURCE PARAMS	103	Integer
	P-wave Pick	INST_TRACE_P_PICK	104	Integer
	P-wave Pick Amplitude	INST_TRACE_P_AMPPICK	105	Integer
	P-wave Pick Velocity	INST_P_PICKVEL	106	Float
	S-wave Pick	INST_TRACE_S_PICK	107	Integer
	S-wave Pick Amplitude	INST_TRACE_S_AMPPICK	108	Integer
	S-wave Pick Velocity	INST_S_PICKVEL		Float
	S2-wave Pick	INST_TRACE_S2_PICK		Integer
	Rotation Type	INST_ROTATION_TYPE		Integer
	Rotation Algorithm	INST_ROTATION_ALGORI THM		Integer
	Reverse Azimuth	INST_REVERSE_AZIMUTH		Integer
	Rotation Source Vector North	INST_ROT_SOURCE_VEC		Integer

		_N		
	Rotation Source Vector East	INST_ROT_SOURCE_VEC _E		Integer
	Rotation Source Vector Down	INST_ROT_SOURCE_VEC _D		Integer
	Rotation Linearity	INST_ROT_LINEARITY		Float
	Single-station-locationDis- tance	INST_SSL_DISTANCE		Integer
	Single-station-locationDis- tance Units	INST_SSL_DIST_UNITS		Integer
		INST_PFREQ_PARAMS		Integer
	P-wave Manual Low- frequency Plateau	INST_PLOWFREQ_PLATE- AU		Integer
	P-wave Manual Corner Fre- quency	INST_PCORNER_FREQ		Integer
	P-wave Manual Energy Flux	INST_PENERGY_FLUX		Integer
		INST_SHFREQ_PARAMS		Integer
	S _H -wave Manual Low- frequency Plateau	INST_SHLOWFREQ_PAR AMS		Integer
	S _H -wave Manual Corner Fre- quency	INST_SHCORNER_FREQ		Integer
	S _H -wave Manual Energy Flux	INST_SHENERGY_FLUX		Integer

Block	Parameter Name	enum declaration	Data Code	Format
		INST_SVFREQ_PARAMS		Integer
	Sy-wave Manual Low-frequency Plateau	INST_SVLOWFREQ_PARAMS		Integer
	Sy-wave Manual Corner Frequency	INST_SVCORNER_FREQ		Integer
	Sy-wave Manual Energy Flux	INST_SVENERGY_FLUX		Integer

2.7.5 Implementation for InSite

For InSite a VVF is written depending on the available data in the event. Each event can contain parameters in the range of data codes 0 to 424 given in Table 2.7.

2.10 The ASCII Trace File ATF

An ATF is a text data file containing a waveform (Figure 2.8). ATFs can be output from InSite, as well as imported. They are therefore a simple method of sharing data between users, and between software, without giving an ESF (Event Storage File).

The ATF file name has the following structure:

ComponentName_EventNumber_ChannelNumber.ATF

The first three lines are header lines.

Line 1 – the ATF version number

Line 2 – a string of initialisation parameters

Line 3 – the string [TraceData]

The lines following this contain the amplitude at each data point in floating-point format.

If the start time and lengths of the traces belonging to a single event are not uniform, waveforms are padded on import to give the same absolute start and end times to all traces.

Figure 2.8: ATF file format

The initialisation parameters in Line 2 are defined by a character name (case sensitive) followed by the parameter value and are delimited by semi-colons (spaces can also be included). Valid parameter names are contained in Table 2.9.

When InSite reads a raw waveform data format, the waveform amplitudes are often stored as a voltage. In the ATF, the amplitude can be the voltage already converted to e.g. acceleration. The AmpToVolts parameter is the scaling relationship that has been used to convert from volts, V_i , to the ATF amplitude, W_i .

$$W_i = [\text{AmpToVolts}] V_i \quad (2.1)$$

If, for example, the amplitude stored in the ATF is acceleration then AmpToVolts has units of $\text{m}\cdot\text{s}^{-2}/\text{volt}$. The Receiver Properties set within InSite (see Section 3.4 of the User Manual) contain the parameters: receiver polarisation, p ; sensitivity, s gain, g . Amplitude scaling of the waveform (W_i) is performed using the following equation. Thus, the AmpToVolts parameter is a combination of these three settings.

$$W_i = p \cdot \frac{V_i}{s \cdot g} \quad (2.2)$$

$$[\text{AmpToVolts}] = \frac{p}{s \cdot g} \quad (2.3)$$

The TraceMaxVolts parameter is the voltage at which the waveforms would 'clip' and is generally the acquisition range of the hardware used in the original monitoring. This is equivalent to the Full-Scale Volts parameter, V_{max} , on the Receiver Properties. The Clipping Amplitude is thus given by

$$W_{\text{max}} = [\text{AmpToVolts}] [\text{TraceMaxVolts}] \quad (2.4)$$

When InSite reads a raw data format it scales the waveform amplitudes read according to the four parameters p ; s ; g ; V_{max} set on the appropriate receiver in the Default Receiver Array. For an ATF, the parameters are explicitly set in the file, so instead of copying across these parameters from the Default Receiver Array they are set directly from the

information in the ATF. This means there is no automatic re-scaling of the waveforms when an ATF is imported, but instead the waveform amplitudes stored there are used directly.

Amplitude re-scaling can be performed after importation by updating the receiver properties in the events (see Section 3.4 of the User Manual).

Table 2.9: Valid initialisation parameters contained in an ATF.

Parameter Name	Format	Description
Date	MM-DD-YYYY	Date of event that waveform is from
Time	HH:MM:SS	Time of event that waveform is from
TracePoints	Int	Number of data points in the waveform
TSamp	Float	Sampling time on the waveform
TimeUnits	Float	Time units on the waveform InSite's valid time units include: 1.0 1.0e-03 1.0e-06
AmpToVolts	Float	Conversion factor used to scale the raw voltage to the contained amplitude Amplitude = (AmpToVolts)\Volts
TraceMaxVolts	Float	Maximum \pm voltage scale for the receiver

2.11 Reading the Nicolet WFT File

It is important to have the WFT file names appropriately formatted so as to correctly read the data. The file name defines whether the data is passive or active.

For a passive event the file name must have the following example format:

MyExp_nic10001.wft

This file name can be split into four parts.

1. MyExp_ defines the component name that the data will be read into. The underscore character _ splits this from the rest of the filename.

2. nic1 (a four-character string) defines the Nicolet on which the data was recorded. This implicitly refers to a receiving channel number as channel numbers 1-4 are contained on nic1, channel numbers 5-8 are contained on nic2 etc.

3. 0001 defines the channel number and 'autocycle' number the file refers to.

0001 to 0004 refers to nicolet channels 1-4 in autocycle number 1.

0005 to 0008 refers to nicolet channels 1-4 in autocycle number 2.

0009 to 0012 refers to nicolet channels 1-4 in autocycle number 3.

Each autocycled file can contain more than one event. For instance, it is common to use a 20-event autocycle. Event numbers 41 to 60 would then be contained in autocycle number 3. The waveforms for these events recorded on receiving channel number 7 (nicolet channel number 3 on nic2) can then be found in the file MyExp_nic2011.wft.

4. .wft is the extension of the file

A small utility program (WCopy) can be obtained by emailing support-asc@appliedseismology.co.uk to easily format numerous file names into this structure.

For an active event the file name must have the following example format:

vel10207.wft

The file name can be split into four parts.

1. vel1 defines the component name that the data will be read into (the survey base name).
2. 02 defines the transmitter number that the waveform data originates from.
3. 07 defines the receiver channel number the waveform data was recorded on.
4. wft is the extension of the file.

The second and third parts therefore define the transmitter-receiver ray path on which the data was recorded.

Each WFT file has a text header that contains almost all the information that InSite requires to read and configure the data. The header can be viewed by using the 'type' command at a dos console. Following the data is a binary string containing the waveform data stored as 2-byte (16-bit) integers.

The Voltage Scale (V_{max}) set on the Nicolets at acquisition time is not explicitly recorded in the WFT file header. This V_{max} is important so that the correct amplitude is defined on the waveform. InSite obtains this information by implicitly assuming the following relationship,

$$V_{max} = \pm 32768.0 V_{norm}; \quad (2.5)$$

where V_{norm} is the voltage between 'normalisation levels' (as defined in the WFT file) and 32768 are the number of normalisation levels in a 16-bit integer.

The V_{max} is then automatically set on the receiver information when the data is imported. It is important to recognise that the V_{max} configuration for the channel that is read will then be different to the V_{max} on the user-configured Default Receiver Array. This can be checked by opening the Receiver Properties for that particular channel.

If the user updates the receiver settings at a later date from the Default Receiver Array without considering the true V_{max} then the waveform amplitude will be automatically adjusted.

2.12 The FFT Format File

The FFT Format File contains an amplitude spectrum from a Fast Fourier Transform (FFT). This is described in two columns delimited by 'tabs'. The first column contains the frequency component, and the second contains the amplitude. There is one header line containing titles for the columns. The title of the amplitude column is defined by the response type visualised in the FFT processor at the time of writing. Two additional columns are given in the file for the real and imaginary parts of the FFT.

The FFT file name has the following structure:

ComponentName_EventNumber_ChannelNumber.FFT

Freq.	Velocity
244.141	6.44430e-008
488.281	6.50138e-008
732.422	4.80808e-008
976.563	2.96657e-008
1220.70	4.39954e-008
1464.84	1.86902e-008
1708.98	3.89252e-008
1953.13	4.87367e-008
2197.27	4.51046e-008
2441.41	4.40963e-008
2685.55	2.42948e-008
2929.69	1.56491e-008
3173.83	1.02885e-008
3417.97	5.37986e-009
3662.11	1.01177e-008
3906.25	1.49432e-008
4150.39	1.61121e-008

Figure 2.9: Example FFT file format.

2.13 The Report RPT Format File

A report file is a text file containing detailed information output by InSite's processing algorithms. This gives the user the ability to record and inspect the process of every calculation.

```

msdata2001.rpt - Notepad
File Edit Format View Help

***LOCATION RUN***
Processing Event 3 (14-08-2001 22:00:01) in Component 1drive with 32P and 27S
Processed wed 21-08-2002 14:41:12
Using both P-wave and S-wave arrivals
Arrival times processed by Instrument

Sensor Information Used (units = 1.000000e+000m):
C/I      N      E      D      dN      dE      dD      Tp      Ts
1        422.460  721.380  143.490  0.639  -0.424  0.642  15.200  19.400
4        447.700  777.130  135.490  0.647  0.325  0.690  17.440  31.120
7        408.280  691.340  121.420  0.474  0.712  0.518  19.300  26.180
10       412.210  707.040  110.460  -0.763  0.393  0.514  17.300  31.300
13       455.450  752.490  143.630  0.838  0.347  0.421  14.400  0.000
16       396.280  699.780  121.350  -0.281  -0.422  0.862  19.820  27.260
19       442.880  765.200  110.550  0.695  -0.719  0.000  15.720  19.700
23       446.320  747.130  121.590  0.871  0.491  0.000  12.200  0.000
25       413.350  724.560  121.450  -0.803  0.559  0.208  15.500  20.060
33       428.960  751.460  110.560  -0.651  -0.646  0.398  14.440  18.000
34       416.020  764.230  143.550  0.392  0.731  0.559  17.720  0.000
37       398.180  716.880  110.450  0.701  0.592  0.397  18.580  25.060
40       402.310  732.440  121.360  0.772  -0.561  0.299  17.140  22.960
43       387.860  702.250  143.600  0.021  -0.961  0.277  21.300  30.100

Number of Independent Sensors = 14
Starting position at array centroid.
using the Simplex algorithm (LPNorm=1) ...

PASS #1:
Number of Independent Sensors = 14
Starting Simplex: N= 419.876 E= 732.379 D= 125.639 with units 1.000000e+000m
Starting Dimension = 6.219
Simplex located at: N= 447.913 E= 731.627 D= 122.789 with units 1.000000e+000m
Dimension = 4.708
Outlier Identification OFF
Event Source Time From Mean of Arrivals = 10.2
Arrival Residuals are:
P1=-0.798 S1=-0.784 P4=-0.721 S4=7.21 P7=-0.422 S7=-0.427 P10=-0.496 S10=8.02 P13=-0.937 P16=-0.600 S16=-0.550 P19=-0.566 S19=-0.981 P23=-0
RMS residual = 2.25
RMS error = 7.79
Arrival S10 has largest residual above 0.400 ... dropping arrival!

PASS #2:
Number of Independent Sensors = 14
Starting Simplex: N= 419.876 E= 732.379 D= 125.639 with units 1.000000e+000m
Starting Dimension = 6.219
Simplex located at: N= 446.640 E= 732.919 D= 123.525 with units 1.000000e+000m
Dimension = 3.067
Outlier Identification OFF
Event Source Time From Mean of Arrivals = 10.1
Arrival Residuals are:
P1=-0.498 S1=-0.367 P4=-0.342 S4=7.76 P7=-0.296 S7=-0.311 P10=-0.347 P13=-0.625 P16=-0.401 S16=-0.307 P19=-0.243 S19=-0.525 P23=-0.293 P25=
RMS residual = 1.62
RMS error = 5.47
Arrival S4 has largest residual above 0.400 ... dropping arrival!

PASS #3:
Number of Independent Sensors = 14
Starting Simplex: N= 419.876 E= 732.379 D= 125.639 with units 1.000000e+000m
Starting Dimension = 6.219
Simplex located at: N= 447.690 E= 732.589 D= 123.810 with units 1.000000e+000m
Dimension = 1.561

```

Figure 2.10: Example RPT file format

2.14 The Trigger Report File TRF

The Trigger Report File (TRF) contains basic information on each event contained in the project including date, time and signal amplitude on each channel. When the trigger report file is written each event that contains waveforms is interrogated. Components and events can be omitted from the process by disabling them in the Data Visualiser.

For each channel in the event a RMS (Root Mean Square) and Peak amplitude are obtained (Peak is from DC to maximum). The information is then written to a column-format text file with one row being a single event. The columns include (from left to right): Component Name, Date, Time, Event Number, followed by three-column groups of Channel Number, RMS amplitude and Peak amplitude. The last two columns in each row are the mean RMS and mean absolute Peak amplitude across all the channels.

```

Trigger report file created in Insite:
conc-01 03-08-2001 06:12:50.360 0001 Channels(#, RMS, Peak) = 01 1.7149e-003 -1.3454e-002 02 2.6028e-003 -1.0164e-002
conc-01 03-08-2001 06:13:16.780 0002 Channels(#, RMS, Peak) = 01 1.6720e-003 -1.6615e-002 02 2.1443e-003 -1.2408e-002
conc-01 03-08-2001 06:13:21.500 0003 Channels(#, RMS, Peak) = 01 1.8041e-003 7.3303e-003 02 3.3284e-003 -1.3773e-002
conc-01 03-08-2001 06:13:30.730 0004 Channels(#, RMS, Peak) = 01 1.8470e-003 -1.3248e-002 02 3.0071e-003 1.2814e-002
conc-01 03-08-2001 06:14:29.440 0005 Channels(#, RMS, Peak) = 01 1.7358e-003 7.0969e-003 02 2.9024e-003 -1.4952e-002
conc-01 03-08-2001 06:14:40.210 0006 Channels(#, RMS, Peak) = 01 1.6665e-003 -6.9763e-003 02 3.1218e-003 -1.3338e-002
conc-01 03-08-2001 06:15:28.490 0007 Channels(#, RMS, Peak) = 01 1.6753e-003 1.6487e-002 02 2.6441e-003 -1.2225e-002
conc-01 03-08-2001 06:15:30.470 0008 Channels(#, RMS, Peak) = 01 1.6629e-003 1.7828e-002 02 2.4849e-003 -1.4731e-002
conc-01 03-08-2001 06:15:38.320 0009 Channels(#, RMS, Peak) = 01 1.7161e-003 6.0242e-003 02 4.1658e-003 -2.3517e-002
conc-01 03-08-2001 06:15:40.240 0010 Channels(#, RMS, Peak) = 01 1.9623e-003 -1.3182e-002 02 8.4230e-003 6.0281e-002
conc-01 03-08-2001 06:19:02.310 0011 Channels(#, RMS, Peak) = 01 1.9183e-003 -1.8204e-002 02 2.1317e-002 1.6227e-001
conc-01 03-08-2001 06:19:33.020 0012 Channels(#, RMS, Peak) = 01 1.8028e-003 -7.2617e-003 02 1.9241e-002 1.5150e-001
conc-01 03-08-2001 06:19:52.900 0013 Channels(#, RMS, Peak) = 01 1.7965e-003 5.9937e-003 02 3.3526e-003 1.5799e-002
conc-01 03-08-2001 06:21:33.630 0014 Channels(#, RMS, Peak) = 01 1.9238e-003 1.5630e-002 02 3.1992e-003 1.3011e-002
conc-01 03-08-2001 06:21:40.330 0015 Channels(#, RMS, Peak) = 01 1.7475e-003 -1.5477e-002 02 3.3593e-003 1.5288e-002
conc-01 03-08-2001 06:22:10.430 0016 Channels(#, RMS, Peak) = 01 1.7628e-003 -6.1310e-003 02 4.1021e-003 1.7781e-002

```

Figure 2.11: Example TRF file format.

2.15 Event Parameters File EPF

An EPF file is an ASCII text file that contains event data in tab-delimited column format. Each line is a single event.

This file format allows easy data import into mathematical software such as MS Excel.

There are three header lines:

- Line 1 is a description of the file
- Line 2 is a description of the file version
- Line 3 contains titles for every column delimited by a tab character

Each line contains data columns (

```

Event Parameter File for msdata2001.pcf created on Fri 23-08-2002 13:58:58
Format Version 2.09
Component      Date      Time      Event      Label      North      East      Down      Units      LocMag.      NLT      NIT      RMS resid.
ldrive 14-08-2001 22:00:01.00 0003 TS3 446.941 733.202 124.127 1.0000 -1.3568 24 14 0.12897
ldrive 15-08-2001 16:36:28.00 0004 TS4 411.669 706.757 98.267 1.0000 -1.6183 12 11 0.022776
ldrive 15-08-2001 23:56:27.00 0009 TS9 424.842 747.423 102.970 1.0000 -1.3960 21 11 0.067979
ldrive 16-08-2001 06:34:00.00 0014 TS14 416.950 705.835 95.990 1.0000 -1.2987 22 14 0.082507
ldrive 16-08-2001 23:55:06.00 0015 TS15 412.679 685.497 123.643 1.0000 -1.4376 23 14 0.11759
ldrive 17-08-2001 01:23:56.00 0016 TS16 413.043 685.272 123.592 1.0000 -1.3316 21 14 0.10918
ldrive 17-08-2001 04:46:11.00 0017 TS17 436.951 762.668 147.235 1.0000 -1.1851 18 11 0.11390
ldrive 17-08-2001 22:28:40.00 0031 TS31 427.644 706.794 124.817 1.0000 -1.7244 22 14 0.065566
ldrive 17-08-2001 22:28:54.00 0032 TS32 428.095 706.252 124.971 1.0000 -0.93816 28 15
ldrive 19-08-2001 00:34:00.00 0035 TS35 391.131 694.606 123.914 1.0000 -1.7047 10 10 0.029418
ldrive 19-08-2001 04:35:09.00 0036 TS36 391.266 695.186 123.391 1.0000 -1.4855 16 12 0.12573
ldrive 19-08-2001 10:25:59.00 0037 TS37 410.689 675.677 72.385 1.0000 -0.66360 11 11
ldrive 19-08-2001 12:49:47.00 0039 TS39 416.179 705.554 97.826 1.0000 -1.6486 22 13 0.086201
ldrive 21-08-2001 21:30:21.00 0052 TS52 422.714 717.446 109.714 1.0000 -1.2600 12 10 0.20563
ldrive 22-08-2001 01:12:17.00 0055 TS55 390.779 695.589 123.188 1.0000 -1.6624 17 12 0.092150
ldrive 22-08-2001 22:26:56.00 0056 TS56 446.495 751.050 146.584 1.0000 -0.68054 20 11
ldrive 23-08-2001 00:38:25.00 0057 TS57 391.293 694.915 123.104 1.0000 -1.4515 23 14 0.092528
ldrive 23-08-2001 21:19:37.00 0058 TS58 417.774 735.954 152.693 1.0000 -0.98773 18 14
ldrive 24-08-2001 01:12:17.00 0059 TS59 419.126 737.973 100.049 1.0000 -1.2308 11 11 0.013959
ldrive 25-08-2001 04:53:46.00 0064 TS64 476.468 713.945 115.399 1.0000 -0.75728 12 12
ldrive 25-08-2001 05:16:46.00 0068 TS68 554.749 684.609 92.479 1.0000 -0.40310 13 13

```

Figure 2.12: Example EPF file format

Table 2.10) and ends with the character string "EOL".

Component	Date	Time	Event	Label	North	East	Down	Units	LocMag.	NLT	NIT	RMS resid.
ldrive	14-08-2001	22:00:01.00	0003	TS3	446.941	733.202	124.127	1.0000	-1.3568	24	14	0.12897
ldrive	15-08-2001	16:36:28.00	0004	TS4	411.669	706.757	98.267	1.0000	-1.6183	12	11	0.02277E
ldrive	15-08-2001	23:56:27.00	0009	TS9	424.842	747.423	102.970	1.0000	-1.3960	21	11	0.06797E
ldrive	16-08-2001	06:34:00.00	0014	TS14	416.950	705.835	95.990	1.0000	-1.2987	22	14	0.082507
ldrive	16-08-2001	23:55:06.00	0015	TS15	412.679	685.497	123.643	1.0000	-1.4376	23	14	0.11759
ldrive	17-08-2001	01:23:56.00	0016	TS16	413.043	685.272	123.592	1.0000	-1.3316	21	14	0.10918
ldrive	17-08-2001	04:46:11.00	0017	TS17	436.951	762.668	147.235	1.0000	-1.1851	18	11	0.11390
ldrive	17-08-2001	22:28:40.00	0031	TS31	427.644	706.794	124.817	1.0000	-1.7244	22	14	0.06556E
ldrive	17-08-2001	22:28:54.00	0032	TS32	428.095	706.252	124.971	1.0000	-0.93816		28	15
ldrive	19-08-2001	00:34:00.00	0035	TS35	391.131	694.606	123.914	1.0000	-1.7047	10	10	0.02941E
ldrive	19-08-2001	04:35:09.00	0036	TS36	391.266	695.186	123.391	1.0000	-1.4855	16	12	0.12573
ldrive	19-08-2001	10:25:59.00	0037	TS37	410.689	675.677	72.385	1.0000	-0.66360		11	11
ldrive	19-08-2001	12:49:47.00	0039	TS39	416.179	705.554	97.826	1.0000	-1.6486	22	13	0.086201
ldrive	21-08-2001	21:30:21.00	0052	TS52	422.714	717.446	109.714	1.0000	-1.2600	12	10	0.20563
ldrive	22-08-2001	01:20:58.00	0055	TS55	390.779	695.589	123.188	1.0000	-1.6624	17	12	0.09215C
ldrive	22-08-2001	22:26:56.00	0056	TS56	446.495	751.050	146.584	1.0000	-0.68054		20	11
ldrive	23-08-2001	00:38:25.00	0057	TS57	391.293	694.915	123.104	1.0000	-1.4515	23	14	0.09252E
ldrive	23-08-2001	21:19:37.00	0058	TS58	417.774	735.954	152.693	1.0000	-0.98773		18	14
ldrive	24-08-2001	01:12:17.00	0059	TS59	419.126	737.973	100.049	1.0000	-1.2308	11	11	0.01395E
ldrive	25-08-2001	04:53:46.00	0064	TS64	476.468	713.945	115.399	1.0000	-0.75728		12	12
ldrive	25-08-2001	05:16:46.00	0068	TS68	554.749	684.609	92.479	1.0000	-0.40310		13	13

Figure 2.12: Example EPF file format

Table 2.10: Description of data columns contained in an EPF format file, version 2.12.

Column Title	Parameter Description
Component	Component Name
Date	Date of Event (DD-MM-YYYY)
Time	Time of Event
Event	Event Number
Label	Event Label
NP Picks	Number of P waves picked
NS Picks	Number of S waves picked
North	Location northing
East	Location easting
Down	Location depth
Units	Units of location (referenced to 1m)
T0 Time	T0 Time
LocMag.	Location Magnitude

Column Title	Parameter Description
NLT	Number of arrivals used in the location
NIT	Number of independent sensors used
RMS resid.	RMS residual time
RMS error	RMS location error
%ISO, %DC, %CLVD	Percentage decomposition of the moment tensor (MT)
MTMom	Relative seismic moment from the MT
NA	Number of amplitudes used in inversion
TVal	T-value from decomposition
DTVal	Uncertainty in the T value
KVal	k-value from decomposition
DKVal	Uncertainty in k value
AResid	RMS amplitude residual
CondNo	Condition number
MT Component 1 - 6	Six Moment Tensor Components
MT Uncertainty 1 - 6	Six Moment Tensor Uncertainties
EFactor	RMS Error Factor
qIndex	Quality index
SMTType	Source mechanism type flag 1 – double couple; 0 – exotic
MTDecom_FP1_Strike, MTDecom_FP1_Dip, MTDecom_FP1_Rake	By Moment Tensor Decomposition, First fault plane azimuth, dip and rake (degrees)
MTDecom_FP2_Strike, MTDecom_FP2_Dip, MTDecom_FP2_Rake	By Moment Tensor Decomposition, Second fault plane azimuth, dip and rake (degrees)
Peig, PeigN, PeigE, PEigD	Pressure axis eigenvalue and eigenvector (NED)
Beig, BeigN, BeigE, BEigD	Null axis eigenvalue and eigenvector (NED)

Teig, TeigN, TeigE, TEigD	Tension axis eigenvalue and eigenvector (NED)
LocErA, LocErAN, LocErAE, LocErAD	Location error ellipsoid – A axis magnitude and orientation (NED)
LocErB, LocErBN, LocErBE, LocErBD	Location error ellipsoid – B axis magnitude and orientation (NED)
LocErC, LocErCN, LocErCE, LocErCD	Location error ellipsoid – C axis magnitude and orientation (NED)
IMag	Instrument Magnitude
SMom	Seismic Moment
MMag	Moment Magnitude
Energy	Event radiated energy
StoP	S:P energy ratio
Radius	Source radius
ASTress	Apparent stress
SSDrop	Static stress drop
Pfc	P-wave corner frequency
Sfc	S-wave corner frequency

2.16 Waveform Pick Results PED Format File

A PED format file is an ASCII text file that contains channel information for every event. On exporting the file the user is asked to choose the channel parameter type to be written out. Only one parameter type can be written. This file format gives a simple method of exchanging channel data between software.

- CCR P-wave Velocity Difference
- CCR P-wave Window Maximum Amplitude
- CCR S-wave Arrival Time
- CCR S-wave Arrival Time Difference
- CCR S-wave Velocity
- CCR S-wave Velocity Difference
- CCR S-wave Window Maximum Amplitude
- Rotation Source Vector North
- Rotation Source Vector East
- Rotation Source Vector Down
- Rotation Linearity
- Single-station-location Distance
- P-wave Manual Low-frequency Plateau
- P-wave Manual Corner Frequency
- P-wave Manual Energy Flux
- S_H-wave Manual Low-frequency Plateau
- S_H-wave Manual Corner Frequency
- S_H-wave Manual Energy Flux
- S_V-wave Manual Low-frequency Plateau
- S_V-wave Manual Corner Frequency
- S_V-wave Manual Energy Flux

The File has three header lines:

- Line 1 is a description of the file
 - Line 2 is the channel parameter type that the file contains
-

- Line 3 is a title line for the column data

The data then follows the header lines. Each event is contained on a single line. The first 52 characters of the line contain information on the event:

- Component Name
- Component Location (where available)
- Event Number
- Date (DD-MM-YYYY)
- Time

Following the event information is the output parameter, where available, for each channel contained in a 19-character block. The first three characters of each block contain the channel number. If no data exists for a particular channel, then the block is output as blank spaces.

2.17 File for Mechanism Axes DIP format

A DIP format file outputs source mechanism orientation information to an ASCII text file. The format of the file allows direct input into the DIPS program [[RocScience Inc., Toronto, Canada](#)] enabling graphical display of the information.

The first eight lines are header lines describing the data that is included in the file. The data is then contained in five columns.

- Event number
- Azimuth (degrees east of north)
- Plunge (degrees down from the horizontal)
- Axis label
- Component Name

The user is given the choice as to which mechanism parameter to write to the file.

- Pressure, P axis
- Null, B axis
- Tension, T axis
- Slip vector 1
- Slip vector 2

```

MOMENT TENSOR INVERSION RESULTS
P-axes
0 TRAVERSES
TREND/PLUNGE
0 DECLINATION
NO QUANTITY
2 EXTRA DATA COLUMNS
NUMBER ; TREND ; PLUNGE ; AXIS ; COMPONENT ;
6      229.5  27.5  P   tsxcluster3
11     18.6   15.3  P   tsxcluster3
13     0.6   10.7  P   tsxcluster3
16     19.6   9.9   P   tsxcluster3
17     210.2  4.5   P   tsxcluster3
-1

```

Figure 2.14: Example DIP file format.

2.18 Column Format Survey File DCF

A DCF file contains survey data (components containing active events). The data is contained in space-delimited columns. The file format gives a straightforward table of information that can be easily imported into mathematical software such as MS Excel.

Each survey (component) is contained on a single line. The user selects the ray paths to export. These are defined by a range of shots/transmitters (defined by the Event Number) and a range of receivers (defined by the Channel Number). Each column then contains one Shot-to-receiver ray path described in the column title (e.g. 0002 < 001 denotes Shot 2 to Receiver 1).

The user selects which data parameter type, associated with the surveys, to output to the file. The file only contains a single parameter type. When no parameter is available for a given ray path then the table position is left blank (filled with spaces).

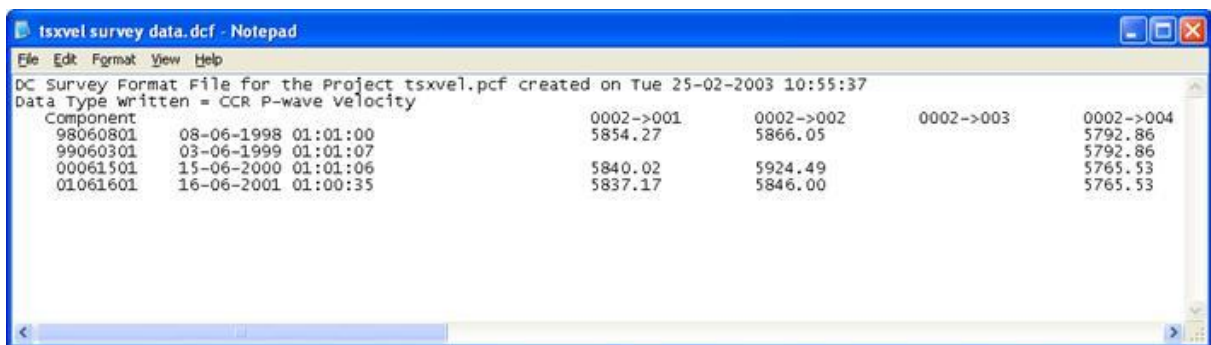
- CCR P-wave Arrival Time
- CCR P-wave Arrival Time Difference
- CCR P-wave Velocity
- CCR P-wave Velocity Difference
- CCR P-wave Window Maximum Amplitude
- CCR S-wave Arrival Time
- CCR S-wave Arrival Time Difference
- CCR S-wave Velocity

- CCR S-wave Velocity Difference
- CCR S-wave Window Maximum Amplitude
- P-wave Velocity from Pick
- P-wave Amplitude from Pick
- P-wave Amplitude from Picking Window
- S-wave Velocity from Pick
- S-wave Amplitude from Pick
- S-wave Amplitude from Picking Window

The file has three header lines:

- Line 1 is a description of the file
- Line 2 gives the data type that the file contains
- Line 3 gives a description of the columns

Each data line also contains information on the component including the name, date (DD-MM-YYYY), time and location (where available).



```

DC Survey Format File for the Project tsxvel.pcf created on Tue 25-02-2003 10:55:37
Data Type written = CCR P-wave Velocity
Component
98060801 08-06-1998 01:01:00 0002->001 0002->002 0002->003 0002->004
99060301 03-06-1999 01:01:07 5854.27 5866.05 5792.86
00061501 15-06-2000 01:01:06 5840.02 5924.49 5792.86
01061601 16-06-2001 01:00:35 5837.17 5846.00 5765.53

```

Figure 2.15: Example DCF file format.

2.19 Stereonet Format Survey File SSF

A SSF format file contains data from a single survey (components containing active events). The data is contained in space-delimited columns and is formatted so as to be easily read into stereonet plotting software.

The user selects which data parameter type, associated with the surveys, to output to the file. The file only contains a single parameter type.

- P-wave Pick Time
 - S-wave Pick Time
 - Waveform RMS Amplitude
 - Waveform Peak Amplitude
 - Time of Waveform Peak
 - Frequency at Waveform Peak
 - Signal to Noise
 - P-wave Amplitude Pick Time
 - P-wave Amplitude from Pick
 - S-wave Amplitude Pick Time
 - S-wave Amplitude from Pick
 - P-wave Velocity from Pick
 - P-wave Amplitude from Picking Window
 - S-wave Velocity from Pick
 - S-wave Amplitude from Picking Window
 - CCR P-wave Arrival Time
 - CCR P-wave Arrival Time Difference
 - CCR P-wave Velocity
 - CCR P-wave Velocity Difference
-

- CCR P-wave Window Maximum Amplitude
- CCR S-wave Arrival Time
- CCR S-wave Arrival Time Difference
- CCR S-wave Velocity
- CCR S-wave Velocity Difference
- CCR S-wave Window Maximum Amplitude

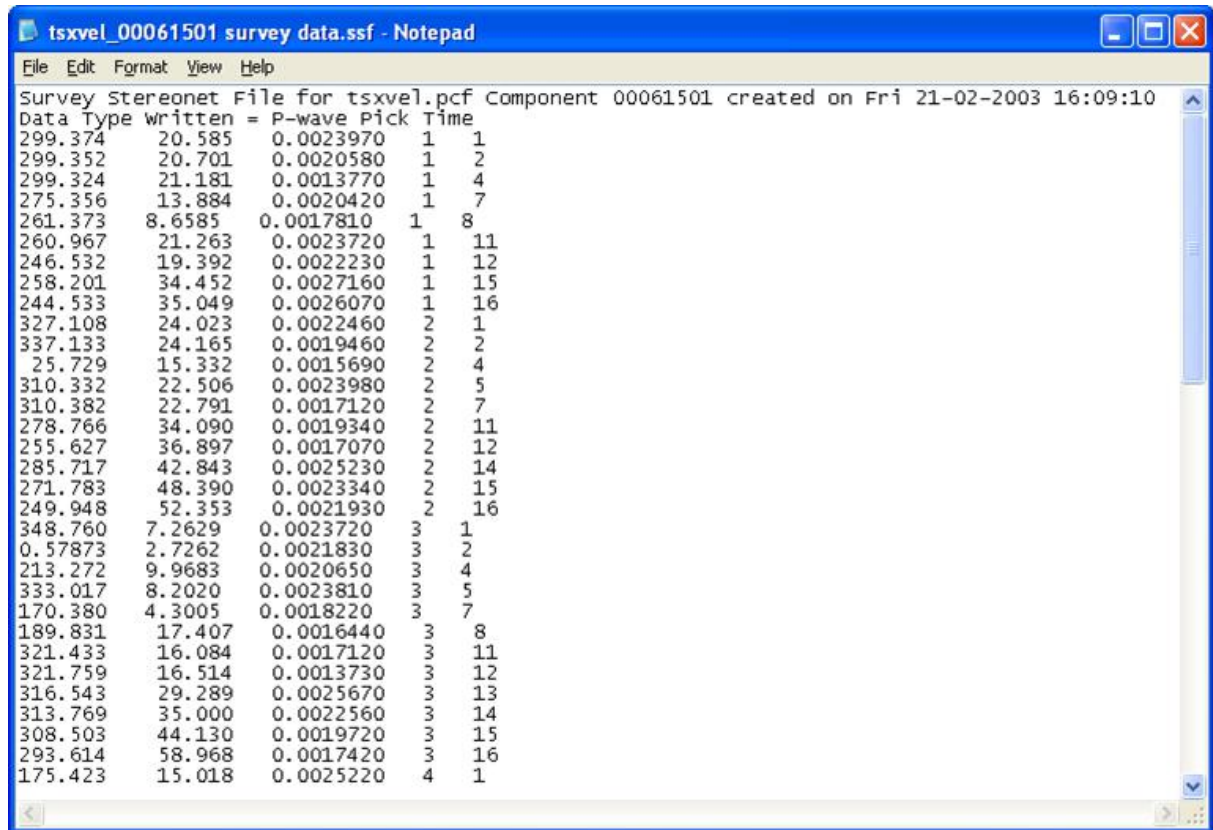
The file contains two header lines:

- Line 1 is a description of the file
- Line 2 gives the data type that the file contains

The data is then contained in the following columns.

- Azimuth of ray path (in degrees east of north)
- Plunge of ray path (in degrees down from the horizontal)
- The data parameter
- The shot/transmitter number (defined by the Event Number)
- The receiver number (defined by the Channel Number)

The ray path is defined as a straight line between the shot location and the receiver location, so as to plot on a lower-hemisphere stereonet (plunge is always positive).



```
tsxvel_00061501 survey data.ssf - Notepad
File Edit Format View Help
Survey Stereonet File for tsxvel.pcf Component 00061501 created on Fri 21-02-2003 16:09:10
Data Type written = P-wave Pick Time
299.374 20.585 0.0023970 1 1
299.352 20.701 0.0020580 1 2
299.324 21.181 0.0013770 1 4
275.356 13.884 0.0020420 1 7
261.373 8.6585 0.0017810 1 8
260.967 21.263 0.0023720 1 11
246.532 19.392 0.0022230 1 12
258.201 34.452 0.0027160 1 15
244.533 35.049 0.0026070 1 16
327.108 24.023 0.0022460 2 1
337.133 24.165 0.0019460 2 2
25.729 15.332 0.0015690 2 4
310.332 22.506 0.0023980 2 5
310.382 22.791 0.0017120 2 7
278.766 34.090 0.0019340 2 11
255.627 36.897 0.0017070 2 12
285.717 42.843 0.0025230 2 14
271.783 48.390 0.0023340 2 15
249.948 52.353 0.0021930 2 16
348.760 7.2629 0.0023720 3 1
0.57873 2.7262 0.0021830 3 2
213.272 9.9683 0.0020650 3 4
333.017 8.2020 0.0023810 3 5
170.380 4.3005 0.0018220 3 7
189.831 17.407 0.0016440 3 8
321.433 16.084 0.0017120 3 11
321.759 16.514 0.0013730 3 12
316.543 29.289 0.0025670 3 13
313.769 35.000 0.0022560 3 14
308.503 44.130 0.0019720 3 15
293.614 58.968 0.0017420 3 16
175.423 15.018 0.0025220 4 1
```

Figure 2.16: Example SSF file format.

2.20 Filter File FLT

A FLT file contains either a user defined time domain filter or a user defined frequency domain filter.

The main points to note about the filter file are:

- The first line should contain either the text "Type:Time" for a time domain filter, or the text "Type:Frequency" for a frequency domain filter.
- The next lines in the file should contain the filter data.
- With a time-domain filter (

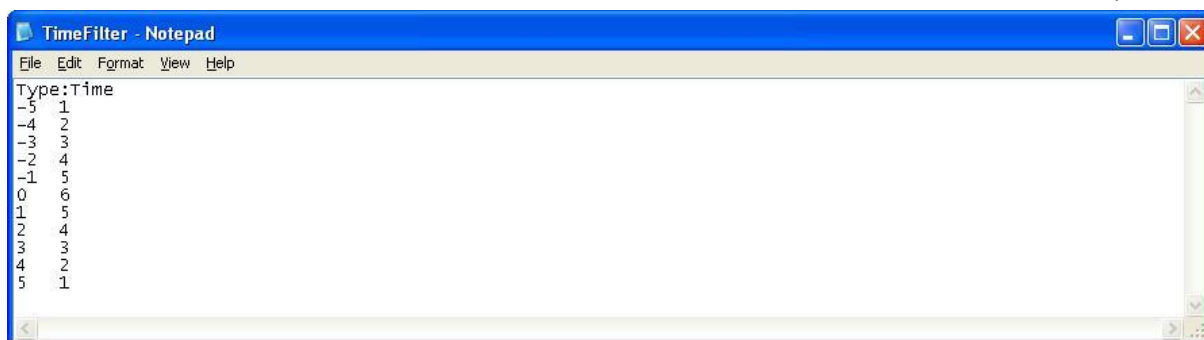


Figure 2.17):

1. Each filter data line in the file should contain two space delimited numbers.
2. The first column contains an x-axis value and the second column contains a y-axis value.
3. There should be an odd number of filter data lines.
4. The x-axis values should go from negative to positive integer values with an even number of positive and negative values and one 0 value.
5. It is important that the filter has fewer data values than the number of sample points in a waveform that is being filtered.

The filter is applied to a waveform in the following way:

1. We initially have two waveforms, the filter waveform and the channel waveform.
2. We create a new filter waveform that is the same length (in data points) as the channel waveform. The first part of this new waveform consists of the filter file's positive x-axis values. The last part of this waveform consists of the filter file's negative x-axis values. The gap in between these two parts is padded out with zeros.
3. We then perform an FFT on the channel waveform and on the new filter waveform.

4. We multiply the two FFTs together in order to convolve them.
5. We then do an inverse FFT of the convolved waveform in order to get the filtered channel waveform.

• With a frequency domain filter (

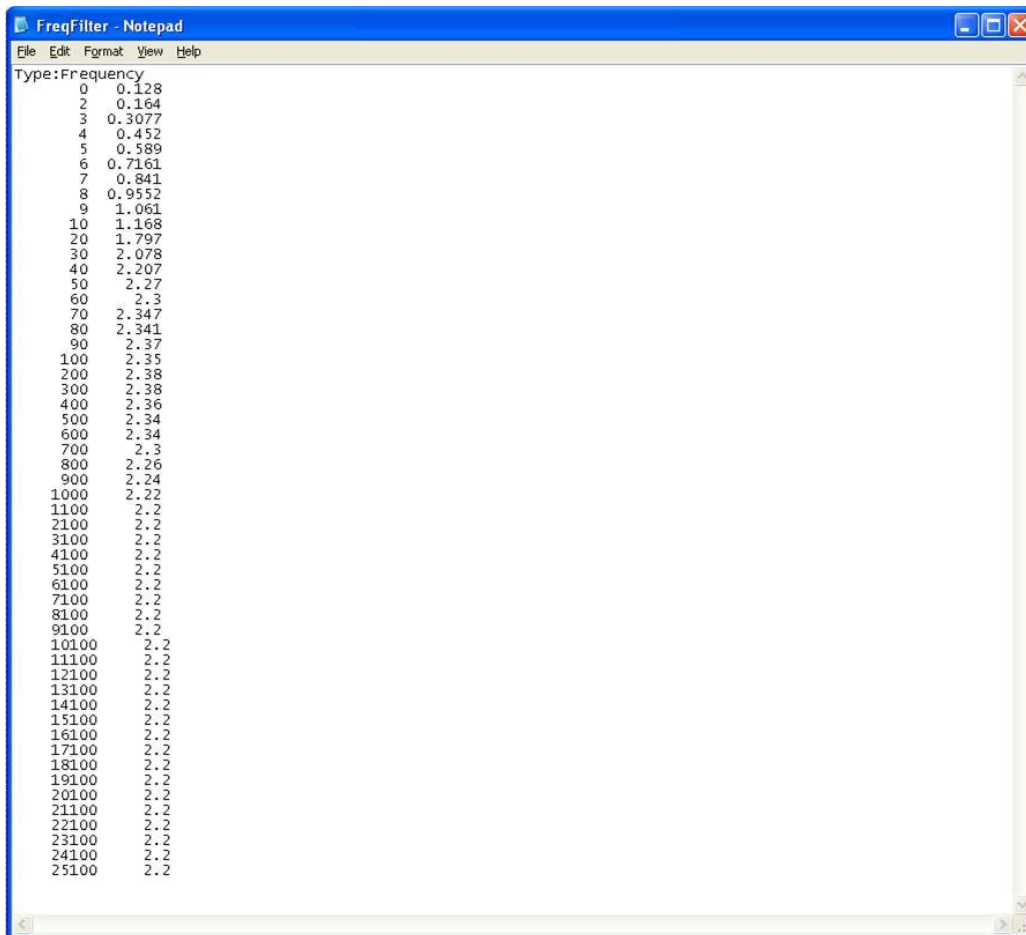


Figure 2.18):

1. Each filter data line in the file should contain two space delimited numbers.
 2. The first column contains a frequency value and the second column contains an amplitude value.
- The filter is applied to a waveform in the following way:
 1. We perform an FFT on the channel waveform.
 2. We interpolate the data points in the filter using a spline so that we can get an amplitude for every frequency in the filter's range.
 3. We multiply the real and imaginary parts of the FFT at each frequency by the amplitude contained in the spline of the filter at that frequency.
 4. We then do an inverse FFT to get the filtered channel waveform.

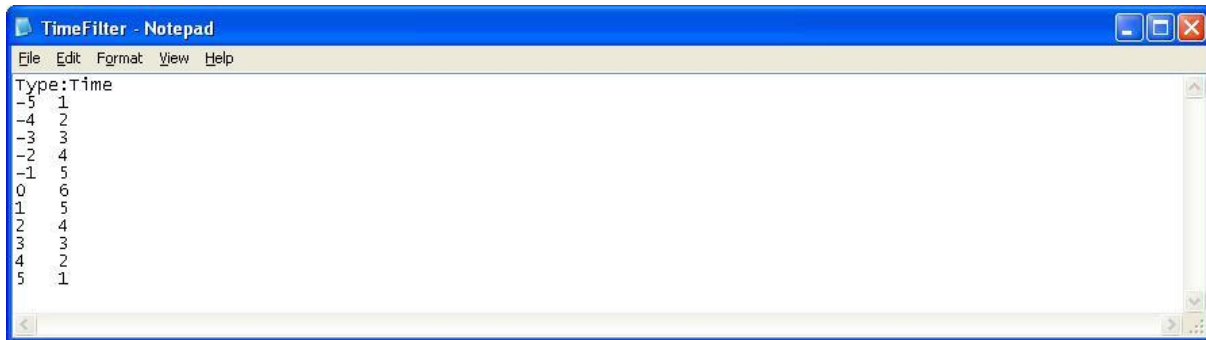


Figure 2.17: Example of a time domain FLT file. Note: With this filter, every waveform must be constructed from more than 11 data points.

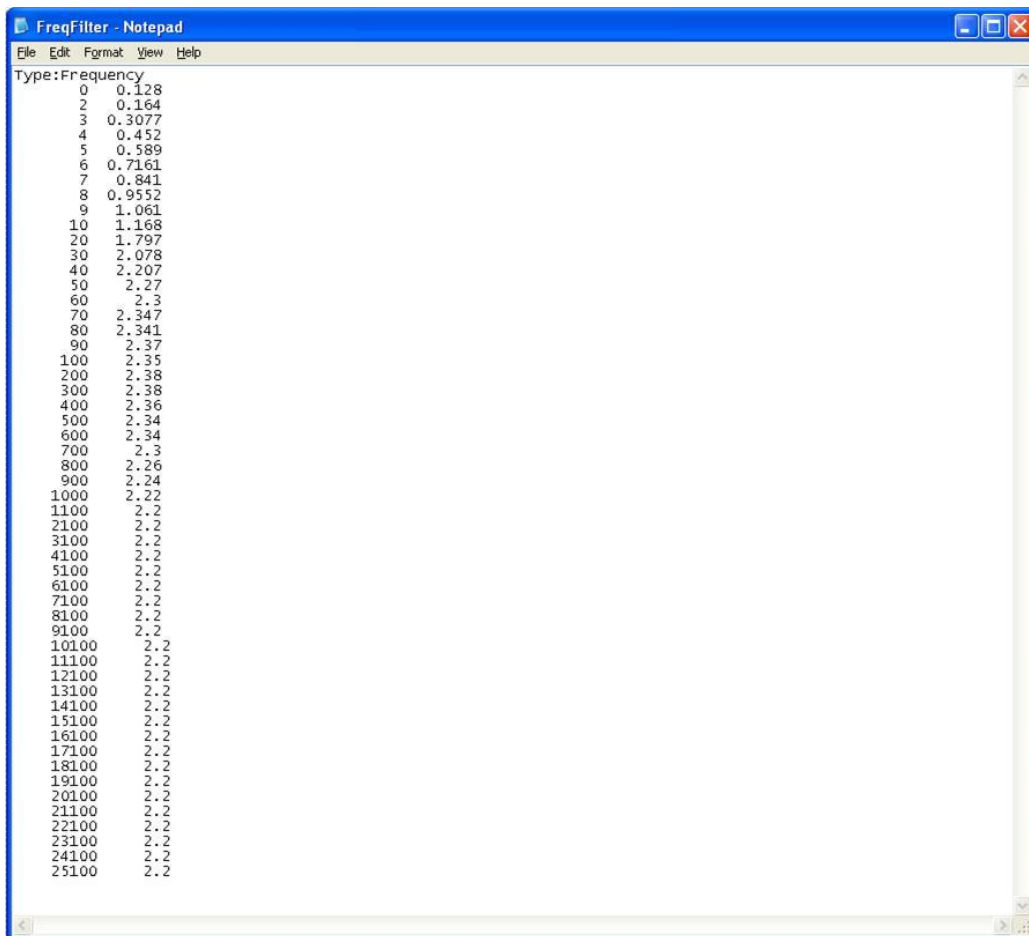


Figure 2.18: Example of a frequency domain FLT file.

2.21 CSV Event Data File

A CSV event data file is an ASCII text file that contains event data in comma-delimited column format. Each line is a single event. This file format allows easy data import into mathematical software such as MS Excel.

There is one header line which contains titles for every column delimited by a comma character. Lines contain a number of data columns (Table 2.11). Through the pop-up dialogue that appears when the user selects to export, the user can select which blocks of information are to be exported to the file and can additionally select to only export enabled and located events.

The CSV event data file can also be imported into InSite. In this case, not all columns need to be present, but the header fields in the first row of the file must accord with the columns contained in the rows of data.

The CSV event data file can be used to update information for existing events in the InSite project. In this case not all columns need to be present, but the header fields in the first row of the file must accord with the columns contained in the rows of data. The events must be associated with valid ESFs.

Note: Distance and time units are set in the “Processing Settings” dialog. Distance units can be switched between Metric and Imperial by navigating to the “Event Initialisation” tab and checking/unchecking the “Use Imperial Distance Units” check box. Distance units can be further configured to either “Kilo” (10^3), “Normal”, “Centi” (10^{-2}), or “Milli” (10^{-3}). Time units can be configured to either “Normal”, “Milli” (10^{-3}), or “Micro” (10^{-6}). Output parameters using either distance or time will adjust to the type and magnitude.

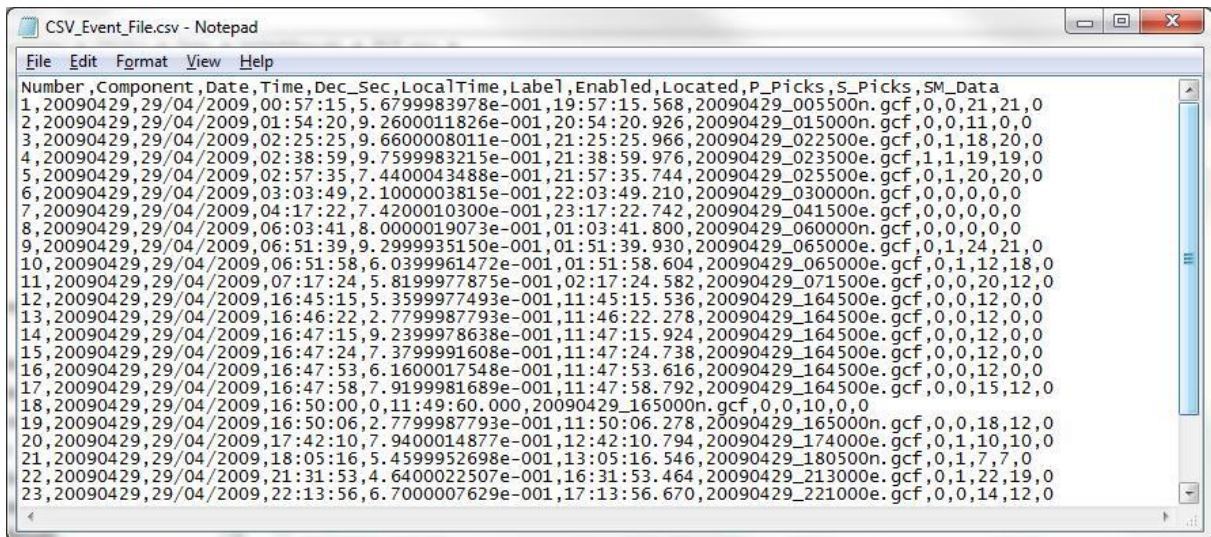


Figure 2.19: Example of a CSV format event data file.

Table 2.11: Description of data columns contained in a CSV format event data file.

Information Block	Column Title	Parameter Description	Units
Basic	Number	Event Number	
	Component	Component Name	
	Date	Date of Event (DD/MM/YYYY)	
	Time	Time of Event (HH:MM:SS)	
	Dec_Sec	Decimal Seconds of Event Time	
	Local Time	Local Time of Event (HH:MM:SS.SSS)	
	Label	Event Label	
	P_Picks	Number of P-waves picked	
	S_Picks	Number of S-waves picked	
Comp_Colour	Component colour	Colour used in 3D visualiser and charts	

Information Block	Column Title	Parameter Description	Units
Location Data	North	Location northing	See note above
	East	Location easting	See note above
	Down	Location depth	See note above
	Loc_Units	Units of location (referenced to 1m)	
	Loc_Error	Location RMS error	
	Num_Arrivals	Number of picks used in location algorithm (both P and S picks)	
	> Num_Ind	Number of independent instruments	
	Num_LR	Number of P-wave source vectors used in the location algorithm	
	Num_NLQ	Number of S-wave source vectors used in the location algorithm	
	T0	Event location T0 time	
	Cluster_Index	Event cluster index	
	Num_Neighbours	Number of interacting neighbours	
	Cluster_ID	Cluster Label	
	Confidence	Confidence Number	
	Radial_Distance	Radial Distance	
Loc_Mag	Location Magnitude		
Mom_Mag	Moment Magnitude		
Ins_Mag	Instrument Magnitude		
Seis_Mom	Seismic Moment	Nm	
Energy	Event radiated energy	J	
Es:Ep	S:P energy ratio		
P_fc	P-wave corner frequency	dB	

Source Parameters	S_fc	S-wave corner frequency	dB
	App_Stress	Apparent stress	Pa
	St_Sdrop	Static stress drop	Pa
	Radius	Source radius	
	Mon_Dist	Mean ray path length	
	SNR	Signal to Noise Ratio	
	RMS_Noise	RMS Noise Level	
	P_Noise	P-wave Signal to Noise Ratio	
	S_Noise	S-wave Signal to Noise Ratio	
	PSH	Mean P to SH amplitude ratio	
	PSV	Mean P to SV amplitude ratio	
Residual	Residual	Location RMS residual in travel times	
	Ang_Residual	RMS angular residual from source vectors	
	SM_Data	Flag to indicate there is source mechanism data	
	%ISO	Percentage of isotropic component in the moment tensor decomposition	
	%DC	Percentage of double-couple component in the moment tensor decomposition	
	%CLVD	Percentage of compensated linear vector dipole component in the moment tensor decomposition	

Information Block	Column Title	Parameter Description	Units
Source Mechanism Data	T_Value	T-value from source decomposition	
	K_Value	k-value from source decomposition	
	T_Error	Uncertainty in the T-value	
	K_Error	Uncertainty in the k-value	
	FP1_Azimuth	Azimuth of Fault Plane 1	Degrees
	FP1_Plunge	Plunge of Fault Plane 1	Degrees
	FP2_Azimuth	Azimuth of Fault Plane 2	Degrees
	FP2_Plunge	Plunge of Fault Plane 2	Degrees
Waveform Stats	Amp_Peak	Peak amplitude of the event	Sensor units
	P_AutoFunc	P-wave auto-pick function	
	S_AutoFunc	S-wave auto-pick function	

2.22 CSV Instrument Data File

A CSV instrument data file is an ASCII text file that contains instrument data in comma-delimited column format.

Each line is a single instrument. This file format allows easy data import into other software.

There is one header line which contains titles for every column delimited by a comma character. Lines contain a number of data columns (Table 2.12).

The CSV instrument data file can be used to update information for existing events in the InSite project. In this case not all columns need to be present, but the header fields in the first row of the file must accord with the columns contained in the rows of data. The events must be associated with valid ESFs.

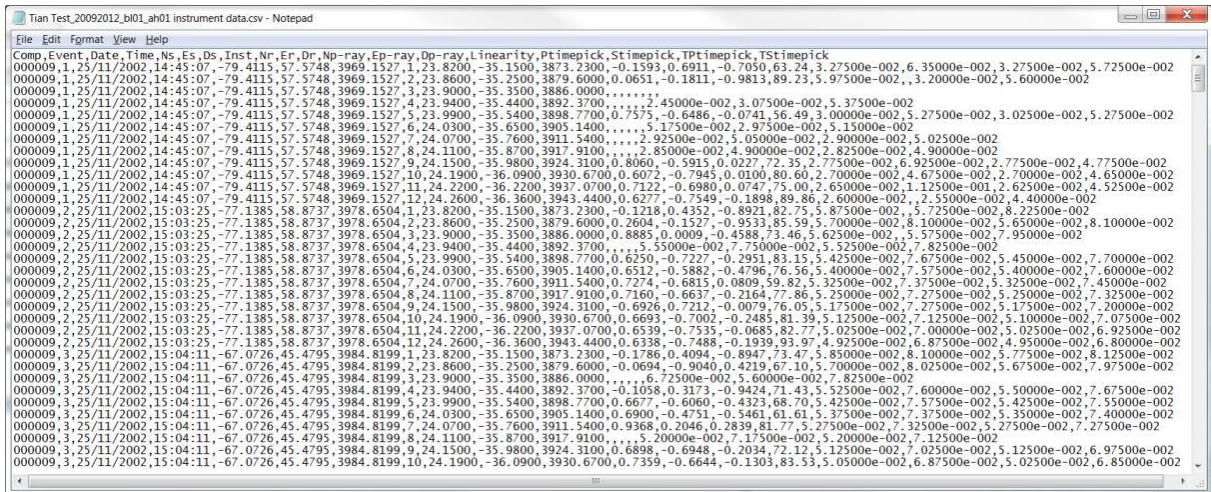


Figure 2.20: Example of a CSV format instrument data file

Table 2.12: Description of data columns contained in a CSV format instrument data file

Column Title	Parameter Description
Comp	Component Name
Event	Event Number
Date	Date of Event (DD/MM/YYYY)
Time	Time of Event (HH:MM:SS)
Ns	Location northing
Es	Location easting
Ds	Location depth
Inst	Instrument number
Nr	Instrument position northing
Er	Instrument position easting
Dr	Instrument position depth
Np-ray	P-ray source vector north component
Ep-ray	P-ray source vector east component
Dp-ray	P-ray source vector depth component
Linearity	P-ray source vector linearity
Ptimepick	Instrument P-wave pick time
Stimepick	Instrument S-wave pick time
Tp-timepick	Instrument theoretical P-wave pick time
Ts-timepick	Instrument theoretical S-wave pick time
PSH	Instrument P/SH amplitude ratio
PSV	Instrument P/SV amplitude ratio
SVSH	Instrument SV/SH amplitude ratio

RayLength	Instrument to event ray length
RayAzimuth	Instrument to event azimuth angle (degrees)
RayTakeoff	Instrument to event takeoff angle (degrees)

2.23 Well Visualiser Object File

This CSV file is used to import the different well object types (formation tops, packers, sleeves, and plugs) into the wells in Frac Manager. The file can contain all object types belonging to any well.

1	Type, Name, OwnerWell, MD, Length, Radius, ColourR, ColourG, ColourB
2	FORMATIONTOP, FT1, Frac_6A_16_79_22W5, 600, 200, 1, 255, 0, 0
3	PACKER, Packer1, Frac_6A_16_79_22W5, 1600, 100, 10, 43, 191, 186
4	SLEEVE, Sleeve1, Frac_6A_16_79_22W5, 1200, 100, 12, 243, 15, 14
5	PLUG, Plug1, Frac_6A_16_79_22W5, 1200, 20, 8, 144, 213, 60

Figure 2.21 shows an example file with an object of each type.

1	Type, Name, OwnerWell, MD, Length, Radius, ColourR, ColourG, ColourB
2	FORMATIONTOP, FT1, Frac_6A_16_79_22W5, 600, 200, 1, 255, 0, 0
3	PACKER, Packer1, Frac_6A_16_79_22W5, 1600, 100, 10, 43, 191, 186
4	SLEEVE, Sleeve1, Frac_6A_16_79_22W5, 1200, 100, 12, 243, 15, 14
5	PLUG, Plug1, Frac_6A_16_79_22W5, 1200, 20, 8, 144, 213, 60

Figure 2.21: Well visualiser object CSV file example

The first line of the file is the header, and each subsequent line contains data for a single well object. The columns from left to right as shown below:

Column Header	Description	Format
Type	Object type ID string: FORMATIONTOP – formation top object PACKER – packer object SLEEVE – sleeve object PLUG – plug object	Text string
Name	Well object name	Text string
OwnerWell	Owner well name – used to match it to an existing well in Frac Manager	Text string
MD	Measured depth – this value is used to interpolate along the well trak to calculate the true vertical depth	Float
Length	Object length	Float
Radius	Object Radius (does not affect formation top objects)	Float
ColourR	Red intensity (0-255) using RGB colour	Integer
ColourG	Green intensity (0-255) using RGB colour	Integer
ColourB	Blue intensity (0-255) using RGB colour	Integer
StartTime(dd/mm/yyyy hh:mm:ss)	The start time when the object appears in the 3D visualiser in the playback function.	String
EndTime(dd/mm/yyyy hh:mm:ss)	The end time when the object disappears in the 3D visualiser in the playback function.	String

2.24 PFC File Format

The PFC format is an ASCII text file used to import waveform data from Itasca's PFC software. The file consists of a three-line header followed by trace data arranged in columns.

Line 1 is the version number of PFC.

Lines 2 and 3 contain the event and trace information separated by commas (this information can be spread across both lines and in any order).

Header	Description
Date=DD-MM-YYY	Date in day-month-year format
InstrumentNums=int	Number of instruments (integer)
TracePoints=int	Number of points in each trace (integer)
TSamp=double	Sampling time (double)
TimeUnits=double	Time units (double) e.g. 1.0 is seconds
AmpToVolts=double	Amplitudes to volts ration (double)
TraceMaxVolts=double	Maximum voltage of the trace (double)
TimeUTC=int	Number of seconds elapsed since midnight 01/01/1970 (integer) – this is used as the start time of the event
Axes=int	Number of axes belonging to each instrument (integer)

Lines 4 downward contain the trace data. Traces are arranged by columns, that is column 1 corresponds to channel 1, column 2 corresponds to channel 2, column N corresponds to channel N, and so on. The total number of columns should be equal to the number of instruments multiplied by the number of axes each instrument has.

The component name is set from the parent folder containing the PFC files and the event label is set from the PFC file name. An example file is shown in Figure 2.22.

```

1 PFC-v1.00
2 Date=23-09-2014,InstrumentNums=125,TracePoints=100,TSamp=3.87603390744e-008,TimeUnits=1.0,
3 AmpToVolts=1.0000,TraceMaxVolts=4.00000000000e+001,TimeUTC=1414431752,Axes=3,
4 2.16300365439e-005,3.66876079980e-005,1.99452967226e-005,2.11642781177e-005,3.63459347521e-005,
5 1.91539010218e-005,3.32682683952e-005,1.85182296295e-005,1.88063178257e-005,3.33393297031e-005,
6 1.71559417248e-005,3.05370194094e-005,1.73221903687e-005,1.68448235534e-005,3.09624517260e-005,
7 1.55217194533e-005,2.83732276873e-005,1.62470839038e-005,1.51918695361e-005,2.91305812844e-005,
8 1.41620334553e-005,2.66420998642e-005,1.52202643754e-005,1.38190015064e-005,2.77755025738e-005,
9 1.30156783966e-005,2.52379713914e-005,1.42092046850e-005,1.27463929310e-005,2.68656029099e-005,
10 1.20487909829e-005,2.41006645250e-005,1.32172238213e-005,1.20152837572e-005,2.63917567516e-005,
11 1.12509522706e-005,2.32129621161e-005,1.22736872381e-005,1.16565983333e-005,2.63403978662e-005,
12 1.06285210474e-005,2.25864146826e-005,1.14212747362e-005,1.16682616405e-005,2.66708937528e-005,
13 1.01962658128e-005,2.22424373441e-005,1.07032273505e-005,1.20088705538e-005,2.73070860571e-005,

```

Figure 2.22: PFC file format

2.25 Waveform Parameters File

The [Waveform Parameters](#) file contains the waveform parameters for each trace belonging to an event. The file is comma separated value (CSV) format. Each line represents the waveform parameters for a single trace. The columns are as follows:

Column Name	Description
COMPONENT	Name of the component
EVENT	Event number
INSTRUMENT	Instrument number
TRACE	Channel/trace number
MAX_V	Maximum voltage of the signal
MAX_dB	Maximum decibel of the signal
COUNT	Number of threshold crossings
TIME_DURATION	Time between the first and last threshold crossing of the signal
TIME_RISE	Time between the first threshold crossing and the maximum amplitude of the signal
FREQ_AVG	Average frequency of the counts
FREQ_REVERB	Frequency of the counts after the maximum peak
FREQ_INITIAL	Frequency of the counts before the maximum peak
RA	Ratio of the rise time to the maximum amplitude
ENERGY	Measured Area under Rectified Signal Envelope (MARSE)

Chapter 3 Source Location Algorithms

[Velocity Models](#)

[Simplex Routine](#)

[Geiger Routine](#)

[Collapsing Grid Search Routine](#)

[Analysis of Source Location Uncertainties](#)

[Post Analysis of Source Locations](#)

[Clustering Analysis Techniques](#)

[Relative Location Routine](#)

[Advanced Source Location Algorithms](#)

3.1 Velocity Models

[Homogeneous Isotropic](#)

[Time-Dependent Homogeneous Isotropic](#)

[Transversely Isotropic](#)

[Time-Dependent Transversely Isotropic](#)

3.1.1 Homogeneous Isotropic

The homogeneous-isotropic velocity structure assumes a half space containing the source and all receivers. Each raypath is calculated as a vector between the source and the receiver. Velocity is homogeneous throughout the volume (i.e. no variations with position) and isotropic (no variations with direction). The following parameters are then required:

- P-wave transmission velocity
- S-wave transmission velocity
- P-wave anelastic attenuation factor (Q_p)

- S-wave anelastic attenuation factor (Q_S)

The anelastic attenuation factors are used in source parameter and mechanism studies.

3.1.2 Time-Dependent Homogeneous Isotropic

Identical to the Homogeneous-Isotropic model outlined [previously](#), but a different model can be defined for each time span. The model is input using a csv file which must have no time interval unaccounted for between the earliest start time and the latest end time and no overlapping if the time intervals. To progress to another time interval, the end time of the first must equal the start time of the next, and so on. The structure of the file is outlined in Table 2.1. Note that the file is used for both time-dependent homogeneous-isotropic and transversely isotropic models, but the anisotropy factors and directions are only used for transversely isotropic models but must still be defined in the file when using the homogeneous-isotropic model (with any direction and anisotropy factor =1).

Table 3.1: Time dependent homogeneous-isotropic model file structure

Column Number	Header	Description
1	VP	P-wave transmission velocity
2	VS	S-wave transmission velocity
3	PAni	P-wave anisotropy factor (Transversely Isotropic only)
4	Sani	S-wave anisotropy factor (Transversely Isotropic only)
5	Azimuth	Axis azimuth (from North) (Transversely Isotropic only)
6	Dip	Axis dip (Transversely Isotropic only)
7	Start_Year	Start time year (4-digit integer)
8	Start_Month	Start time month (2-digit integer)
9	Start_Day	Start time day (2-digit integer)
10	Start_Hour	Start time hour (2-digit integer in 24-hour time)
11	Start_Minute	Start time minute (2-digit integer)
12	Start_Second	Start time second (2-digit integer)
13	End_Year	End time year (4-digit integer)
14	End_Month	End time month (2-digit integer)
15	End_Day	End time day (2-digit integer)

16	End_Hour	End time hour (2-digit integer in 24-hour time)
17	End_Minute	End time minute (2-digit integer)
18	End_Second	End time second (2-digit integer)

3.1.3 Transversely Isotropic

The transversely-isotropic velocity structure assumes a half space containing the source and all receivers. Each raypath is calculated as a vector, r , between the source and the receiver, as in the homogeneous-isotropic structure described above.

Velocity varies depending on the ray orientation through the volume with respect to an axis of symmetry. The direction of the axis is defined by a vector, a , in a North-East-Down Cartesian coordinate system using an azimuth from North and a plunge down from the horizontal.

The velocity (P or S wave) along the raypath (V_r) is given by:

$$V_r = \left(\frac{V_{\parallel} + V_{\perp}}{2} \right) - \left(\frac{V_{\parallel} - V_{\perp}}{2} \right)_{\perp} \cos [\pi - 2\theta] \quad (3.1)$$

where θ is the angle (in radians) between r and a . V_{\parallel} is the phase velocity parallel to the axis of symmetry and is the input velocity used for homogeneous-isotropic. V_{\perp} is the velocity perpendicular to the axis and is calculated using,

$$V_{\perp} = \alpha \cdot V_{\parallel} \quad (3.2)$$

Where α is the Anisotropy Factor (allowed to vary in the range $0 < \alpha < 10$). Thus, if $\alpha < 1.0$ then the velocity along the axis represents a fast direction and if $\alpha > 1.0$ the velocity along the axis represents a slow direction. If $\alpha = 1.0$ then $V_{\perp} = V_{\parallel}$ and the velocity structure becomes homogeneous-isotropic.

3.1.4 Time-Dependent Transversely Isotropic

Identical to the [Transversely Isotropic](#) model, but a different model can be defined for each time span. The structure of the file is described in Table 3.1.

3.2 Simplex Routine

[Methodology](#)

[Specific Configuration Parameters](#)

3.2.1 Methodology

The Downhill Simplex Method (Nelder and Mead, 1965; Press et al., 1994) is an iterative procedure that searches the error space for a minimum. The method uses a geometrical shape termed a simplex. In three-dimensions this is a tetrahedron with each vertex defined by its spatial coordinates (x, y, z). The error space is calculated at each vertex and, depending upon the relative values of the error space, the simplex is then instructed to move or deflate for the next iteration. As such the simplex wanders around the error space until it settles into a minimum.

The application of the method to seismic studies used here is developed from Chow (1992) and Falls (1993). An error value is calculated for each arrival time (P or S-wave) on each sensor used in the algorithm, from the sum of the travel-time residuals across the array. The error space at the given point in space is then the mean of these arrival error values. The residuals can be either raised to the power of 1 (L1Norm) or to the power of 2 (L2Norm). The P and S-wave errors can be weighted, in a relative sense, so as to diminish the effect of either of the phase travel times.

An outlier identification procedure can be utilised to remove the arrival with the highest error above an error threshold defined by the "Arrival Error Factor".

3.2.2 Specific Configuration Parameters

There are a number of configuration parameters the user can set, as described in Table 3.2.

Table 3.2: Configuration parameters for the Simplex Location Algorithm

Parameter	Description	Example Value
Simplex Tolerance	The tolerance stops the simplex iteration when the fractional change in the error space across the simplex becomes less than the tolerance value.	0.1
LPNorm	Power to raise the travel time residuals.	1
P-wave Weighting	Relative weighting of the P-wave arrival errors	1.0
S-wave Weighting	Relative weighting of the S-wave arrival errors	0.5
Arrival Error Factor	The factor that an arrival error has to be greater than the mean error to produce an outlier	2.0

3.3 Geiger Routine

[Methodology](#)

[Error Ellipsoids](#)

[Specific Configuration Parameters](#)

3.3.1 Methodology

The Geiger algorithm solves for the origin time t_0 , and source location (x_0, y_0, z_0) , such that the sum of the square of the residuals is a minimum, where the residual r is equal to the observed time minus t_0 minus the calculated time at (x_0, y_0, z_0) . The algorithm iterates towards the correct location using the magnitudes of the time derivatives (the change in time for small changes in x , y , or z).

The routine uses SVD (Singular Value Decomposition) inversion from P- and/or S-wave arrival times. Advantages of the Geiger method are: (1) The robustness of the inversion is estimated by the condition number, and (2) advanced analysis of the covariance matrix can be made to give error ellipsoids (X, Y, Z error estimates).

The Geiger method is an inverse least-square (L2 norm) problem. The source location is defined by four parameters:

$$\theta = (t_0, x_0, y_0, z_0) \quad (3.3)$$

Namely the coordinates $h = (x_0, y_0, z_0)$, and the origin time, t_0 .

The time residual, r_i (where i equals 1 to the number of stations n) is the difference between the calculated arrival times, T_i , and the observed arrival times, t_i , corrected to the time zero of the event, t_0 :

$$r_i = t_i - t_0 - T_i \quad (3.4)$$

The function relating the arrival times and the location is nonlinear since there is no single step approach to find the best event location. The standard technique is to linearise the problem:

$$\theta = \theta^* + \Delta\theta \quad (3.5)$$

Where θ^* is a source location estimate near the true location, and $\Delta\theta$ is a small perturbation.

Using the first term in the Taylor series expansion, the observed times may be approximated by:

$$t_i = t_0^* + \delta t_0 + T_i(h^*) + \frac{\partial T_i}{\partial h} \Delta h \quad (3.6)$$

The time residuals at the location h are given by:

$$R_i(h^*) = t_i - t_0^* - T_i(h^*) \quad (3.7)$$

Combining the above equations gives:

$$r_i(h^*) = \Delta t_0 + \frac{\partial T_i}{\partial h} \Delta h r_i(h^*) = \frac{\partial T_i}{\partial \theta} \Delta \theta \quad (3.8)$$

And in matrix notation,

$$\mathbf{r} = \mathbf{A} \cdot \Delta \boldsymbol{\theta} \quad (3.9)$$

Following Gibowicz and Kijko (1994), the minimization of the sum of the squared time residuals can be given by:

$$\mathbf{b} = \mathbf{B} \cdot \Delta \boldsymbol{\theta} \quad (3.10)$$

where $\mathbf{b} = \mathbf{A}^T \mathbf{r}$, and $\mathbf{B} = \mathbf{A}^T \mathbf{A}$ and is termed the covariance matrix.

The Geiger location is found by choosing a starting location, solving the matrix problem (e.g. by Singular Value Decomposition) for $\Delta \boldsymbol{\theta}$, and then iterating until this adjustment parameter reaches a user set minimum.

3.3.2 Error Ellipsoids

Examples of data analysis using error ellipsoids can be seen in Young et al. (2002).

The solution covariance matrix, $\boldsymbol{\varepsilon}$, for the source location can be determined by Aki and Richards (2002):

$$\boldsymbol{\varepsilon} = \sigma^2 \cdot \mathbf{B}^{-1} \quad (3.11)$$

Where σ^2 is the data variance.

Jones and Stewart (1997) just consider the 3D spatial uncertainty ellipsoid, and apply a scaling factor, $\chi_3^2(\alpha)$, to the calculation:

$$\boldsymbol{\varepsilon}_s = \sigma^2 \mathbf{B}_s^{-1} \chi_3^2 \quad (3.12)$$

where the subscript s denotes the spatial partition, and $\chi_3^2(\alpha)$ is the value of the χ^2 distribution with degrees of freedom for some confidence level α .

$\boldsymbol{\varepsilon}_s$ is a 3x3 matrix which can be diagonalized (e.g. using Householder Reduction), with the resulting diagonal elements being the solution variance. The square roots of these three diagonal elements are used to define the

uncertainty ellipsoid in terms of the maximum, intermediate, and minimum error magnitudes (e_1 , e_2 , e_3) and directions.

Pavlis (1986) notes that for a single event least squares source location estimate, the total error is a combination of three terms: (1) measurement error, (2) velocity model error, and (3) a nonlinear term. As a first assumption many studies choose to ignore the terms 2 and 3 suggesting that the main source of error is from measurement

uncertainty. Jones and Stewart (1997) use this assumption and estimate the data variance from the standard error of the location residuals:

$$\sigma^2 = \frac{\sum_{i=1}^n r_i^2}{n - 4} \quad (3.13)$$

3.3.3 Specific Configuration Parameters

There are a number of configuration parameters the user can set.

Table 3.3 gives a description of each.

Table 3.3: Configuration parameters for the Geiger Location Algorithm

Parameter	Description	Default Value
Geiger Tolerance	Criteria for convergence. The summation of the differences between (x0, y0, z0) from two successive iterations needs to be less than the 'Tolerance' for a successful convergence.	1.0 in distance units
Step Size	Used in the time derivative calculation. Rule of thumb: For functions whose characteristic scale [e.g. x, y, or z] is of order unity, the 'Step Size' is typically set to be a few tenths.	0.1 in distance units
Maximum Iterations	On rare occasions, the Geiger routine can end up in an infinite loop never reaching convergence, and the Maximum Iterations parameter is used to stop this. If the maximum number of iterations is reached with no convergence, then residuals are calculated using the final source location, and the arrival time from the P- or S-wave with the highest residual is dropped. The Geiger routine then begins again from the starting location. If maximum iterations is reached again, then a second arrival time is dropped, and the routine restarts. If maximum iterations is reached for a third time, then the Geiger routine gives up, and says that no source location can be found.	100
Condition Number Limit	An idea of the robustness and reliability of the inversion process, related to how well conditioned the set of linear equations is. The lower the number the better, minimum is 1. In relation to the source location routine, it has a proportionality to the completeness of the seismic array geometry. Recommendation is to set this limit high so you do not initially exclude any data. Then analyse and compare the condition numbers from different events to look for outliers (possible data to exclude or to check the picks).	10000000

3.4 Collapsing Grid Search Routine

A collapsing grid search algorithm, using a single-velocity model, is implemented within InSite's Location Tool. In this case the algorithm operates in a similar manner to both the Geiger and Simplex algorithms in that:

1. the algorithm calculates a single velocity for a ray path defined between a position P and a receiver R;
2. the ray path is calculated as the three-dimensional vector (with azimuth, plunge and length) connecting the two positions P and R (i.e. a direct ray path);
3. the velocity for the ray path can be calculated using either the homogeneous-isotropic or transversely-isotropic velocity models defined by the user on the Transmission Medium dialogue box.

The algorithm searches a three-dimensional space defined by the user for the minimum misfit between the measured travel times picked for every receiver and the theoretical travel times calculated from the ray path and given velocity model. The search is performed over a regularly spaced grid.

Grid search algorithms are notoriously slow to perform in three dimensions because the algorithm must search every depth position within every easting position within every northing position. For a regular grid of equal dimensions, the number of searches to perform is thus N^3 , where N is the number of 3D-location nodes in each direction. For a location resolution of 1m in a 1km location cube the algorithm must therefore search 10^9 (1 billion) times, resulting in time consuming computations.

InSite therefore uses a collapsing approach in order to produce a more efficient grid search algorithm. In this case an initial coarse grid is first searched for the minimum misfit position, E_{m1} . The algorithm then assumes that this minima is spatially close to the global minima and automatically generates a smaller and finer grid (a collapsed grid) around this position. The minimum misfit within this grid is then found, E_{m2} , and another collapsed grid defined. The algorithm continues until a user-specified location resolution is obtained.

The algorithm assumes that the deepest minima it finds within a particular grid is not a local minima positioned at some distance from the required global minima. For a poor array geometry (where local minima might be expected) a user can mitigate against being caught by local minima by defining the extent to which the grid is collapsed (the Collapsing Buffer) and thus the size of volume in which the new search is performed. Larger collapsed grids result in more collapsing loops and thus longer computations.

The user defines the following configuration parameters on the Location Algorithm Properties dialogue:

1. The Northing, Easting and Depth limits of the grid-search volume;
 2. The Cell Dimension (Dc_1) – the starting cell dimension for the initial coarse grid;
 3. The Desired Resolution – the final resolution the user wishes to obtain;
 4. The Collapsing Buffer (Bc) – the half-width, in uncollapsed cells, of the new collapsed grid.
-

A collapsed grid has a size (volume limits) and cell dimension defined by the previous grid. The dimension of a collapsed cell is given by

$$D_{C_{i+1}} = \frac{D_{C_i}}{R} \quad (3.14)$$

where D_{C_i} is the Cell Dimension of the previous grid and R is the Collapsing Ratio.

The starting grid defined by the user can have any cuboid volume (rectangular prism), whereas collapsed grids are always a cube with side length given by

$$D_{g_{i+1}} = N \cdot D_{C_{i+1}} \quad (3.15)$$

, where N is the average number of axial cells in the starting grid. The number of axial cells along each direction

(North, East and Down) in the starting grid is e.g. $\frac{D_x}{D_{C_1}}$ and must always be 10. The cubic collapsed grid is defined D_{C_1} by the user-specified Collapsing Buffer, B_c , as illustrated in

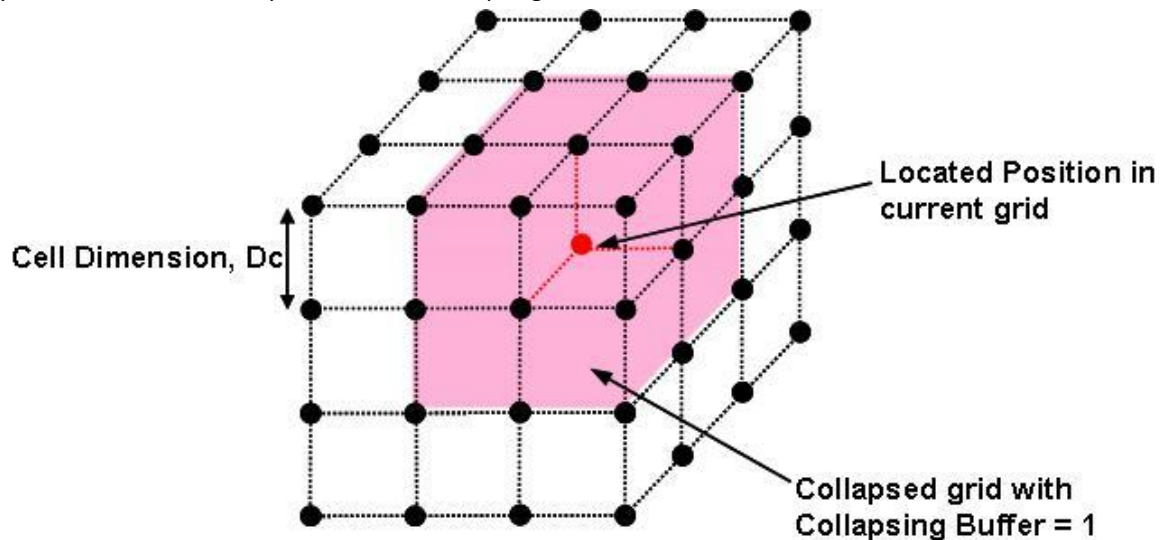


Figure 3.1, where B_c is the half-width of the new grid measured in number of uncollapsed cells. The new grid thus has a side length of $D_{g_{i+1}} = 2 \cdot B_c \cdot D_{C_i}$ centred on a given position in the uncollapsed grid. Thus, if $B_c = 1$ then the new collapsed grid has a volume consisting of 8 uncollapsed cells ($2 \times 2 \times 2$) centred on the given position. If $B_c = 2$ then the new collapsed grid consists of 64 uncollapsed cells ($4 \times 4 \times 4$) and so forth.

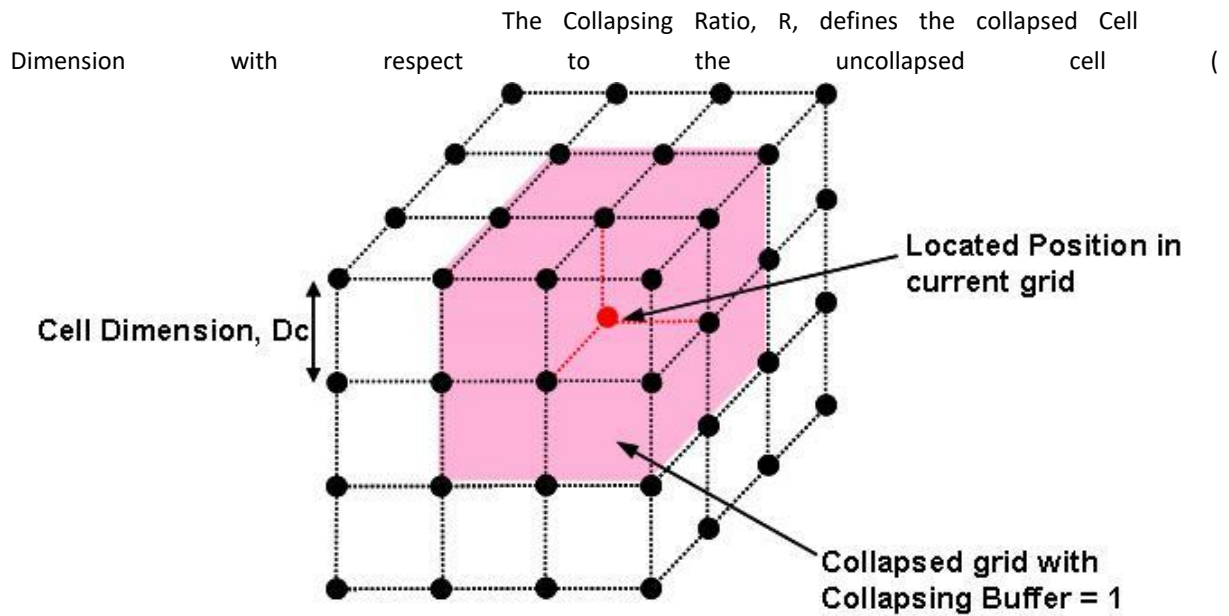


Figure 3.1).

The ratio is defined by:

$$R = \frac{N}{2 \cdot B_C} \tag{3.16}$$

The Collapsing Ratio must always have a value $R \geq 2.0$ to ensure effective and efficient collapsing.

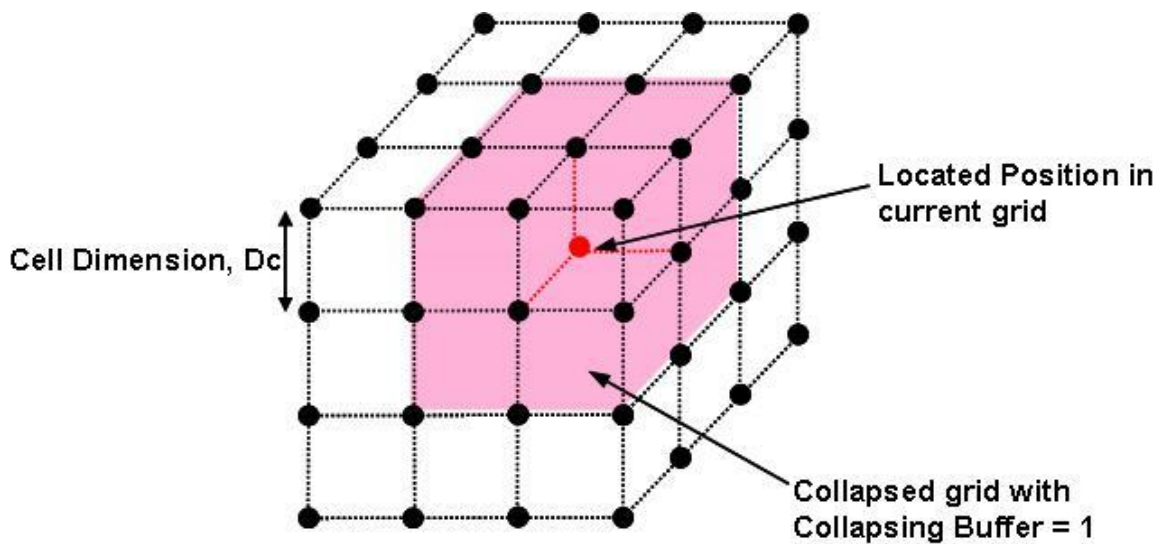


Figure 3.1: Definition of a collapsed grid volume

3.5 Analysis of Source Location Uncertainties

Two standard parameters are produced from the location algorithm to give a simple estimate of the location accuracy.

[Residuals and the RMS Residual](#)

[The RMS Error](#)

3.5.1 Residuals and the RMS Residual

A residual, ΔT_i , is calculated for each arrival time (P or S-wave) used in the location. The residual is the difference between the measured arrival time and the theoretical time calculated from the source location. The RMS Residual is then given by:

$$\Delta T_{RMS} = 0.5 \left(\sqrt{\frac{\sum_{i=1}^{N^P} \Delta T_i^P}{N^P}} + \sqrt{\frac{\sum_{i=1}^{N^S} \Delta T_i^S}{N^S}} \right) \quad (3.17)$$

where N^P is the number of P-wave arrivals over the array. NB: The 0.5 multiplication factor is dropped if there are zero P or S-wave arrivals used

3.5.2 The RMS Error

A source location error estimate is calculated from the residuals knowing the velocity structure of the medium

$$E_{RMS} = 0.5 \left(\sqrt{\frac{\sum_{i=1}^{N^P} \Delta T_i^P}{N^P}} v_P + \sqrt{\frac{\sum_{i=1}^{N^S} \Delta T_i^S}{N^S}} v_S \right) \quad (3.18)$$

NB: the 0.5 multiplication factor is dropped if there are zero P or S-wave arrivals used

3.6 Post Analysis of Source Locations

A number of the calculated output variables can be used to filter the source locations to isolate the locations with the highest accuracy. The source location error (or average residual) can be used with lower values generally representing more accurate source locations. However, it should be noted that the location error will generally be smaller if for example 5 arrival times are used compared to 15 arrival times. Therefore, it is also important to consider the number of arrival times used (accounting for the arrival times dropped).

3.7 Clustering Analysis Techniques

A large seismic data set can be difficult to interpret since the location error on each event is different and may vary significantly. Clustering, also known as collapsing, techniques are a method of perturbing events within their uncertainty ellipsoids, to search for structures (such as faults) in a dataset.

The method makes two main assumptions. The first is that each event analysed has an accurately calculated uncertainty ellipsoid. The second is that structures in a seismic data set are as simple as possible. One disadvantage of this is that two closely placed fractures may show as one structure following clustering, if the corresponding error ellipsoids are larger than the fracture spacing. The magnitude of the uncertainty ellipsoids is therefore a major factor on the resolution of the technique.

The method used here is modified from Jones and Stewart (1997). The first step is to calculate the uncertainty ellipsoid (equation) that bounds a required confidence level (i.e. a certain number of standard deviations). Jones and Stewart (1997) choose to use four standard deviations. An iterative method is then performed consisting of an inner and outer loop. The inner loop involves:

1. considering one seismic event location, the object event;
2. finding all events that locate inside the uncertainty ellipsoid of the object event;
3. calculating the centroid (mean location) of all these events including the object event location;
4. moving the object event a fraction of the distance towards the centroid.

Jones and Stewart (1997) suggest using a movement factor of 0.61803 of the total distance to the centroid, however they note the actual value does not seem to be critical. These four steps are taken for all individual events, resulting in a set of new calculated locations (which may or may not be different from the original locations). The outer loop involves calculating the total movement that has occurred, and if this value is above a user set value, then the inner loop is repeated using the new event locations. It has been found that after each successive iteration, the sum of the movements of the event locations is less than for the iteration before, indicating the locations are collapsing towards stable positions. In comparison to the convergence criteria used here, Jones and Stewart (1997) compare the distribution of event movements with a χ^2 distribution, and continue the algorithm until a user set misfit is reached.

An important point about this method is that the uncertainty ellipsoid of an object event is always centred about its original unclustered location, and therefore no event can move outside of its uncertainty volume.

3.8 Relative Location Routine

[Methodology](#)

[Specific Configuration Parameters](#)

3.8.1 Methodology

The Master event relative location method locates events relative to a previously located reference event (master), the location of which is assumed to be accurate. This method assumes that the ray paths from the master and process events to the sensors have approximately the same take off angle and traverse the same structures. With this assumption, the differential travel times solely depend on the seismic wave velocity in the volume between the master and the relocated event (Yang et al., 2002). These assumptions are approximately true if the hypocentral separation between the events is much smaller than their distance to the sensors and the sizes of the inhomogeneities on the ray paths between the event and the sensors.

The current version uses a set of P wave arrival-time differences recorded at stations $i = 1, \dots, N$ between the master and process event. The travel time for the master event r at station i can be written as (Gibowicz and Kijko, 1994) :

$$t_{ri} = t_{r0} + T(h_r, \langle m_i \rangle) + \Delta T(h_r, m_i) + \varepsilon_{ri} \quad (3.19)$$

where t_{r0} is the origin time, $T(h_r, \langle m_i \rangle)$ is the theoretical travel time from hypocentre h_r to station i calculated using an average velocity model $\langle m_i \rangle$, $\Delta T(h_r, m_i)$ is the difference between the travel time calculated using the average velocity model and the true model, and ε_{ri} is the reading error (assumed to follow a Gaussian distribution).

Similarly, for the process event we can write:

$$t_i = t_0 + T(h, \langle m_i \rangle) + \Delta T(h, m_i) + \varepsilon_{ri} \quad (3.20)$$

The last expression can be approximated by its first order Taylor expansion around the reference event (with hypocentre coordinates x_0, y_0, z_0) as follows:

$$t_i = t_{r0} + T(h_r, \langle m_i \rangle) + \frac{\partial T_i}{\partial x_0} \delta x_0 + \frac{\partial T_i}{\partial y_0} \delta y_0 + \frac{\partial T_i}{\partial z_0} \delta z_0 + \Delta T(h, m_i) + \varepsilon_i \quad (3.21)$$

A discussion on the effect of neglecting the higher order terms in the Taylor expansion on the accuracy of the relative location of teleseismic events can be found in Spence (1980).

Subtracting the above equation for t_i and the equation t_{ri} we get:

$$\delta t_i = t_i - t_{ri} = \delta t_0 + \frac{\partial T_i}{\partial x_0} \delta x_0 + \frac{\partial T_i}{\partial y_0} \delta y_0 + \frac{\partial T_i}{\partial z_0} \delta z_0 \quad (3.22)$$

where $\Delta T(h_r, m_{ri})$ and $\Delta T(h, m_i)$ have cancelled as we can assume that the process and reference events are close enough so that the ray paths to station i traverse the same structure. In matrix notation can be written as:

$$A\delta\theta = \delta t \quad (3.23)$$

where δt is the 4×1 matrix of unknown deviations from the reference arrival time t_0 , and hypocentre coordinates x_0, y_0, z_0 ; A is the $N \times 4$ partial derivatives matrix and t is the $N \times 1$ matrix formed by the arrival time differences at the N sensors.

If the velocity model described by the first arrival velocity (v_{Pi}), the travel time equation for sensor i can be simplified as $T_i = \frac{d_i}{v_{Pi}}$, where d_i is the Euclidean distance between the master event and sensor i . Hence, the derivatives in matrix A can be expressed as (Gibowicz and Kijko, 1994):

$$\begin{aligned} \frac{\partial T_i}{\partial y_0} &= \frac{1}{v_{Pi}} \frac{\partial [\sqrt{(x_0 - x_i)^2 + (y_0 - y_i)^2 + (z_0 - z_i)^2}]}{\partial y_0} = \frac{y_0 - y_i}{v_{Pi} d_i} \\ \frac{\partial T_i}{\partial z_0} &= \frac{1}{v_{Pi}} \frac{\partial [\sqrt{(x_0 - x_i)^2 + (y_0 - y_i)^2 + (z_0 - z_i)^2}]}{\partial z_0} = \frac{z_0 - z_i}{v_{Pi} d_i} \\ \frac{\partial T_i}{\partial x_0} &= \frac{1}{v_{Pi}} \frac{\partial [\sqrt{(x_0 - x_i)^2 + (y_0 - y_i)^2 + (z_0 - z_i)^2}]}{\partial x_0} = \frac{x_0 - x_i}{v_{Pi} d_i} \end{aligned} \quad (3.24)$$

In the inversion, v_{Pi} is taken as the average P-wave velocity in the volume around the master event, which is similar to that from the Geiger procedure for single event location, however the solution vector does not converge to zero, and no iteration is required in the process of reaching a solution, as the location of the master event is assumed to be accurate. Also, the derivative matrix only has to be calculated once, as it depends solely on the location of the master event and the array's sensors. The final relative coordinates of the process event are $t_0 + \delta t_i$, $x_0 + \delta x_i$, $y_0 + \delta y_i$ and $z_0 + \delta z_i$. The vector is the solution of the system of linear equations

$$A^T A \delta\theta = A^T \delta t \quad (3.25)$$

which can be solved using Singular Value Decomposition (SVD) inversion as described in **Press et al. (1994)**. In this method, the matrix $B = A^T A$ is decomposed as the product:

$$B = U W V^T \quad (3.26)$$

where U and V are orthogonal, and W is a diagonal matrix. The column matrix can be calculated as:

$$\delta\theta = V \cdot W^{-1} U^T A^T \delta t \quad (3.27)$$

The process follows two iterations. The second iteration rejects the instruments with large residuals, in order to remove the effect of outliers in the final location. Removed instruments are those with residual:

$$r_i \geq r + \alpha \cdot \sigma_r \quad (3.31)$$

where r_i is the residual for the i^{th} instrument, r is the median of residuals for all instruments, σ_r is the standard deviation for the residual distribution. α is a factor that defines the rejection level at standard deviations.

3.8.2 Specific Configuration Parameters

A number of configuration parameters described in the [previous section](#) can be set by the user.

Table 3.4: Configuration parameters for the Relative Location Algorithm.

Parameter	Description
Maximum Separation from Master Event (m)	Limits the maximum distance from the master event for events to be processed. This distance is dependent on the nature of the experiment and needs to be at least an order of magnitude smaller than the average separation between the master and receivers. Formal location error is found to be dependent on the distance to the master (Reyes-Montes et al., 2005)
Local P-wave Velocity (m/s)	Velocity in the volume between Master and Process events
Outlier Dropping Factor	α in $r_i \geq r + \alpha \cdot \sigma_r$ see equation (3.31) .
Condition Number Limit	Similar to the number used in the Geiger_Routine .

3.9 Advanced Source Location Algorithms

[TravelTime Modelling Wavefront Construction](#)

[Tian Raytrace](#)

[VTI Tian Raytrace](#)

[Fast Sweeping Method](#)

[Source Scan Method](#)

3.9.1 Travel Time Modelling Wavefront Construction

The travel-time modelling uses a "dynamic" Wavefront Construction (WFC) technique in two-dimensions (2D), based on Vinje et al. (1993). The principle behind the WFC technique is to dynamically produce a grid of travel times across the volume of interest. The grid is defined by a set of time "nodes" connected by wavefronts and raypaths (

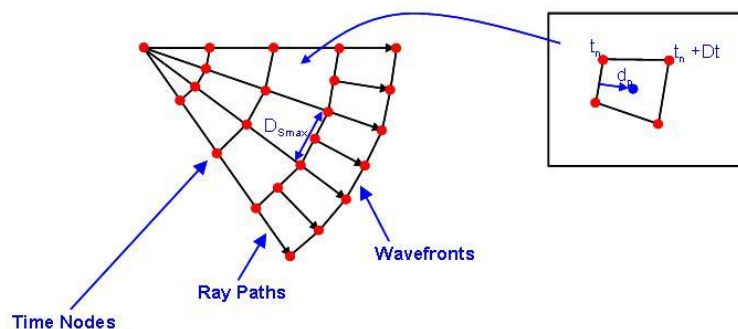


Figure 3.2). A wavefront consists of nodes at equal travel time and a ray path consists of nodes at equal take-off angle. Time nodes are positioned in 2D space (distance from source versus depth) and are obtained by propagating the ray path forward a small time interval t . One of the key features of the method presented by Vinje et al. (1993) is that once two adjoining nodes are separated by a distance greater than a fixed parameter D_S , then a new time node is interpolated between the two nodes on the wavefront. Thus, as the ray paths are stretched out by geometrical spreading new ray paths are brought into the grid so as to retain accuracy when interfaces are met. This interpolation feature is included in the WaveBuilder algorithm, although other features of the Vinje method have been adapted for more efficient processing. These most notably include:

- Ray paths are propagated forward the small time step assuming a homogeneous path;
- Ray paths are bent once a discrete interface is observed along the ray segment – bending follows Snell's law and removes the requirement for a smoothed velocity model;
- The wavefront segment between adjoining time nodes is linear.

A time pixel is defined as four adjoining time nodes (on two ray paths and two wavefronts) – the travel time to any position within the grid can then be calculated (

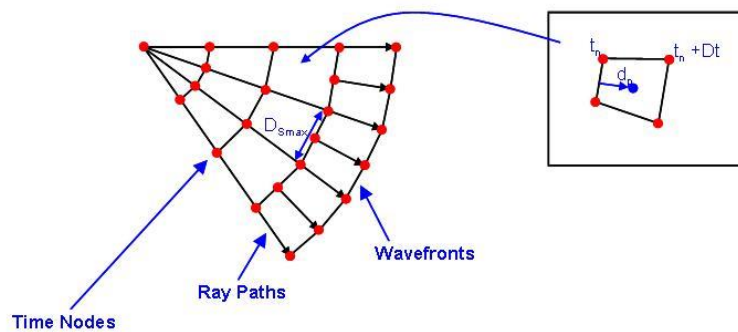


Figure 3.2) where the velocity within the pixel is assumed to be homogeneous.

The time grid is completely propagated across a 2D plane that extends from the instrument beyond the maximum boundaries of a 3D location volume defined by the user. The time-node information is stored to a binary ALT file (and to a TXT file if the number of nodes is sufficiently small, < 100000). Two examples of travel-time grids produced by the WaveBuilder algorithm are provided in the user manual (section 7.4 of the user manual).

The time grid is completely "pixelated" by determining all time pixels across the plane. Information on the pixels is stored to a binary ALP file and enables faster searching for any four bounding time nodes on the grid.

Finally, a "coarse" 3D location grid is defined through the location volume given by the user. Each 3D position on the location grid is translated to a 2D position on the time grid and the bounding time pixel found. References to the time pixels across the entire 3D location grid are stored to a binary ALL file.

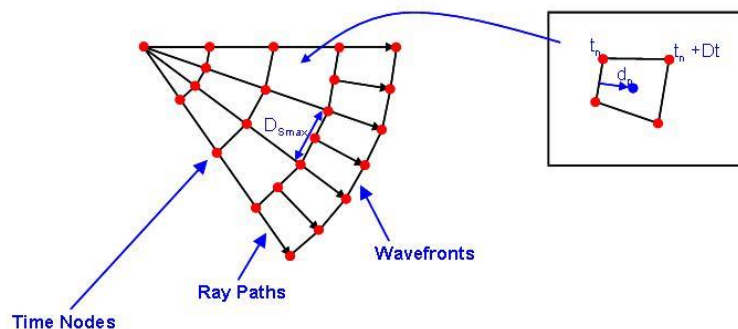


Figure 3.2: Illustration of a travel-time grid produced through Wavefront Construction (WFC) in InSite. Wavefronts are separated by a constant time difference, Dt . The inset box illustrates a time pixel bounded by four nodes.

The travel time to any arbitrary position on the time grid can be computed by calculating the orthogonal distance, d_p , from the wavefront t_n . A homogeneous velocity is assumed in the time pixel.

3.9.2 Tian Raytrace

This algorithm model travel times using a two-point ray tracing for a horizontally layered velocity model. The method is based on the algorithm presented by Tian and Chen (2005), expanding the algorithm to account for potential headwaves generated from total reflection at the interface of the different layers.

The algorithm loops through all grid points within the search volume computing the raypaths connecting to each station, taking into account the diffraction at each interface. The algorithm searches iteratively for the ray parameter p that solves the epicentral equation:

$$\Delta = p \sum_{k=1}^L \frac{\tilde{h}_k v_k}{\sqrt{1 - p^2 v_k^2}} \quad (3.32)$$

where L is the number of layers, \tilde{h}_k is the equivalent thickness (thickness of the portion of layer traversed by the ray) and v_k is the propagation velocity of the phase considered. The solution ray parameter p is constant for the raypath and provides the take-off angle θ_k at each layer k (provided the wave velocity v_k is known).

The algorithm also computes travel times for potential headwaves travelling up and down from the source. A head-wave can be generated at every interface between two layers provided that there is a strong contrast of transmission velocity, with higher impedance (lower velocity) in the second layer.

The final travel time is calculated as the lowest between that of the direct (refracted) ray and every possible head-wave. The process is repeated following the same method for S-waves.

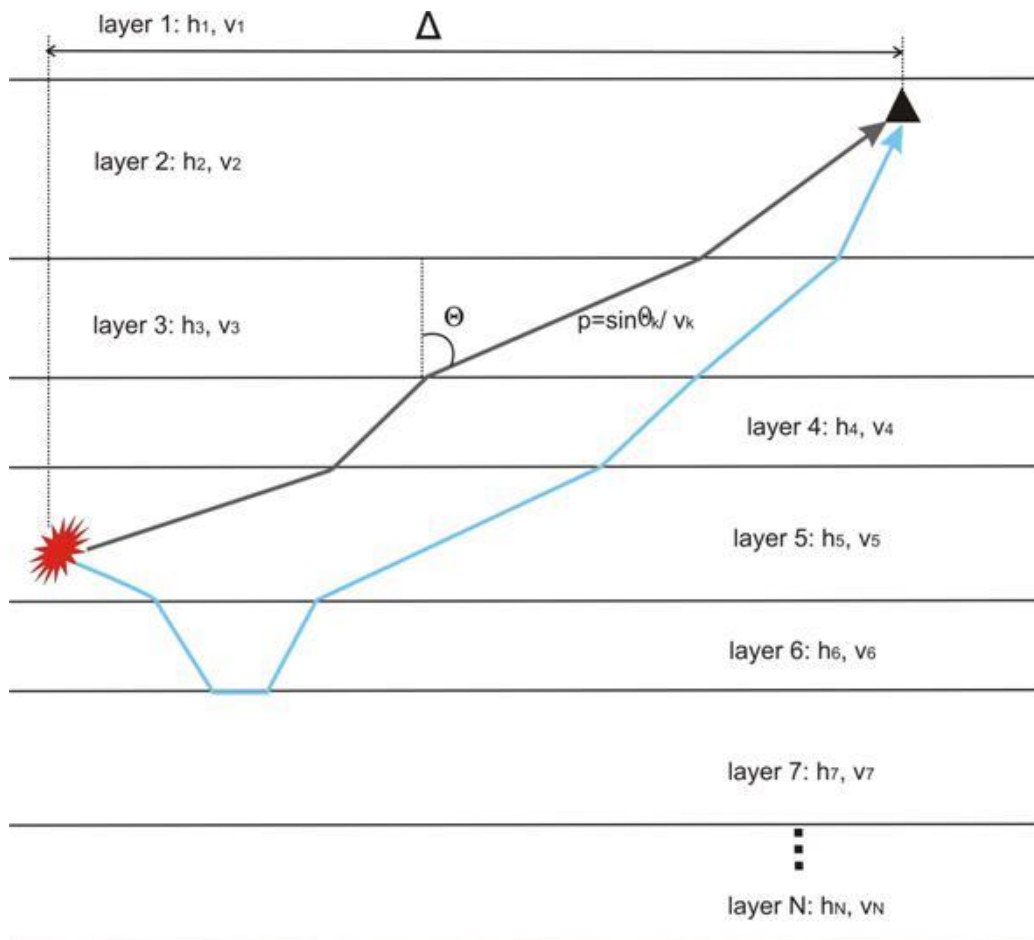


Figure 3.3: Illustration of a travel-time grid produced through Tian Raytracing

An extension to the Tian two-point ray tracing method allows for travel-time computation through a model of parallel dipping layers. The ray tracing is performed in a section containing the receiver and source position. This section is at an arbitrary azimuth to the strike of the layers and thus ray tracing is performed through interfaces that have an apparent dip calculated from the provided strike and true dip.

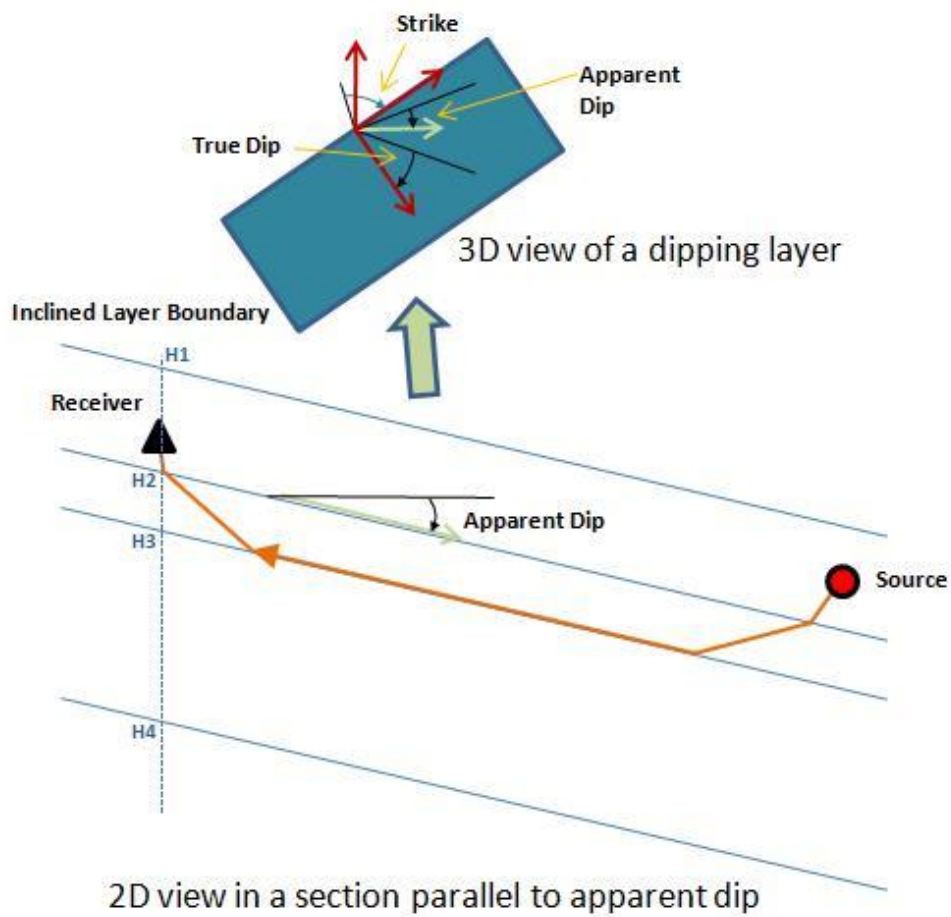


Figure 3.4: Illustration of two-point ray tracing through parallel dipping layers. H1 to H4 are interface heights vertically above and below the receiver.

3.9.3 VTI Tian Raytrace

The general anisotropic medium is fully characterized by 21 independent elastic parameters. Following the Voigt recipe, the four-order density normalized elastic tensor a_{ijkl} ($i, j, k, l = 1, 2, 3$) can be expressed by a 6x6 symmetric matrix:

$$A = \begin{bmatrix} A_{11} & A_{12} & A_{13} & A_{14} & A_{15} & A_{16} \\ A_{12} & A_{22} & A_{23} & A_{24} & A_{25} & A_{26} \\ A_{13} & A_{23} & A_{33} & A_{34} & A_{35} & A_{36} \\ A_{14} & A_{24} & A_{34} & A_{44} & A_{45} & A_{46} \\ A_{15} & A_{25} & A_{35} & A_{45} & A_{55} & A_{56} \\ A_{16} & A_{26} & A_{36} & A_{46} & A_{56} & A_{66} \end{bmatrix} \quad (3.33)$$

The element of the Christoffel matrix $\Gamma_{3 \times 3}$ are given by

$$\begin{aligned} \Gamma_{11} &= A_{11}p_1^2 + A_{66}p_2^2 + A_{55}p_3^2 + 2(A_{16}p_1p_2 + A_{15}p_1p_3 + A_{56} + p_2 + p_3) \\ \Gamma_{22} &= A_{66}p_1^2 + A_{22}p_2^2 + A_{44}p_3^2 + 2(A_{26}p_1p_2 + A_{46}p_1p_3 + A_{24} + p_2 + p_3) \\ \Gamma_{33} &= A_{55}p_1^2 + A_{44}p_2^2 + A_{33}p_3^2 + 2(A_{45}p_1p_2 + A_{35}p_1p_3 + A_{34} + p_2 + p_3) \\ \Gamma_{12} &= A_{16}p_1^2 + A_{26}p_2^2 + A_{45}p_3^2 + (A_{12} + A_{66})p_1p_2 + (A_{14} + A_{56})p_1p_3 + (A_{46} + A_{25})p_2p_3 \\ \Gamma_{13} &= A_{15}p_1^2 + A_{46}p_2^2 + A_{35}p_3^2 + (A_{14} + A_{56})p_1p_2 + (A_{13} + A_{55})p_1p_3 + (A_{36} + A_{45})p_2p_3 \\ \Gamma_{23} &= A_{56}p_1^2 + A_{24}p_2^2 + A_{34}p_3^2 + (A_{46} + A_{25})p_1p_2 + (A_{36} + A_{45})p_1p_3 + (A_{23} + A_{44})p_2p_3 \end{aligned} \quad (3.34)$$

where p_1, p_2, p_3 are the three components of the slowness vector p , i.e., the direction of the phase velocity and normal to the wavefront. The matrix has three eigenvalues $G_m(x, p)$ and eigenvectors $g_m(x, p)$ with $x \in R_3$, $m = 1$ for qP-wave and $m = 2; 3$ for two qS-waves. If the slowness vector is known at a given point x in the model, the ray vector $r = [r_1; r_2; r_3]^T$ generally not in parallel with the slowness vector can be calculated as (Cerveny, 2001):

$$r_i = a_{ijkl}p_l g_j g_k \quad (3.35)$$

The direction of the ray vector and the magnitude represent the direction and the value of the group velocity, respectively.

Using the element values of transversely isotropic medium,

$$A^{TI} = \begin{bmatrix} a_{11} & a_{11} - 2a_{66} & a_{13} & 0 & 0 & 0 \\ a_{11} - 2a_{66} & a_{11} & a_{13} & 0 & 0 & 0 \\ a_{13} & a_{13} & a_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & a_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & a_{44} & 0 \\ 0 & 0 & 0 & 0 & 0 & a_{66} \end{bmatrix} \quad (3.36)$$

and the direction of phase velocity (θ, ϕ) , where θ is the inclination, i.e., the angle between the direction and the positive vertical z -axis, and ϕ is the azimuth, i.e., the angle between the direction and the positive x -axis, we can express the phase velocity for P, SH, and SV wave as a function of (θ, ϕ) :

$$C_P(\theta) = \sqrt{\frac{(a_{33} + a_{44}) \cos^2(\theta) + (a_{11} + a_{44}) \sin^2(\theta) + A}{2}} \quad (3.37)$$

Where

$$A = \sqrt{(a_{33} - a_{44})^2 \cos^4(\theta) + (a_{11} - a_{44})^2 \sin^4(\theta) + 2((a_{11} + 4a_{13} + a_{33})a_{44} - a_{11}a_{33} + 2a_{13}^2 + a_{44}^2) \sin^2(\theta) \cos^2(\theta)} \quad (3.38)$$

$$C_{SV}(\theta) = \sqrt{\frac{(a_{33} - a_{11}) \cos^2(\theta) + (a_{11} + a_{44}) - B}{2}} \quad (3.39)$$

Where

$$B = \sqrt{-4(a_{11} + 2a_{13} + a_{33})a_{44} + 4a_{13}^2 - a_{11}^2 - 2a_{11}a_{33} - a_{33}^2 \cos^4(\theta) + D} \quad (3.40)$$

And

$$D = 2((3a_{11} + 4a_{13} + a_{33})a_{44} - a_{11}^2 - a_{11}a_{33} + 2a_{13}^2) \cos^2(\theta) + (a_{11} - a_{44})^2 \quad (3.41)$$

$$C_{SH}(\theta) = \sqrt{(a_{44} - a_{66}) \cos^2(\theta) + a_{66}} \quad (3.42)$$

We notice that the phase velocity is independent from the azimuth in VTI media

Given the slowness vector expressed as

$$\mathbf{p} = [p_1, p_2, p_3]^T = \left[\frac{\sin(\theta) \cos(\phi)}{c(\theta)}, \frac{\sin(\theta) \sin(\phi)}{c(\theta)}, \frac{\cos(\theta)}{c(\theta)} \right]^T \quad (3.43)$$

the value of the group velocity can be calculated from the above equation for r_i . The phase velocity takes the value from the above equation for $c(\theta)$ depending on the ray type of interest.

In the VTI Tian Raytracing method, the ray path from the Tian Raytracing method in isotropic media is taken as the phase direction (θ, ϕ) and the group velocity is calculated in this direction. The travel time is then recalculated based on the ray path and the group velocity for the ray path direction. This first order approximation works well for weak anisotropy (see definition of weak anisotropy in Thomsen (1986), i.e. the error in the travel time is usually less than 10 ms for weak to moderate anisotropic media. The Thomsen's parameters $(\epsilon, \gamma, \delta)$ together with V_p , and V_s along the symmetry axis is uniquely related to the five elastic parameters in the above equation for A^T .

3.9.4 Fast Sweeping Method

The isotropic version of fast sweeping method models travel times by solving Eikonal equation for a 3D heterogeneous isotropic velocity model. The Eikonal equation belongs to the class of static Hamilton-Jacobi equations. In seismology, it can be derived from the full elastic wave equation with the high frequency approximation as expressed in equation:

$$|\nabla T(x)|^2 = \frac{1}{c^2(x)}, \quad x \in \Omega \setminus \Gamma \quad (3.44)$$

With the boundary condition:

$$T(x_s) = g(x_s), x_s \in \Gamma \quad (3.45)$$

where $T(x)$ is the traveltimes at location x , and $c(x)$ are the propagation velocity, respectively, $\Omega = R^n$ for a n -dimensional problem, and Γ is the boundary with known function of $g(x_s)$. In the case of point source, the boundary values are defined at a point for first arrivals.

There are generally two different methods used to solve a static Hamilton-Jacobi equation. One converts the above [Eikonal Equation](#) from a static to time-dependent state with the level set method (Osher, 1993) or the paraxial formulation (Qian and Symes, 2002). A variety of numerical methods can be applied to solve the time-dependent Hamilton-Jacobi equations (see [Shu \(1997\)](#) and references therein). The other method is to directly discretize the [Eikonal Equation](#) as a stationary boundary value problem and tracks the wavefront by following the causality of wave propagation. The fast marching method (FMM) and fast sweeping method (FSM) fall in this category. FMM was initially developed in the field of computational mathematics to solve interface evolution problems ([Sethian, 1996](#)). Starting from the boundary, FMM stores all active nodes (the wavefront) in a heap data structure named Narrow Band and uses an upwind finite difference method to calculate traveltimes at all the neighbouring nodes. To ensure the causality, only the grid point with minimum traveltimes is updated and the wavefront is moved one step forward. The complexity of FMM is $O(N \log N)$ for a N grid model, where $O(\log N)$ is from the heap sorting algorithm. FSM (Tsai et al., 2003; Zhao, 2004) uses Gauss-Seidel iterations and alternating sweeping orders such that each sweeping direction covers a class of wave propagation directions. During each sweep, the traveltimes can only monotonically increase to ensure the causality of wave propagation and the complexity drops to $O(N)$. We provide a brief description on how FSM is used to solve both direct and reflected traveltimes from the above [Eikonal Equation](#).

The gradient of misfit function of direct and reflected arrivals is then derived using adjoint method.

We apply the fast sweeping algorithm to solve the Eikonal equation with a point source in a three-dimensional (3-D) medium:

$$|\nabla T(x)|^2 = \frac{1}{c^2(c)}, x \in R^3 \setminus x_s \quad (3.46)$$

$$T(x_s) = 0 \quad (3.47)$$

where x_s is the location of the point source and $c(x)$ is the propagation velocity.

Using the first order Godunov upwind finite difference scheme (Rouy and Tourin, 1992), we discretize the above equation as follows:

$$\left[\frac{(T(x) - T(x)^{x_{min}})^+}{\Delta x} \right]^2 + \left[\frac{(T(x) - T(x)^{y_{min}})^+}{\Delta y} \right]^2 + \left[\frac{(T(x) - T(x)^{z_{min}})^+}{\Delta z} \right]^2 = \frac{1}{c^2(x)} \quad (3.48)$$

where $T(x)$ is the discrete approximation at the grid $x = (i; j; k)$, the indices of $i; j; k$ are running from 0 to $N_p - 1$, and $T(x)_{p_{min}}$ is the minimum traveltime value of two adjacent neighbours of $T(x)$ along the axis of p (p is x, y , or z). For example, $T(x)_{x_{min}} = \min(T(x + \Delta x, y, z); T(x - \Delta x, y, z))$. The $\Delta x, \Delta y$, and Δz are the grid intervals along x, y and z directions, respectively, and the total number of grid points of the model is $N_x \times N_y \times N_z$. The superscript '+' is an operator taking the positive value only and expressed as:

$$(A)^+ = \begin{cases} A, & A > 0 \\ 0, & A \leq 0 \end{cases} \quad (3.49)$$

where A is an arbitrary value evaluated by the operator. Note that one-sided finite difference must be used when x is at the edge of the modelling region (e.g., $x = (0, j, k)$ or $x = (N_x - 1, j, k)$). In this case, the first term in the equation becomes:

$$\left[\frac{(T(0, j, k) - T(l, j, k))^+}{\Delta x} \right]^2 \text{ or } \left[\frac{(T(N_x - 1, j, k) - T(N_x - 2, j, k))^+}{\Delta x} \right]^2 \quad (3.50)$$

Here, we start the array index from zero.

The fast sweeping algorithm can thus be implemented as follows.

1. Initialize the point source condition $T(x_s) = 0$ and assign very large positive values to the rest of the grid points which will be updated.
2. Update grid points with Gauss-Seidel iterations with 8 alternating sweeping orders. At each node, the solution from equation 3.50 denoted as T^* is compared with the old value (T^{old}) and the traveltime at this node T^{new} is updated with the smallest value between the old and calculated traveltime value, i.e., $\min(T^{old}; T^*)$. It is important to mention that the sweeping order is not necessary applied in sequential order and can be performed in parallel on multiple processors.
3. Test the convergence by checking the criterion $\|T^{n+1} - T^n\|_{L1}$ point-wise, for a given convergence criterion $\epsilon > 0$.

During the sweeping of the whole domain (step 2), a unique solution must be obtained from the equation to update node $T(i; j; k)$. The following algorithm (steps 2.1 to 2.3) is used to systematically find that unique solution:

2.1 Assign the values of $T(x)^{x_{min}}$, $T(x)^{y_{min}}$; and $T(x)^{z_{min}}$ to u , v , and w such that $w \leq v \leq u$ and let $T^* = w + 1/c(x)$. For clarity of annotation in steps 2.2 and 2.3, u , v , and w are intermediate variables used to store the values of $T(x)^{x_{min}}$, $T(x)^{y_{min}}$, and $T(x)^{z_{min}}$.

2.2 If T^* is less than v , then T^* is the solution. Otherwise, let,

$$T^* = \frac{v + w + \sqrt{-v^2 - w^2 + 2vw + \frac{2}{c^2}}}{2} \tag{3.51}$$

2.3 If T^* is less than u , then T^* is the solution. Otherwise, the solution is:

$$T^* = \frac{2(u + v + w) + \sqrt{4(u + v + w)^2 - 12(u^2 + v^2 + w^2 - \frac{1}{c^2})}}{6} \tag{3.52}$$

Here we show an example of seismic wave propagation in a 3-D layered media model comprising 10^6 grid points obtained with the fast sweeping algorithm (

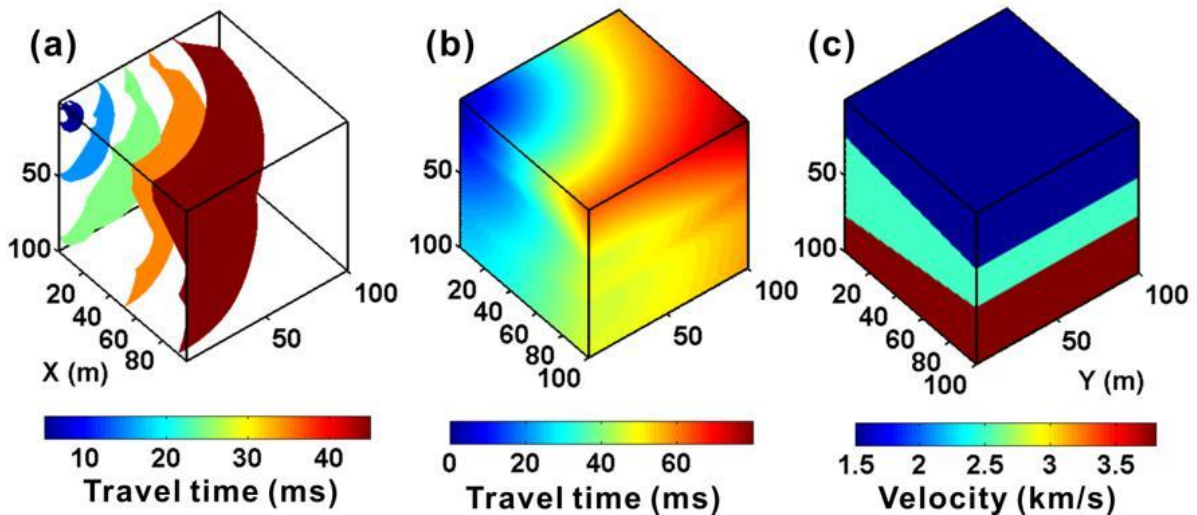


Figure 3.5). The iso-surface of different arrival times are rendered using different colours (

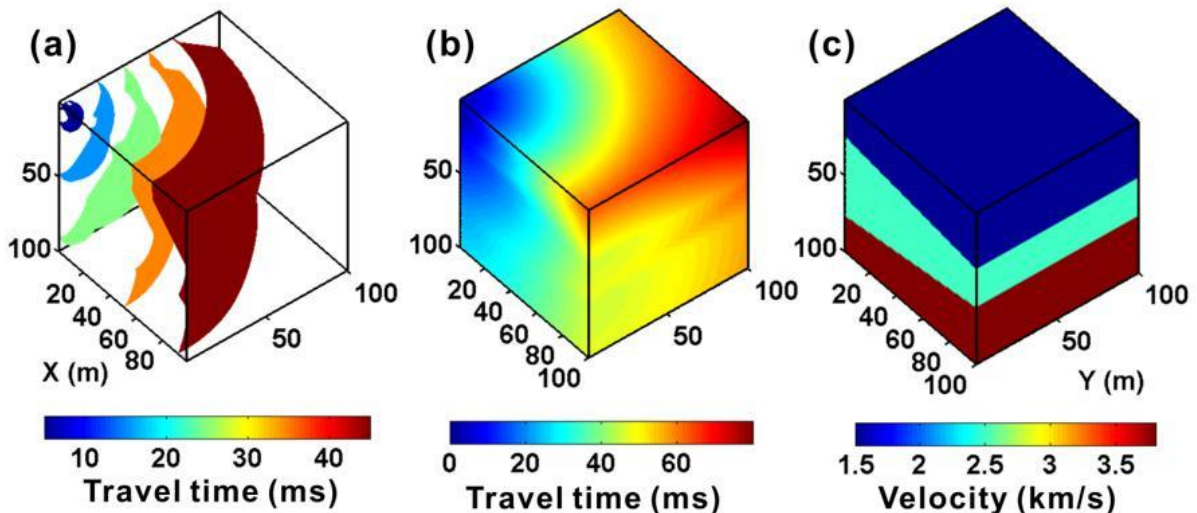


Figure 3.5a) and the first arrivals are obtained at each grid point of the volume (

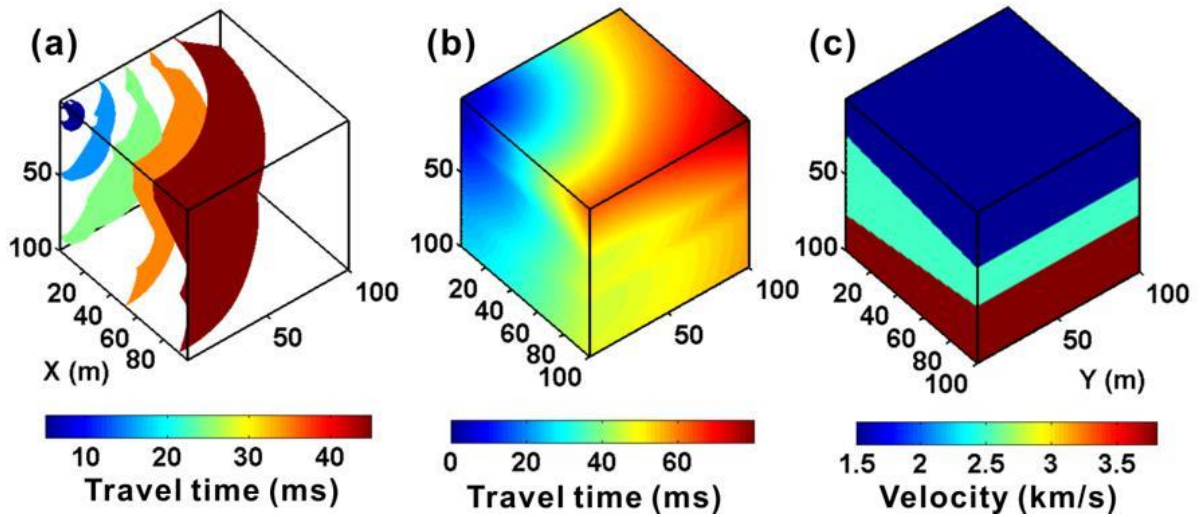


Figure 3.5b). In this example, the source is located at the (5, 5, 10) grid point and a total of 4 iterations were completed on a single CPU in 7 seconds.

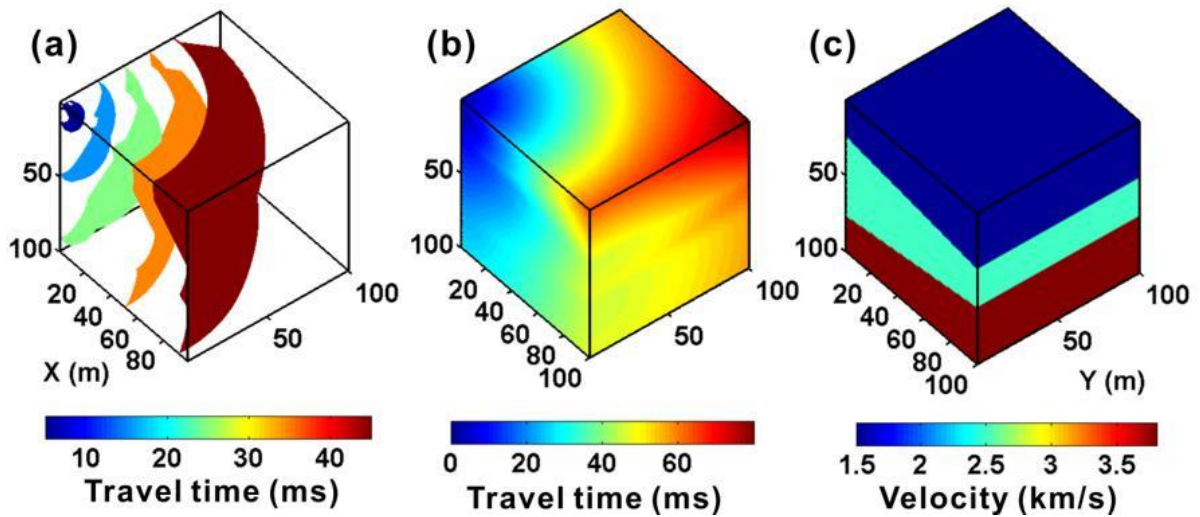


Figure 3.5: An example of seismic wave propagation in a 3-D model with 106 grid points. The traveltime is in millisecond and the distance is in meter. The travelttime field is calculated in 7 seconds. (a) The isosurface of different arrival times are rendered using

3.9.5 Source Scan Method

Unlike the pick-based methods in previous sections, the source scan method is a waveform based microseismic location technique. The development is motivated by the observation that first arrival picking is tedious and error prone, even though certain improvement has been made in automatic phase detection. Instead of relying on the time picks, the source scan method checks the fitness of the theoretical arrival to the waveform for every grid point in the search volume and consider the grid that can maximize the fitness as the source location. The fitness is defined as the stacked energy from the waveform. There are several variants of stacking in the literature for the waveform-based location method. For example, the original source scanning algorithm (Kao and Shan, 2004) stacks the absolute amplitude. The Continuous Map Migration (CMM) method patented by Schlumberger Technology Corp (US7391675B2) stacks the STA/LTA trace instead of the raw waveform. Other waveform-based methods stack Kurtosis trace (e.g. Langet et al., 2014), or weight the stacked energy by semblance (Huang et al., 2013; Zhang and Zhang, 2013).

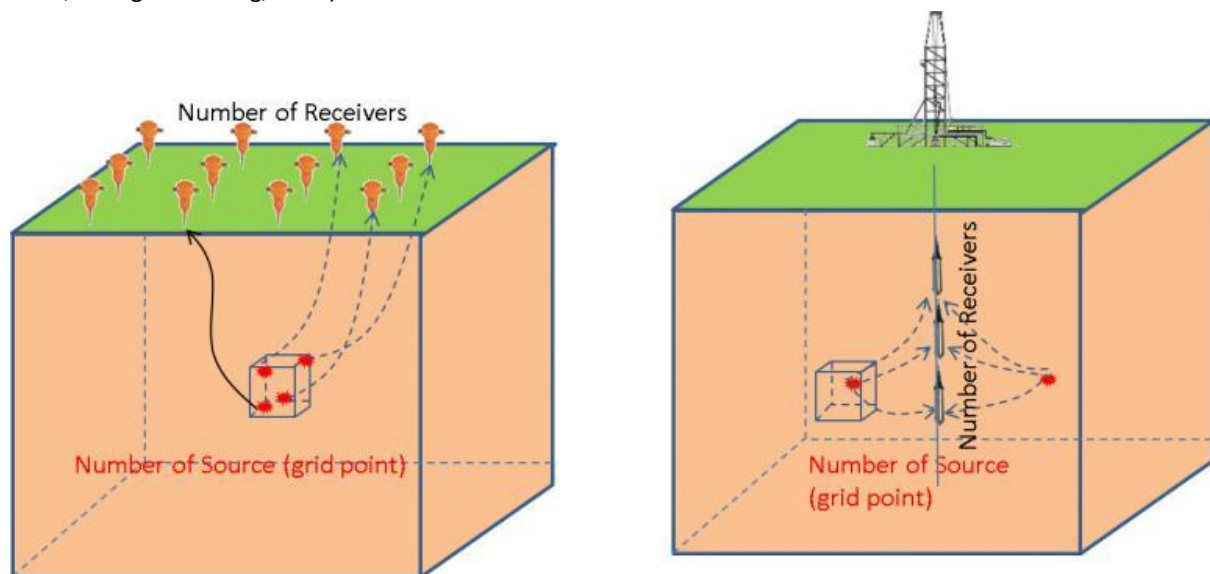


Figure 3.6: A cartoon demonstrating the search volume for travel-time table calculation for surface or downhole monitoring. For single borehole (right), the azimuthal ambiguity can be reduced by including source vector information.

The source scan method in InSite stacks semblance weighted auto-function of the waveform. The auto-function is user defined. It can be the STA/LTA trace based on RMS waveform or waveform envelope, or the AIC trace. Depending on the data quality, modified AIC and the STA/LTA trace are more robust than other methods. When P-wave source vector is used, the match between the measured polarization and theoretical polarization is used to further modify the stacked energy. The typical workflow is:

1. Calculate a traveltime table for both P- and S-wave direct arrivals (e.g., fast sweeping method for 3D heterogeneous model, or [fast sweeping method](#) for 1D layered model). For single borehole case, source vector can be used to reduce azimuthal ambiguity.
2. For each grid point and each original time, align the auto-function trace following the theoretical travel time. The semblance is measured for each grid point and is multiplied to the 4D function S . The function S is then summed along the time axis. The maximum stacked energy (the grid with the maximum value) is considered as the likely source position.

3. If source vector is used, the difference between the theoretical polarization and the measured polarization is applied to function S as additional weight.
4. The theoretical arrivals from the maximum grid point usually match the waveform and may be inconsistent with the polarity which is sensitive to signal quality. The uncertainty can be further reduced by a conventional collapsed grid search.

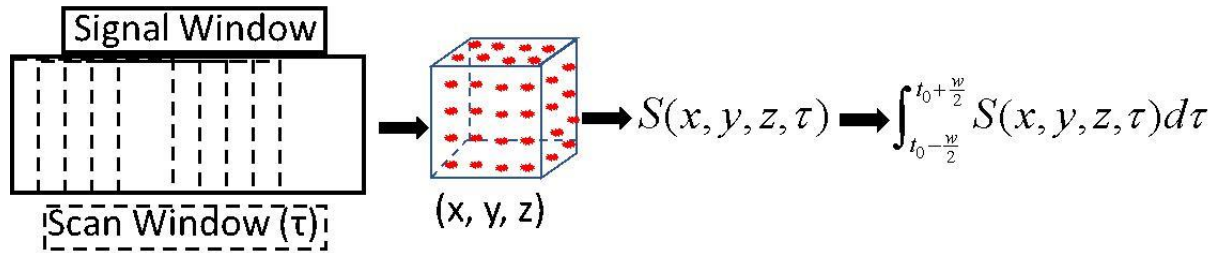


Figure 3.7: The basic workflow of stacking.

To speed up the global search in step 2, the Gaussian Particle Swarm Optimization (Krohling, 2004) is used. Unlike the conventional Particle Swarm Optimization (PSO), the GPSO significantly improves the convergence performance with less tuning parameters. For example, the inertia weight, the maximum velocity of the particles and the accelerating constants required in PSO are no more needed in GPSO. Specifically, step 2 is further divided into the following sub-steps:

2a. Randomly generate many particles, each with four attributes i.e. North, East, Depth and Origin Time. The particles are uniformly distributed within the grid search volume. If no first arrival picks, each particle's time attribute also has a uniform distribution. Otherwise, the picks will be honoured in the search along time axis.

2b. For each particle, calculate the stacked energy. Each particle is aware of the global maximum energy and its own best maximum energy in the past iterations.

3c. In the following iterations, the spatial temporal attributes of each particle are updated according to the global maximum and its own best maximum energy. After certain number of iterations, majority particles will be clustered in a single region. This location and origin time are thus considered as the position of microseismicity.

As a demonstration, GPSO was used to find the global maximum of the Rastrigin function (

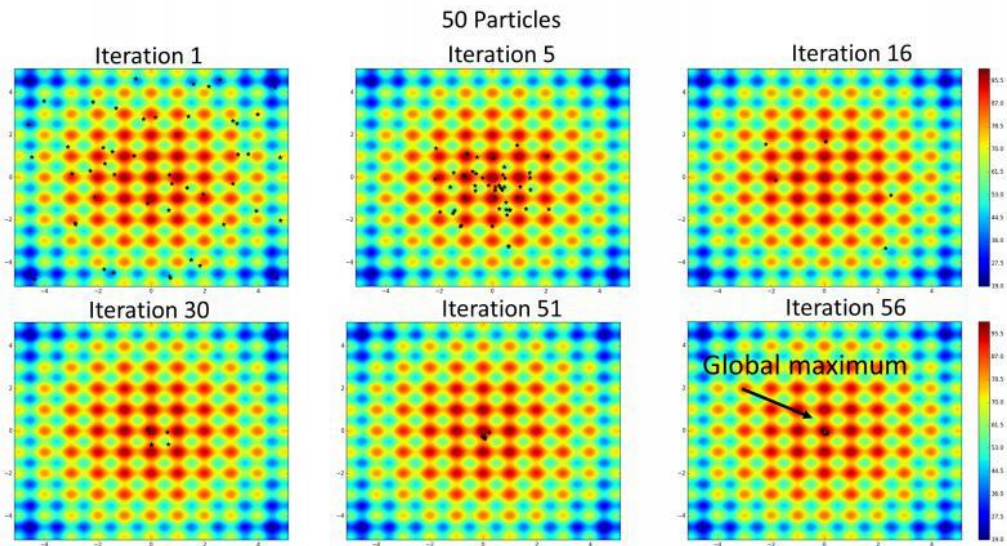


Figure 3.8). With 50 particles, the global maximum was detected after a few iterations and the optimization process converged after 80 iterations. The computation time increases with the number of particles and the maximum allowed iteration number. Empirically, the combination of large number of particles and a small number of iterations are effective for source scan problems to detect the region with global maximum before convergence.

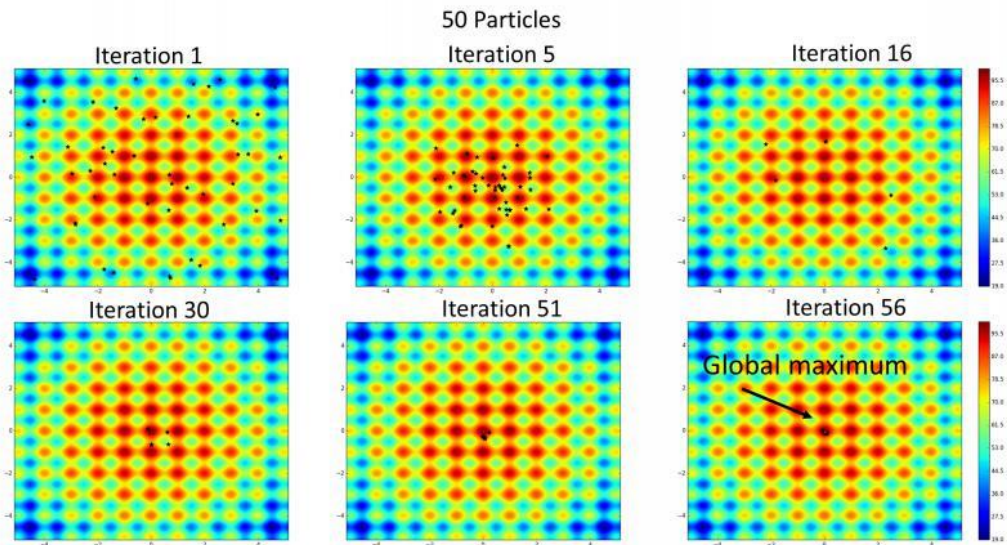


Figure 3.8: An example of GPSO searching for the global maximum in a space with numerous local minima. The black dots are particles and the colour represent the value of the function. The global maximum can be detected with a smaller number of iterations given sufficient number of particles.

Chapter 4 Source Parameter Algorithms

[Data Initialisation for Automatic Source Parameters](#)

[Instrument Magnitude](#)

[FrequencySourceModel Parameters Calculated in the Time Domain](#)

[Calculation of Source Parameters](#)

4.1 Data Initialisation for Automatic Source Parameters

Automatic source parameters are calculated at the end of the source location algorithm.

- Source parameters are only calculated when the software is run in Instrument Mode.
- Only instruments used in the source location are used in the source parameter algorithms.
- The instruments must have a known response to either volts, acceleration, velocity or displacement.
- Velocity and displacement waveforms are obtained for each instrument automatically. A band-pass filtering of the waveform is performed.
- The algorithms use the RMS Waveform generated from the M channels on the sensor (where $1 \leq M \leq 4$).

The RMS Waveform is calculated using,

$$W_i = \sqrt{\frac{\sum_{m=1}^{m=M} W_{im}^2}{M}} \quad (4.1)$$

where W_{im} is the waveform amplitude on channel m at data point i.

- Automatic source parameters use a window of data around the P-wave and S-wave arrivals. The windows are constructed using a front and a back-window length, with reference to the theoretical travel times for the located event. The window extremities are displayed on the Waveform Visualiser. The theoretical travel times are calculated knowing the velocity structure of the medium and the source and receiver locations.

- Instruments that have 'clipped' waveforms are discarded from the algorithm. The criterion for clipping is that the signal amplitude equals or exceeds the amplitude range of the receiver, set in the receiver properties.
- The source parameters can include a correction for anelastic attenuation by multiplying measured signal amplitudes by the correction factor, P_k (see [equation 5.3](#)). This requires a single dominant frequency input from the receiver properties.
- It is possible to calculate all source parameters for an event using a single triaxial instrument, however, in general, the greater the number of instruments used in the analysis the more accurate the results as averaging reduces the effect of uncertainties.

4.2 Instrument Magnitude

The instrument magnitude is calculated for every instrument using,

$$M_I = A \cdot \log_{10} \left(r \cdot W_{V_{rmspeak}} \right) + B \quad (4.2)$$

,where r is the source to receiver distance. A single instrument magnitude is calculated for the event by averaging over the instruments used in the source location.

$W_{V_{rmspeak}}$ is the peak signal amplitude (DC to peak) on the RMS Velocity Waveform.

A and B are user configured constants.

The magnitude is calculated irrespective of the number of channels contained in the instrument. This calculation therefore makes maximum use of the data available across the array.

4.3 Frequency-Source-Model Parameters Calculated in the Time Domain

There are three Frequency-Source-Model Parameters that must be measured for a phase arrival in order to calculate a set of source parameters for an event (Collins and Young, 2000) :

- Low-frequency Spectral Plateau
- Corner Frequency
- Energy Flux

These can be manually picked from a Fast Fourier Transform (FFT) of rotated waveform data, however this requires lengthy manual processing. Urbancic(1996) provides an automatic method of obtaining these parameters from the time domain waveforms. The method uses RMS Velocity and Displacement Waveforms on a fully-functioning triaxial instrument. Both P-wave and S-wave windows are constructed around the arrivals and the Low-frequency Spectral Plateau (Ω_{0c}), Corner Frequency (f_{0c}) and Energy Flux (J_c) are subsequently calculated.

$$\Omega_{0c} = \sqrt{4 \cdot S_{D2c}^{3/2} \cdot S_{V2c}^{-1/2}} \quad (4.3)$$

$$f_{0c} = \frac{1}{2\pi} \sqrt{\frac{S_{V2c}}{S_{D2c}}} \quad (4.4)$$

$$J_c = S_{V2c} \quad (4.5)$$

The 'c' subscript refers to the phase type (P or S). S_{D2c} and S_{V2c} are calculated using:

$$S_{D2c} = 2 \cdot t_s \cdot \sum_{i=1}^{i=N} W_{Drmsi}^2 \quad (4.6)$$

$$S_{V2c} = 2 \cdot t_s \cdot \sum_{i=1}^{i=N} W_{Vrmsi}^2 \quad (4.7)$$

, where N is the number of data points in each waveform window.

W_{Drmsi} is the amplitude of the RMS Displacement Waveform at data point i.

W_{Vrmsi} is the amplitude of the RMS Velocity Waveform at data point i.

t_s is the sampling time used on the waveform.

4.4 Calculation of Source Parameters

Urbancic et al. (1996) discusses source parameter calculations and gives an example case study.

The Seismic Moment (M_0) is directly proportional to the low-frequency spectral level:

$$M_{0c} = \frac{4\pi\rho V_c^3 r \Omega_{0c}}{F_c} \quad (4.8)$$

Where ρ is the density, V_c is the arrival velocity, r is the path length between source and receiver, and F_c is the arrival's average radiation coefficient. The latter is user configured but is often assumed for a shear source as $F_P = 0.52$ and $F_S = 0.63$ (Boore and Boatwright, 1984). A mean seismic moment, M_0 , for the event is given by averaging all of the P and S-wave results across the instruments. A Moment Magnitude, M_w , is then given by:

$$M_w = A \cdot \log_{10}(M_0) + B \quad (4.9)$$

Where A and B are user-set constants, often taken as $A = 2/3$ and $B = -6$.

The Radiated Seismic Energy is given for each phase by:

$$E_c = 4\pi\rho V_c \frac{\sum_{n=1}^N J_{c_n} r_n^2}{N} \quad (4.10)$$

, where N is the total number of instruments used (Collins and Young, 2000).

The total radiated seismic energy for the event is then $E = E_P + E_S$ and the S:P Wave Energy Ratio is $E_{S:P} = \frac{E_S}{E_P}$

Assuming a circular fault plane, the Source Radius is calculated from the corner frequency using:

$$r_{0c} = \frac{K_c V_c}{2\pi f_{0c}} \quad (4.11)$$

,where K_c is a constant. Madariaga (1976) gives average values for the constant as $K_p = 2:01$ and $K_s = 1:32$, assuming a rupture velocity of $0.9V_s$.

A mean source radius, r_0 , is calculated for the event by averaging across the instruments.

Two estimates of stress release are then calculated; the Static Stress Drop, $\Delta\tau$, and the Apparent Stress, τ_a ,

$$\Delta\tau = \frac{7M_0}{17r_0^3} \tau_a = \frac{\mu E}{M_0} \quad (4.12)$$

μ is the shear modulus for the material in the source region.

Chapter 5 Source Mechanism Algorithms

[Inversion](#)

[Decomposition](#)

[Error Analysis](#)

[Fault Plane Solution from First Motion Polarities](#)

5.1 Inversion

A Moment Tensor (MT) inversion is used to obtain the source mechanism of an event. This approach does not assume a shear (double-couple) type mechanism, but can also resolve more complex mechanisms that might be associated with, for example, opening and closing cracks. It is widely used in mining seismology (Gibowicz and Kijko, 1994). The MT inversion procedure has been developed and tested using various data sets including Pettitt (1998), Young et al. (2000), Collins et al. (2000). The procedure follows earlier work by Stump and Johnson (1977), De Natale and Zollo (1989), Feignier and Young (1993), and Ohtsu (1991).

Jost and Herrman (1989) give a thorough introduction to MT theory and the inversion of seismograms. The method uses amplitude data taken from recorded waveforms and an estimate of the source-receiver propagation effects to invert for the nine equivalent forces at the source. These are represented as a real and symmetric 3x3 matrix, M . The MT can then be diagonalised into its three principal forces represented by three eigenvalues and corresponding eigenvectors. The eigenvalues give a measure of the source type (e.g. whether the source is a shear crack or whether it has an isotropic component that might be associated with opening or closing along it). The eigenvectors are the principal axes of the source and, depending upon the magnitude of the corresponding eigenvalue, give the orientations of the pressure, p , null, b and tension, t axes. The pressure axis is the direction of maximum compressive motion on the fracture surface and the tension axis is the direction of the maximum tensional motion. The null axis is the vector orthogonal to both the pressure and tension axes and in a double couple solution represents the direction where no motion occurs on the fracture surface. The use of a moment tensor method gives the ability not only to see the location of damage within a rock mass but also the micromechanics that are driving it.

The method applied here uses P-wave first-motion polarities and amplitudes. These are manually picked from the recorded waveforms. Generally, first-break times are also picked, and the event relocated to get more accurate source locations

Assuming a point source and a simple delta type source-time function then a set of recorded amplitudes can be related to the nine equivalent forces at the source by the linear equation,

$$u = Gm \quad (5.1)$$

,where u is a column vector containing the n recorded amplitudes, G is a $n \times 6$ matrix containing the Green's functions components and m is a column vector containing the six required MT elements (only six are required due to tensor symmetry).

The Green's functions take into consideration the radiation pattern of the source, the propagation effects along the path and the receiver's response effects. We simplify these by assuming that the receivers are in the far-field (usually estimated as 10 wavelengths) and that the propagation medium is homogeneous and isotropic. As we use P-wave amplitudes from the time domain then G contains the P-wave radiation components which can easily be calculated from the known locations of the source and receiver (De Natale and Zollo,1989).

Knowing the source-receiver path length, r , each amplitude in the vector u is first corrected for path and sensor effects using

$$u_k^0 = S_k \cdot P_k \cdot F_k \cdot \frac{1}{C_{w_k}} \cdot G_{S_k} \cdot u_k \quad (5.2)$$

, where u_k is the amplitude measured on sensor k and u_k^0 is the corrected amplitude.

- Sensor polarity, S_k – Sensors have different response polarities, $S_k = \pm 1$. For a sensor that responds to a first motion compression with a positive signal then $S_k = 1$. If the opposite is true then $S_k = -1$.

- Geometrical spreading, G_{S_k} – A multiplication by the distance travelled, $G_{S_k} = 1/r_k$, accounts for the spreading of the wave's energy as it travels further from the source.

- Anelastic attenuation, P_k – This accounts for dissipation of energy as the wave travels through the medium. The amount of dissipation is described by the materials Attenuation Quality Factor, Q_k is the dominant frequency of the recording sensor.

- Free-surface amplification, F_k – This corrects for an apparent amplification of the wave as it reflects at a free surface; θ is the incident azimuth, p is the ray parameter, and j is the angle of reflection of a mode-converted Sv-wave off the free-surface (Aki and Richards, 2002). This correction does not need to be applied to a sensor grouted into the medium as no free surface is present.

- Receiver coupling weight, C_{w_k} – This is an estimate of the sensitivity of a particular transducer depending upon how efficiently it is coupled to the rock surface. It can be estimated from P-wave amplitudes acquired from known seismic sources. The coupling weights for an array of sensors should use $C_w = 1.0$ as a datum. Sensors with better coupling should have a weight $C_w > 1.0$ and sensors with reduced coupling should have a weight $C_w < 1.0$.

$$P_k = \frac{1}{e^{\frac{-\pi \cdot f_k \cdot r_k}{Q \cdot v_p}}} \quad (5.3)$$

$$F_k = \frac{\left[\left(\frac{1}{v_s^2} - 2p^2 \right) + 4p^2 \left(\frac{\cos \theta}{v_p} \right) \left(\frac{\cos j}{v_s} \right) \right]}{\left(\frac{-2v_p}{v_s^2} \right) \left(\frac{\cos \theta}{v_p} \right) \left(\frac{1}{v_s^2} - 2p^2 \right)} \quad (5.4)$$

$$p = \frac{\sin \theta}{v_p} = \frac{\sin j}{v_s} \quad (5.5)$$

The matrix G can then be inverted using Singular Value Decomposition (Stump and Johnson, 1977) and the vector m subsequently calculated. The six elements of m are then used to fill the real and symmetric 3x3 MT matrix, M .

5.2 Decomposition

The decomposition of the MT provides an analysis of the type and orientation of the source mechanism. Further information for this procedure can be obtained from Pettitt (1998).

5.3 Error Analysis

A quality assessment of the inversion is also performed. A higher quality solution will be more robust and have less uncertainties associated with it. A series of parameters are provided to try to quantify the quality of the solution.

1. **Amplitude Residuals** - The resulting MT is used to calculate amplitudes expected at the array of receivers. These are compared to the actual amplitudes recorded; the difference between the two being the amplitude residual. Amplitudes that have a residual of > 0.2 should be checked and either re-picked or discarded from the inversion. A final RMS Amplitude Residual is produced from all those used in the inversion to describe the overall fit of the solution to the data.
2. **Error Factors** - Estimated uncertainties in the amplitude measurements are compared to the amplitude residuals from the inversion. Amplitudes that have residuals much greater than would be expected from the uncertainties (Error Factor $\gg 1$) should be re-assessed. Nodal or small amplitudes often produce very large Error Factors that are just an artefact of a small amplitude being used in the inversion. An RMS Error Factor is produced for the final solution.
3. **Quality Index** - The 6x6 co-variance matrix of the inversion is obtained. This assesses the quality of the inversion of the Green's functions matrix, G and so does not consider the amplitude data. A poor inversion is often produced when receivers have a poor focal coverage. In these cases elements in the co-variance matrix become large. In order to reduce this assessment down to one usable value the quality index is calculated, defined as the sum of the squares of the co-variance elements. A large quality index is hence a warning of a poor inversion.
4. **Condition number** - The condition number allows a qualitative assessment of the robustness of the inverse method (how well it can tolerate a few bad data). Generally, condition numbers of a few tenths to a few units indicate a very robust inversion.

The error analysis needs to generate an uncertainty in each amplitude reading used in the inversion. The uncertainty must include a contribution from each of the corrections applied in;

$$u_k^0 = S_k \cdot P_k \cdot F_k \cdot \frac{1}{C_{wk}} \cdot G_{sk} \cdot u_k \quad (5.6)$$

, defined [here](#). The total amplitude uncertainty Δu_k^0 is estimated using

$$\frac{\Delta u_k^0}{u_k^0} = \sqrt{\left(\frac{\eta_k}{u_k}\right)^2 + \left(\frac{\Delta G_{sk}}{G_{sk}}\right)^2 + \left(\frac{\Delta P_k}{P_k}\right)^2 + \left(\frac{\Delta F_k}{F_k}\right)^2 + \left(\frac{\Delta C_{wk}}{C_{wk}}\right)^2} \quad (5.7)$$

The factor (η_k/u_k) is the inverse of a signal:noise ratio calculated from the waveform. The remaining factors are contributions from each of the amplitude corrections. To simplify the analysis these are input simply as estimated percentage errors. Pettitt (1998) assumed a 5% error in the geometric spreading and anelastic attenuation corrections, and a 10% error in the free-surface and coupling weight corrections.

A statistical measure of the model fit in the procedure can then be obtained by calculating the standard deviations,

$\sigma[M_j]$ of the MT elements, M_j ($j = 1 \dots m$ where $m = 6$). This then allows estimates of absolute error bounds to be obtained for the source mechanism. From Stump and Johnson (1977) the variance on each of the elements, $\text{var}[M_j]$ is given by:

$$\text{var}[M_j] = \sum_{k=1}^n (G_{jk}^{-1})^2 \sigma_k^2 \quad (5.8)$$

where G_{jk}^{-1} is the inverted Green's function matrix with $j = 1 \dots m$ rows and $k = 1 \dots n$ columns and σ_k^2 are the variances on each of the n amplitudes, u_k^0 .

The standard deviation on each of the elements can then be obtained as $\sigma[M_j] = \sqrt{\text{var}[M_j]}$.

As only a single measurement is performed for each amplitude, it is assumed that the standard deviations in the amplitudes approximate to the estimated uncertainties in the amplitude measurements, $\sigma_k^2 \approx (\Delta u_k^0)^2$.

A set of six standard deviations can then be produced for the six MT elements. Assuming that these standard deviations accurately represent the uncertainties in each of the MT elements then we have $M_j \pm \sigma[M_j]$ where M_j is one standard deviation. The problem now is how to represent uncertainties in the source mechanism from uncertainties in the MT. To obtain a quantitative uncertainty is not trivial, as it requires propagation of the errors through the complete decomposition process. Instead, an error space sampling procedure has been developed (Pettitt, 1998). This consists of the following stages.

1. The six MT elements, $M_j \pm \Delta M_j$ are sampled for every combination of the MT with elements adjusted by + or $-\Delta M_j$.

2. For each sample, the MT is decomposed to get a mechanism type and associated eigenvector orientations. This then gives a distribution of mechanisms that fit within the error bounds estimated for the MT and is then an estimate of the error space in the solution.
3. The error space is represented as standard deviations in each of the key parameters, calculated from the variance in the set of solutions.

5.4 Fault Plane Solution from First Motion Polarities

The solution for the fault plane is also calculated using the first-motion polarities of the P-waves. This method is widely used in seismology and examples applied to different scales can be found in the literature (e.g. Gibowicz and Kijko (1994), Udias (1989)).

The program searches for the pair of orthogonal nodal planes that optimally separates the first motions into compressional and tension quadrants on the focal sphere. Following this search, the program selects all potential solutions with the minimum number of misfits (stations with polarity reversed with respect to the quadrant's). The solution presented in the visualiser and rpt file is the closest solution to the average of all solutions with minimum number of misfits. Apart from the two orthogonal nodal planes, the following results are also output in the Mechanism visualiser:

- B axis: the intersection of the two nodal planes, representing the null axis.
 - P axis: the compression axis, situated in the tension quadrant and oriented at 45 with respect to both nodal planes.
 - T axis: the tension axis, situated in the compressional quadrant and oriented at 45 with respect to both nodal planes.
 - X1 and X2: normal vectors to the solution nodal planes, also interpreted as the slip vectors.
-

Chapter 6 Cross-Correlation Algorithms

[Introduction to the CCR Technique](#)

[Description of the CCR procedure](#)

[Data Input](#)

[Construction of the CCR Data Arrays](#)

[CCR Processing](#)

[Velocity Calculations](#)

[Data Output](#)

[Testing of the CCR Routines](#)

6.1 Introduction to the CCR Technique

A cross-correlation approach has been developed to provide accurate and precise measurements of velocity change between active (velocity) surveys. The approach has been developed for measuring both P-wave and S-wave velocity change in laboratory experiments and in situ investigations of rock and concrete structures.

The cross-correlation method was first developed for velocity surveys conducted in laboratory experiments and in the Zone of Excavation Damage Experiment (ZEDEX) at SKB's Äspö Hard Rock Laboratory (HRL). The method was then developed further to give an automatic approach to velocity survey processing, whereby hundreds of surveys taken over months or years could be processed without time-consuming manual picking, but with higher accuracy than would have been achieved if manual picking had been used. The approach has subsequently been applied to further investigations at AECL's Underground Research Laboratory, Canada and at the HRL. An example application is described in Pettitt et al. (2002).

There are two principal objectives in the cross-correlation (CCR) technique.

- Very precisely measure the time shift of two P-wave (or S-wave) arrivals along the same ray path but at different acquisition times. This could be, for instance, velocity surveys recorded before and after excavation close to the ray path position. The advantage of the technique is that by mathematically comparing the two arrivals a much more precise and accurate time difference is measured than if both arrivals were manually picked. This is because each manual pick has a relatively large uncertainty due to where the operator can visibly pick the arrival time from the noise level. By over-sampling the data the CCR is able to measure a time shift that can be smaller than a single real sample point. Such high
-

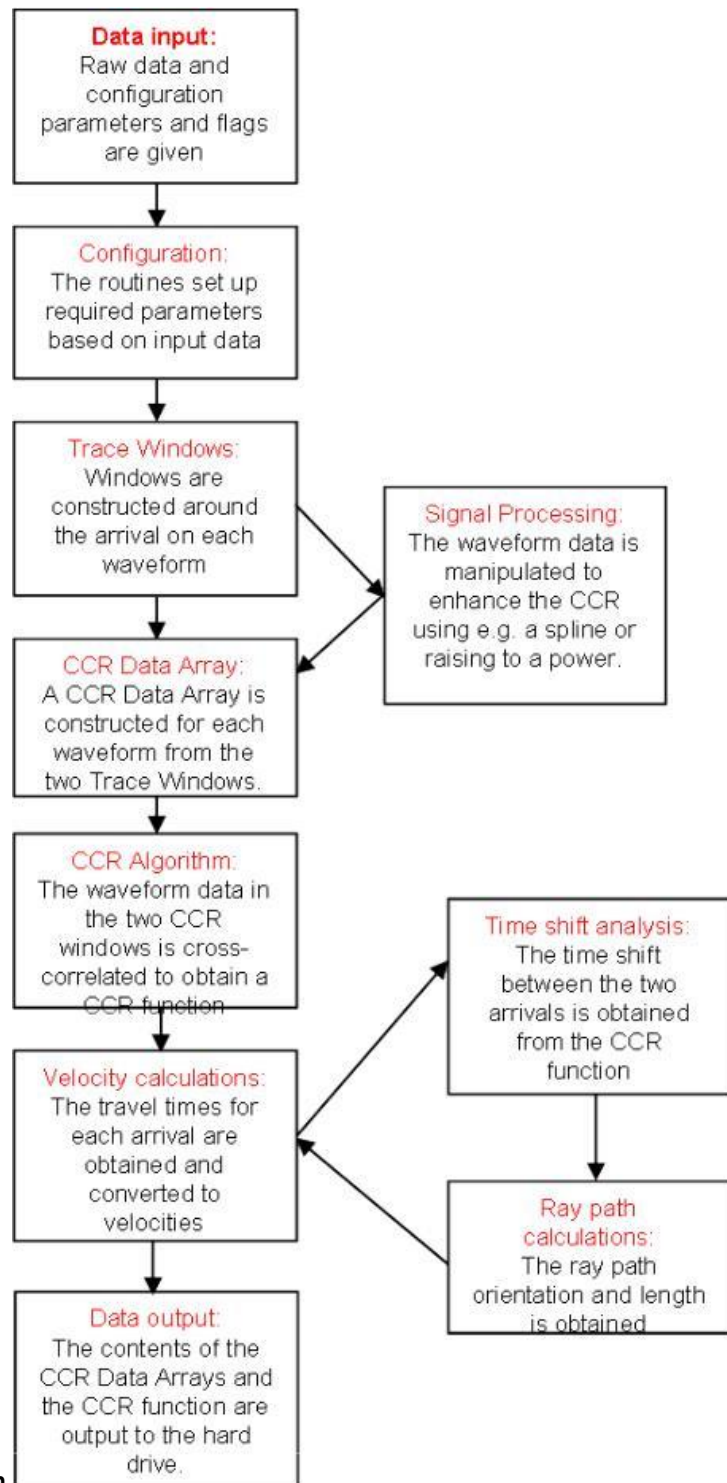
precision is required so as to obtain sensitivity to small changes in apparent velocity. These small changes may actually be associated with high changes in absolute velocity that occur along only a small portion of the ray path. A drop of e.g. 300m/s along 5% of a ray path in granite will be observed as approximately 15m/s in the apparent velocity.

- Provide a method of measuring velocity changes automatically across numerous velocity surveys performed during an experiment. The CCR approach uses a window of waveform data around the Pwave (or S-wave) arrival and mathematically compares it to a similar window of data around the arrival on a pre-defined reference survey. The windows are constructed around picks of the arrival time. The time picks can either be manually picked, or picked through an automatic picking algorithm. Current automatic picking algorithms can pick arrivals (on clear P- or S-waves) to within approximately one eighth of a cycle, depending on the data quality. This is far too inaccurate to generate meaningful velocity measurements directly from the times of the automatic picks. Testing has shown that using CCR, automated picks will produce output velocities that are generally identical to velocities calculated from a CCR using manual picks. This is because a CCR window generated around an automatic pick does not differ considerably to one generated around a manual pick. As such the CCR approach is a good method of providing velocities automatically from P- or S-wave arrivals.

The CCR technique has also been extended so that it can be used on passive events for “Master Event Source Location”. In this case, waveform arrivals on individual events are compared with a Master Event. No velocity information is generated, but instead the CCR travel times that are obtained are made available for input into the source location algorithm. This results in more accurate source locations of events relative to the location of the Master, and hence is ideal for investigating more detailed structure in clusters of events.

6.2 Description of the CCR procedure

The data processing functions performed during the



complete CCR procedure are shown in

Figure 6.1. The procedure can be split into five main components that are described in more detail here.

1. Data input
2. CCR Data Array construction
3. CCR processing
4. Velocity calculations
5. Data output

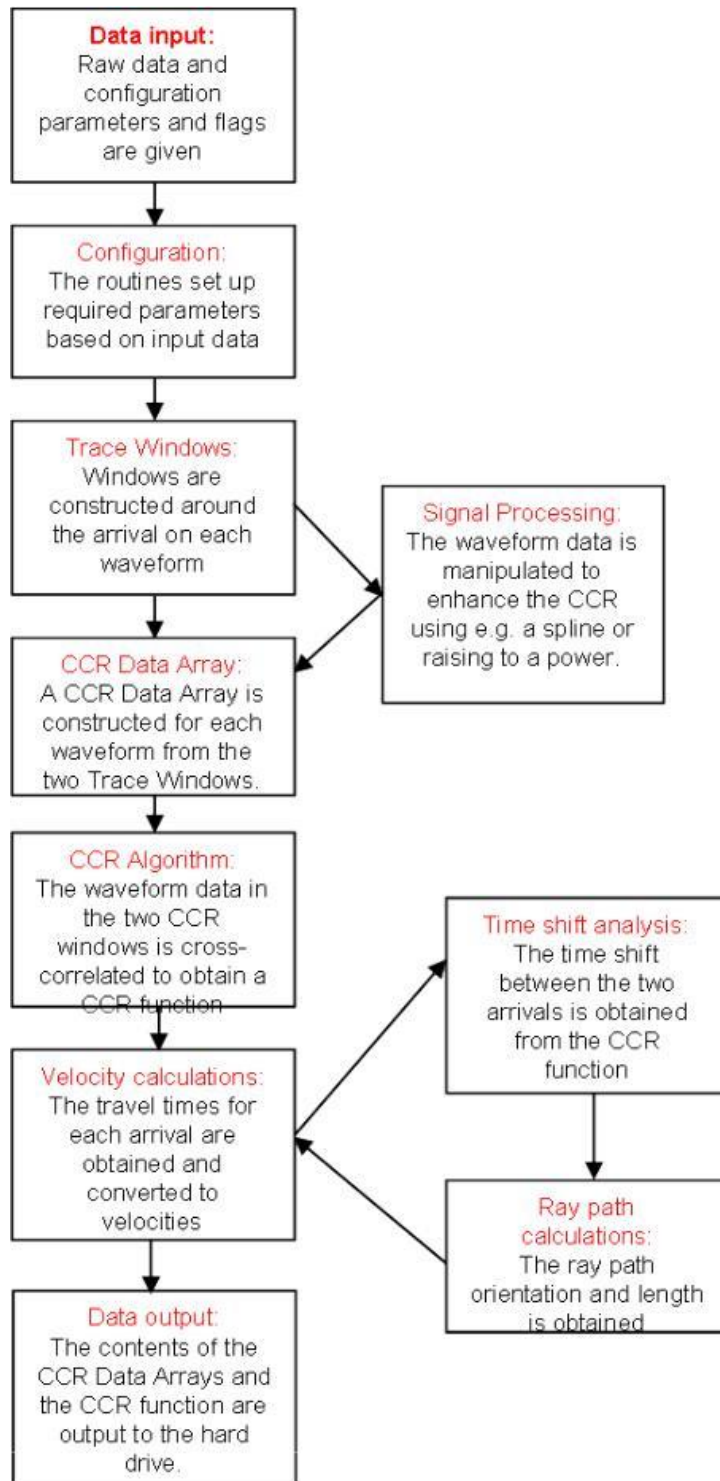


Figure 6.1: The principal data processing functions performed during the CCR procedure.

6.3 Data Input

The following data are required.

[Trace Data](#)

[Source and Receiver Parameters](#)

[Configuration Parameters](#)

[Configuration Flags](#)

6.3.1 Trace Data

Two waveforms are required; the Reference Waveform and the Process Waveform (a 'waveform' is an amplitude time series). The waveforms will each have a window constructed around the phase arrival of interest, and the data within the windows will be cross-correlated. Data manipulation functions, such as frequency filtering, should be performed before the waveform is passed to the CCR functions.

Each waveform is just one member of a set of data. This is collectively known as a 'Trace'. In addition to the waveform each Trace includes the following input parameters:

- The number of data points in the waveform
- The sampling frequency on the waveform
- The first-break pick time, T_B
- The amplitude pick time, T_A (if necessary)
- The time-zero, T_0

The first-break pick time on the Reference Waveform is used to generate absolute velocities for both the Reference and Process traces. Hence, this arrival time should be as accurate a representation of the true arrival time as possible so as to provide accurate absolute velocities. The first-break pick time on the Process Waveform is not used to produce a velocity (this is calculated from the CCR), however it is often used to produce a Trace Window (Table 6.1)

6.3.2 Source and Receiver Parameters

To calculate the orientation and length of the ray path the source and receiver must be defined. The following inputs are required for each.

- The (X, Y, Z) coordinates of the source/receiver location in a North-East-Down (NED) system.
- The (i, j, k) direction cosines defining the sensor orientation vector (if necessary).
- The coordinate units as a multiple of metres (e.g. 1000 for kms)
- The sensor diameter (if necessary)

6.3.3 Configuration Parameters

To process the CCR the routines require some additional parameters.

- The front window length, ΔT_f (if necessary)
- The back window length, ΔT_b (if necessary)
- The window size expansion (if necessary)
- The sampling frequency of the spline function to apply to the waveforms (if necessary)

6.3.4 Configuration Flags

The CCR routines have a number of options that can be applied to the procedure. These options are set through the following flags.

- The type of Trace-Window construction method
 - The type of Trace-Window adjustment when producing the CCR Data Array
 - Whether to fit a spline function to the waveforms
 - Whether to convert the waveform data to absolute values
 - Whether to raise the waveform data to a power
-

6.4 Construction of the CCR Data Arrays

[Window construction](#)

[Waveform signal processing](#)

[Filling the CCR Data Array](#)

A CCR Data Array contains the time-series data from around a defined arrival on a waveform. In the CCR algorithm two sets of time-series data are mathematically compared to produce a CCR Function (this is explained in later sections). The maximum positive peak in the CCR Function corresponds to a time-shift at which the two sets of time-series data are most similar. When comparing arrivals along the same ray path but from two different seismic surveys, the time shift represents a change in velocity between the surveys.

The time-series data input into the CCR algorithm could be the complete raw waveform, however this may produce a meaningless result as seismic waveforms are complex and contain many phases, reflections, and scattered energy that would complicate the CCR function and hide the result for the phase of interest. The time-series data must also be manipulated so that the CCR algorithm does not produce mathematical artefacts from frequency end effects generated from an abrupt ending of the time-series data. Therefore, only a small portion of the waveform containing the phase of interest is used in the CCR. This has an additional benefit of significantly reducing the processing time. The following procedure is used to produce a CCR Data Array that is a window of the waveform that includes the complete phase arrival of interest.

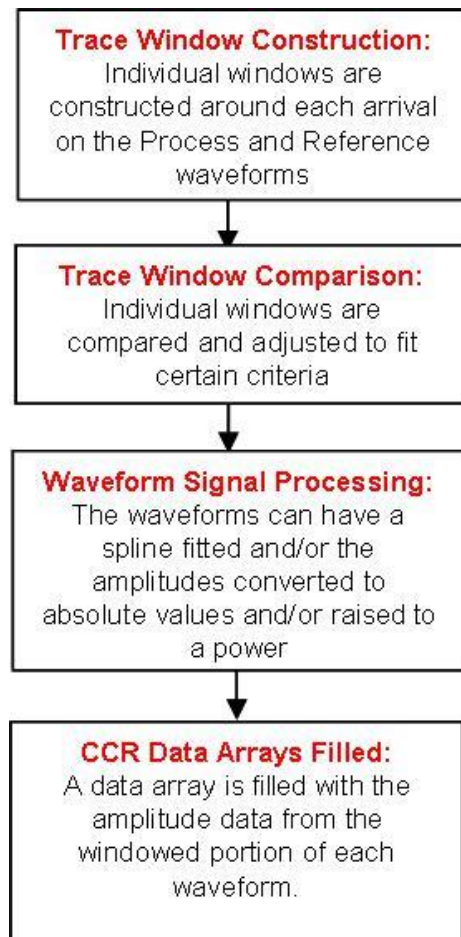


Figure 6.2: The procedure for construction of CCR Data arrays from input Trace data.

6.4.1 Window construction

A Trace Window is constructed that defines the start and end of the waveform data that is of interest (time T_s and T_e respectively). Two methods have been written that produce a Trace Window: the 'FrontWin/BackWin method' and the 'Rise-time method'.

[The FrontWin/BackWin method](#) and [the Rise-time method](#) describe the two window construction methods and the parameters that must be initialised when using each. Both methods use the first-break pick times from the two traces, so it is essential that these are initialised. It is not essential that the pick time(s) on the Process Waveform is accurate as long as a) the window size is configured so as to consider uncertainties in the picking, or b) a suitable window adjustment criteria (Table 6.1) is set. Both window construction methods produce a window size that is configured by the user. The user can therefore choose how much of the arrival to include in the CCR. From experience, it is advised that the window size is:

1. Sufficiently wide so that at least the first half a wavelength of the arrival is included and;
2. the window is not too large to include numerous artefacts in the wavetrain and that will considerably slow the processing.

Window Comparison Criteria	Usefulness of the Comparison Criteria
Set individual Trace Windows	A Trace Window is set on each waveform using one of the methods described in the text. This is the normal operating mode when good manual time picks are obtained for both Reference and Process waveforms.
Set a joint Trace Window	The start and end times for both Trace Windows are set to the minimum and maximum of the two individual Trace Windows. This is useful when an auto-picking algorithm is used that may occasionally pick at a slightly later or slightly earlier time on the Process waveform as it provides a better opportunity for the arrival to be completely included in the window.
Set the Process Trace Window to the Reference Trace Window	The start and end times for both the Trace Windows are set to the Reference Trace Window (the Process Trace Window is ignored). This is useful when the arrival on the Process Waveform is guaranteed to be in the Reference Trace Window, but auto-picking the Process Waveform gives spurious arrival times.
Use a fixed window length	This is only useful in the Rise-time window construction method. The Process Trace is given the same window length constructed from the Rise time on the Reference Trace. This is useful when a good auto-pick of T_B is generally obtained but the auto-pick of T_A is often spurious.

Table 6.1: Description of the window comparison criteria

6.4.1.1 The FrontWinBackWin method

The required input parameters are described in Table 6.2. The method requires one pick time (T_B) to be defined for each Trace. The start and end times (T_S and T_E respectively) of the window are then defined by

$$T_S = T_B - \Delta T_b \quad T_E = T_B + \Delta T_f \quad (6.1)$$

T_f and T_b are the front and back window sizes respectively and define the size of the window to apply.

Table 6.2: Parameters required for the FrontWin/BackWin window construction method

Symbol	Description
T_B	The first-break pick time on each trace used.
T_f	The front window length set in the WSurveyControl class
T_b	The back window length set in the WSurveyControl class

6.4.1.2 The Risetime method

The required input parameters are described in Table 6.3. The method requires two pick times; referred to as a first-break time (T_B) and an amplitude time (T_A). The rise time is defined as the difference between the two ($T_R = T_A - T_B$) and is used to calculate the size of the window to apply as defined in

$$\Delta T_h = \frac{R}{2} \cdot T_R \quad (6.2)$$

ΔT_h is the half window size and is applied symmetrically around the first-break time (i.e. in this case $\Delta T_h = \Delta T_f = \Delta T_b$). R is an expansion multiplier and must be set to 2, 4, 8 or 16 (i.e. the smallest window size that can be applied is $2 T_R$ and the largest is $16 T_R$).

The start and end times for the Trace Window are then defined as in equations in the [previous section](#) with

$$\Delta T_f = \Delta T_b = \Delta T_h$$

Table 6.3: Parameters required for the Rise-time window construction method.

Symbol	Description
T_B	The first-break pick time on each trace used
T_A	The amplitude pick time on each trace used
R	The expansion multiplier to apply to the rise-time

6.4.2 Waveform signal processing

There are three different signal-processing options that can be applied to the waveform.

- Fit a spline function
- Convert the waveform to absolute values
- Apply a power to the waveform amplitudes

[Fitting a spline function](#)

[Converting the waveform to absolute values](#)

[Applying a power to the waveform amplitudes](#)

6.4.2.1 Fitting a spline function

A 'natural cubic spline' can be fitted to the waveform data. This is an interpolation through a set of data points and is described in detail by Press et al. (1994), Chapter 3. The effect of the spline is to over-sample (fit new data points between the existing data points) the waveform. This gives it much better definition in cases where the Nyquist frequency of the acquisition sampling is close to the receiver response. The spline allows the CCR routines to provide much higher precision in the velocity results it outputs in these cases.

A spline is fitted independently to each of the input traces. The spline has a sampling frequency Δf_w and the raw waveform has an acquisition sampling of Δf_s . The over-sampling applied is defined by the ratio $\Delta f_s / \Delta f_w$. The set f_w spline sampling frequency MUST BE a multiple of the Trace sampling frequency (i.e. a 2MHz signal can have a spline of e.g. 4MHz but not e.g. 5MHz). The CCR routines allow two traces with different acquisition sampling frequencies to be processed for their CCR when splines are fitted to both (as the spline sampling frequency is the same for both). For this to be performed the spline sampling frequency MUST BE a multiple of both of the Trace sampling frequencies (i.e. two signals of 2MHz and 3MHz respectively can have a spline of e.g. 6MHz applied, but not e.g. 4MHz).

6.4.2.2 Converting the waveform to absolute values

The CCR routine measures a shift in the first arrival. The arrival must be the same polarity on the Process waveform as the arrival on the Reference waveform. This stipulation is fine when a source with constant signature is used in the experiment, however if the source varies (for instance in real earthquakes) then the CCR will obtain the time difference between the nearest in-phase arrivals. In order to change this, the amplitudes on both the Reference and Process waveforms can be converted to absolute values. This is performed by applying the C function `fabs(...)` to every waveform data point.

6.4.2.3 Applying a power to the waveform amplitudes

This is performed by applying the C function `pow(...)` to every waveform data point. The effect is to increase large amplitudes relative to small amplitudes and so make the CCR results more biased to the changes in the large amplitude data. Applying an even power (e.g. 2, 4 etc) causes the whole waveform to have absolute values and is an alternative to that described in [Converting the waveform to absolute values](#).

6.4.3 Filling the CCR Data Array

In order for the CCR to work effectively, data at the ends of each Trace Window must be reduced to zero. There must be no abrupt break in the amplitude at the ends of the window as these cause frequency end effects. A bell function is applied to the waveform amplitudes within the Trace Window, symmetric about the centre of the window (this is the first-break pick time, T_B , for a window with $T_f = T_b$). This has the effect of increasing the waveform amplitudes around the window's centre with respect to amplitudes towards T_S and T_E . A "Hann Window" is the bell function that is used (see Press et al. (1994), Chapter 13). All data outside of the window is set to zero amplitude. For computational efficiency, the CCR Data Array is then filled with only the waveform section of interest. This is defined by the minimum T_S and the maximum T_E of the two Trace Windows.

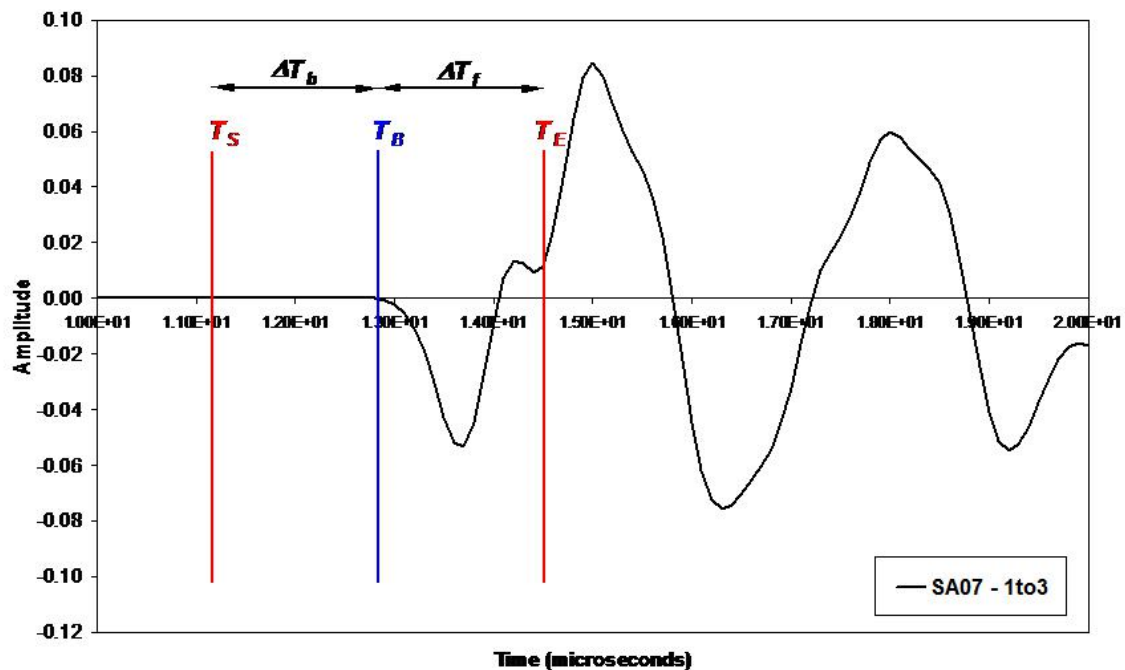


Figure 6.3: Example of window construction around an arrival using the FrontWin/BackWin method

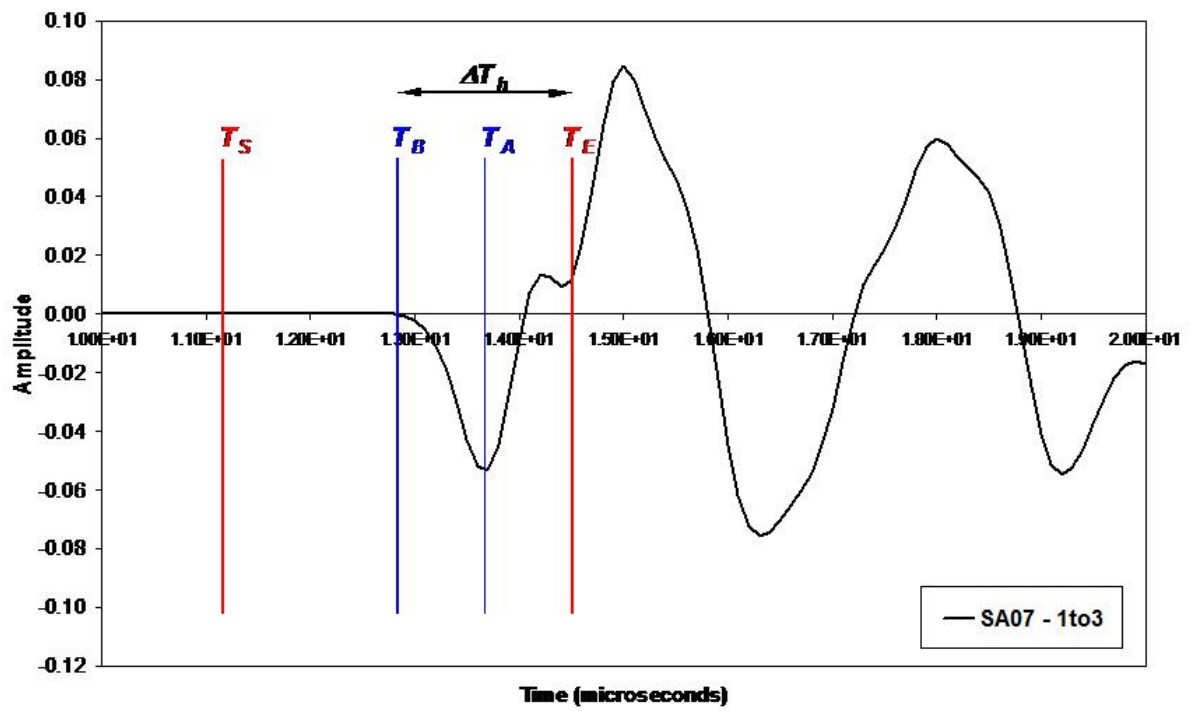


Figure 6.4: Example of window construction around an arrival using the Rise-time method. An expansion multiplier, $R=4$ is shown.

6.5 CCR Processing

The CCR function $\varphi(\tau)$ is a measure of the similarity between two time-series data sets $x(t)$ and $y(t)$ and is given by

$$\varphi_{xy}(\tau) = \sum_k x_k y_{k+\tau} \quad (6.3)$$

, see also Telford et al. (1990).

The time series is sampled at points k .

τ is the data point displacement of $y(t)$ with respect to $x(t)$.

The CCR function is smoothly varying and has a maximum positive peak where the data is most similar and a maximum negative peak where it is most dissimilar.

The data point at which the maximum peak in the CCR function occurs corresponds to a time lag of one arrival compared to the other. It is converted to a time (T_B) knowing the sampling frequency. By definition the CCR routines output a negative time when the Process waveform is shifted to the right with respect to the Reference waveform. This represents a slower (decrease) in velocity.

6.6 Velocity Calculations

To calculate velocities the CCR routines first calculate the ray path length, r (the ray path orientation (azimuth and plunge) is also calculated). The length can be corrected for a transducer diameter; this is useful during high-frequency ultrasonic laboratory experiments. The following assumptions are made in the calculations.

- The ray path length is assumed not to change between the Reference and Process measurements.
- The ray path is a 'direct' ray (straight line) between the source and receiver.
- The first response of an ultrasonic transducer (with diameter D) to an incoming wave occurs when a line segment of the wavefront first hits its outer diameter (see Pettitt (1998), Chapter 2).

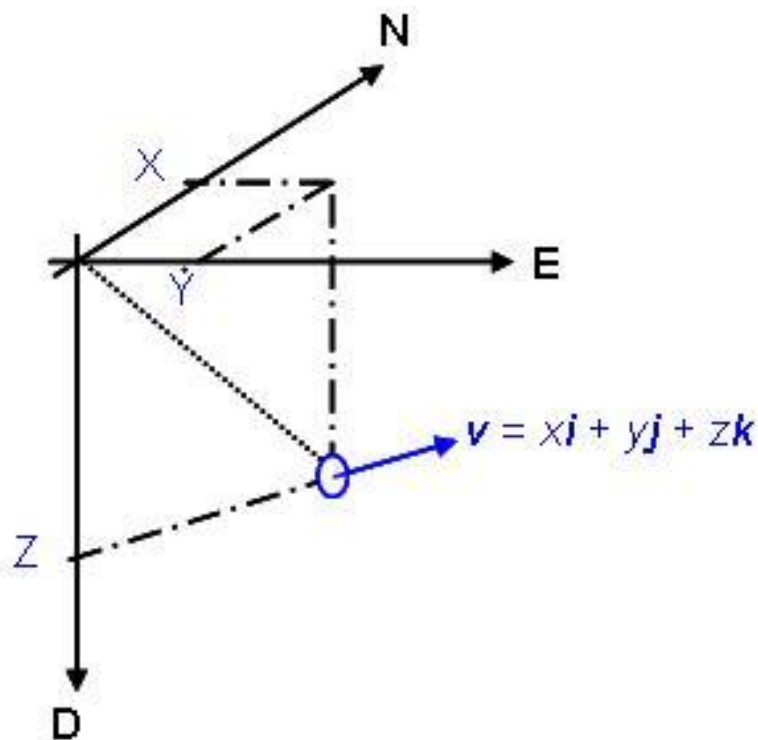


Figure 6.5: Definition of a sensor location and orientation.

The source/receiver locations must be defined in a North-East-Down co-ordinate system (as shown). When correcting for a transducer diameter a three-dimensional orientation vector, v is defined pointing along the longitudinal axis of the transducer. Using unit vectors, a transducer orientated due north would be given direction cosines $(x, y, z) = (1, 0, 0)$, and similarly for a transducer orientated directly upwards $(0, 0, 1)$. A transducer orientated Northwest with a 60 plunge would have the vector $(0:354, 0:354, 0:866)$.

Absolute velocities are then calculated from the Reference Trace arrival time, T_{ref} . The following applies to both P and S-wave arrivals

$$T_{ref} = T_B - T_{0ref} - T_{Cref} \quad (6.4)$$

T_B is the first-break (manual) pick time on the Reference Trace, T_{0ref} is the user-defined source time ('time-zero') for the Reference Trace and T_{Cref} is the user-defined arrival-time correction for the Reference Trace. The velocity for this arrival is then given by:

$$v_{ref} = \frac{r}{(T_{ref})} \quad (6.5)$$

The Process Trace arrival time, T_{proc} is then calculated from the time lag (ΔT_B) produced by the CCR (see [CCR Processing](#))

$$T_{proc} = T_{ref} - \Delta T_B - \Delta T_0 - T_{Cproc} \quad (6.6)$$

ΔT_0 is the difference between the user-defined source time on the two channels $\Delta T_0 = T_{0proc} - T_{0ref}$, and T_{Cproc} is the user-defined arrival-time correction for the Process Trace.

The velocity for the Process Trace arrival (v_{proc}) is then given in the same manner as for v_{ref} . The change in velocities between the two Trace data is then defined as the following

$$\Delta v_B = v_{proc} - v_{ref} \quad (6.7)$$

6.7 Data Output

The following are the principal outputs from the CCR routines.

- The time shift between the two defined arrivals (ΔT_B)
- The travel times for each arrival (T_{ref} and T_{proc})
- The absolute velocity on each arrival (v_{ref} and v_{proc})
- The difference in velocity between the two arrivals (Δv_B)
- The contents of the two CCR Data Arrays and the CCR Function

- The ray path orientation (length, azimuth and plunge)

6.8 Testing of the CCR Routines

This [figure](#) shows some simple tests of the CCR routines. The figures show the contents of the two CCR Data Arrays ([Construction of the CCR Data Arrays](#)) tested at three time offsets. The CCR Data Arrays contain the data input for cross-correlation. The figures on the right are the resulting CCR Functions ([CCR Processing](#)). This data is output by the routines in data files and has been plotted in MS EXCEL. In each of the tests the CCR routines arrived at exactly the correct answer. In the first test a waveform was cross-correlated with itself ('auto-correlation'). After correlation the CCR Function shows a time shift of 0 s (the maximum positive peak is at $x=0$ on the graph). In the second test (b) the process waveform was artificially shifted by $7\ \mu\text{s}$ to the left (an increase in velocity). The maximum positive peak in the resulting CCR Function moves by $+7\ \mu\text{s}$ in the positive direction indicated on the plot.

In the third test, the opposite was applied (a shift of the process waveform $7\ \mu\text{s}$ to the right) with the resulting CCR Function moving $7\ \mu\text{s}$ in the negative direction.

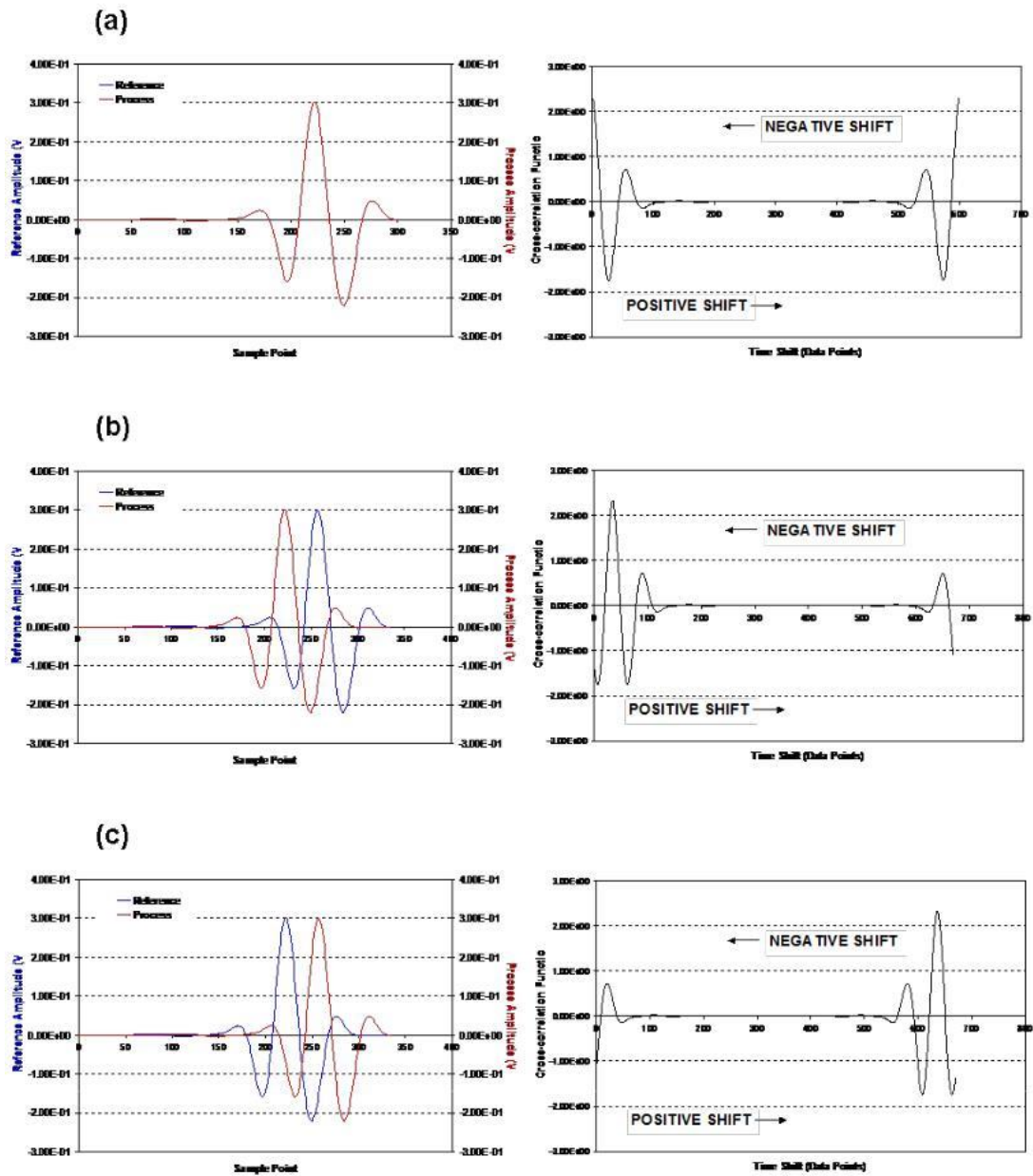


Figure 6.6: Results from testing of the CCR routines. a) Comparison of two identical waveforms with zero offset ($\Delta T=0$ micro seconds). b) Comparison of two identical waveforms with the process waveform shifted 7 micro seconds to the left ($\Delta T=+7$ micro seconds). c) Comparison of two identical waveforms with the process waveform shifted 7 micro seconds to the right ($\Delta T=-7$ micro seconds).

Chapter 7 Advanced Waveform Processing

[Waveform Rotation](#)

[Spectral Processing Algorithms](#)

[Spectral Autopicking](#)

[Waveform Parameters](#)

[Confidence Number](#)

[Akaike Information Criterion AIC](#)

7.1 Waveform Rotation

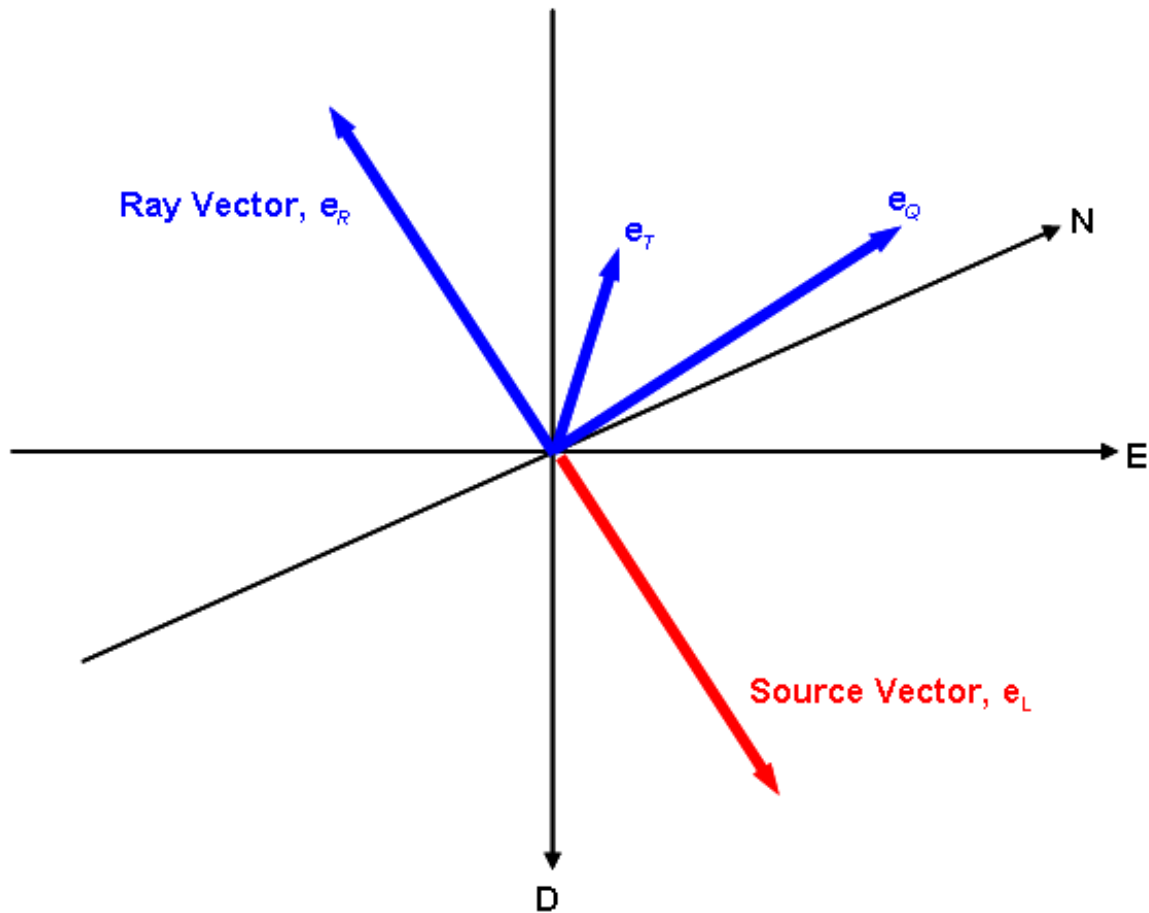
[Coordinate System Geometry](#)

[The Montalbetti-Kanasewich Method](#)

[Single station Location](#)

7.1.1 Coordinate System Geometry

The ray coordinate system is defined here following Pujol and Hermann (1990) and Plesinger et al. (1986) is shown in Figure 7.1 and Figure 7.2.



Two unit vectors are defined. The Source Vector, e_L , points in the direction receiver to source location. The Ray Vector, e_R , points in the direction source location to receiver and represents the direction of the P-wave. The correct amplitude and sense of motion on the P-wave is obtained when the waveforms are rotated into the Ray Vector.

The two vectors have an azimuth (measured east of north) and an incidence angle (measured from the down direction). The angles (α , i) define the source vector and the angles (ϕ , θ) define the ray vector where:

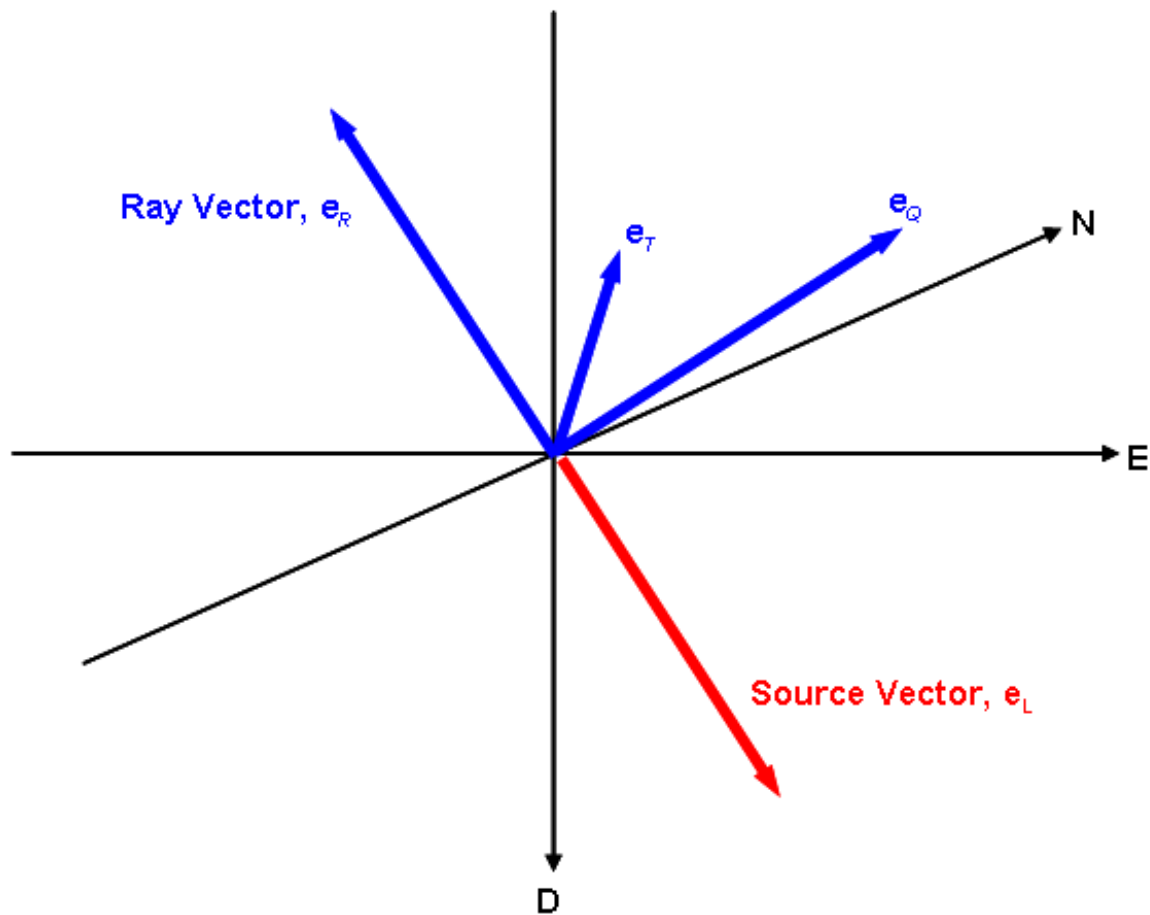
$$0 \leq \phi \leq 360^\circ \quad (7.1)$$

$$0 \leq \theta \leq 360^\circ \quad (7.2)$$

The azimuth of the source vector is termed the Back Azimuth and for the ray vector is termed the Forward Azimuth.

The plane of S-wave polarisation is orthogonal to the ray vector. It is useful to decompose the S-wave into two orthogonal vectors, e_T and e_Q ; e_Q is in a vertical plane containing the source and instrument (S_V direction) and e_T is in a horizontal plane (S_H direction). The three vectors then define the (e_R ; e_T ; e_Q) right-handed coordinate

system described in



. These vectors are obtained with the equations:

$$e_T = \frac{e_z \otimes e_R}{|e_z \otimes e_R|} \quad (7.3)$$

and

$$e_Q = e_T \otimes e_R \quad (7.4)$$

Where e_z is a unit vector in the down direction $e_z = (0, 0, 1)$.

Alternatively, they can be obtained using the Azimuth and Incidence angles through the following equations

$$e_R = (\sin\theta \cos\varphi, \sin\theta \sin\varphi, \cos\theta) \quad (7.5)$$

$$e_T = (-\sin\varphi, \cos\varphi, 0) \quad (7.6)$$

$$e_Q = (\cos\theta \cos\varphi, \cos\theta \sin\varphi, -\sin\theta) \quad (7.7)$$

If the time series X_i , Y_i and Z_i represent the waveforms on each axis (1, 2, 3) of an instrument then the resulting waveform, W_i , from a rotation onto a unit vector is given by the cosines ($\gamma_1, \gamma_2, \gamma_3$) of the angles between the vector and each of the unit vectors defining the instrument coordinate system

$$W_i = X_i \gamma_1 + Y_i \gamma_2 + Z_i \gamma_3 \quad (7.8)$$

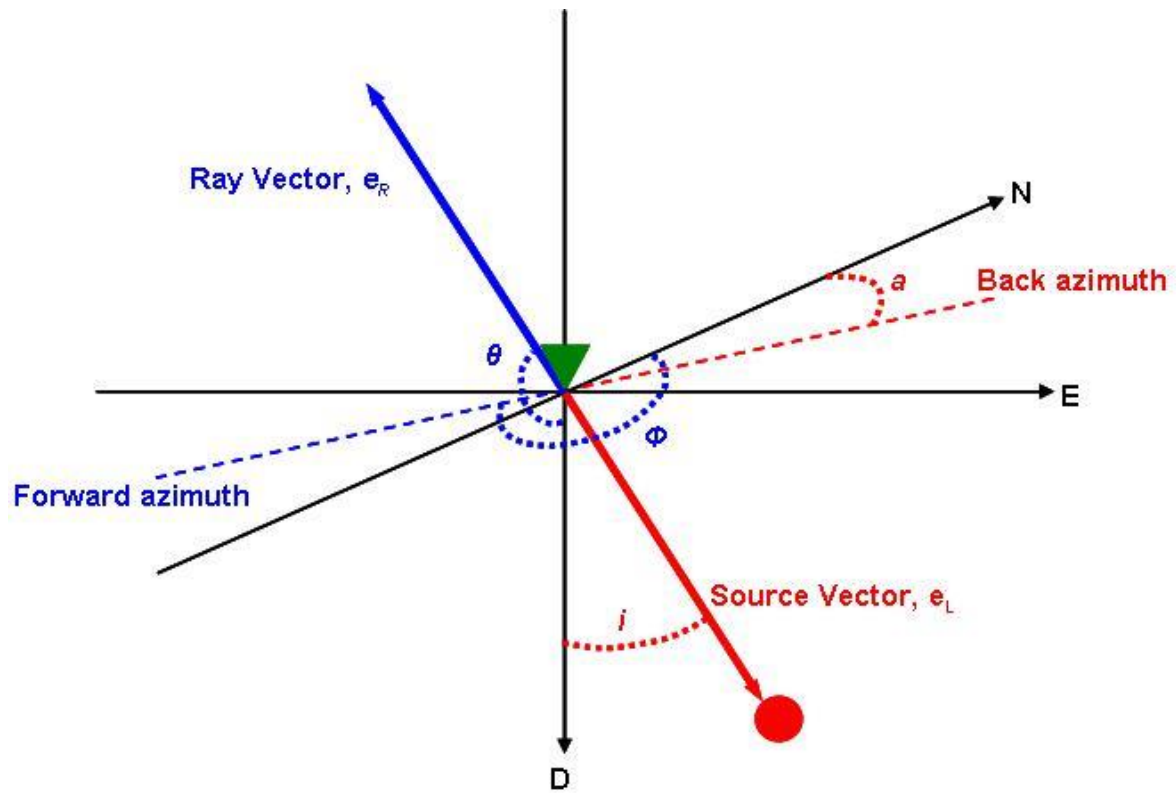


Figure 7.1: Definition of the ray and source vectors. The red circle is the source location and the green triangle is the receiver.

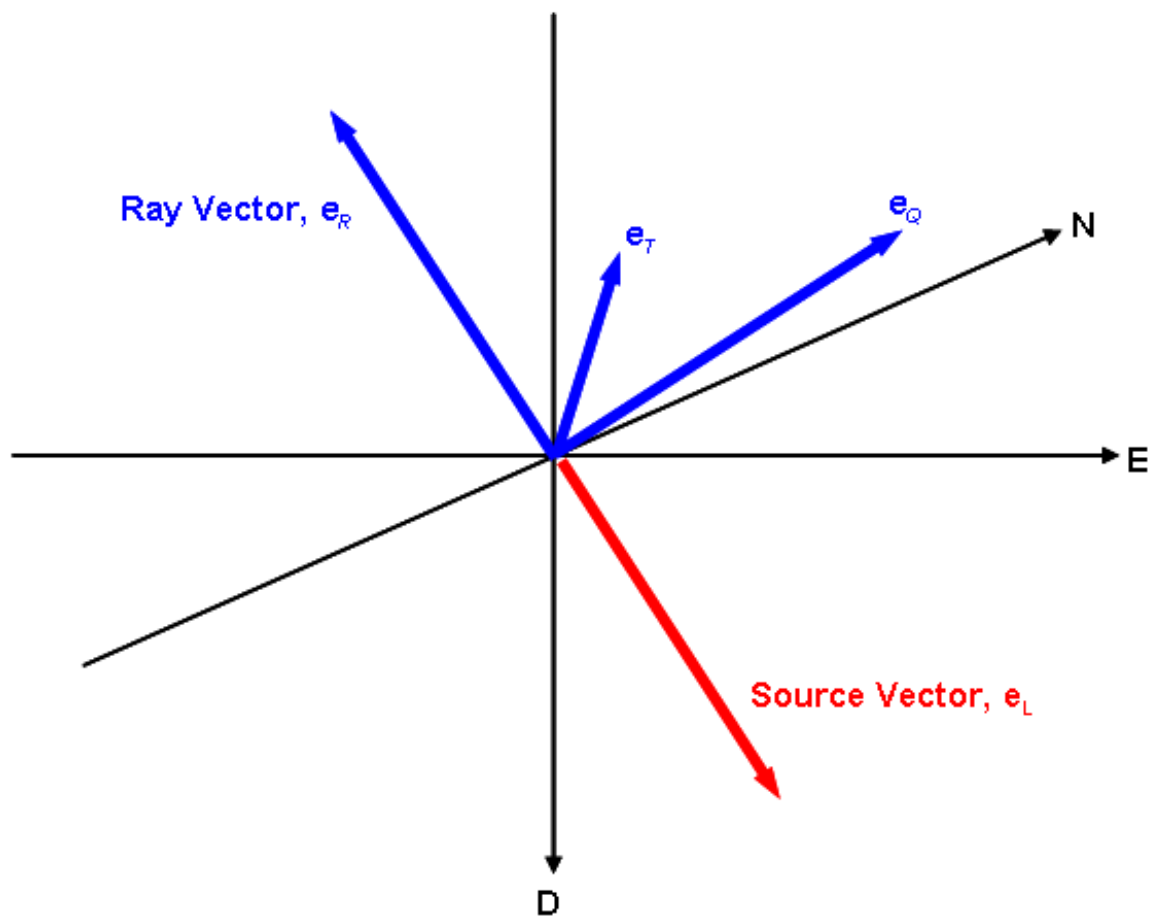


Figure 7.2: The P-SH-SV ray coordinate system represented by the e_R , e_T , e_Q unit vectors.

7.1.2 The Montalbetti-Kanasewich Method

This method is described further by Montalbetti and Kanasewich (1970). The method uses the covariance matrix constructed from the waveform data recorded on the three instrument axes.

If the time series X_i , Y_i and Z_i represent the data in the Rotation Window on each of the waveforms then

$$V = \begin{bmatrix} \text{Var}[X] & \text{Cov}[X, Y] & \text{Cov}[X, Z] \\ \text{Cov}[X, Y] & \text{Var}[Y] & \text{Cov}[Y, Z] \\ \text{Cov}[X, Z] & \text{Cov}[Y, Z] & \text{Var}[Z] \end{bmatrix} \quad (7.9)$$

defines the covariance matrix.

$\text{Cov}[X, Y]$ is the covariance between the waveform data in X and Y . $\text{Var}[X]$ is the variance in the waveform data X .

$$\text{Cov}[X, Y] = \frac{1}{N} \sum_{i=1}^N (X_i - \mu_X)(Y_i - \mu_Y) \quad (7.10)$$

μ_X is the mean of the waveform data in X .

The covariance matrix is then transformed into its three eigenvalues and eigenvectors. The eigenvector associated with the largest eigenvalue represents the P-wave polarisation direction. This could be either the ray or source vector. The other two eigenvectors represent the S_1 and S_2 directions since they are in the S-wave polarisation plane.

A linearity coefficient is calculated after Samson and Olson (1980). This represents the effectiveness of the rotation by considering that the maximum eigenvalue should be much greater than the other two. If $\lambda_1, \lambda_2, \lambda_3$ are the three eigenvalues ordered such that $\lambda_1 < \lambda_2 < \lambda_3$ then the rotation linearity is obtained from:

$$P = \frac{(\lambda_1 - \lambda_2)^2 + (\lambda_1 - \lambda_3)^2 + (\lambda_2 - \lambda_3)^2}{2(\lambda_1 + \lambda_2 + \lambda_3)^2} \cdot 100 \quad (7.11)$$

The obtained eigenvectors are in the instrument coordinate system defined by the three axial unit vectors. In order to get absolute polarisation directions the eigenvectors must be transformed into the (N,E,D) coordinate system by multiplying each eigenvector with the axial unit vectors.

7.1.3 Single station Location

A single-station location is computed from the P-to-S separation time, $T_S - T_P$. The S-wave arrival time, T_S is computed as the mean of the two S_1 and S_2 arrival picks. The distance to the source, D , is then given by:

$$D = \frac{(T_P - T_S) \cdot v_p \cdot v_s}{v_p - v_s} \quad (7.12)$$

Knowing the Instrument Location, the source vector (a unit vector) and the distance to the source, a location for the source can then be obtained.

7.2 Spectral Processing Algorithms

[Taper Types](#)

[Power Spectrum Estimation Methods](#)

[Polarisogram Algorithm](#)

7.2.1 Taper Types

A taper is used to remove leakage frequencies from the Fourier Transform of a time window in the waveform. Leak-age frequencies are produced by frequency components either side of a discrete frequency bin. These leakage frequencies would produce artefacts if they were not attenuated using the taper method. A taper is applied to a waveform in the time domain.

The equations for the two taper types used in InSite are as follows:

Hann taper:

$$w_j = \frac{1}{2} \left[1 + \cos \left(\frac{2\pi j}{N} \right) \right] \quad (7.13)$$

where N is number of data points in the waveform window and j is the jth data point.

Hildyard taper:

$$w_j = e^{-10\tau^2} \quad (7.14)$$

Where the parameter τ is given by the following:

$$1 \leq j \leq N_\tau, \tau = \frac{N_\tau - j}{N_\tau + 1} \quad (7.15)$$

$$(N_w - N_\tau) < j \leq N_w, \tau = \frac{j - (N_w - N_\tau)}{N_\tau + 1} \quad (7.16)$$

$$N_\tau \leq j \leq (N_w - N_\tau), \tau = 0 \quad (7.17)$$

where j is the j th data point, N is the taper's Minimum-to-maximum transition-length ($N_\tau = 10$) and N_w is the window length.

7.2.2 Power Spectrum Estimation Methods

The two power spectrum estimation methods used are the Fast-Fourier-Transform and the Maximum Entropy method. Below, we briefly detail the equations used in these methods (Press et al., 1994).

The Fourier Transform equations are as follows:

$$H(f) = \int_{-\infty}^{\infty} h(t) e^{2\pi i f t} dt \quad (7.18)$$

And

$$h(t) = \int_{-\infty}^{\infty} H(f) e^{-2\pi i f t} df \quad (7.19)$$

where $h(t)$ is a function in the time domain with parameter t (time) and $H(f)$ is a function in the frequency domain with parameter f (frequency).

The above integral is approximated by the following discrete sum:

$$H(f_n) = \int_{-\infty}^{\infty} e^{2\pi i f_n t} dt \approx \sum_{k=0}^{N-1} h_k e^{2\pi i f_n t_k \Delta} = \Delta \sum_{k=0}^{N-1} h_k e^{\frac{2\pi i k n}{N}} \quad (7.20)$$

Where $f_n = \frac{n}{N\Delta}$ and Δ is the sampling frequency.

Let us denote the final summation by H_n ,

$$H_n \equiv \sum_{k=0}^{N-1} h_k e^{\frac{2\pi i k n}{N}} \quad (7.21)$$

Then the inverse Fourier transform is given by:

$$h_k = \frac{1}{N} \sum_{n=0}^{N-1} H_n e^{-\frac{2\pi i k n}{N}} \quad (7.22)$$

The Fast Fourier Transform method is given by the Danielson-Lanczos Lemma:

$$\begin{aligned} F_k &= \sum_{j=0}^{N-1} e^{\frac{2\pi i j k}{N}} f_j \\ &= \sum_{j=0}^{\frac{N}{2}-1} e^{\frac{2\pi i k(2j)}{N}} f_{2j} + \sum_{j=0}^{\frac{N}{2}-1} e^{\frac{2\pi i k(2j+1)}{N}} f_{2j+1} \\ &= \sum_{j=0}^{\frac{N}{2}-1} e^{2\pi i k j / N/2} f_{2j} + e^{\frac{2\pi i k}{N}} \sum_{j=0}^{\frac{N}{2}-1} e^{2\pi i k j / N/2} f_{2j+1} \\ &= F_k^e + W^k F_k^o \end{aligned} \quad (7.23)$$

Where $W = e^{\frac{2\pi i}{N}}$, F_k^e denotes the k^{th} component of the Fourier transform of length $N/2$ formed from the even components of the original f_j 's, while F_k^o is the corresponding transform of length $N/2$ formed from the odd components.

In the FFT algorithm this Lemma is used recursively to produce smaller and smaller sums. The FFT algorithm has order $O(N \log(N))$.

The Maximum Entropy Method equations are as follows:

First, we need to transform the complex f -plane to a new plane, called the z -transform plane, by the relation:

$$z \equiv e^{2\pi i f \Delta}$$

where Δ is the sampling interval in the time domain. The Nyquist interval on the real axis of the f -plane maps one-to-one onto the unit circle in the complex z -plane.

The FFT power spectrum estimate for any real sampled function $c_k \equiv c(t_k)$ can be written as:

$$P(f) = \left| \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} c_k z^k \right|^2 \quad (7.24)$$

This is only an estimate of the true power spectrum of the underlying function $c(t)$. We can see in two related ways why the estimate is not likely to be exact. First, in the time domain, the estimate is based in only a finite range of the function $c(t)$. There is a possibility that this function could have continued on the interval $t = (-\infty, +\infty)$. Second, in the z -plane of this equation, the finite Laurent series offers only an approximation to a general analytic function of z . A formal expression for representing "true" power spectra is:

$$P(f) = \left| \sum_{k=-\infty}^{\infty} c_k z^k \right|^2 \quad (7.25)$$

This is an infinite Laurent series which depends on an infinite number of values c_k , which is just one kind of analytic approximation to the analytic function of z . The model spectrum can have zeros in the z -plane, but no poles.

We can approximate the equation for z using a better method which involves a rational function, one with a series in both the numerator and the denominator

$$P(f) = \frac{1}{\left| \sum_{k=-M/2}^{M/2} b_k z^k \right|^2} = \frac{a_0}{\left| 1 + \sum_{k=1}^M a_k z^k \right|^2} \quad (7.26)$$

Here the second equality brings in a new set of coefficients a_k 's, which can be determined from the b_k 's using the fact that z lies on the unit circle. The b_k 's can be thought of as being determined by the condition that power series expansion of this equation agree with the first $M + 1$ terms of the Laurent series.

The Laurent series and the second approximation have very different character. The most notable fact is that the second approximation can have poles, corresponding to infinite power spectral density, on the unit z -circle, i.e., at real frequencies in the Nyquist interval. Such poles can provide an accurate representation for underlying power spectra that have sharp, discrete "lines" or delta-functions. By contrast, the Laurent series can only have zeros, not poles, at real frequencies in the Nyquist interval, and thus must attempt to fit sharp spectral features with, essentially, a polynomial. The second approximation is the definition for our maximum entropy method (MEM).

The number of poles should be limited to a few times the number of sharp spectral features that one desires to fit. With this restricted number of poles, the method will smooth the spectrum somewhat, but this is often a desirable property.

7.2.3 Polarisogram Algorithm

Moriya and Niitsuma (1996) present the polarisation algorithm used here. A covariance matrix in the frequency domain can be used to evaluate the particle motion of a seismic wave. Samson (1997, 1983) used a spectral matrix as a function of frequency. Moriya and Niitsuma (1996) treat the terms in the spectral matrix as a joint function of time and frequency. The spectral matrix is defined with the time-frequency distributions represented by:

$$a_p(t_j, f_k) = \begin{pmatrix} S_{xx}(t_j, f_k) & S_{xy}(t_j, f_k) & S_{xz}(t_j, f_k) \\ S_{yx}(t_j, f_k) & S_{yy}(t_j, f_k) & S_{yz}(t_j, f_k) \\ S_{zx}(t_j, f_k) & S_{zy}(t_j, f_k) & S_{zz}(t_j, f_k) \end{pmatrix} \quad (7.27)$$

where $S_{im}(t_j; f_k)$ [$i, m = x, y, z$] are the time-frequency distributions calculated from a three component signal. If $A(f)$ and $B(f)$ are the Fourier transforms of the waveforms on two channels $a(t)$ and $b(t)$ then the time-frequency distributions are calculated using the following equations:

$$S_{aa} = A(f)A(f)^*, \quad S_{bb}(f) = B(f)B(f)^*, \quad S_{ab} = A(f)B(f) \quad (7.28)$$

where $S_{aa}(f)$ and $S_{bb}(f)$ are the power spectra and $S_{ab}(f)$ is the cross spectrum. The asterisk in the equations denote the complex conjugate.

The spectral matrix is generally complex but is always a Hermitian matrix. This matrix is considered to be a covariance matrix in both the time and frequency domains. Particle motion can be quantitatively evaluated using eigenvalue/eigenvector analysis of the matrix.

Benhama et al. (1988) presented a function using eigenvalues of the covariance matrix to characterize the 3-D particle motion. Replacing this function with eigen values from the spectral matrix, enables us to evaluate the linearity of particle motion as a joint function of time and frequency. From Moriya and Niitsuma (1996) use the following function:

$$C_R(t_j, f_k) = \frac{(\lambda_1 - \lambda_2)^2 + (\lambda_1 - \lambda_3)^2 + (\lambda_2 - \lambda_3)^2}{2(\lambda_1 + \lambda_2 + \lambda_3)^2} \quad (7.29)$$

Where $\lambda_i = \lambda_i(t_j, f_k)$ ($\lambda_1 > \lambda_2 > \lambda_3, i=1, 2, 3$)

The λ_i are the eigenvalues of the spectral matrix in this [equation](#). The range of $C_R(t_j, f_k)$ is, $0 \leq C_R(t_j, f_k) \leq 1$ and is equal to zero when the particle motion is spherical in space and approaches the value one as the particle motion becomes rectilinear. $C_R(t_j; f_k)$ can be treated as a 3-D cross-correlation coefficient.

The eigenvector associated with the largest eigenvalue represents the P-wave polarisation direction. This could be either the ray or the source vector. The obtained eigenvectors are in the instrument coordinate system defined by the three axial unit vectors. In order to get absolute polarisation directions, the eigenvectors must be transformed into the (n, e, d) coordinate system by multiplying each eigenvector with the axial unit vectors.

7.3 Spectral Autopicking

The spectral auto-picking algorithm in InSite automatically processes phase arrivals from spectrogram information.

See Section 3.9.3 of the User Manual for further information.

In the spectral auto-picking algorithm the spectral estimation method is always set as FFT, as the MEM cannot evaluate frequency phase data used in the polarisogram functions. The GateLength, GateType and TaperType are user set from the processing properties. A StepSize of 1 is always used so that an FFT is produced at every data point in the waveform (except in the last GateLength number of data points that are invalid). A frequency range is applied in the analysis, which defines the minimum and maximum frequency limits across which the spectral picking function is evaluated.

For each data point in the waveform a "spectral density", S is calculated for each of the four spectrograms to provide a two-dimensional function (S versus time) that can be used as an auto-picking function. The spectral density is calculated as the mean of the data contained in the frequency spectrum at that waveform data point,

$$[S_{polarisation}]_i = [mean(FFT_{polarisation})]_i = \frac{\sum_{j=1}^{NFFT} [FFT_{polarisation}]_j}{NFFT} \quad (7.30)$$

where $FFT_{polarisation}$ is the polarisation frequency spectra evaluated for waveform data point i between the user-defined frequency limits (containing $NFFT$ spectra data points, j).

Four spectral density functions are thus calculated using this approach: $S_{polarisation}$; $S_{azimuth}$; S_{plunge} ; $S_{sonogram}$.

Generally, azimuth and plunge functions do not have a character that can be directly used as an auto-picking function – i.e. containing peaks where phase arrivals are positioned. A further two spectral density functions have therefore been developed as options. These are calculated as a function of the variance in the azimuth and plunge spectrograms and operate on the observation that when a signal arrives its variation in e.g. azimuth across the spectra becomes low. The inverse of the variance is calculated in order to provide peaks in the function when the variance is low.

$$[S_{var(azimuth)}]_i = \frac{NFFT}{\sum_{j=1}^{NFFT} |[FFT_{azimuth}]_j - [mean(FFT_{azimuth})]_i|} \quad (7.31)$$

An auto-picking equation is defined by the user, which can include any combination of the four spectral density functions. The functions are defined as variables in the equation in the properties sheet: pol ; azm ; plu and son where e.g. pol is the polarisation spectral density function multiplied by a user-defined coefficient, a :

$$pol_i = a \cdot [S_{polarisation}]_i \quad (7.32)$$

Three further coefficients, b , c , d , are defined respectively for azimuth, plunge and sonogram.

$$azm_i = b \cot[S_{azimuth}]_i \quad \text{or} \quad azm_i = b \cdot [S_{var(azimuth)}]_i$$

$$plu_i = c \cdot [S_{plunge}]_i \quad \text{or} \quad plu_i = c \cdot [S_{var(plunge)}]_i$$

$$son_i = d \cdot [S_{sonogram}]_i \quad (7.33)$$

A fifth auto-picking variable, rms, can also be included in the auto-picking equation, along with an accompanying coefficient, e. This is the RMS (root-mean-square) signal from the given waveforms on the instrument;

$$W_i = \sqrt{\frac{\sum_{m=1}^{m=M} W_{im}^2}{M}} \quad rms_i = e \cdot W_i \quad (7.34)$$

where W_{im} is the waveform amplitude on channel m at point i ($1 \leq M \leq 4$).

Thus, an equation of the form given in the equation below can be defined by the user where F_i is a function at waveform data point i .

$$F_i = a \cdot [S_{polarisation}]_i + b \cdot [S_{azimuth}]_i + c \cdot [S_{plunge}]_i + d \cdot [S_{sonogram}]_i + e \cdot W_i \quad (7.35)$$

The user also has the option of applying the "Moving-window Technique" (a variation on the STA/LTA format) to the function to create the final function to be used in the auto-picking, autofunc (equation below where MW (F) designates application of the moving-window technique to function F).

$$autofunc = MW(F) \text{ or } autofunc = F \quad (7.36)$$

The auto-picking function in InSite's standard auto-picking algorithm is given by

$$autofunc = M W (W) \quad (7.37)$$

, where W is defined by as above.

Thus, the standard auto-picking functionality can be created in the advanced auto-picking equation by setting $F = rms$; $e = 1.0$ and enabling the Moving-window Technique.

7.4 Waveform Parameters

The waveform parameters are as follows:

Amplitude: the maximum amplitude of a waveform.

Rise Time: the time between the first threshold crossing and the maximum amplitude of the signal.

Duration: the time between the first and final threshold crossings of the signal.

MARSE: Measured area under rectified signal envelope: the rectified signal envelope is determined from the first threshold crossing, subsequent peaks and the final threshold crossing. The area under this envelope is calculated using the trapezium rule for polygons defined by the peaks of the signal.

Counts: the count is the number of times that the signal exceeds the defined threshold.

Average frequency: this is the average frequency of the counts. It is calculated as:

$$\text{Average frequency} = \frac{\text{number of counts}}{\text{duration of signal}} \quad (7.38)$$

Reverberation frequency: the frequency of counts during the signal after the maximum peak. It is calculated as:

$$\text{Reverberation frequency} = \frac{\text{number of counts after max peak}}{\text{duration} - \text{rise time}} \quad (7.39)$$

Initial frequency: the frequency of counts before the maximum peak of the signal has been reached. It is calculated as:

$$\text{Initial frequency} = \frac{\text{number of counts before peak}}{\text{rise time}} \quad (7.40)$$

RA value: is calculated as:

$$\text{RA value} = \frac{\text{rise time}}{\text{max amplitude}} \quad (7.41)$$

7.5 Confidence Number

The confidence number for an event is assigned based on the signal to noise ratio, number of sensors picked, arrival time residuals, etc. The confidence number consists of 5 separate numbers, each between 0 and 1, which are added together for a total score between 0 and 5. The exception to this is if the average signal to noise ratio is less than 1, the score can be negative.

The 5 terms are:

- Signal to noise ratio for P-waves
- Signal to noise ratio for S-waves
- Azimuth residual
- P-wave arrival time residuals
- S-wave arrival time residuals

$$N_{\text{confidence}} = N_{p \text{ amp ratio}} + N_{s \text{ amp ratio}} + \max(\text{penalty}_p, \text{penalty}_s) \cdot [N_{ax \text{ res}} + N_{p \text{ res}} + N_{s \text{ res}}]$$

(7.42)

Signal to noise ratio for P- and S- waves (Terms 1 and 2):

$$N_{amp\ ratio} = Num_{picked} \frac{[n_{avg\ snr} - 1]}{10} \quad (7.43)$$

$$Num_{picked} = \begin{cases} \frac{n_{picked}}{8}, & n_{picked} < 8 \\ 1, & n_{picked} \geq 8 \end{cases} \quad (7.44)$$

Where Num_{picked} is the number of P-(S-) waves picked for this event.

$$n_{avg\ snr} = \begin{cases} Avg(snr), & Avg(snr) < 11 \\ 11, & Avg(snr) \geq 11 \end{cases} \quad (7.45)$$

In applying these equations it becomes clear that the maximum contributions to the confidence number occur when:

- At least 8 picks
- Average signal to noise ratio is at least 11

Azimuth residual (Term 3)

$$N + az\ res = \max(\text{penalty}(P, S)) \cdot Az_{res} \quad (7.46)$$

$$\text{penalty} = \begin{cases} \frac{Avg(snr)}{3}, & Avg(snr) < 3 \\ 1, & Avg(snr) \geq 3 \end{cases} \quad (7.47)$$

$$Az_{res} = \begin{cases} \frac{Num_{Az} \cdot (30 - StdDev(Az))}{25}, & StdDev(Az) \leq 30 \\ 0, & StdDev(Az) > 30 \end{cases} \quad (7.48)$$

$$Num_{Az} = \begin{cases} \frac{Num_{Az_picked}}{8}, & Num_{Az_picked} < 8 \\ 1, & Num_{Az_picked} \geq 8 \end{cases} \quad (7.49)$$

Arrival time residuals (Terms 4 and 5)

$$N_{res} = \begin{cases} \frac{Num_{Az_picked} \cdot (2.5 - Avg(res))}{2.25}, & Avg(res) \leq 2.5 \\ 0, & Avg(res) > 2.5 \end{cases} \quad (7.50)$$

$$Avg(res) = \begin{cases} 0.25, & Avg(res) < 0.25 \\ Avg(res), & Avg(res) \geq 0.25 \end{cases} \quad (7.51)$$

7.6 Akaike Information Criterion AIC

The AR-AIC Picker uses Autoregressive (AR) modelling to provide improved estimates of the pick position.

[AIC Simple](#)

7.6.1 AIC Simple

This method calculates a simple AIC algorithm for the entire waveform. It is calculated by:

$$AIC(K) = K * \log(\text{var}(A[1, k])) + (N - K - 1) * \log(\text{var}(A[k + 1, N])) \quad (7.52)$$

Where:

- K is the position in the waveform array
- N is the number of points in the waveform
- A is the waveform array
- $\text{var}(X)$ is the variance of input X

References

- K Aki and P G Richards. Quantitative seismology. University Science Books, 2002.
- A. Benhama, C. Cllet, and M. Dubesset. Study and Applications of Spatial Directional Filtering in Three-Component Recordings. *Geophysical Prospecting*, 36(6):591–613, aug 1988. ISSN 0016-8025. doi: 10.1111/j.1365-2478.1988.tb02182.x. URL <http://doi.wiley.com/10.1111/j.1365-2478.1988.tb02182.x>.
- D. M. Boore and J. Boatwright. Average body-wave radiation coefficients. *Bulletin of the Seismological Society of America*, 74(5):1615–1621, 1984. ISSN 19433573. URL <http://www.bssaonline.org/content/74/5/1615.short>.
- V. Cerveny. *Seismic ray theory*. Cambridge University Press, 2001.
- T.M. Chow. Concurrent ultrasonic imaging and acoustic emission in solid materials. PhD thesis, Queen's University, Kingston, Ontario, Canada, 1992.
- D. S. Collins and R. P. Young. Lithological controls on seismicity in granitic rocks. *Bulletin of the Seismological Society of America*, 90(3):709–723, 2000.
- D.S. Collins, W.S. Pettitt, and R.P. Young. High-resolution mechanics of a micro-earthquake sequence. *Pure and Applied Geophysics*, 159:197–219, 2000.
- G. De Natale and Aldo Zollo. Earthquake focal mechanisms from inversion of First P and S wave motions. In R. Cassini, G. Nolet, and G.F. Panza, editors, *Digital Seismology and Fine Modelling of the Lithosphere*, pages 319–419. 1989.
- S.D. Falls. Ultrasonic Imaging and Acoustic Emission Studies of Microcrack Development in Lac Du Bonnet Granite. PhD thesis, Queen's University, Kingston, Ontario, Canada,, 1993.
- B. Feignier and R.P. Young. Failure mechanisms of microseismic events generated by a breakout development around an underground opening. In R.P. Young, editor, *Proceedings of the 3rd International Symposium on Rockbursts and Seismicity in Mines*, pages 181–186, Kingston, 1993.
- S. J. Gibowicz and A. Kijko. *An Introduction to Mining Seismology*. Academic Press, 1994. URL <http://linkinghub.elsevier.com/retrieve/pii/B9780122821202500054>.
- Huang, J. J.M. Reyes-Montes and R.P. Young. Automated microseismic event location using finite difference traveltimes calculation and enhanced waveform stacking. *EAGE*, 2013. doi: 10.3997/2214-4609.20130398. URL <http://earthdoc.eage.org/publication/publicationdetails/?publication=69273>.
- R. H. Jones and R. C. Stewart. A method for determining significant structures in a cloud of earthquakes. *Journal of Geophysical Research*, 102(B4):8245, 1997. ISSN 0148-0227. doi: 10.1029/96JB03739.
- M.L. Jost and R.B. Herrmann. Department of Earth and Atmospheric Sciences Saint Louis University P. O. Box 8099 St. Louis, MO 63156. *Seismological Research Letters*, 60(2):37–57, 1989.
- Honn Kao and Shao-Ju Shan. The source-scanning algorithm: mapping the distribution of seismic sources in time and space. *Geophysical Journal International*, 157(2):589–594, 2004. doi: 10.1111/j.1365-246X.2004.02276.x. URL <http://dx.doi.org/10.1111/j.1365-246X.2004.02276.x>.
- R Krohling. Gaussian particle swarm with jumps. *Proc. IEEE Congr. Evol. Comput*, 2:1226–1231, sep 2005.
- N. Langet, A. Maggi, A. Michelini, and F. Brenguier. Continuous kurtosis-based migration for seismic event detection and location, with application to piton de la fournaise volcano, la réunion continuous kurtosis-based migration for seismic event detection and location. *Bulletin of the Seismological Society of America*, 104(1):229, 2014. doi: 10.1785/0120130107. URL <http://dx.doi.org/10.1785/0120130107>.
- W.H.K. Lee. *Toolbox for Seismic Data Acquisition, Processing, and Analysis*. International Association of Seismology and Physics of the Earth's Interior, 1989.
- R. Madariaga. Dynamics of an Expanding Circular Fault. *Bulletin of the Seismological Society of America*, 66(3): 639–666, 1976.
- J. F. Montalbetti and E R Kanasevich. Enhancement of Teleseismic Body Phases with a Polarization Filter. *Geophysical Journal International*, 21(2):119–129, nov 1970. ISSN 0956-540X. doi: 10.1111/j.1365-

- 246X.1970.tb01771.x. URL <https://academic.oup.com/gji/article-lookup/doi/10.1111/j.1365-246X.1970.tb01771.x>.
- H. Moriya and H. Niitsuma. Precise detection of a P -wave in low S/N signal by using time-frequency representations of a triaxial hodogram. *GEOPHYSICS*, 61(5):1453–1466, sep 1996. ISSN 0016-8033. doi: 10.1190/1.1444071. URL <http://library.seg.org/doi/10.1190/1.1444071>.
- J. A. Nelder and R. Mead. A Simplex Method for Function Minimization. *The Computer Journal*, 7(4):308–313, jan 1965. ISSN 0010-4620. doi: 10.1093/comjnl/7.4.308. URL <https://academic.oup.com/comjnl/article-lookup/doi/10.1093/comjnl/7.4.308>.
- Nicolet. System 400 Benchtop Waveform Acquisition System Operation Manual. Technical report, Madison, Wisconsin, 1989.
- M. Ohtsu. Simplified moment tensor analysis and unified decomposition of acoustic emission source: Application to in situ hydrofracturing test. *Journal of Geophysical Research: Solid Earth*, 96(B4):6211–6221, apr 1991. ISSN 01480227. doi: 10.1029/90JB02689. URL <http://doi.wiley.com/10.1029/90JB02689>.
- S. Osher. A Level Set Formulation for the Solution of the Dirichlet Problem for Hamilton–Jacobi Equations. *SIAM Journal on Mathematical Analysis*, 24(5):1145–1152, 1993. doi: 10.1137/0524066. URL <https://doi.org/10.1137/0524066>.
- G. L. Pavlis. Appraising Earthquake Hypocenter Location Errors - a Complete, Practical Approach for Single-Event Locations. *Bulletin of the Seismological Society of America*, 76(6):1699–1717, 1986.
- W.S. Pettitt, C. Baker, R.P. Young, L.-O. Dahlström, and G. Ramqvist. The Assessment of Damage Around Critical Engineering Structures Using Induced Seismicity and Ultrasonic Techniques. *Pure and Applied Geophysics*, 159 (1):179–195, jan 2002. ISSN 0033-4533. doi: 10.1007/PL00001250. URL <http://link.springer.com/10.1007/PL00001250>.
- W.S. Pettitt. Acoustic emission source studies of microcracking in rock. PhD thesis, Keele University, Keele, Staffordshire, UK, 1998.
- A. Plesinger, M. Hellweg, and D. Seidl. Interactive high-resolution polarization analysis of broad-band seismograms. *Journal of Geophysics*, 59:129–139, 1986.
- W. H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C The Art of Scientific Computing*. Cambridge University Press, Cambridge, 1994. ISBN 0521431085.
- J. Pujol and R B Herrmann. A Student’s Guide to Point Sources in Homogeneous Media. *Seismological Research Letters*, 61(3-4):209–224, 1990. ISSN 08950695. doi: 10.1785/gssrl.61.3-4.209. URL <http://srl.geoscienceworld.org/content/61/3-4/209.abstract>.
- J. Qian and W.W. Symes. Finite-difference quasi-P traveltimes for anisotropic media. *Geophysics*, 67(1):147–155, 2002. ISSN 00168033. doi: 10.1190/1.1451438.
- J.M. Reyes-Montes, A Rietbrock, D.S. Collins, and R.P. Young. Relative location of excavation induced microseismicity at the Underground Research Laboratory (AECL, Canada) using surveyed reference events. *Geophysical Research Letters*, 32(5):L05308, 2005. ISSN 0094-8276. doi: 10.1029/2004GL021733. URL <http://doi.wiley.com/10.1029/2004GL021733>.
- E. Rouy and A. Tourin. A Viscosity Solutions Approach to Shape-From-Shading. *SIAM Journal on Numerical Analysis*, 29(3):867–884, 1992. doi: 10.1137/0729053. URL <https://doi.org/10.1137/0729053>.
- J. C. Samson. Matrix and Stokes vector representations of detectors for polarized waveforms: theory, with some applications to teleseismic waves. *Geophysical Journal International*, 51(3):583–603, dec 1977. ISSN 0956-540X. doi: 10.1111/j.1365-246X.1977.tb04208.x. URL <https://academic.oup.com/gji/article-lookup/doi/10.1111/j.1365-246X.1977.tb04208.x>.
- J. C. Samson. The spectral matrix estimates and principle components in the analysis the multichannel geophysical data. *Annales Geophysics*, 1:115–119, 1983.
- J. C. Samson and J. V. Olson. Some comments on the descriptions of the polarization states of waves. *Geophysical Journal International*, 61(1):115–129, apr 1980. ISSN 0956-540X. doi: 10.1111/j.1365-246X.1980.tb04308. x. URL <https://academic.oup.com/gji/article-lookup/doi/10.1111/j.1365-246X.1980.tb04308.x>.
-

- J A Sethian. A fast marching level set method for monotonically advancing fronts. *Proceedings of the National Academy of Sciences of the United States of America*, 93(4):1591–1595, 1996.
- Chi-Wang Shu. High order numerical methods for time dependent Hamilton-Jacobi equations. *Mathematics and Computation in Imaging Science and . . .*, 2007.
- W. Spence. Relative Epicenter Determination using P-Wave Arrival Time Differences. *Bulletin of the Seismological Society of America*, 70:171–183, 1980.
- B.W. Stump and L.R. Johnson. The determination of source properties by the linear inversion of seismograms. *Bulletin of the Seismological Society of America*, 67:1489–1502, 1977.
- W.M. Telford, L.P. Geldart, and R.E. Sheriff. *Applied Geophysics*. Cambridge University Press, Cambridge, UK, 1990.
- L. Thomsen. Weak elastic anisotropy. *Geophysics*, 51(10):1954–1966, oct 1986. ISSN 0016-8033. doi: 10.1190/1.1442051. URL <http://library.seg.org/doi/abs/10.1190/1.1442051>.
- Y. Tian and X. Chen. A rapid and accurate two-point ray tracing method in horizontally layered velocity model. *Acta Seismologica Sinica*, 18(2):154–161, mar 2005. ISSN 1000-9116. doi: 10.1007/s11589-005-0062-4. URL <http://link.springer.com/10.1007/s11589-005-0062-4>.
- Yen-Hsi Richard Tsai, Li-Tien Cheng, Stanley Osher, and Hong-Kai Zhao. Fast Sweeping Algorithms for a Class of Hamilton–Jacobi Equations. *SIAM Journal on Numerical Analysis*, 41(2):673–694, 2003. ISSN 0036-1429. doi: 10.1137/S0036142901396533.
- A. Udias. *Development of fault-plane studies for the mechanism of earthquakes*. University of California Press, Berkeley.
- T. I. Urbancic, C. I. Trifu, R. A. Mercer, A. J. Feustel, and J. A G Alexander. Automatic time-domain calculation of source parameters for the analysis of induced seismicity. *Bulletin of the Seismological Society of America*, 86(5): 1627–1633, 1996. ISSN 00371106.
- V. Vinje, E. Iversen, and H. Gjøystdal. Traveltime and amplitude estimation using wavefront construction. *GEOPHYSICS*, 58(8):1157–1166, aug 1993. ISSN 0016-8033. doi: 10.1190/1.1443499. URL <http://library.seg.org/doi/10.1190/1.1443499>.
- Z. Yang, Y. Chen, and H. Zhang. Relocation and seismogenic structure of the 1998 Zhangbei-Shangyi earthquake sequence. *Acta Seismologica Sinica*, 15(4):383–394, jul 2002. ISSN 1993-1344. doi:10.1007/s11589-002-0031-0. URL <https://doi.org/10.1007/s11589-002-0031-0>.
- R P. Young, J.F. Hazzard, and W.S Pettitt. Seismic and micromechanical studies of rock fracture. *Geophysical Research Letters*, 27(12):1767–1770, jun 2000. ISSN 00948276. doi: 10.1029/2000GL011547. URL <http://doi.wiley.com/10.1029/2000GL011547>.
- W. Zhang and J. Zhang. Microseismic migration by semblance-weighted stacking and interferometry. *SEG*, 2003. doi: 10.1190/segam2013-0970.1. URL <https://doi.org/10.1190/segam2013-0970.1>.
- H. Zhao. A fast sweeping method for Eikonal equations. *Mathematics of Computation*, 74(250):603–628, 2004. ISSN 0025-5718. doi: 10.1090/S0025-5718-04-01678-3. URL <http://www.ams.org/journal-getitem?pii=S0025-5718-04-01678-3>.