Instruction

- Analysis of CCA Method -

APR. 05, 2021

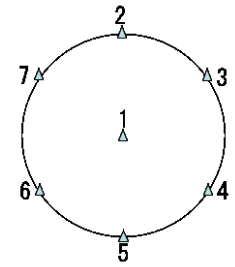
IISEE, BRI, Japan

By T. Yokoi

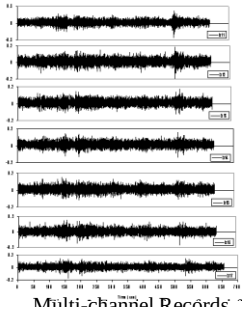
Acknowledgements:
The revision for the 2021 version was partly supported by Japan Science and Technology Agency (JST) and Japan International Cooperation Agency (JICA) under the “Science and Technology Research Partnership for Sustainable Development (SATREPS): Integrated Research on Great Earthquakes andDisaster Mitigation in Nepal Himalaya (FY2016-2021)”.

Procedures of analysis

Field Work

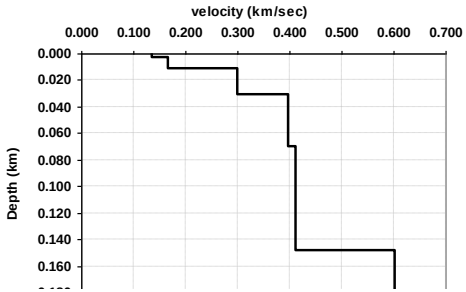


Microtremor Array Measurement
(Vertical Component)



1. Multi-plexing & Resampling

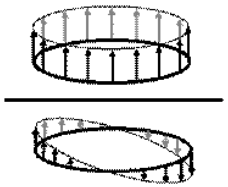
Estimation of Shear Wave Velocity Structure

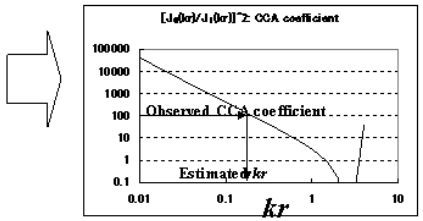


4. Heuristic Search of V_s Structure

Inversion or
Heuristic Search

Calculation of CCA Coefficient

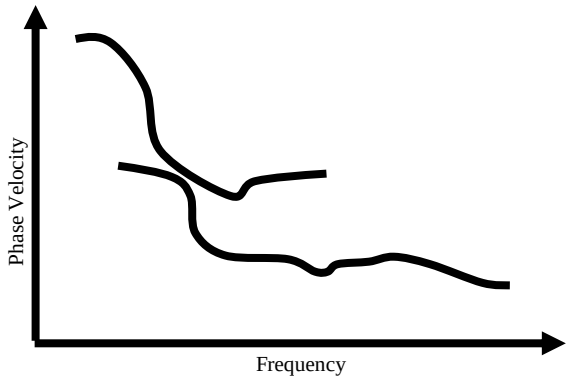




2. Calculation of CCA coefficient

Least Square Fitting to $J_0(kr)$

Calculation of the phase velocity



3. Determination of Dispersion Curve

Note: This version was developed on Linux: Ubuntu 19.10 on Virtual Box 6.1 on Windows10 Home 64bit (Build 14393) for 64bit PC, using gfortran compiler.

Operation on other OS may require additional revision or modification by users themselves.

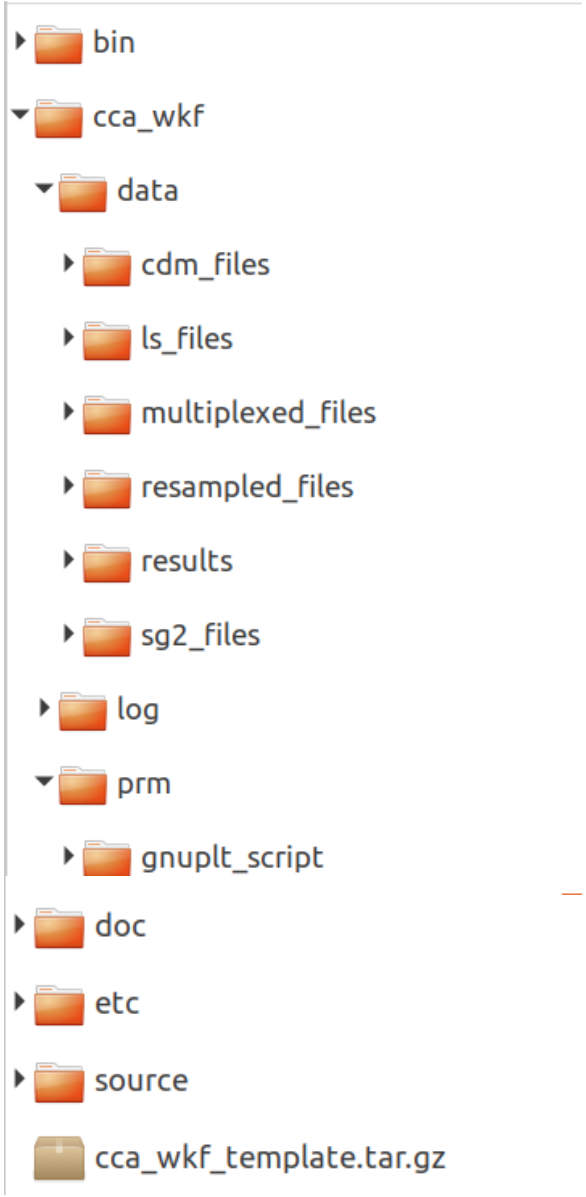
Execution of commands is conducted as

./bin/*executable_file_name*.exe
or
sh *shell_script_file_name*.sh

If it is necessary to leave log file of execution

./bin/*executable_file_name*.exe 2>&1 | tee ./spacwkf/log/*log_file_name*.log
or
sh *shell_script_file_name*.sh 2>&1 | tee ./spacwkf/data/log/*log_file_name*.log

Folder Structure



Every necessary programs and files are stored under the folder “CCA2021”. The command operation must be conducted in the same folder, where shell script files are stored.

The source codes of the programs are stored in the subfolder “source”, document files including this instruction manual in “doc”, executable ones in “bin”.

The subfolder of work space “cca_wkf” contains the subfolder “prm” for parameter files that includes script files of GNUPLOT and the subfolder “data” for data files including graphic ones.

The compressed file “ccawkf.tar.gz” keeps subfolder structure of “cca_wkf” and all parameter files in “cca_wkf/prm”.

Note: GNUPLOT scripts files

Some files of GNUPLOT scripts are stored under the subfolder

"./cca_wkf/prm/gnuplt_scripts"

These can be loaded on GNUPLOT as load '????'
Some programs create the scripts of GNUPLOT that include the command

'set terminal x11' ,

This works on the GNUPLOT on Ubuntu and may be that on Windows.

If any problem on Windows, it is worth to try to replace it with

'set terminal wxt' .

Note: Executable files

The folder “CCA2021” includes several executable files. Their source code files are stored in the subfolder “./source”. Then, the following command is required to re-compile them if necessary. In the folder CCA2021, type in the following command.

gfortran ./source/???.for –o ./bin/???.exe

In case of problems caused by the incompatibility between Fortran77 and Fortran95,

gfortran -ff2c ./source/???.for –o ./bin/???.exe

Executable files are stored in the subfolder “CCA2021/bin”.

Note: Shell script files

The folder “CCA2021” includes several shell script files.
They are composed of few executing commands to reduce the typing tasks in data processing.
The following command can execute the shell script files.

sh *shell_script_file_name*.sh
or
sh ./*shell_script_file_name*.sh

As the contents of the shell script files contained in this program package are simple, they can work as batch files. However, it is necessary to activate batch files using the following.

chmod u+x *shell_script_file_name*.sh

For execution as a batch file,
./*shell_script_file_name*.sh

Note: Cleaning up the subfolders

- Use
  sh clean_all.sh
  in the folder "CCA2021" to delete all files of input data, interim outputs and results for a new processing task.
- Use
  sh clean_sg2.sh
  in the folder "CCA2021" to delete all input files of sg2 format.
- Use
  sh clean_ls.sh
  in the folder "CCA2021" to delete all input files of win format.

Note: Format of Data Files

- Basically, users themselves are responsible to convert the format of the input data files to one of those acceptable formats by this program package.
- The acceptable format is “*.cdm” described in the section “1. Resampling”.
- The following two format converters are provided:

- sg2 (seg2 standard of IEEE)
  ➔./cca_wkf/multiplexed_files/*.dat

- ls (Win format of Hakusan-Kogyo)
  ➔./cca_wkf/cdm_files/*.cdm

Build CCA2021

makefile:

```
FC = gfortran
store= 2>&1 | tee -a ./cca_wkf/log/make_all.log

all:          clean_log
              ${FC} ./source/ch_pivot.for      -o ./bin/ch_pivot.exe  ${store}
              ${FC} ./source/coh_plt.for       -o ./bin/coh_plt.exe   ${store}
              ${FC} ./source/disp_sma1_2.for   -o ./bin/disp_sma1_2.exe ${store}
              ${FC} ./source/fourier_plt.for   -o ./bin/fourier_plt.exe ${store}
              ${FC} ./source/inv_plt.for       -o ./bin/inv_plt.exe   ${store}
              ${FC} ./source/lstocdm2.for      -o ./bin/lstocdm2.exe  ${store}
              ${FC} ./source/mk_title.for      -o ./bin/mk_title.exe  ${store}
              ${FC} ./source/mk_title_post.for -o ./bin/mk_title_post.exe ${store}
              ${FC} ./source/mk_title_pre.for  -o ./bin/mk_title_pre.exe  ${store}
              ${FC} ./source/multi_pre.for     -o ./bin/multi_pre.exe ${store}
              ${FC} ./source/multipx6.for      -o ./bin/multipx6.exe  ${store}
              ${FC} ./source/power_plt.for     -o ./bin/power_plt.exe ${store}
              ${FC} ./source/prm_maker.for     -o ./bin/prm_maker.exe ${store}
              ${FC} ./source/pwrcrs3.for       -o ./bin/pwrcrs3.exe   ${store}
              ${FC} ./source/q_control.for     -o ./bin/q_control.exe ${store}
              ${FC} ./source/resamplec.for     -o ./bin/resamplec.exe ${store}
              ${FC} ./source/resamplec_pre.for -o ./bin/resamplec_pre.exe ${store}
              ${FC} ./source/dc_model.for      -o ./bin/resultc_plt.exe ${store}
              ${FC} ./source/seeblkc.for       -o ./bin/seeblkc.exe ${store}
              ${FC} ./source/seewavc.for       -o ./bin/seewavc.exe ${store}
              ${FC} ./source/seg2read.for      -o ./bin/seg2read.exe ${store}
              ${FC} ./source/vel_model_plt.for -o ./bin/vel_model_plt.exe ${store}

clean:
              cd ./bin/; rm *.exe; cd ..

clean_log:
              rm -f ./cca_wkf/log/make_all.log 2>/dev/null
```

Data Processing

1. Multiplexing & Resampling
    - 1.1 Format Conversion & Multiplexing
        - 1.1.1 *seg2* standard format
        - 1.1.2 *win* format for LS8800
    - 1.2 Resampling & Screening
        - “sh ./resamplec.sh”
    - 1.3 Plot Waveform:
        - “sh ./seewavc.sh”
    - 1.4 Checking the selected time blocks
        - “sh ./seeblkc.sh”
2. Estimating Dispersion Curve
    - 2.1 Calculation of CCA coefficient
        - “sh ./pwrcrs3.sh”
    - 2.2 Plot Power, Fourier Spectra & Coherence
        - “sh ./spectra_all.sh”
    - 2.3 Quality Control & Dispersion Curve
        - “sh ./results.sh”
3. Heuristic Search of Vs Structure
4. Re-arrange graphs *(in preparation)*

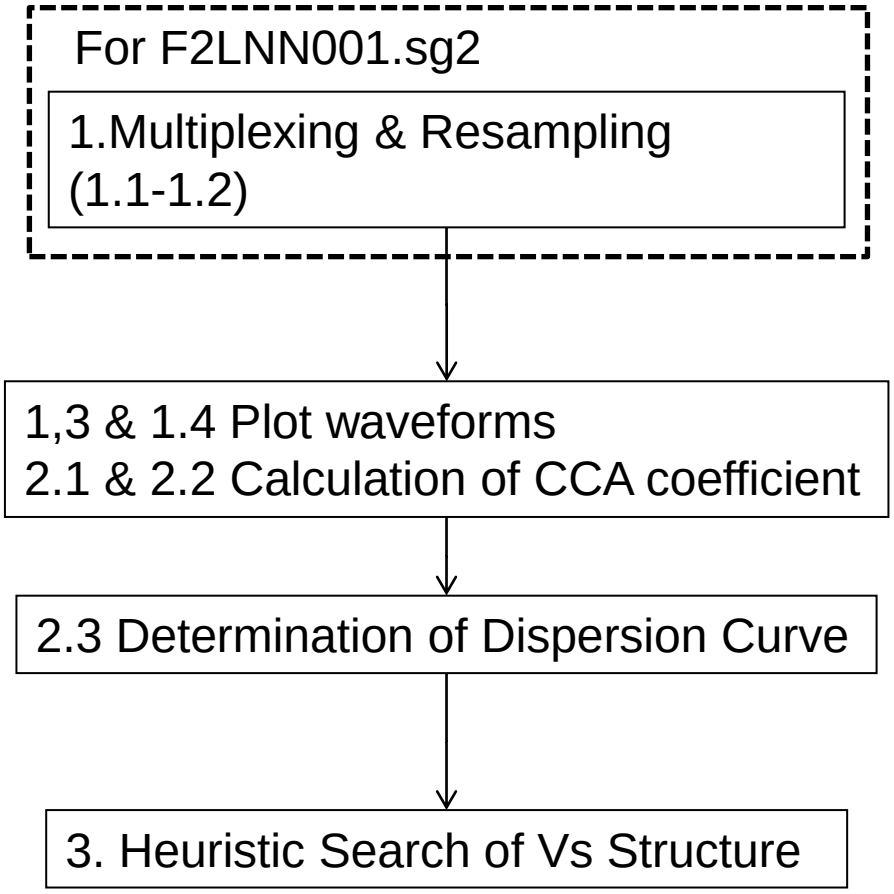
Note: Example1

(Instrumental Correction without sensor at the center)

Example with two data sets:

F2LNN001.sg2: 6 points circular array without one at the center

Both are the seg2 standard format file.



1. Multiplexing & Resampling

1.1 Format conversion & Multiplexing

1.1.1 *seg2* standard format)

Shell Script used:

sh ./seg2read.sh

Program and parameter file used:

./bin/seg2read.exe +./cca_wkf/prm/seg2read.prm

seg2read.exe is prepared for the field data files of seg2 standard format.

Terminology

Multiplexing:

To sort the data individually stored in single channel files into a multi-channel file of the time-sequential format.

Here, *seg2* standard (multiplexed binary) format (IEEE) is explained.

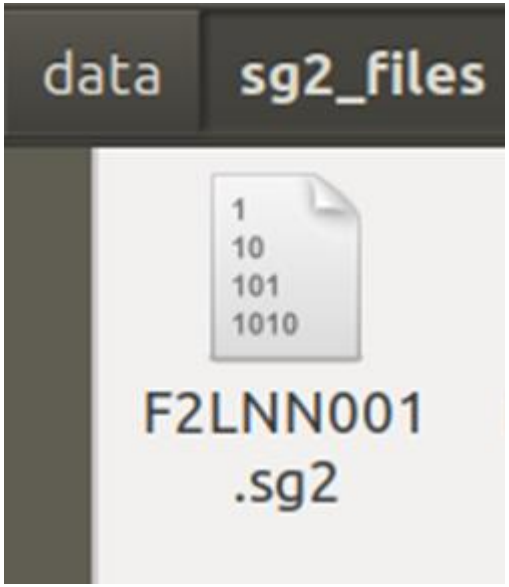
The expected input files of seg2 format are, for example, those obtained in the field using multi-channel data logger designed for the exploration geophysics.

seg2read.exe

+ Convert data format from seg2 standard format (binary & multi-channel) in the sub-folder “cca_wkf/data/sg2_files" to cdm format (ascii text, multi-channel),
+ Channel pivoting and extraction
+ Store the output files into “cca_wkf/data/multiplexed_files”

First: Copy all the seg2 format files to be converted into the sub-folder "cca_wkf/data/sg2_files".

Example:



```
seg2read.sh:

#! /bin/sh -x
cd cca_wkf/data/sg2_files
ls *.sg2 > sg2file.lst
cd ../../..
./bin/seg2read.exe | tee
cca_wkf/log/seg2read.log
cd cca_wkf/data/multiplexed_files
ls *.dat > mltfile.lst
cd ../../..
./bin/mk_title.exe
```

Shell script executes "ls *.sg2 > sg2file.lst" in this sub-folder and existing sg2 files are listed in the newly created file "sg2file.lst".
All the files listed in it that have the extension specified in the 3rd line of the parameter file “seg2read.prm”.

Finally, the first line of “seg2read.prm” is copied to “graph_title.txt” ” in the subfolder “./cca_wkf/prm”.

Example: seg2read.prm

All the files listed in "sg2file.lst" that have the extension specified in the 3rd line are converted to the output files that have the extension given in the 4th line.
Edit the file "sg2file.lst" using “gedit” or other text edior if necessary.

seg2read.prm

```
LCCM,Field,r=2m,L22D,No_Rs,No_A_amp,D_amp=X1,F2LNN0,Mar.15,2012      :comm(a70)
5.E-3    : scaling factor (for output files in mkine(1.e-3cm/s))
sg2      : extension of input seg2 format files(a3)
dat      : extension of output ascii text files(a3)
0 3 0.1 1.0 1.5  :nfilter(=1:apply),nchara=3:bandpass),fl,fh,fs
pvlist 1 2 3 4 5 6  : Channel Pivoting
```

The array used in the field has 6 sensors, but none of them at the center. They were installed counter-clockwise order. Then, pvlist 1 2 3 4 5 6 :

Explanation: seg2read.prm

All the files listed in "sg2file.lst" that have the extension specified in the 3rd line are converted to the output files that have the extension given in the 4th line.
Edit the file "sg2file.lst" using “gedit” or other text edior if necessary.

seg2read.prm

1st line : comment (a70)
2nd line : scaling factor (use the value that makes the unit of the output files "mkine" (1.e-3 cm/s))
3rd line : extension of input seg2 format files(a3)
4th line : extension of output ascii text files(a3)➔Fix it “.dat”
5th line : nfilter(=0:pass, =1:apply, =2:DC & Trend removal), ncharacter(=2:lowpass,=3:bandpass),fl,fh,fs
6th line : Channel Pivoting
'normal' : no pivoting, all channel used
'rev_al' : all channel used but in reversed order
'rev_fh' : all channel used but former half in reversed order
'rev_lh' : all channel used but latter half in reversed order
'pvlist' 2 1 3 4 6 23 24 : Pivoting list.

Examples of the 6th line of seg2read.prm

```
Use all channels without pivoting:
        normal    : Channel Pivoting
Use all channels but reversed order:
       reverse   : Channel Pivoting
Use the first 7 channels of the input files without changing order:
        pvlist 1 2 3 4 5 6 7 : Channel Pivoting
The same as above but 7th channel moved to the first:
        pvlist 7 1 2 3 4 5 6 : Channel Pivoting
Use only odd numbered channels among 24 without changing order:
        pvlist 1 3 5 7 9 11 13 15 17 19 21 23 : Channel Pivoting

Note: Be sure to put ‘ ‘(blank) before ‘:’(colon), otherwise the
program can have an error in detecting the end of line.
If a channel number in pvlist is negative, the polarity of that
channel is reversed.
```

If a sensor is located at the center of the circler array and used for incoherent noise correction, the corresponding data must be assigned to ch-1. Then, other channels must be listed following the order counter-clockwise. The value of the azimuth of the first peripheral sensor must be given to the parameter ph00 and the in-coming azimuth of the pre-dominant wave is calculated.
“pvlist” can be used to change the channel order.

Format of output files

in ./cca_wkf/data/multiplexed _files:

Users who use single channel recorders or data loggers must multiplex the record files in the following format by themselves.

```
 6  0.0080      0.5500E-01  225000  mkine
LCCM,Field,r=2m,L22D,No_Rs,No_A_amp,D_amp=X1,F2LNN0,Mar.15,2012      :
    0.000000  0.5280000E+01  0.3795000E+01  0.6710000E+01 …
    0.008000  0.9900000E+00  0.2310000E+01 -0.7975000E+01 …
    0.016000  0.6050000E+01  0.1138500E+02 -0.1061500E+02 …
    0.024000  0.3025000E+01  0.7535000E+01 -0.1017500E+02 …
    …
```

1st line: Number of channels, Δt(sec), scale, number of samples,unit
2nd line: Comment (less than 50 characters)
3rd line: Time, 1st-ch sample, 2nd-ch sample, 3rd-ch sample, ….

In the next step (resamplec.for reads this file as follows)

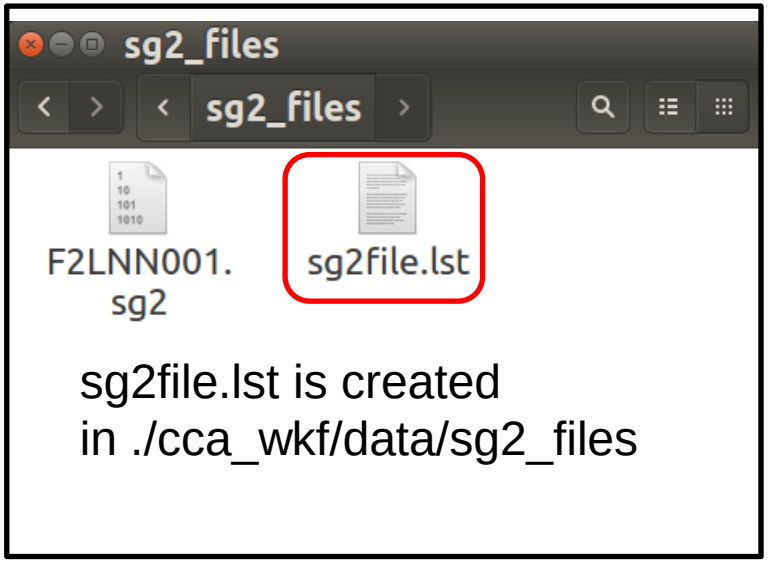
```
read(1,*)nch00,dt00,scale00,ndata00,cunit
…
read(1,'(a50)')comment
…
read(1,*,end=10) xdum,(x(i,j),j=1,nch)
```

Warning!

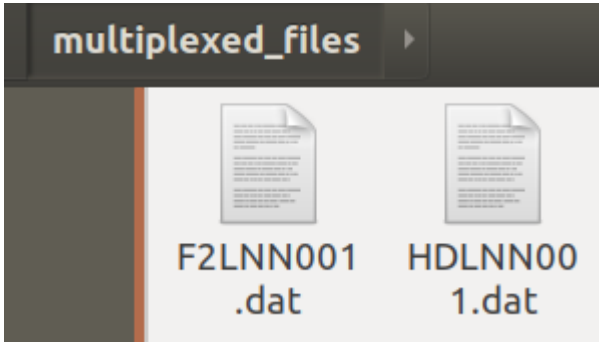
seg2read.exe can handle less than or equal to 25 channels and less than or equal to 500,000 samples in every channel.
Exceedance may result in a significant error.
It is recommendable to split the input data file if too long, for example, into several files of 1 hour or 30 minutes data.

Ref:
500,000 samples correspond to 1.38 hours for 100 Hz sampling,
1.11 hours for 125 Hz sampling,
41.6 minutes for 200 Hz sampling,
33.3 minutes for 250 Hz sampling
16.6 minutes for 500 Hz sampling.

```
yokoi@eoan-ermine:~/Desktop/CCA2020$ sh seg2read.sh
./cca_wkf/prm/seg2read.prm
   4.99999989E-03 mkine
./cca_wkf/data/sg2_files/sg2file.lst
F2LNN001.sg2

./cca_wkf/data/sg2_files/F2LNN001.sg2
./cca_wkf/data/multiplexed_files/F2LNN001.dat
                  :32bit fixed point.
nch= 12 dt= 0.008 nn=  225000
       6  0.0080      0.5000E-02  225000  mkine
LCCM,Field,r=2m,L22D,No_Rs,No_A_amp,D_amp=X1,F2LNN0,Mar.15,2012
     1 files have been converted.
Normal End.
```



Input file is ./cca_wkf/data/sg2_files/F2LNN001.sg2
Multiplexed output is ./cca_wkf/data/multiplexed_files/F2LNN001.dat & another.
Log file is stored in 'cca_wkf/log/seg2read.log'



Note: The converted files are already multiplexed and stored in the subfolder "./cca_wkf/data/multiplexed_files" with the extension ".dat". For these files, the next step is resampling.

1. Multiplexing & Resampling

1.1 Format conversion & Multiplexing

1.1.2 *win* format for LS8800

This is the example of individually recorded data using a tri-axial sensor at each observation point.

Format conversion & Multiplexing must be done by the users prior to the analysis for the case of individual recording at each site.

Here *win* format is explained as an example. The *win* format data files are created, e.g., LS8800 of Hakusan Kogyo.

The multiplexed files must be written in a format that is readable in the next step: resampling.

As it is impossible to cover all existing formats in the world, it is strongly recommended to make their own program for format conversion.

Format conversion is conducted using
sh lstocdm2.sh
in the folder "CCA2021".

Note: Usage of 4 seismographs in a site is assumed.

Preparation:
1) Edit the parameter file "prm_maker.prm"

```
Sitename_    : site name (a9)
 3  3        : numbers of obs_ponts and channels
 10          : duration of each connected file in min.(integer)
17091511.45  : first file name (yymmddhh.mm)
 20          : number of output connected files (integer)
```

2) Copy the data files (binary) of LS8800 into the subfolders of
"spacwkf/data/ls_files" as follows:
"no1"← files from seismograph No.1
"no2"← files from seismograph No.2
"no3"← files from seismograph No.3
"no4"← files from seismograph No.4

lstocdm2.sh

```
#!/bin/sh -x
./bin/prm_maker.exe | tee cca_wkf/log/prm_maker.log
./bin/lstocdm2.exe | tee cca_wkf/log/lstocdm2.log
cd ./cca_wkf/data/cdm_files
rm *.cdm 2>/dev/null
cd ../../..
mv ./cca_wkf/data/ls_files/Combined_Data/*.cdm ./cca_wkf/data/cdm_files
```

Execution:

./bin/prm_maker.exe

➔ "lstocdm2.prm" is created in "cca_wkf/prm".

./bin/lstocdm2.exe

➔ All converted and separated files are stored in "./cca_wkf/data/ls_files/Combined_Data".

All converted and separated files are automatically stored in "cca_wkf/data/ls_files/Combined_Data".

Then, the subfolder "./cca_wkf/data/cdm_files" is cleaned.

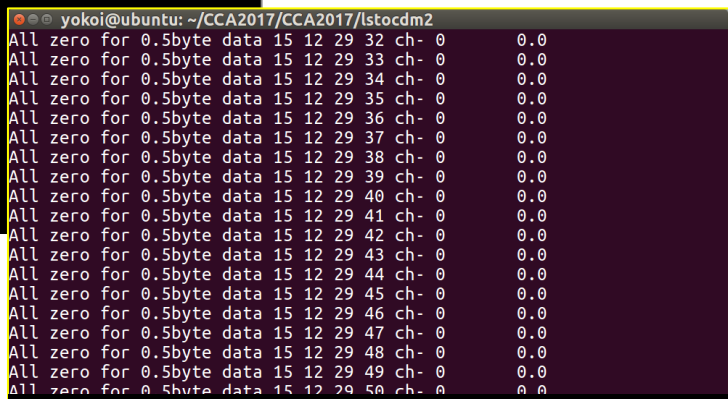
Finally, by the command "mv" at the last line all of the cdm files are moved from "./cca_wkf/data/ls_files/Combined_Data" to "./cca_wkf/data/cdm_files".

Example:

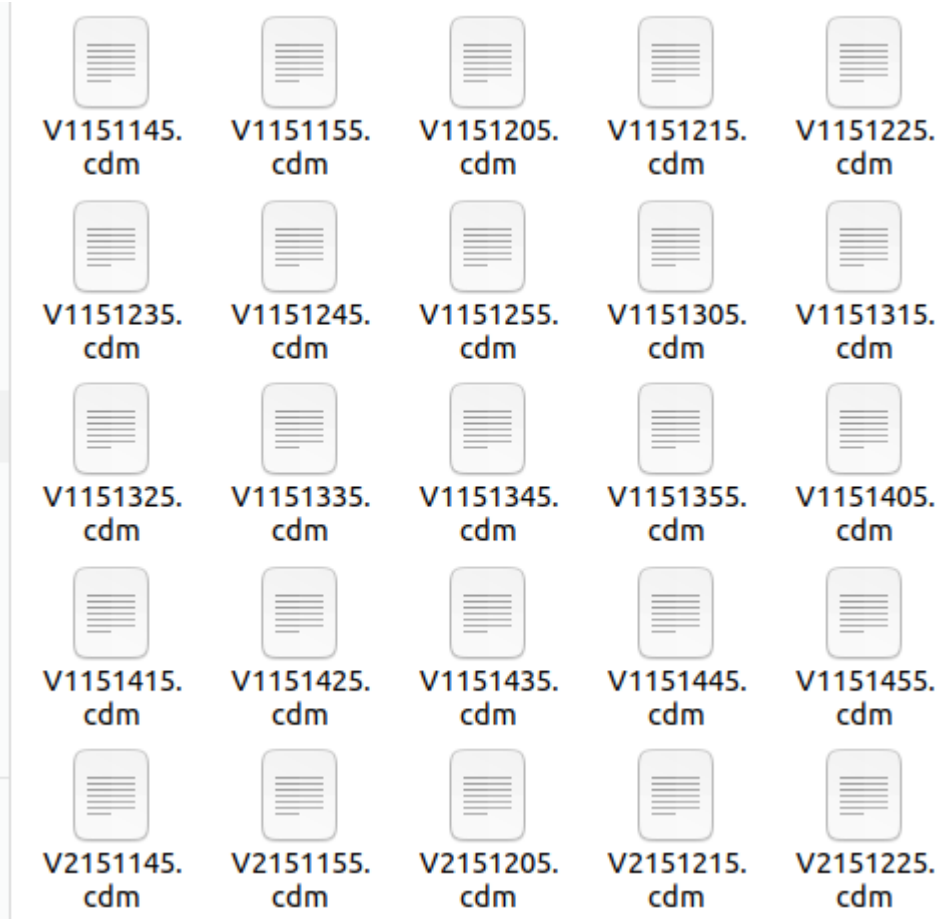
lstocdm2.prm: automatically created by lstocdm2.sh

```
 3                      : case numbers
                        : This blank line is necessary
City_Hallno1            : Data Folder Name(a12)
  794.  792.  798.      : #1 T44002 Sensitivity (V/M/s)
NS EW V                 : channel labels
0.15646E-06             : voltage per Least Significant Bit (V/LSB)
V1    10                : (a2) channel identifier of name; duration(min); of output file
 3 1 2 1                : channel pivoting, channel selection (0=all, 1=1st channel only)
17091511.45  20         : 1st File name(a11),number of output files
                        : This blank line is necessary
City_Hallno2            : Data Folder Name(a12)
  798.  798.  796.      : #2 T44003 Sensitivity (V/M/s)
NS EW V                 : channel labels
0.15646E-06             : voltage per Least Significant Bit (V/LSB)
V2    10                : (a2) channel identifier of name; duration(min); of output file
 3 1 2 1                : channel pivoting, channel selection (0=all, 1=1st channel only)
17091511.45  20         : 1st File name(a11),number of output files
                        : This blank line is necessary
City_Hallno3            : Data Folder Name(a12)
  796.  794.  798.      : #3 T44004 Sensitivity (V/M/s)
NS EW V                 : channel labels
0.15646E-06             : voltage per Least Significant Bit (V/LSB)
V3    10                : (a2) channel identifier of name; duration(min); of output file
 3 1 2 1                : channel pivoting, channel selection (0=all, 1=1st channel only)
17091511.45  20         : 1st File name(a11),number of output files

Constants
  794.  792.  798.      : #1 T44002 Sensitivity (V/M/s)
  798.  798.  796.      : #2 T44003 Sensitivity (V/M/s)
  796.  794.  798.      : #3 T44004 Sensitivity (V/M/s)

pivoting
 1 2 3 1                : NS output
 2 3 1 1                : EW output
 3 1 2 1                :  V output
```

```
yokoi@eoan-ermine:~/Desktop/CCA2020$ sh lstocdm2.sh
tee: cca_cwkf/log/prm_maker.log: No such file or directory
tee: cca_cwkf/log/lstocdm2.log: No such file or directory
 Parameter file: ./cca_wkf/prm/lstocdm2.prm

Name of the site: City_Hal
No. of station: no1
 Sensitivity(M/s):    794.000000       792.000000       798.000000
 Components: NSEWV
 voltage per Least Significant Bit   1.56460004E-07 (V/LSB)
   1.97052905E-05   1.97550507E-05   1.96065157E-05
Output 1 channel  V
Start from: 17091511.45 ,    10 file
 Output file: ./cca_wkf/data/ls_files/Combined_Data/V1151145.cdm

 Input file: ./cca_wkf/data/ls_files/no1/17091511.45

 Output file: ./cca_wkf/data/ls_files/Combined_Data/V1151155.cdm

 Input file: ./cca_wkf/data/ls_files/no1/17091511.55

 Output file: ./cca_wkf/data/ls_files/Combined_Data/V1151205.cdm

 Input file: ./cca_wkf/data/ls_files/no1/17091512.05
```



This error message means the clipping of data. Check the time and eliminate the corresponding part.



File name (E1151145.cdm) includes the following information:

1st letter: component

2nd letter: numbering of seismograph (=numbering of station)

3rd & 4th: Date in (i2)

5th & 6th: hour in (i2)

7th & 8th; minutes in (i2)

Example of a converted file

Open | V1151145.cdm ~/Desktop/SPAC2020/spacwkf/data/cdm_files | Save

```
 1 File=17091511.45_17091511.54
 2                          UNIT=mkine
 3                                V
 4         0 00:00:00.000  -0.2722953E+00
 5         1 00:00:00.010  -0.2532378E+00
 6         2 00:00:00.020  -0.2709032E+00
 7         3 00:00:00.030  -0.3568974E+00
 8         4 00:00:00.040  -0.4600865E+00
 9         5 00:00:00.050  -0.4690271E+00
10         6 00:00:00.060  -0.4567338E+00
11         7 00:00:00.070  -0.6430937E+00
12         8 00:00:00.080  -0.7474200E+00
```

3 lines for header

Data lines:

Numbering, (A8)

time, (A13)

data (e16.7)

The created single channel file by this format conversion program will be read in the next step using “multipx6.exe” as follows.

“multipx6.for”

```
     character cline(3)*80,cdum*13,cdummy*8  ! Declare three
                                              ! character strings
……..
      do i=1,3
        read(1,'(a80)')cline(i)   ! Read 3 lines header
      enddo
      do i=1,nst0-1
       read(1,*)cdummy            ! Skip first nst0 sec data
      enddo
      do i=1,ndur0
c read input data
        read(1,*,end=10)cdummy,cdum,xdum
        x(i,j)=xdum/scale
       enddo
   10  ndur0=i-1                   ! Adjust number of samples
……..
```


Character strings cline(3) are not used further. Neither cdummy nor cdum.

Preparation for Multiplexing

- In the next step, *multipx6.sh* is used with the parameter file "*multipx6.prm*".
- For automatic editing of multipx6.prm, *multi_pre.sh* is prepared.
- Edit the following "*multi_pre.prm*" in "cca_wkf/prm" and run *"sh multi_pre.sh*".

```
CityHall 3P-Array l=35m 2017/09/11:45-2H20M       :comment(A50)
 10          : duration of each connected file in min.(integer)
V1151145     : first file name (yymmddhh.mm)
cdm          : extention (A3)
1 2 3 0      : station pivot (0=4th_ch_not_used)
 20          : number of output connected files(integer)
```

- 1st line: Comment but later used as the title of all graphs showing the results of analysis.
- 2nd line: Duration of connected files same as the 3rd line of "prm_maker.prm".
- 3rd line: The eariest file name for the 1st position (A8)
- 4th line: Extension of the name of files in the folder "cdm_files", i.e., cdm.
- 5th line: Station pivoting list.
- 6th line: Number of connected files same as the 5th line of "prm_maker.prm".

Execution

```
yokoi@eoan-ermine:~/Desktop/CCA2020$ sh multi_pre.sh
  Working Folder=./cca_wkf/prm/
  Default Settings:
    dt      = 0.0
    np      =   3
    tst     = 0.0
    file_out=CC**.dat

V1151145 V2151145 V3151145
V1151155 V2151155 V3151155
V1151205 V2151205 V3151205
V1151215 V2151215 V3151215
V1151225 V2151225 V3151225
V1151235 V2151235 V3151235
```

Automatically created multipx6.prm which will be used in the next step.

```
 1    1                        : Number of cases
 2                             : This blank is necessary
 3   3 0.010                   : Number of obs points,dt
 4          0.0       600.0    : tst,tdur(sec)
 5 1.e0  mkine                 : scale(input data is divided by this scale)
 6  0 3 0.1 1.0 1.5            : nfilter(=1:apply), ncharacter(=2:lowpass,=3:bandpass),fl,fh,fs
 7   4           .cdm          : nattach, cattach
 8  2   CC                     : n_out(A12),cout ("**.dat" is attached)
 9 CityHall 3P-Array l=35m 2017/09/11:45-2H20M       : comment(A50)
10    20            8          : number of measurement in the same array configuration,n_character
11 V1151145 V2151145 V3151145
12 V1151155 V2151155 V3151155
13 V1151205 V2151205 V3151205
14 V1151215 V2151215 V3151215
15 V1151225 V2151225 V3151225
16 V1151235 V2151235 V3151235
17 V1151245 V2151245 V3151245
```

Edit it in an appropriate way if necessary.

multipx6.prm

```
 1                          : Number of cases
                            : This blank is necessary
  3 0.010                   : Number of obs points,dt
        0.0       600.0     : tst,tdur(sec)
1.e0  mkine                 : scale(input data is divided by this scale)
 0 3 0.1 1.0 1.5            : nfilter(=1:apply), ncharacter(=2:lowpass,=3:bandpass),fl,fh,f
  4           .cdm          : nattach, cattach
 2   CC                     : n_out(A12),cout ("**.dat" is attached)
CityHall 3P-Array l=35m 2017/09/11:45-2H20M       : comment(A50)
   20            8          : number of measurement in the same array configuration,n_chara
V1151145 V2151145 V3151145                  ←—— 1st
…                                                  Input filelist
```

Input file names : V1151145.cdm

consist of the character string ‘V1151145’ of 8 characters plus another character string ‘.cdm’ of 4 characters. These character strings and their number of characters are indicated in the 7th line for the latter and the 11th line and below for the former. Program ‘multipx6.exe’ automatically combines them and read the data from the files.

Don’t leave a blank line at the end of input-file-list

multipx6.prm

Output file name: CC01.dat for the 1st measurement. ’01’ shows the numbering of measurement.

…

CH04.dat for the 4th measurement. ’04’ shows the numbering of measurement.

These output file names consist of the character string ‘CH’ of 2 characters as indicated in the 8th line. The following two integers show the numbering of measurement. ‘.dat’ is attached to all automatically.

The data from tst to tst+tdur are processed in every files.

Values read from the input files are divided by the scale factor given in the 5th line. This value must be selected to make the unit of data in the output file is ‘mkine’, i.e., 1.0E-5 M/sec for ground velocity. For ground acceleration ‘gal’, i.e., 1.0E-2 M/sec^2 should be used. Otherwise the amplitudes of the data will be erroneously shown in the output figures.

Execution:

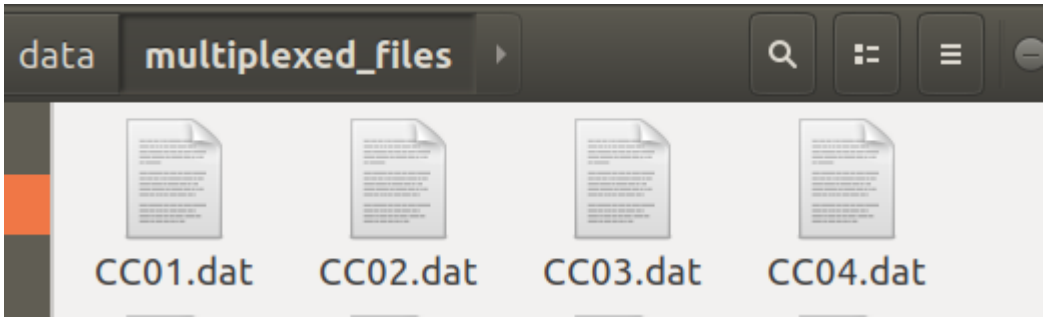
sh multipx6.sh

```
yokoi@eoan-ermine:~/Desktop/CCA2020$ sh multipx6.sh
  Working Folder=./cca_wkf/prm/
  Nch=           3  dt=   9.99999978E-03
  Scale Factor=   1.00000000
  Does this scale convert the unit of data mkine ?
  Please change the value if not.
           0           3  0.100000001       1.00000000       1.50000000
 CityHall 3P-Array l=35m 2017/09/11:45-2H20M
V1151145 V2151145 V3151145
V1151155 V2151155 V3151155
V1151205 V2151205 V3151205
V1151215 V2151215 V3151215
V1151225 V2151225 V3151225
V1151235 V2151235 V3151235
V1151245 V2151245 V3151245
```

multipx6.sh

```
#! /bin/sh -x
./bin/multipx6.exe | tee
cca_wkf/log/multipx6.log
cd cca_wkf/data/multiplexed_files
ls *.dat > mltfile.lst
cd ../../..
./bin/mk_title_pre.exe
```

10 multiplexed data files are created in “./cca_wkf/data/multiplexed_files”



Format of multiplexed file:

Open ▾ CC01.dat ~/CCA2017/CCA2019/cca_wkf/data/multiplexed_files Save

```
1        4  0.0100      0.1000E+01  180000  mkine
2 CCA Array,TESTCASE,  Dec/05/17:00-15H,2017
3     0.000000  0.0000000E+00  0.0000000E+00 -0.0000000E+00  0.0000000E+00
4     0.010000  0.8910510E-02  0.6514205E-02 -0.1340472E-01  0.7184593E-01
5     0.020000  0.2662862E-01  0.1362474E-01 -0.6547699E-02  0.1320289E+00
6     0.030000  0.6811393E-01  0.2614746E-01  0.4290193E-01  0.1220999E+00
7     0.040000  0.1228383E+00  0.4015458E-01  0.1152715E+00  0.5129088E-01
8     0.050000  0.1217078E+00  0.1120007E+00  0.1573336E+00 -0.6455636E-02
```

ch_pivot.exe

This program is provided for changing the order of channels in a group of already multiplexed files, for example, those copied from the data processing of SPAC method.

Edit the parameter file “ch_pivot.prm” in “./cca_wkf/prm”

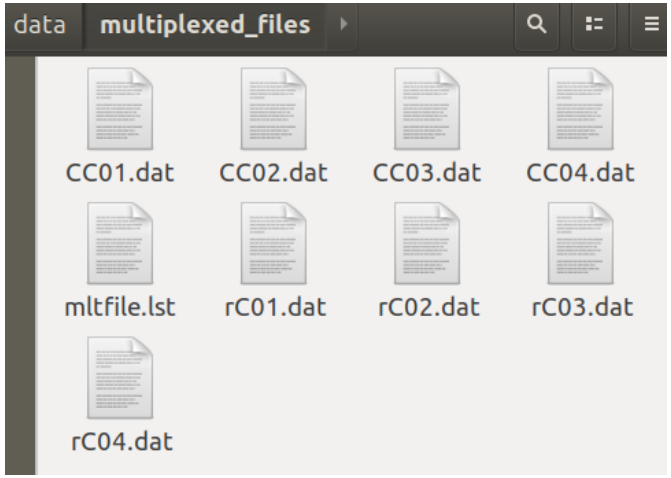
ch_pivot.prm

```
4           : channel number➔ How many channels the file has
4 1 2 3     : pivot list
```

Execute
sh ch_pivot.sh

Then, all files listed in “./cca_wkf/multiplexed_files/mltfile.lst” are processed.

The pivoted files’s name has “r” at the top.



ch_pivot.sh

```
#!/bin/sh
cd ./cca_wkf/data/multiplexed_files
ls *.dat > mltfile.lst
cd ../../..
./bin/ch_pivot.exe
```

1.2 Resampling & Screening

Shell Script used

sh resamplec.sh

Program and parameter file used:

./bin/resamplec.exe +./cca_wkf/prm/resamplec.prm

Terminology

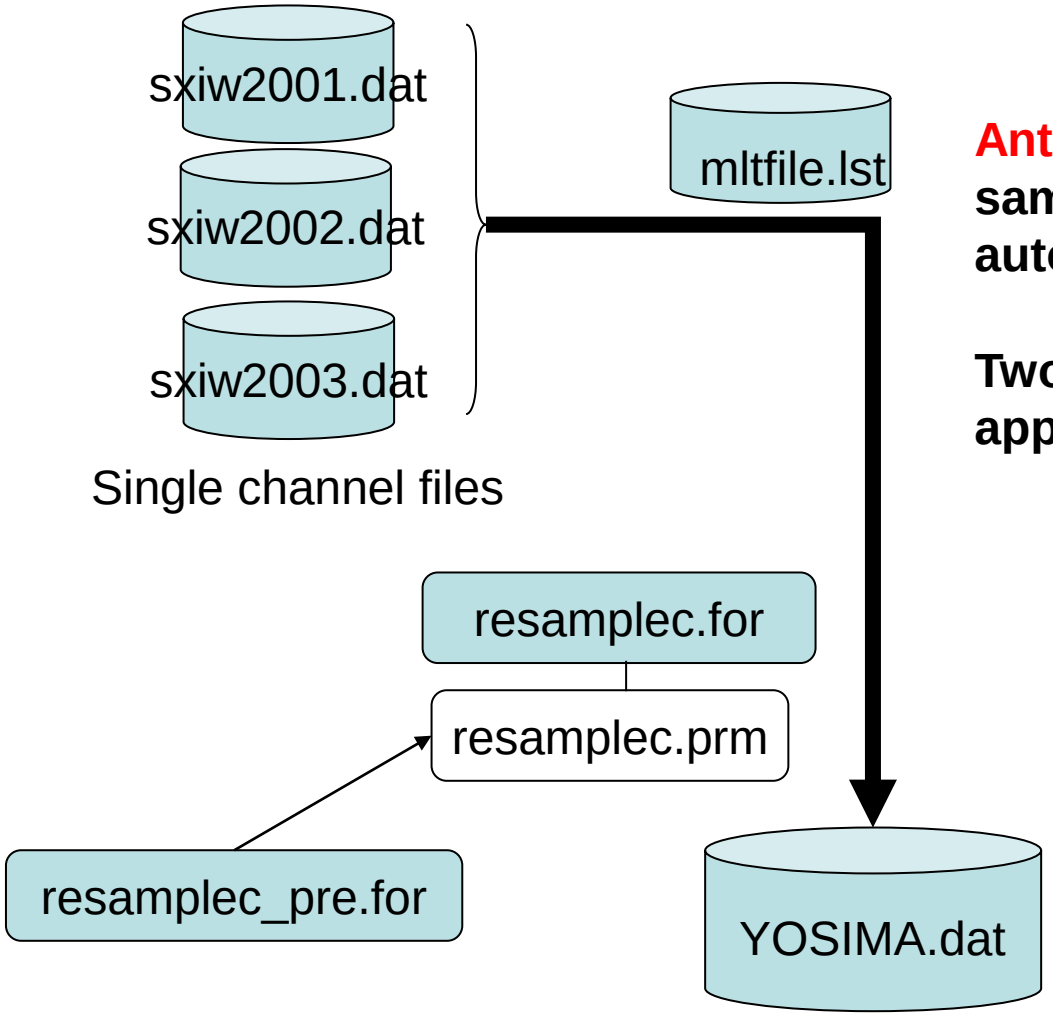
Re-sampling:

It can be done to thin the data out in order to reduce the size of data files and the load to PC for processing. This can cause the aliazing effect. Then, it Is necessary to apply the digital anti-alias filter that has high cut characteristics before thinning out.

Note: Modify the file “./cca_wkf/prm/graph_title.txt”, if you have not started the processing from “*sh seg2read.sh*”. This means that your original data files are not seg2 format.

The contents of the file “./cca_wkf/prm/graph_title.txt” is used for the title of various graphs produced in further processing. It is recommendable to give an appropriate title to the figures to prevent potential confusion.

Multi-channel data files from the same array configuration



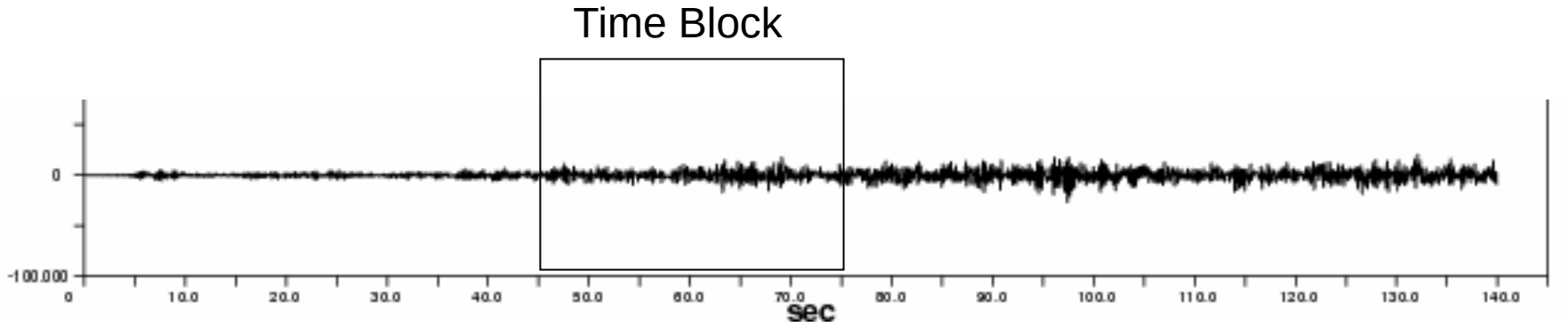
Anti-alias filter for re-sampling is applied automatically.

Two steps for screening are applied.

Resampled & screened multi-channel & multi-measurement data file

Screening: Step-1

Parameter: ajudge



If the maximum amplitude in a time block exceeds the product of "ajudge" to RMS amplitude of the same time block, this time block is not used in analysis.
This is a countermeasure against impulsive noise due to traffic, i. e., vehicles passing near by seismometers.
The bigger value of "ajudge" means looser screening. The smaller value means fewer available time blocks.

Screening: Step-2

Parameter: a_sgm

If the RMS amplitude in a time block deviates more than a_sgm X the standard deviation from the average, this time block is not used in analysis, where the average and the standard deviation are calculated over the all time blocks that survived in the screening step-1.

This is a countermeasure against outliers.
The bigger value of “a_sgm” means looser screening. The smaller value means fewer available time blocks.

resamplc.sh:

```
#! /bin/sh –x
cd ./cca_wkf/data/multiplexed_files
ls *.dat > mltfile.lst
cd ../../..
./bin/resamplec.exe | tee cca_wkf/log/resamplec.log
```

Example of resamplec.prm:

```
 6 0.008  2   0.0  0  2.0   :nch,dt,nchannel,dt,nskip,ph0,ncenter,radius
5.0 3.0                     :ajudge,a_sgm
0.0 1800.0                  :tst,tdur
F2LNN1.dat                  :output file name
1024                        :number of data in one time block after resampling
```

where
nskip: skip number for resampling (1: no resampling, 2: resample at every two samples)
ph0: azimuth from the center(1ch) to 2ch
ncenter: 0 (no sensor at the center) or 1 (1ch at the center) , cannot be bigger than 2
radius: radius of circular array (m)
tst: start time of analysis (sec)
tdur: duration of time window for analysis (sec)

Example of Output (resampled) file format

F2LNN1.dat

1st line:file parameters

1st block

2nd block

```
 6  120 1024 0    0.00    2.00 (i8,f16.4, 6e15.7) mkine
       1          8.1920 -0.2134265E+01 -0.1570028E+01 -0.1464243E+01 -0.1183819E+01 -0.1944440E+01 -0.2460508E+01
       2          8.2080 -0.8561512E+00 -0.1420853E+00  0.5698163E+00 -0.7201288E+00 -0.8229986E+00 -0.1031229E+01
       3          8.2240 -0.1268116E+01 -0.1240067E+01 -0.2134730E+00 -0.2036263E+00 -0.3363592E+00 -0.7179448E+00
       4          8.2400 -0.1921562E+01 -0.1946060E+01 -0.1426458E+01  0.5525613E+00  0.5879311E-01 -0.1037454E+01
       ....
    1024         24.5600  0.2423161E+01  0.9976824E+00  0.2375970E+01  0.6267622E+01  0.5800716E+01  0.4042787E+01
       1         16.3840  0.5473618E+01  0.3885463E+01  0.6374265E+01  0.4033590E+01 -0.3046133E+01  0.5571502E+00
       ….
```

The file include mmblk=120 time blocks of nch=6 channel data. Each time block is composed of nblk=1024 data.
Each line corresponds to a time step. The format used to store each line is cform3='(i8,f16.4, 7e15.7)'.
Unit of data is 'mkine' (=1.0e-3 cm/sec)
These parameters are stored in the 1st line.

As all of the data are delimited by space, this file can be read using free format.

Note: For the data of huddle test, it is necessary to change the value of the 6th parameter of the first line of the resamples files to "0.0" .

1. Multiplexing & Resampling
1.3 Plot Waveform

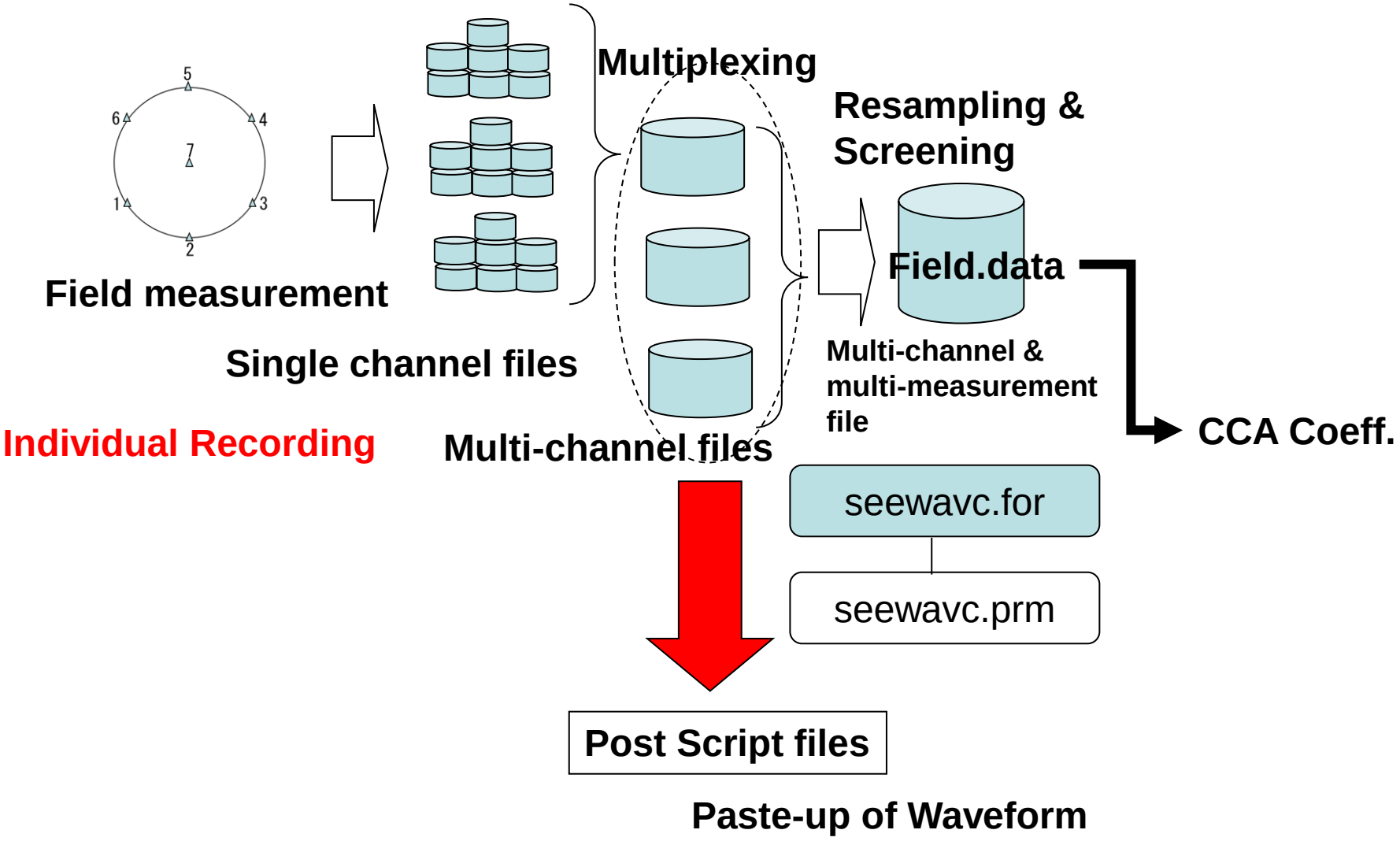
Shell Script used:

sh seewavc.sh

Program and parameter file used:

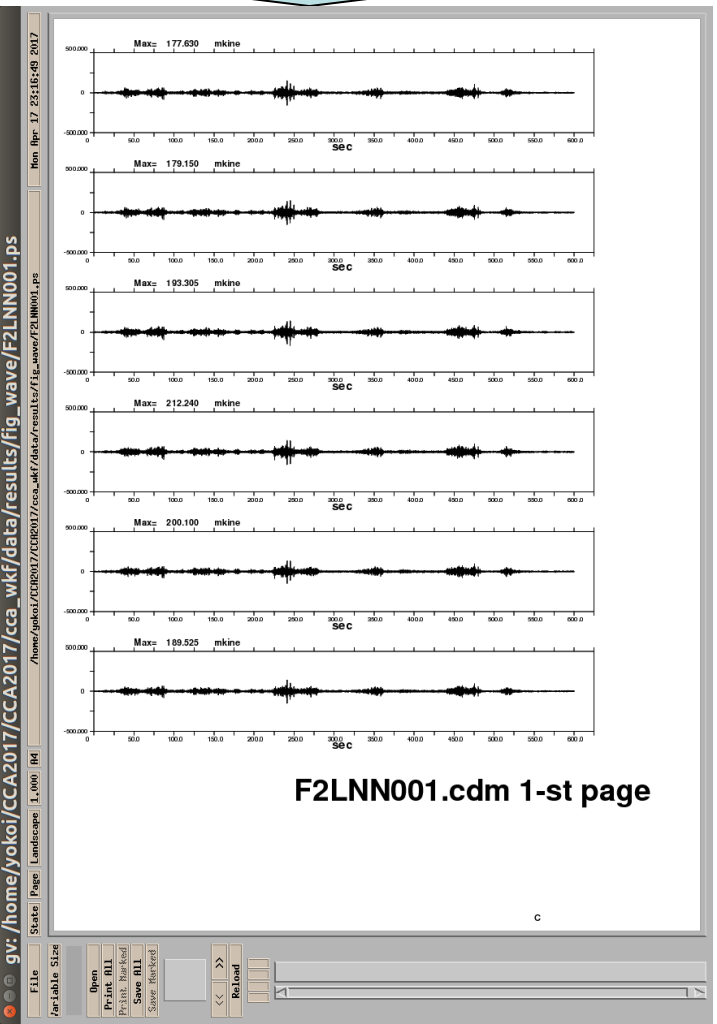
./bin/seewavc.exe +./cca_wkf/prm/seewavc.prm

Flow of the data processing for the conventional CCA



Multi-channel file: F2LNN001.dat

```
 6   0.0080        0.5000E-02   225000   mkine
LCCM,Field,r=2m,L22D,No_Rs,No_A_amp,D_amp=X1,F2LNN
     0.000000   0.4800000E+00   0.3450000E+00   0.6100000E+00   0.5650000E+00   0.3250000E+00   0.3650000E+00
     0.008000   0.9000000E-01   0.2100000E+00  -0.7250000E+00  -0.2000000E-01   0.8600000E+00  -0.1900000E+00
     0.016000   0.5500000E+00   0.1035000E+01  -0.9650000E+00   0.1250000E+00   0.1485000E+01   0.4300000E+00
     ...
```



F2LNN001.cdm 1-st page

seewavc.for

seewavc.prm

Figure in Multi-page Post Script file. Post Script file can be opened, for example, by "gv &".

seewavc.sh:

```
#! /bin/sh -x
cd cca_wkf/data/multiplexed_files
ls *.dat > mltfile.lst
cd ../../..
./bin/seewavc.exe | tee cca_wkf/log/seewavc.log
```

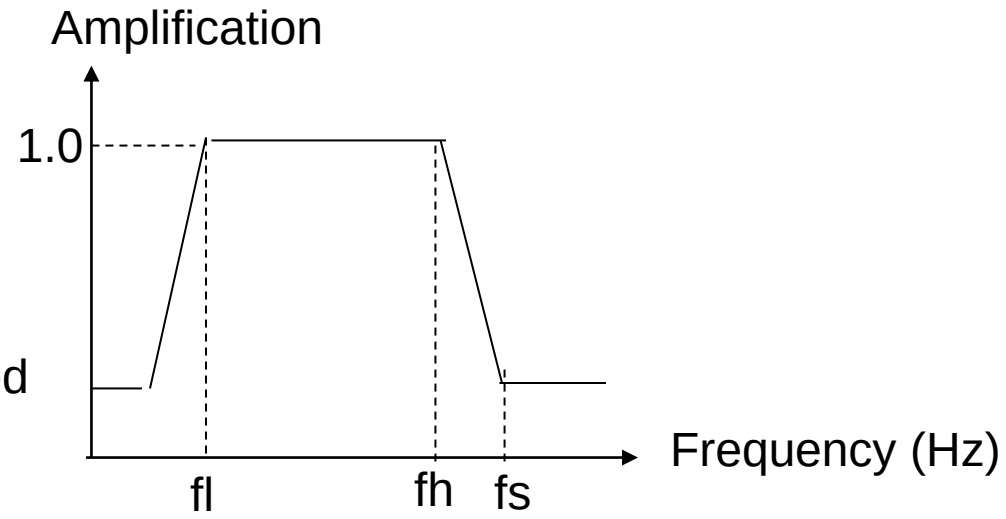
seewavc.prm

dtl denotes the time duration that corresponds to 1cm along the time axis.
In one page, 28*dtl/dt time step can be plotted. If the file has more, new pages are automatically added as much as necessary and multi-page PS file is created.

```
6                                   :nch
0 0.1    1.0 1.5 3                  :nfilter,fl,fh,fs,nchara(=2:lowpass, =3:bandpass)
  25.                               :dtl(sec/cm),25,50==>10,20 min/page
```

nbandp=0: no effect
nbandp=1: bandpass filter is applied

Band Pass Filter



This BPF does not affect to the data files.

Execution

Open ▾ | seewavc.prm
~/CCA2017/CCA2019/cca_wkf/prm

```
1  6                          :nch
2  0 0.1    1.0 1.5 3         :nfilter,fl,fh,fs,nchara(=2:lc
3   25                        :dtl(sec/cm),25,50==>10,20 mir
4
```

multiplexed_files

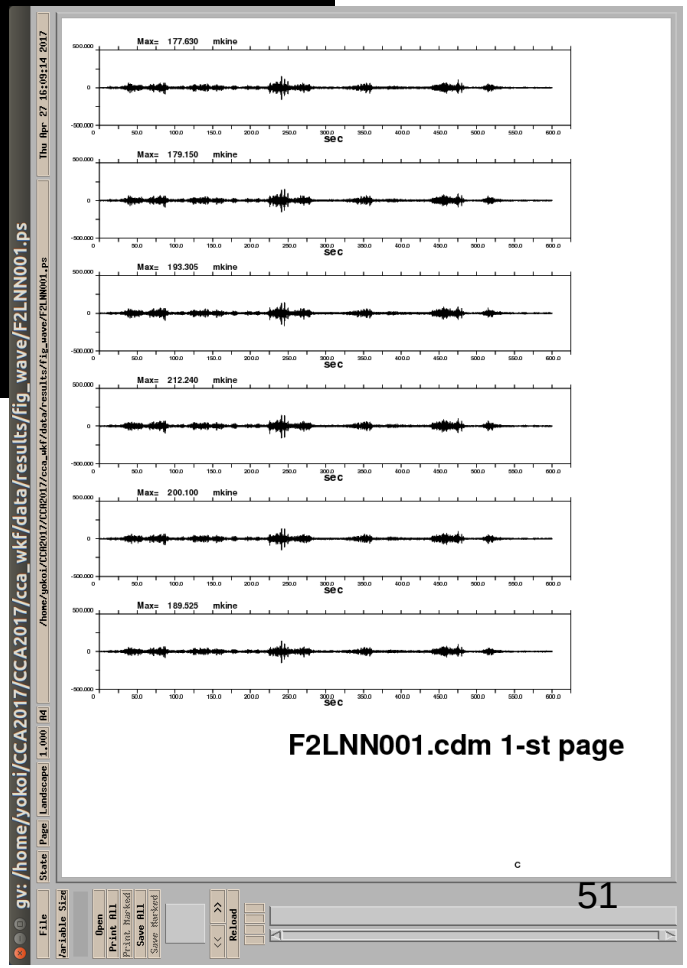
multiplexed_files

F2LNN001.dat

HDLNN001.dat

mltfile.lst

```
yokoi@eoan-ermine:~/Desktop/CCA2020$ sh seewavc.sh
F2LNN001.dat
HDLNN001.dat
  2 measurement
./cca_wkf/data/results/fig_wave/F2LNN001.ps
F2LNN001.dat 1-st page
F2LNN001.dat 2-nd page
F2LNN001.dat 3-rd page
./cca_wkf/data/results/fig_wave/HDLNN001.ps
HDLNN001.dat 1-st page
```



Waveforms are plotted in PostScript files.
Log file is stored in “./cca_wkf/log/seewavc.log

1. Multiplexing & Resampling
1.4 Check the selected time blocks

Shell Script used

sh seeblkc.sh

Program and parameter file used:

./bin/seeblkc.for +./cca_wkf/prm/seeblkc.prm

seeblkc.sh

```
#! /bin/sh -x
cd ./cca_wkf/data/resampled_files
ls *.dat > rsmfile.lst
cd ../../..
./bin/seeblkc.exe | tee cca_wkf/log/seeblkc.log
```

Parameter file:
./cca_wkf/prm/seeblkc.prm

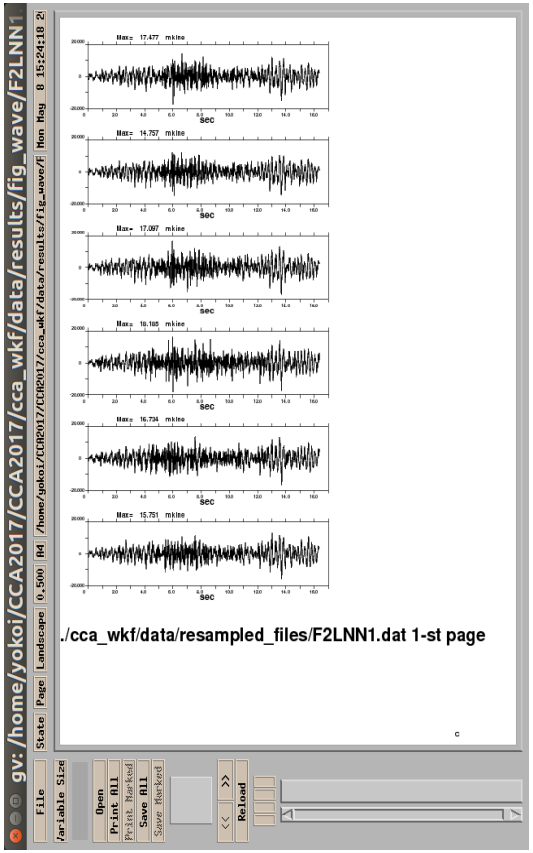
```
 0 0.1 1.0 1.5 3    :nfilter,fl,fh,fs,nchara(=2:lowpass, =3:bandpass)
```

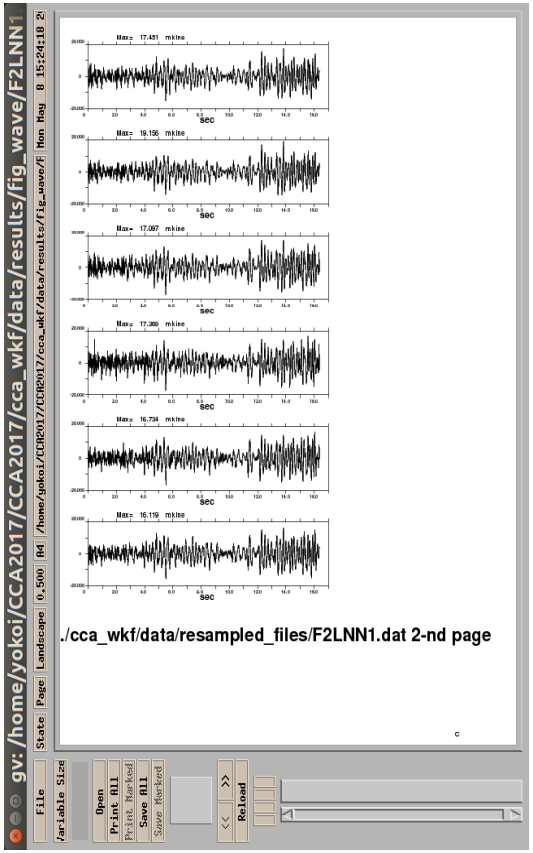
Execution

```
yokoi@eoan-ermine:~/Desktop/CCA2020$ sh seeblkc.sh
 ./cca_wkf/data/resampled_files/rsmfile.lst

./cca_wkf/prm/seeblkc.prm
./cca_wkf/data/resampled_files/F2LNN1.dat
 6  120 1024 0    0.00    2.00 (i8,f16.4, 6e15.7) mkine
 ./cca_wkf/data/results/fig_wave/F2LNN1.ps

./cca_wkf/data/resampled_files/HDLNN1.dat
 6   35 1024 0    0.00    2.00 (i8,f16.4, 6e15.7) mkine
 ./cca_wkf/data/results/fig_wave/HDLNN1.ps
```

Multi-page postscript files are created in
./cca_wkf/data/results/fig_wave
Log file is stored in ./cca_wkf/log/seeblkc.log

PostScript File





Multi-page postscript files are created in ./cca_wkf/data/results/fig_wave

2. Estimating Dispersion Curve
2.1 Calculation of CCA coefficient

Shell Script used

sh pwrcrs3.sh

Program and parameter file used:

./bin/pwrcrs3.exe +./ccawkf/prm/pwrcrs3.prm

Terminology

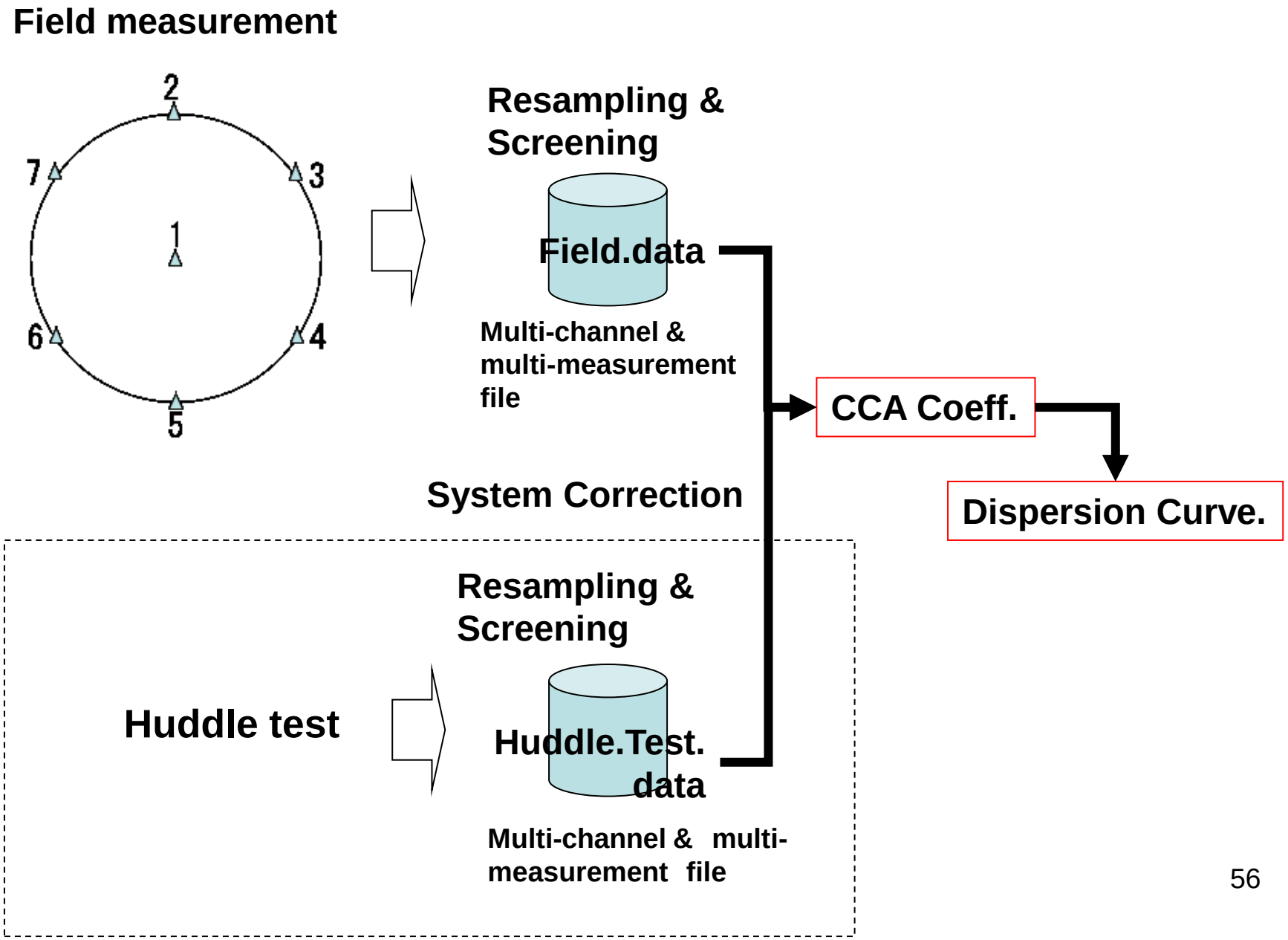
Huddle test:

Common input motion recording to determine the difference of the system characteristics among the recording system and/or channels.
The seismometers used in field measurement are put close each other like a huddle and simultaneous recording is conducted.

System correction:

The difference of the characteristics among the recording system can be corrected using the data obtained by huddle tests (Use CCA2019 if necessary).

Flow of the data processing for CCA



pwrcrs3.sh

```
#! /bin/sh
./bin/pwrcrs3.exe 2>&1 | tee cca_wkf/log/pwrcrs3.log
./bin/mk_title_post.exe
```

Example of pwrcrs3.prm:

```
1.0 25.0 0.016 0.5 5.0 1 :fmin,fmax,dt,bw,smthf,nhide
F2LNN1.dat     1 1       :Field data File name(A12), n_coh,n_pow
0   1                    :n_cor_center, n_mod
```

where

fmin, fmax : minimum and maximum frequencies for analysis

bw : band width of Parzen window

n_huddle : flag for system correction using huddle test data (0:no, 1:yes)

smthf : smoothing parameter

nhide :

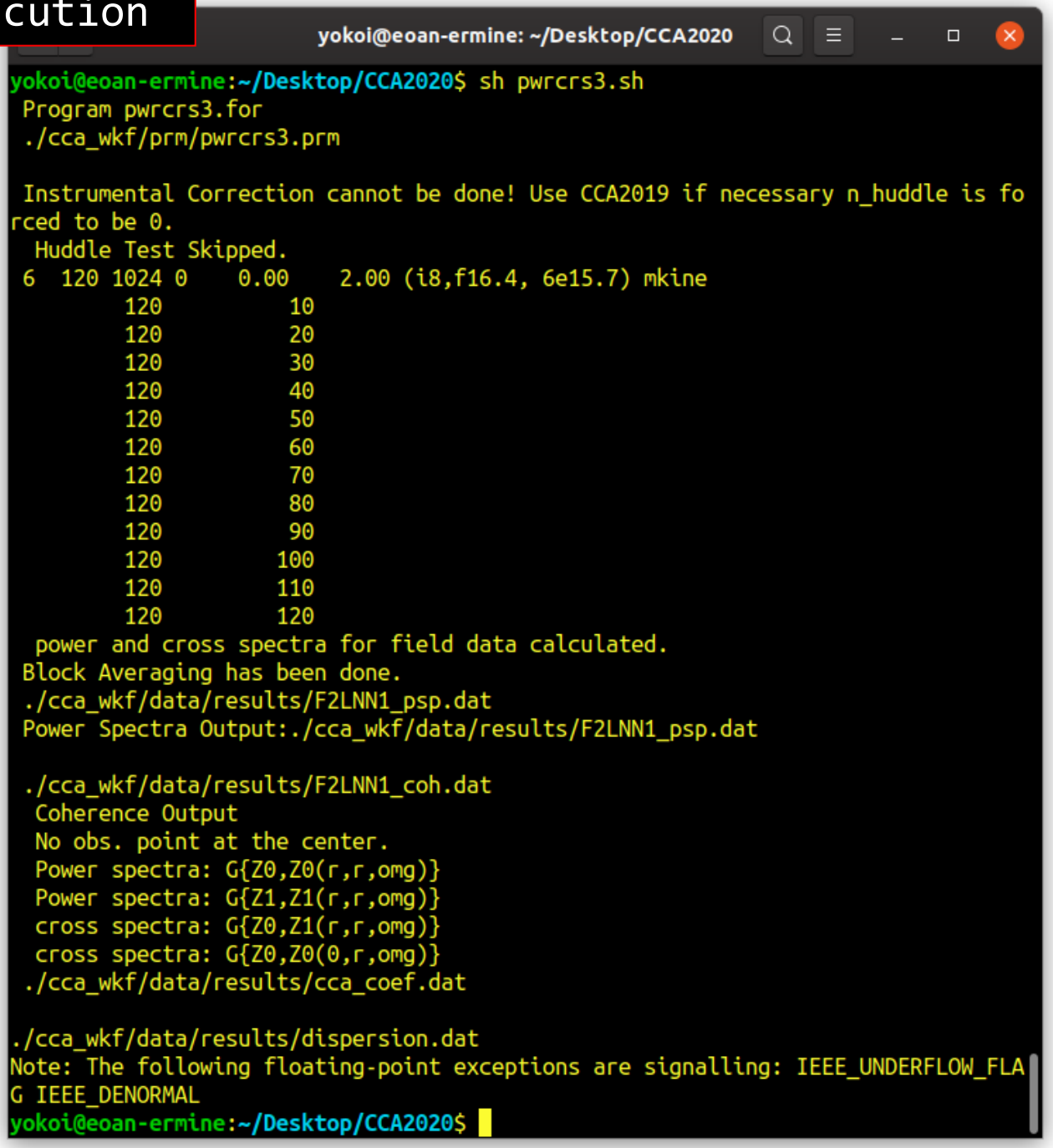
n_coh,n_pow : coherence & power spectra output flag : (0:no output , 1:yes)

n_cor_center : flag for correction using sensor at the center. (0:no output , 1:yes)

n_mod : =0 no , =1 yes for plotting wavelength/3 vs Vs

Note: ‘mk_title_post.exe’ copies the 1st line of ‘graph_title.txt’ to the 4th line of ‘vel_model_plt.prm’.

Example of Execution



CCA coefficient

calculated in the frequency domain

n_huddle=0

$$s_{CCA'}(r,\omega) \approx \frac{\sum_{m=1}^{M}\sum_{m'=1}^{M} E\left[C_{m,m'}(r,\omega)\right]}{\sum_{m=1}^{M}\sum_{m'=1}^{M} E\left[C_{m,m'}(r,\omega)\right]\exp\left\{-i\left(\theta_m-\theta_{m'}\right)\right\}} \approx \frac{J_0^2\left(r\omega/c\right)}{J_1^2\left(r\omega/c\right)}$$

n_huddle=1

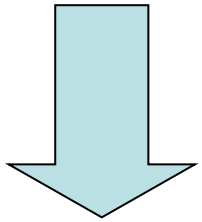
$$s_{CCA}(\omega) = \frac{\sum_{i=1}^{M}\sum_{k=1}^{M} R_{ik}(\omega)}{\sum_{i=1}^{M}\sum_{k=1}^{M} R_{ik}(\omega)exp\left\{-j\left(\theta_i-\theta_k\right)\right\}}$$

$$R_{ik}(\omega) = \frac{C_{00}^{obs}(\omega)E\left[C_{ik}^{obs}(\omega)\right]\overline{Cor_{ik}^{huddle}(\omega)}}{\sqrt{E\left[C_{ii}^{obs}(\omega)\right]E\left[C_{kk}^{obs}(\omega)\right]}}$$

$$\overline{Cor_{ik}^{huddle}(f)} = \exp\left\{ jE\left[Arg\left(\frac{\sqrt{C_{ii}^{huddle}(f)\cdot C_{kk}^{huddle}(f)}}{C_{ik}^{huddle}(f)} \right)\right]\right\}$$

KR derived from CCA coefficient

$$s(r,\omega)=\frac{J_0^2(r\omega/c)}{J_1^2(r\omega/c)}$$



Long wavelength apploximation
(Small value of *kr*) (Cho et al,2006)

$$c(\omega)=\frac{r\omega}{2}\sqrt{s(r,\omega)+2}$$

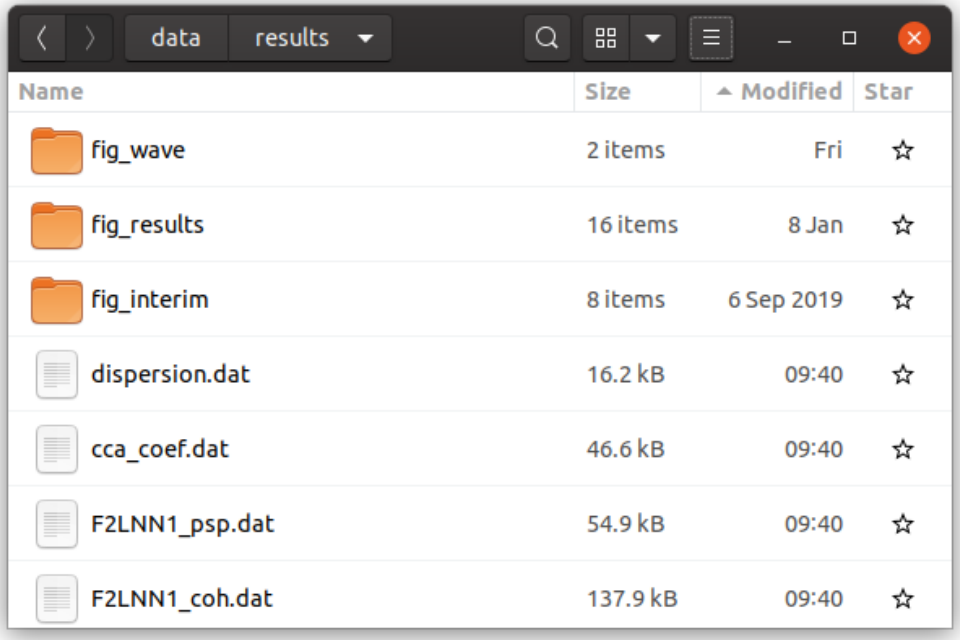
Improved approximation:Yokoi(2012)

$$\mathrm{c}(\omega)/\mathrm{r}\omega=1.0003\left(\frac{\sqrt{\mathrm{s}(\mathrm{r},\omega)+0.97221}}{2.0003}+0.0015245\right)\ -0.0138$$

Estimate c(ω) using the relation KR=2πR/c(ω)

Output Files

in ./ccawkf/data/results




```
cca_coef.dat       : calculated CCA coefficient
dispersion.dat     : estimated dispersion curve
F2LNN1_coh.dat     : coherence among the channels
F2LNN1_psp.dat     : power spectra of all channels
```

2. Estimating Dispersion Curve

2.2 Plot Power, Fourier Spectra & Coherence

Shell Script used

sh spectra_all.sh

Program used

./bin/power_plt.exe

./bin/fourier_plt.exe

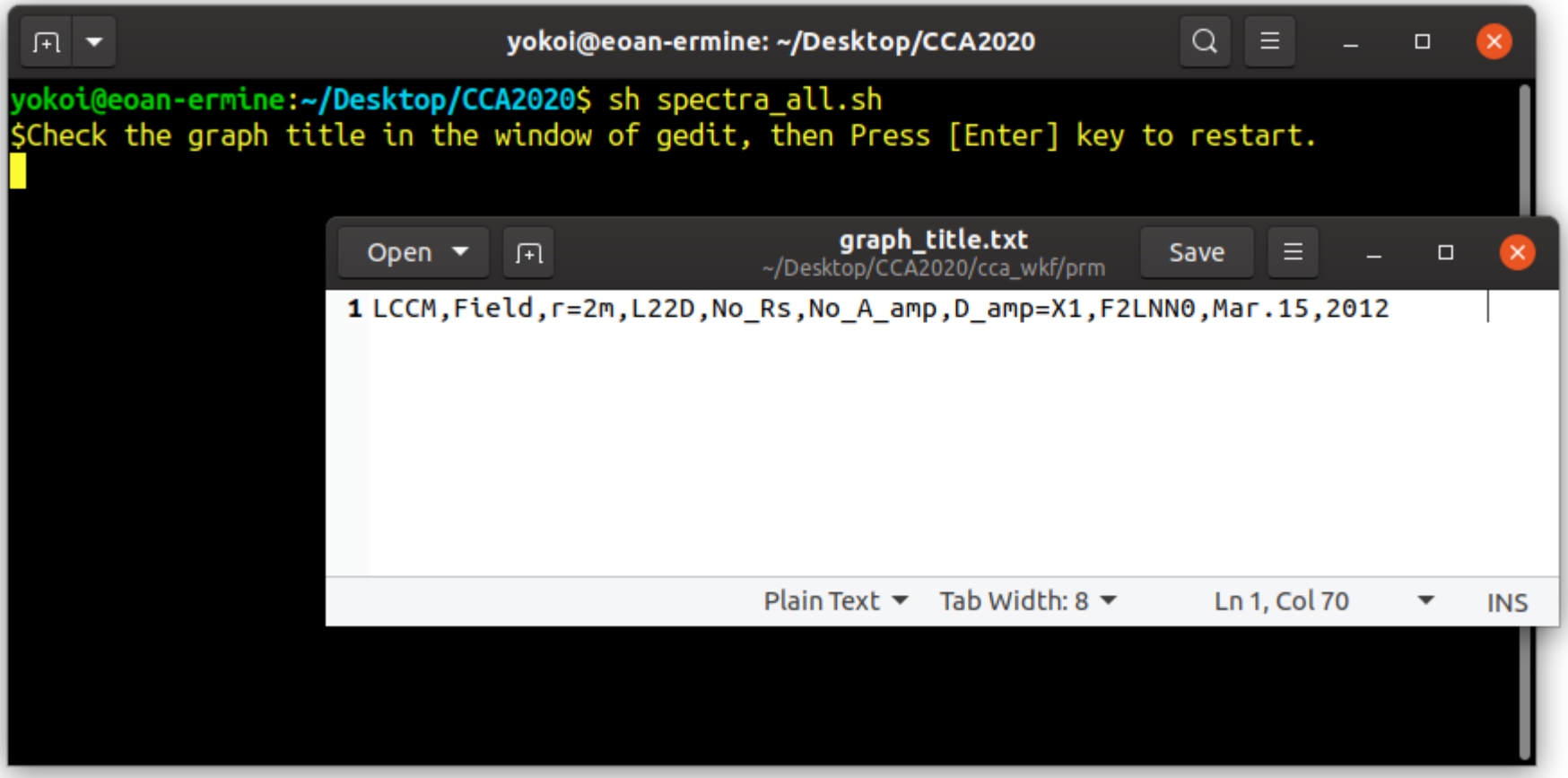
./bin/coh_plt.exe

spectra_all.sh

```
#!/bin/sh -x
echo $'Check the graph title in the window of gedit, then Press [Enter] key to restart.'
gedit ./cca_wkf/prm/graph_title.txt
read Wait
./bin/power_plt.exe | tee cca_wkf/log/spectra_all.log
gnuplot -e " load './cca_wkf/prm/gnuplt_script/power.plt' ; pause -1 "
./bin/fourier_plt.exe | tee -a cca_wkf/log/spectra_all.log
gnuplot -e " load './cca_wkf/prm/gnuplt_script/fourier.plt' ; pause -1 "
./bin/coh_plt.exe | tee -a cca_wkf/log/spectra_all.log
gnuplot -e " load './cca_wkf/prm/gnuplt_script/coherence.plt' ; pause -1 "
rm ./cca_wkf/data/results/temp*.dat 2>/dev/null
rm ./cca_wkf/prm/temp*.dat 2>/dev/null
```

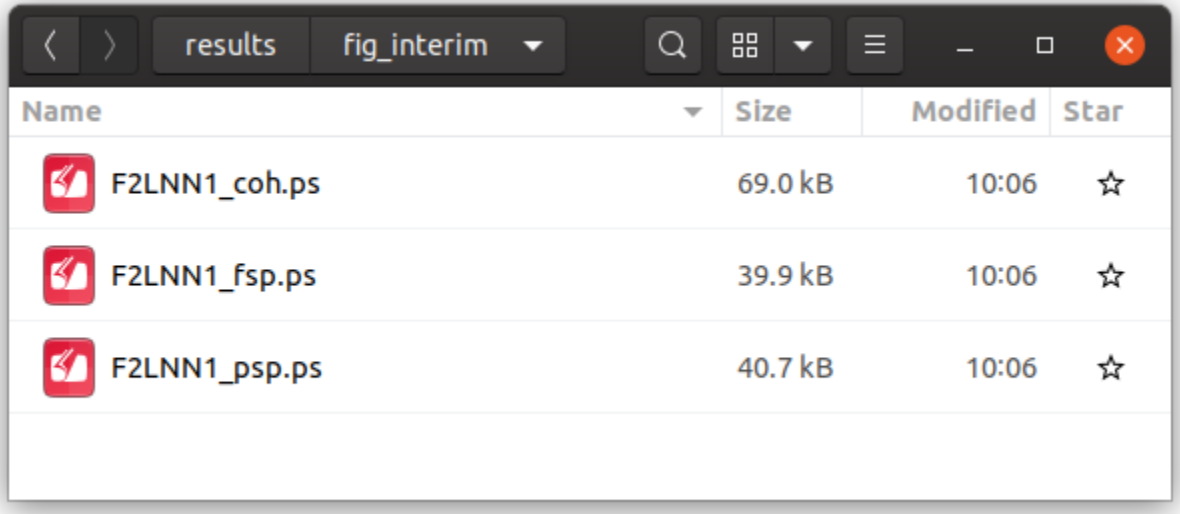
Execution(1)

First, check and modify (if necessary) ‘graph_title.txt’ on gedit window that appears automatically. Then, press [Enter] to continue the process.



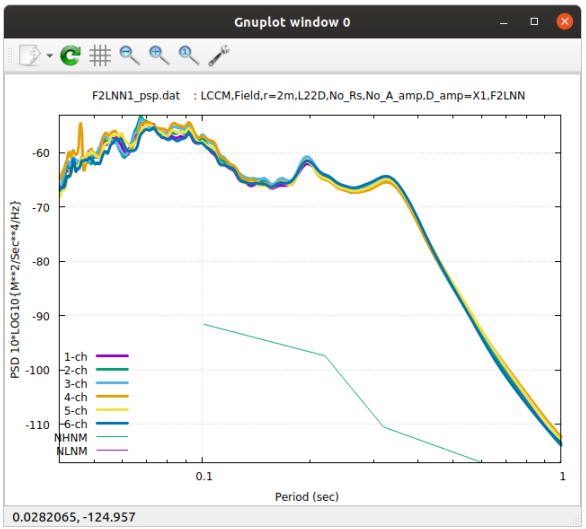
Execution(2)

All output figures displayed in X-windows are stored in PostScript format in the subfolder ./cca_wkf/data/results/fig_interim

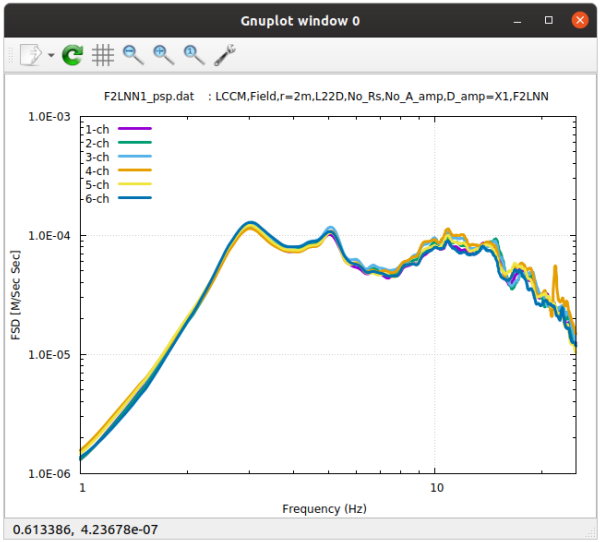


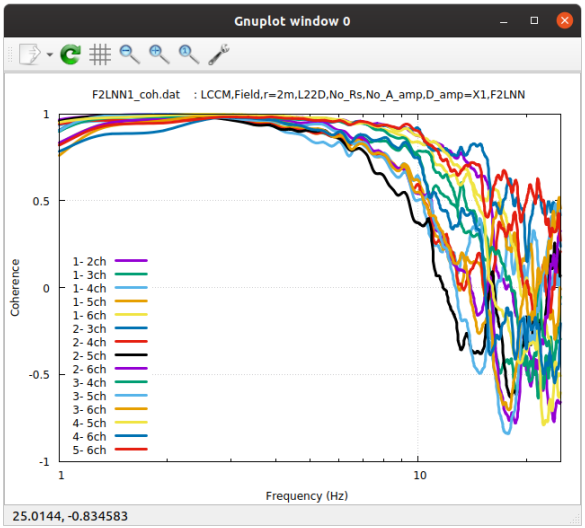
Power & Fourier Spectra

(Velocity) Power Spectra



(Velocity) Fourier Spectra





In the shown example, seismometers of which natural frequency is 2Hz are used, then the power & Fourier spectra decays in the frequency range lower than that frequency.

2. Estimating Dispersion Curve

2.3 Quality Control & Dispersion Curve

Shell Script used

sh dc_model.sh

Program used

./bin/resultc._plt.exe

./bin/q_control.exe

./bin/vel_model_plt.exe

dc_model.sh

```
#!/bin/sh -x
./bin/resultc_plt.exe | tee cca_wkf/log/results.log
gnuplot -e " load './cca_wkf/prm/gnuplt_script/results.plt' ; pause -1 "
./bin/q_control.exe | tee -a cca_wkf/log/results.log
gnuplot -e " load './cca_wkf/prm/gnuplt_script/q_control.plt' ; pause -1 "
./bin/vel_model_plt.exe | tee -a cca_wkf/log/results.log
gnuplot -e " load './cca_wkf/prm/gnuplt_script/vel_model.plt' ; pause -1 "
rm ./cca_wkf/data/results/temp*.dat 2>/dev/null
rm ./cca_wkf/prm/temp*.dat 2>/dev/null
```

cca_coef.dat

```
Freq.        ALL          Z0/ALL       Z1/ALL        Z0/Z1        Azi          err/N          2.000     1.000    25.000 :Radius, fmin, fmax
0.916          0.182E-11    0.869E+00    0.284E-01    0.306E+02   -0.209E+03    0.000E+00
0.977          0.279E-11    0.895E+00    0.223E-01    0.401E+02   -0.210E+03    0.000E+00
1.038          0.429E-11    0.909E+00    0.196E-01    0.465E+02   -0.210E+03    0.000E+00
1.099          0.652E-11    0.927E+00    0.163E-01    0.570E+02   -0.210E+03    0.000E+00
1.160          0.102E-10    0.942E+00    0.133E-01    0.707E+02   -0.210E+03    0.000E+00
1.221          0.164E-10    0.954E+00    0.106E-01    0.899E+02   -0.210E+03    0.000E+00
1.282          0.269E-10    0.964E+00    0.821E-02    0.117E+03   -0.210E+03    0.000E+00
1.343          0.435E-10    0.972E+00    0.633E-02    0.154E+03   -0.210E+03    0.000E+00
1.404          0.674E-10    0.978E+00    0.501E-02    0.195E+03   -0.209E+03    0.000E+00……..
```

1st line : titles of columns (Freq.,All, Zo/ALL, Z1/ALL, Z0/Z1,Azi,err/N), radius,fmin,fmax

Freq. : frequency

ALL : Total power spectra

Z0/ALL :

Z1/ALL :

Z0/Z1 : CCA coefficient

azi : azimuth estimated from array analysis

err/N : actually not used

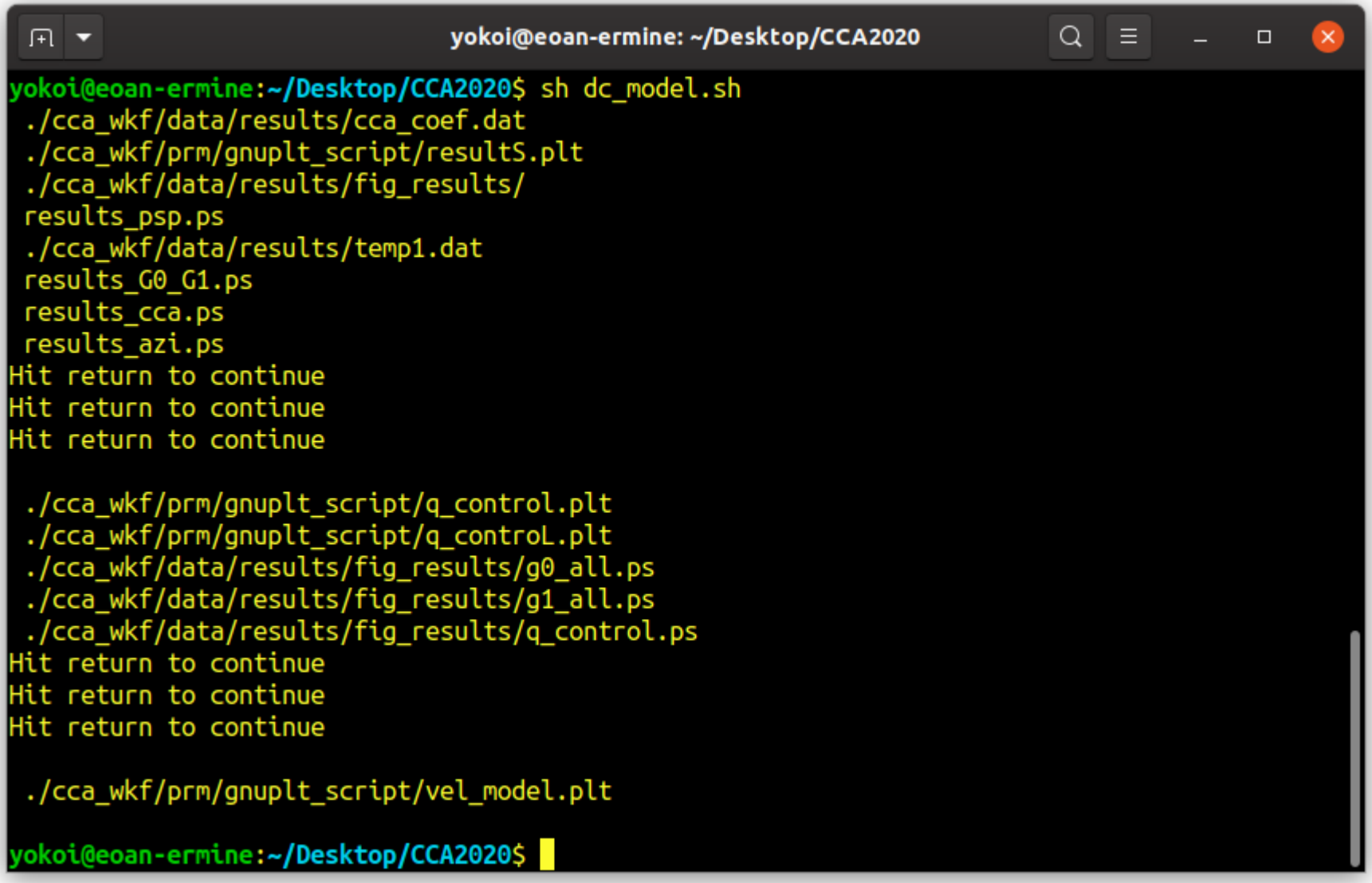
Z0: numerator of CCA coefficient

power of zero order component of Fourier expansion over azimuth

Z1: denominator of CCA coefficient

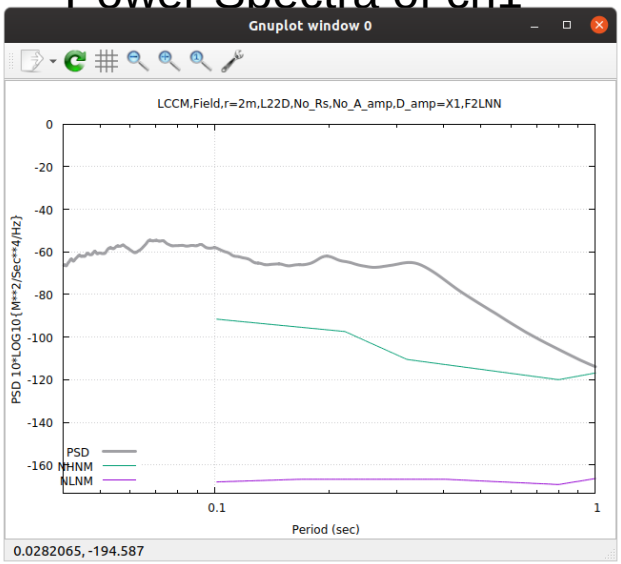
power of first order component of Fourier expansion over azimuth

Execution

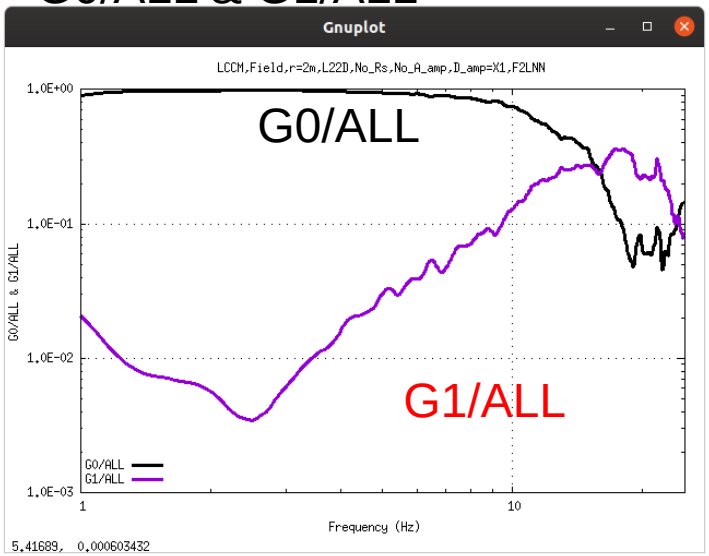

yokoi@eoan-ermine: ~/Desktop/CCA2020
yokoi@eoan-ermine:~/Desktop/CCA2020$ sh dc_model.sh
./cca_wkf/data/results/cca_coef.dat
./cca_wkf/prm/gnuplt_script/resultS.plt
./cca_wkf/data/results/fig_results/
results_psp.ps
./cca_wkf/data/results/temp1.dat
results_G0_G1.ps
results_cca.ps
results_azi.ps
Hit return to continue
Hit return to continue
Hit return to continue
./cca_wkf/prm/gnuplt_script/q_control.plt
./cca_wkf/prm/gnuplt_script/q_controL.plt
./cca_wkf/data/results/fig_results/g0_all.ps
./cca_wkf/data/results/fig_results/g1_all.ps
./cca_wkf/data/results/fig_results/q_control.ps
Hit return to continue
Hit return to continue
Hit return to continue
./cca_wkf/prm/gnuplt_script/vel_model.plt
yokoi@eoan-ermine:~/Desktop/CCA2020$

cca_coef.dat

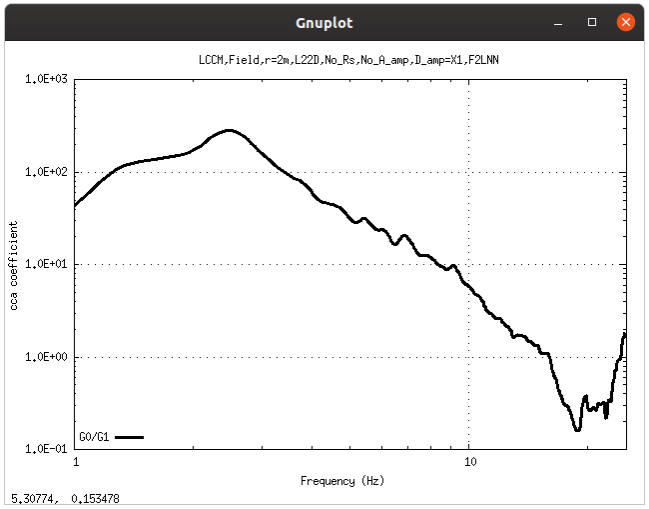
Power Spectra of ch1



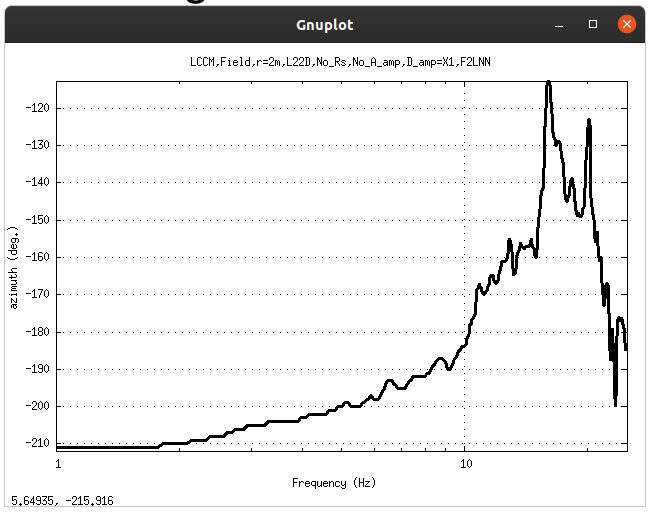
G0/ALL & G1/ALL



CCA coefficient



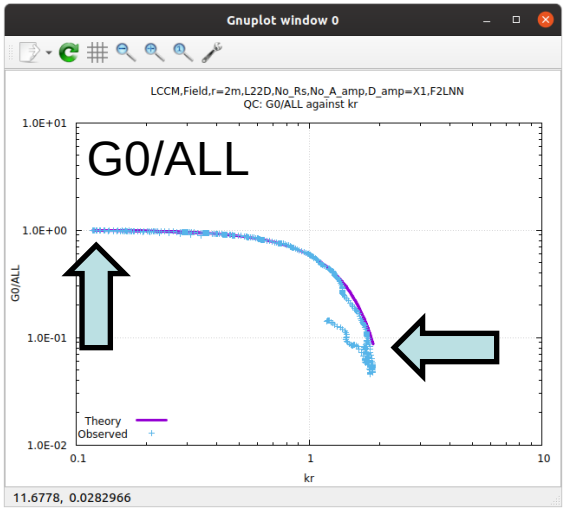
Incoming Azimuth

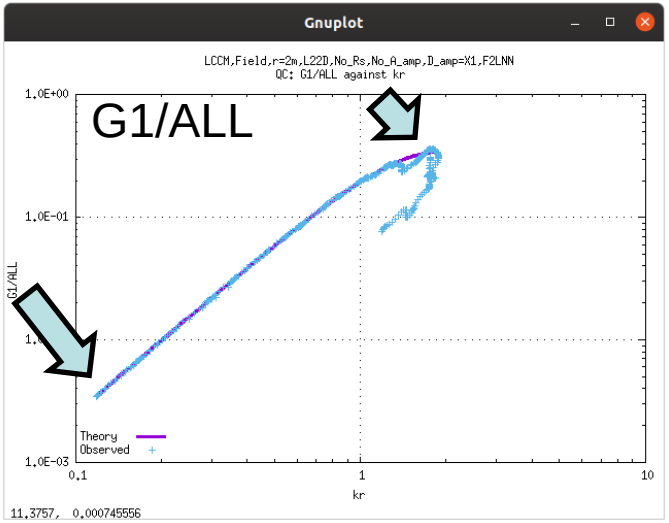


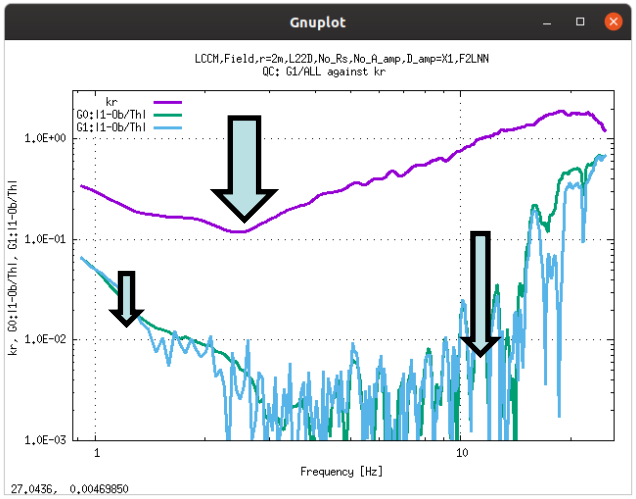
These figures are stored in ./cca_wkf/data/results/fig_results in PostScript format.

Quality Control

Using the values of “kr” in the file “dispersion.dat”, Z0/ALL and Z1/ALL in the file “cca_coef.dat” are compared with the theoretical $J_0(kr)^2$ and $J_1(kr)^2$, respectively. Arrows indicate the limit of the range of “kr” for analysis, where the observed curves run off from the theoretical ones.







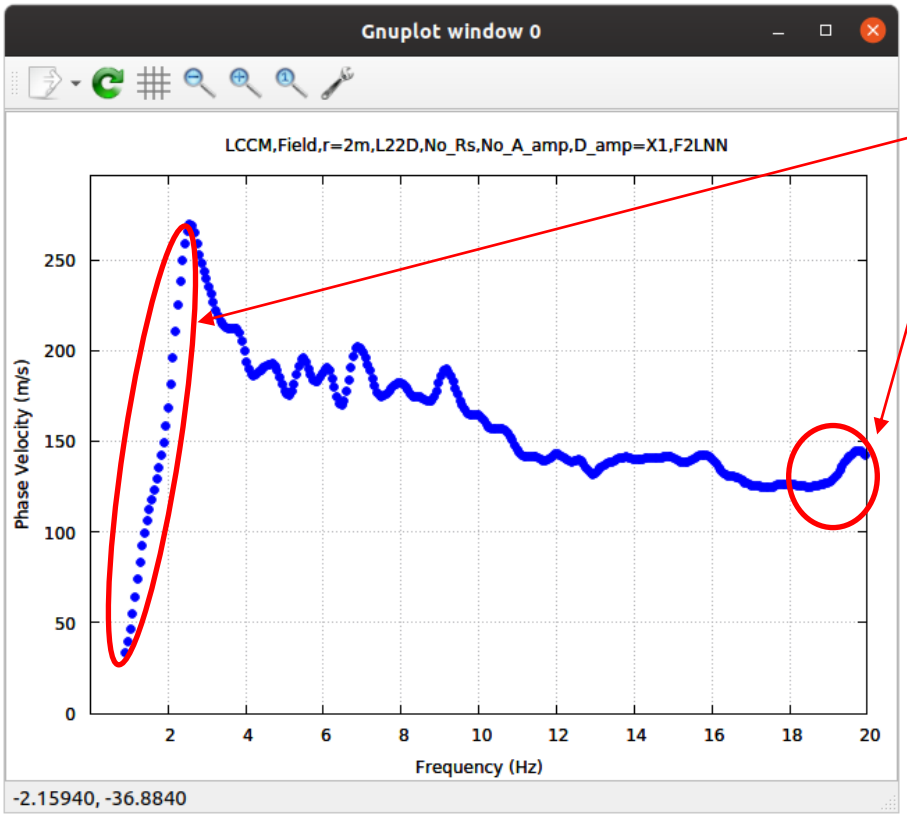
(Example 1-5)

These figures are stored in ./cca_wkf/data/results/fig_results in PostScript format.

dispersion.dat

Frequency	Velocity	Azimuth	KR
0.916	31.498	-208.950	0.365
0.977	37.589	-210.297	0.326
1.038	43.833	-211.355	0.297
1.099	50.411	-211.607	0.274

…

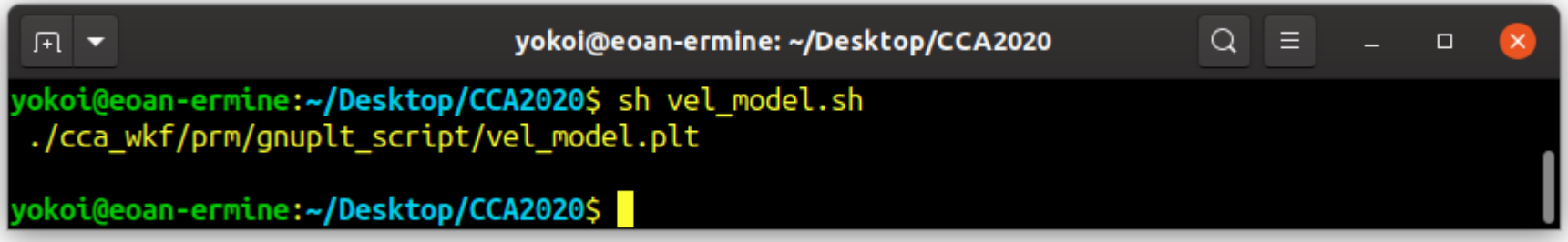


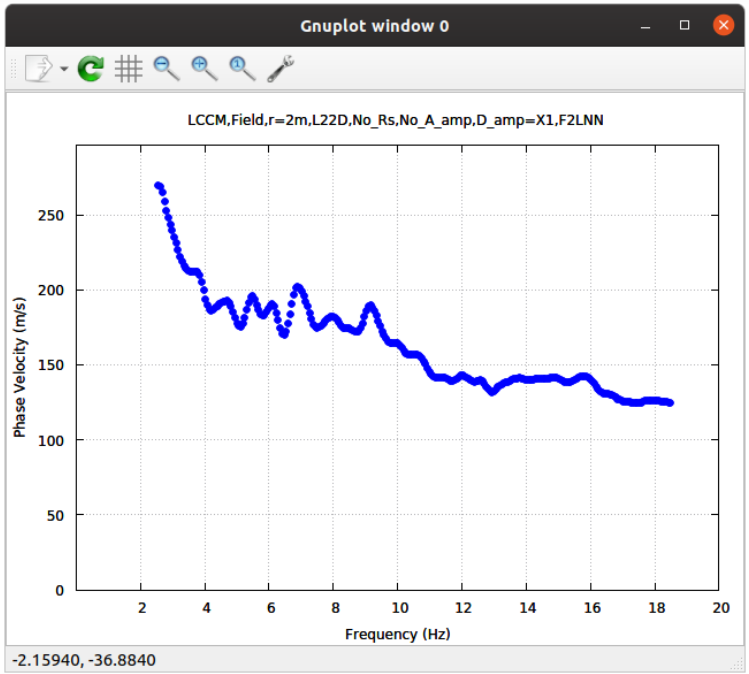
Eliminate strange data using gedit.

This figure is stored in ./cca_wkf/data/results/fig_results in PostScript format.

sh vel_model.sh

Plot and check the edited dispersion curve.





vel_model.sh

./bin/vel_model_plt.exe | tee -a cca_wkf/log/results.log
gnuplot -e " load './cca_wkf/prm/gnuplt_script/vel_model.plt' ; pause -1 "
rm ./cca_wkf/data/results/temp*.dat

4. Heuristic Search of Vs Structure

Shell Script used:

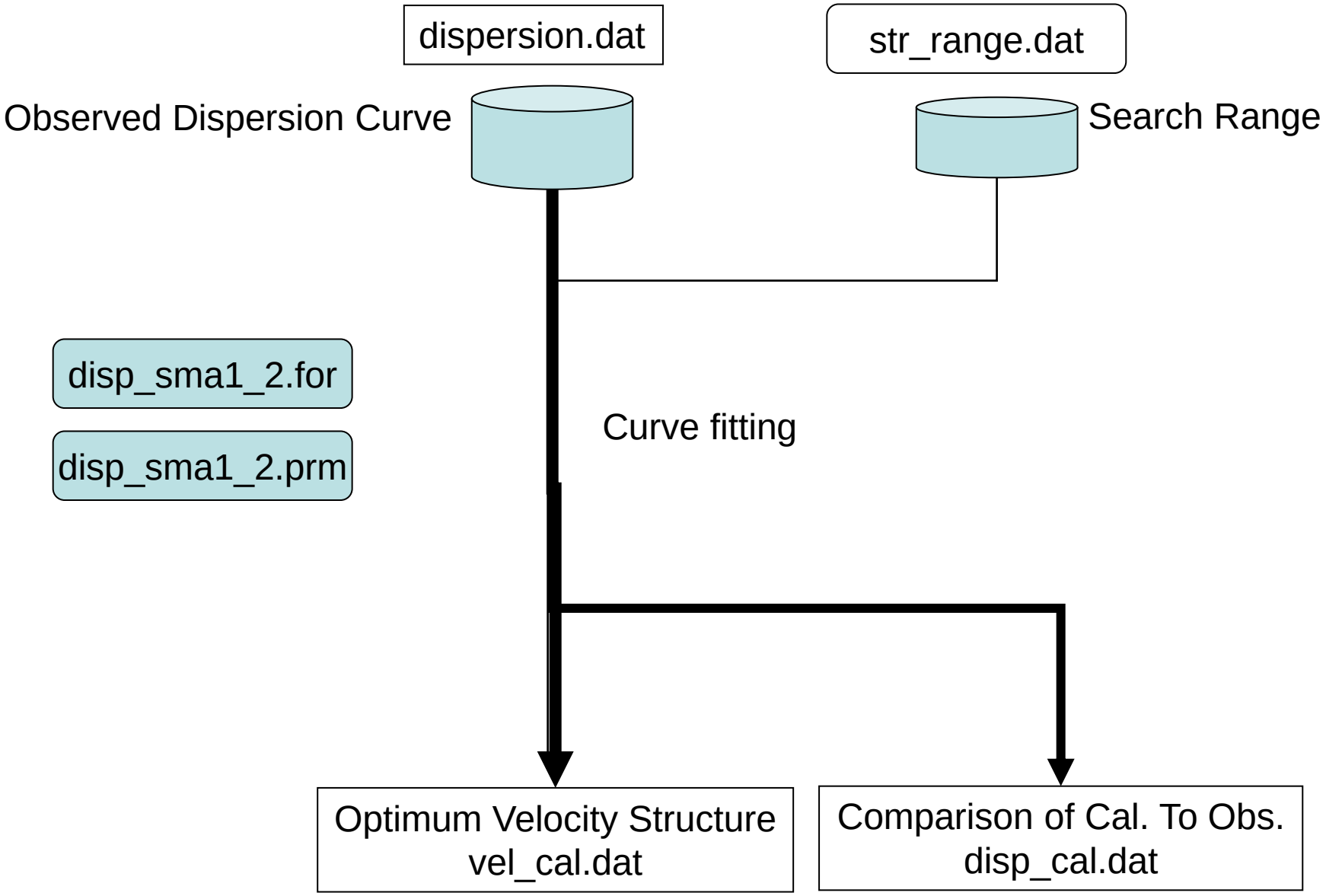
sh inversion.sh

Programs and parameter file used:

./bin/disp_sma1_2.exe + disp_sma1_2.prm

inversion.sh:

```
#!/bin/sh -x
./bin/disp_sma1_2.exe | tee ./cca_wkf/log/inversion.log
./bin/inv_plt.exe | tee -a ./cca_wkf/log/inversion.log
gnuplot -e "load './cca_wkf/prm/gnuplt_script/inv.plt' ; pause -1 "
rm ./cca_wkf/data/results/temp*.dat 2>/dev/null
rm ./cca_wkf/prm/temp*.dat 2>/dev/null
```


dispersion.dat
str_range.dat
Observed Dispersion Curve
Search Range
disp_sma1_2.for
disp_sma1_2.prm
Curve fitting
Optimum Velocity Structure
vel_cal.dat
Comparison of Cal. To Obs.
disp_cal.dat

disp_sma1_1.for Yokoi(2006)

Combination of the Down Hill Simplex Method (Nelder & Mead (1965)) and the Very Fast Simulated Annealing method (Ingber, 1989).

DHSM: Down Hill Simplex Method (Nelder & Mead (1965))
An efficient algorithm to find "local minimum".
Faster than Geiger's method. Partial derivatives are not necessary.
Result is controlled by given initial values and easily captured by local minimum.
Example of application to the microtremor array: Ohori et al(2002)

VFSA: Very Fast Simulated Annealing method (Ingber, 1989)
One of the heuristic search methods.
Analogy of cooling and crystallization process of metals.
Results can escape from local minimum and can get global minimum with some probability.
Time consuming due to the probabilistic search for each parameter.
Example of application to the microtremor array & appropriate values of parameters for this purpose: Yamanaka (2004)

Parameter file: disp_sma1_2.prm

(Example 1-6)

```
 1 1. 0.6 1.3 10000  5   :idum,t0,a,c,ntemp,j0
 0.0025                 :eps0
    1    1              :n_roh,n_vp
    1    0    1         :ini_flg,ndsp_flg,n_err
    0    1              :kflg,jflg
    0    0              :n_vs,n_th
str_range.dat           :File name for the initial velocity model (a25).
dispersion.dat          :File name for the obseved dispersion relation (a25).
vel_cal.dat             :File name for the estimated velocity structure (a25)
disp_cal.dat            :File name for the calculated dispersion relation (a25)
```

Explanation

Control parameter for the simulated annealing method

idum :Random seed (integer)
As the result may depend on the initial velocity model given by random number, it is strongly recommended for users to apply this program several times with various values of random seed and to grasp the scatter of result.

t0 :Initial Temperature

a,c :Coefficients for T=T0*exp(-c*k**a), where k is iteration number

ntemp :Maximum number of temparature change

j0 :Number of iteration for each temperature

threshold for conversion

eps0 : acceptable misfit.

Search Range file: str_range.dat

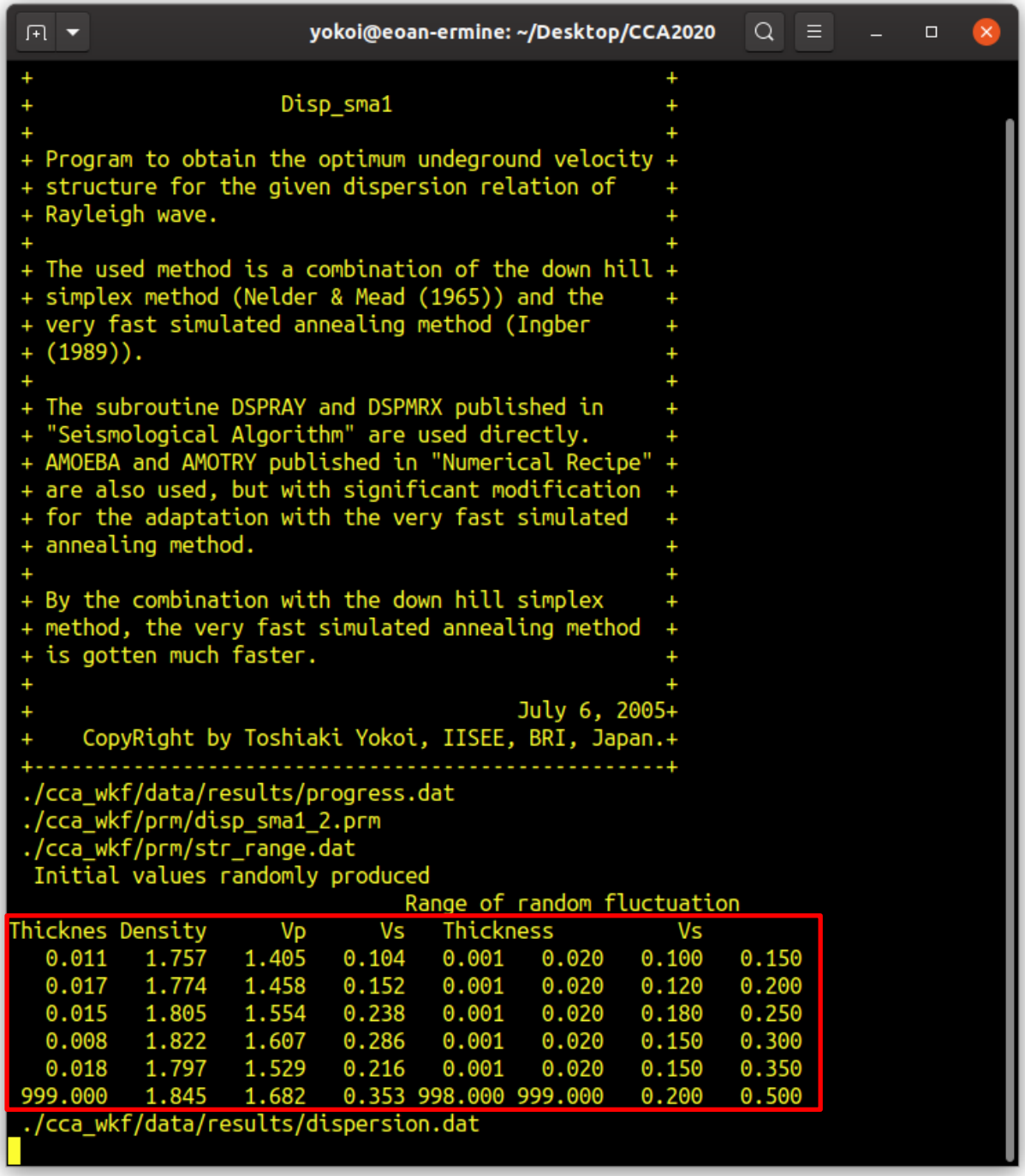
(Example 1-6)

```
LCCM,r=2m,L22D,F2LNN0,12/03/15:Model(a30)
         6                          :IL(I5),Layer Number
1.9   1.5    0.001  0.020  0.10 0.15 :density,Vp,hmin,hmax,vmin,vmax
1.9   1.5    0.001  0.020  0.12 0.20 :density,Vp,hmin,hmax,vmin,vmax
1.9   1.5    0.001  0.020  0.18 0.25 :density,Vp,hmin,hmax,vmin,vmax
1.9   1.5    0.001  0.020  0.15 0.30 :density,Vp,hmin,hmax,vmin,vmax
1.9   1.5    0.001  0.020  0.15 0.35 :density,Vp,hmin,hmax,vmin,vmax
2.0   1.70   998.0  999.0  0.20 0.50 :density,Vp,hmin,hmax,vmin,vmax
```

If n_vp=1 in parameter file, the given values of Vp are not used

If n_roh=1 in parameter file, the given values of density are not used

Execution (Example 1-6).



Initial values randomly selected within the search range.

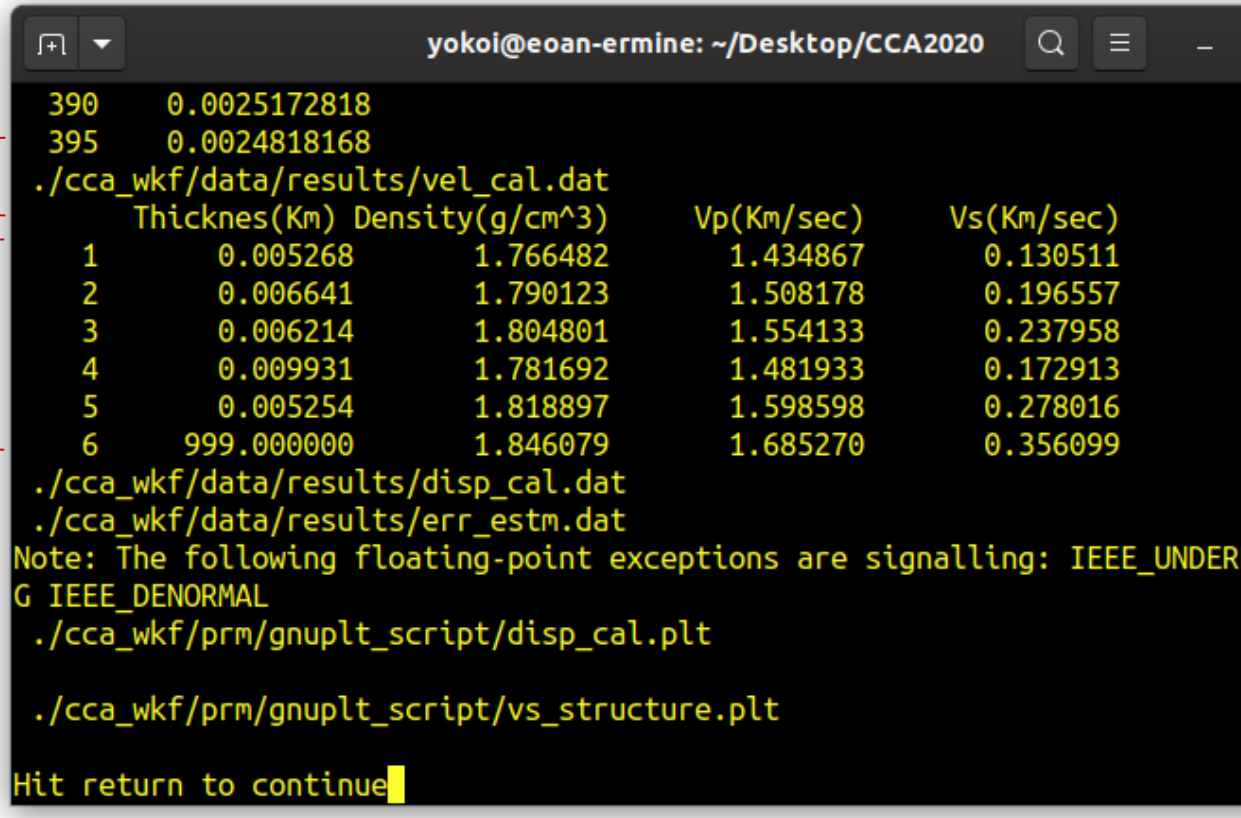
Iteration Starts

Execution (Example 1-6 cont).

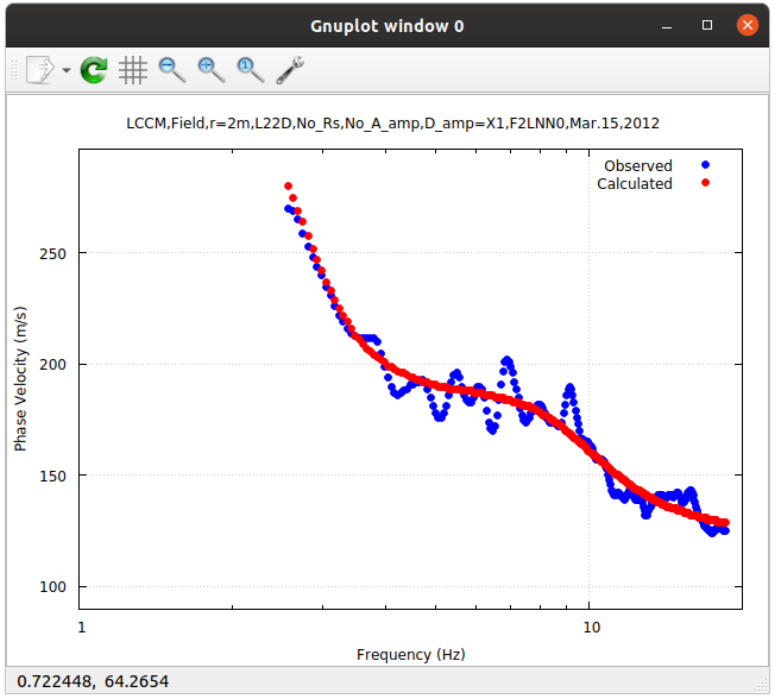
Conversion of misfit to the threshold eps0

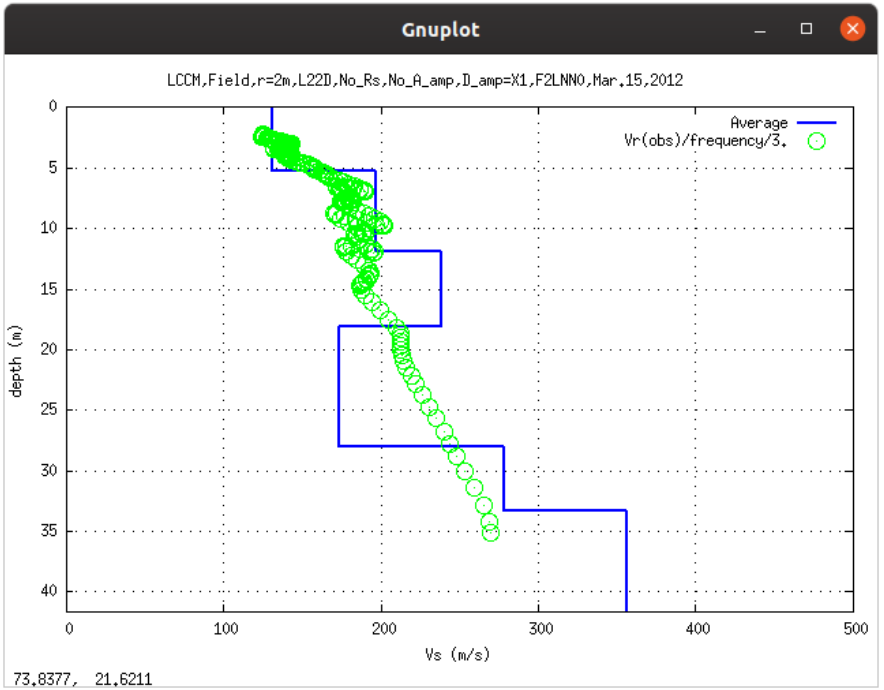
Optimum Underground velocity structure

Plotting



Plotting (Example 1-6 cont).





The same figures are stored in Post Script files:

./cca_wkf/data/results/fig_results/disp_cal.ps

./cca_wkf/data/results/fig_results/vs_structure.ps

Green circles can be eliminated by setting n_mod=0 in pwrcrs3.prm.

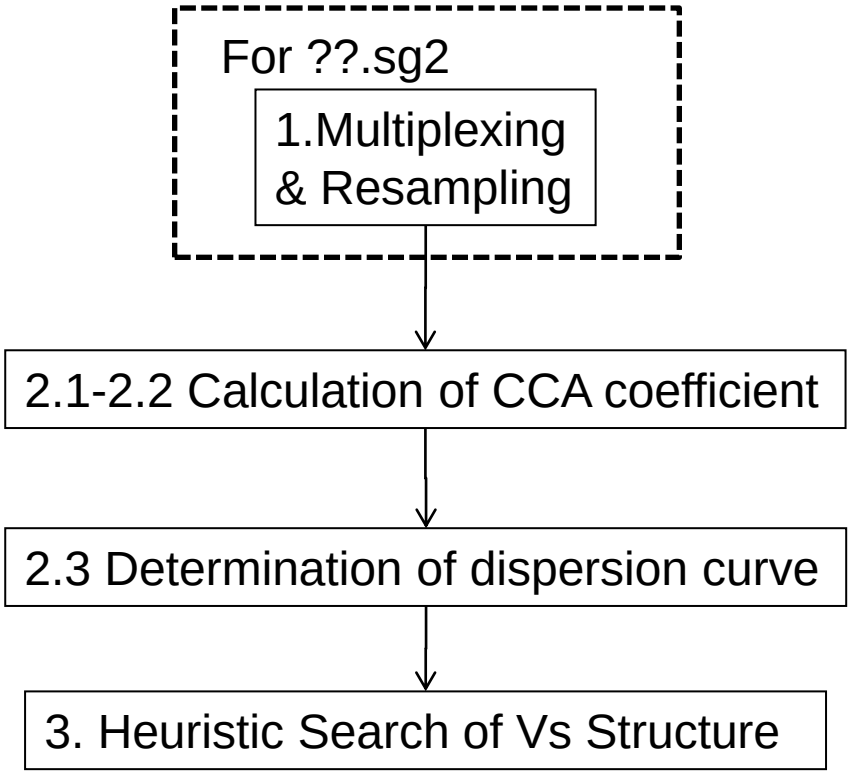
Example2

(Sensor at Center without Instrumental Correction)

Example with two data sets:

7 points circular array, one of which is at the center

Both are the seg2 standard format file.



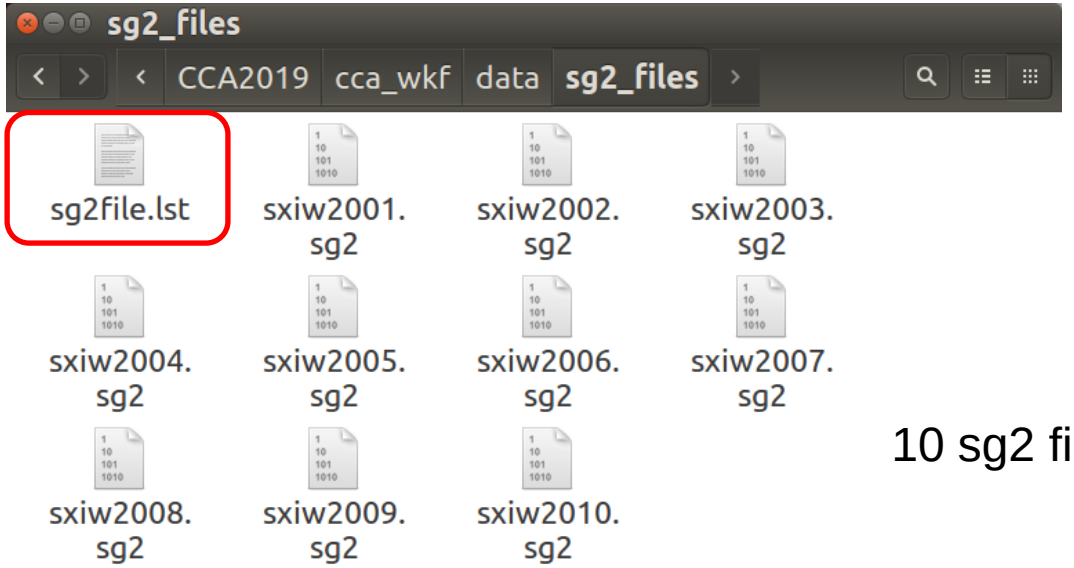
The array used in the field has 7 sensors and 7-th sensor (CH-7) at the center. Other 6 sensors were installed clockwise order. Then,
pvlist 7 6 5 4 3 2 1 :

seg2read.sh: Execution

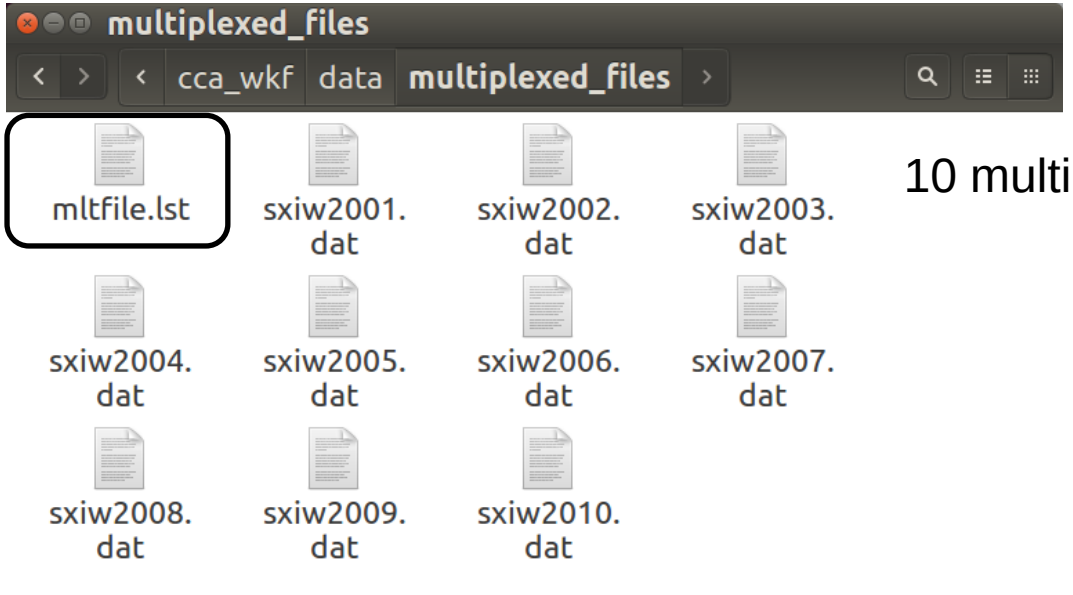
This title is used for every figures

```
seg2read.prm

Yoshima Elementary School, L22, Feb. 15 2016                    :comm(a70)
5.E-3     : scaling factor (for output files in mkine(1.e-3cm/s))
sg2       : extension of input seg2 format files(a3)
dat       : extension of output ascii text files(a3)
0 3 0.1 1.0 1.5  :nfilter(=1:apply),nchara=3:bandpass),fl,fh,fs
pvlist 7 6 5 4 3 2 1 :Channel Pivoting
```

```
yokoi@eoan-ermine: ~/Desktop/CCA2020
yokoi@eoan-ermine:~/Desktop/CCA2020$ sh seg2read.sh
./cca_wkf/prm/seg2read.prm
   4.99999989E-03 mkine
./cca_wkf/data/sg2_files/sg2file.lst
sxiw2001.sg2

./cca_wkf/data/sg2_files/sxiw2001.sg2
sxiw2010.sg2

./cca_wkf/data/sg2_files/sxiw2010.sg2
./cca_wkf/data/multiplexed_files/sxiw2010.dat
                  :32bit floating point.
nch=  7 dt= 0.002 nn=   16384
       7  0.0020      0.5000E-02   16384  mkine
Yoshima Elementary School, L22, Feb. 15 2016                    :
    10 files have been converted.
Normal End.
yokoi@eoan-ermine:~/Desktop/CCA2020$
```

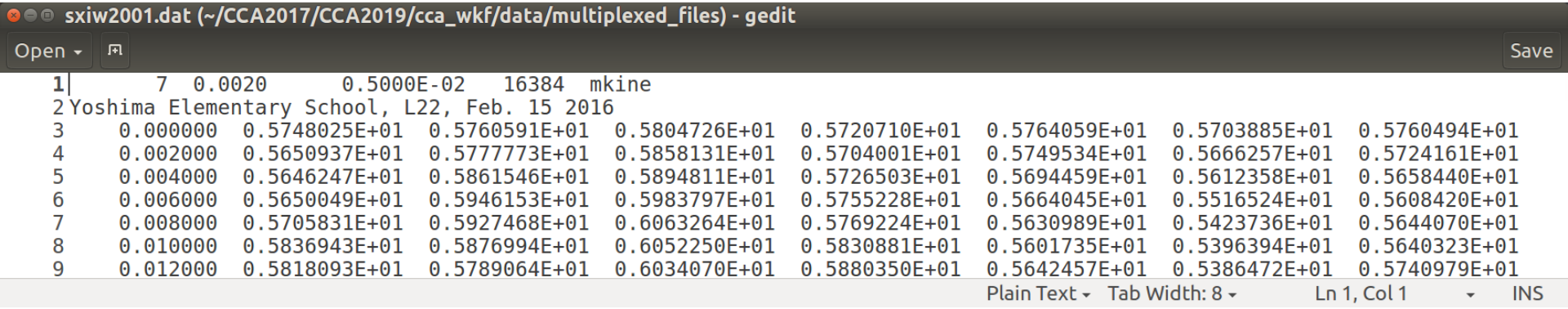


10 sg2 files



10 multiplexed files

cdm files in ./cca_wkf/data/multiplexed_files



7 channels
Sampling interval dt = 0.002 sec (0.5kHz)
Scaling factor = 0.005
16384 samples in each file

➔ 16.384 sec in each file

resamplec_pre.sh: Execution

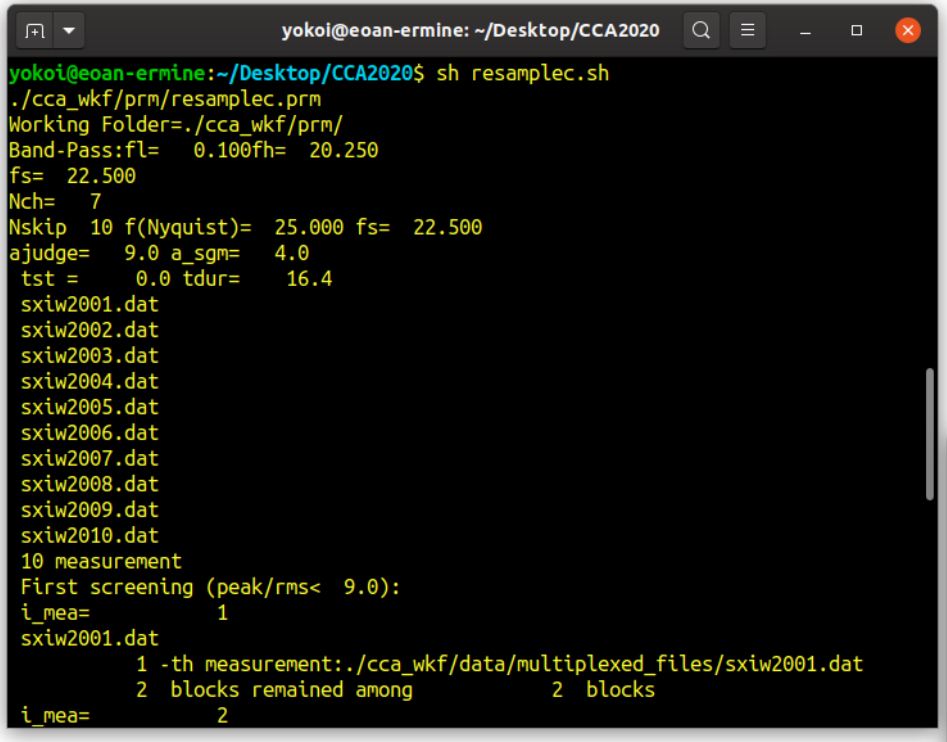
```
yokoi@eoan-ermine: ~/Desktop/CCA2020
yokoi@eoan-ermine:~/Desktop/CCA2020$ sh resamplec_pre.sh
 Number of multiplexed files:           10
 Check point: previous parameter setting:
 +++++++++++++++++++++++++++++++++++++++++
 7 0.002 10   0.0  1  1.0   :nnch,dt,nchannel,dt,nskip,ph0,ncenter,radius
   9.000   4.000            :ajudge,a_sgm.
   0.000  16.383            :tst,tdur (start time and duration).
YOSIMA.dat                  :output (output file name).
 1024                       :ndata (in a time block after resampling).
 +++++++++++++++++++++++++++++++++++++++++++++
 Do they look fine? ("yes" or "no")
no
 Modify parameters.
 Which parameter do you want modify? Type "e" to finalize setting renewal.
ndata 512
 Which parameter do you want modify? Type "e" to finalize setting renewal.
e

 Check point: renewed parameter setting:
 7 0.002 10   0.0  1  1.0   :nnch,dt,nchannel,dt,nskip,ph0,ncenter,radius
   9.000   4.000            :ajudge,a_sgm.
   0.000  16.383            :tst,tdur (start time and duration).
YOSIMA.dat                  :output (output file name).
  512                       :ndata (in a time block after resampling).

 Do they look fine? ("yes" or "no")
yes
 +++++++++++++++++++++++++++++++++++++++++
 Selection of data files. Currently there are:
   10 files in ~/multiplexed_files/
sxiw2001 sxiw2002 sxiw2003 sxiw2004 sxiw2005 sxiw2006 sxiw2007 sxiw2008
sxiw2009 sxiw2010
 +++++++++++++++++++++++++++++++++++++++++
 Do they look fine? ("yes" or "no")
yes
 +++++++++++++++++++++++++++++++++++++++++
 +++++++++++++++++++++++++++++++++++++++++
 7 0.002 10   0.0  1  1.0   :nnch,dt,nchannel,dt,nskip,ph0,ncenter,radius
   9.000   4.000            :ajudge,a_sgm.
   0.000  16.383            :tst,tdur (start time and duration).
```

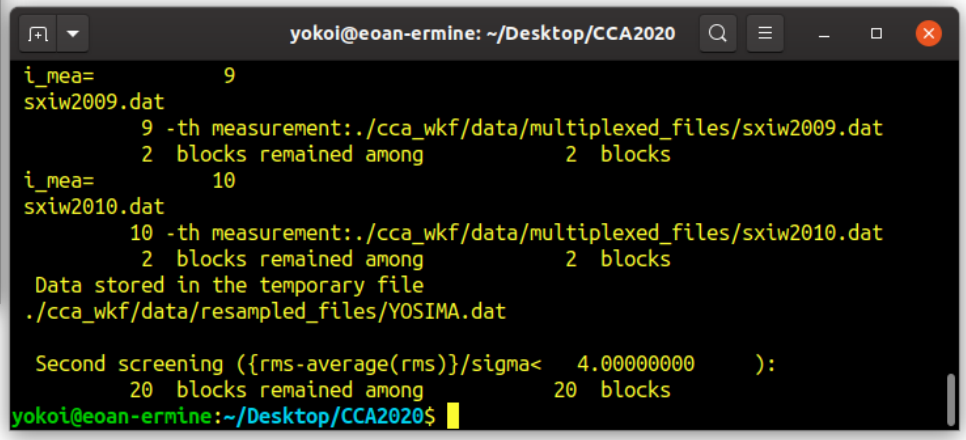
```
yokoi@eoan-ermine: ~/Desktop/CCA2020
 +++++++++++++++++++++++++++++++++++++++++
 +++++++++++++++++++++++++++++++++++++++++
 7 0.002 10   0.0  1  1.0   :nnch,dt,nchannel,dt,nskip,ph0,ncenter,radius
   9.000   4.000            :ajudge,a_sgm.
   0.000  16.383            :tst,tdur (start time and duration).
YOSIMA.dat                  :output (output file name).
  512                       :ndata (in a time block after resampling).
 10                         :nfile (no. of measurement in the same array.
sxiw2001
sxiw2002
sxiw2003
sxiw2004
sxiw2005
sxiw2006
sxiw2007
sxiw2008
sxiw2009
sxiw2010
 +++++++++ written in resamplec.prm ++++++++
 +++++++++++++++++++++++++++++++++++++++++++
 ./cca_wkf/prm/resamplec.prm

yokoi@eoan-ermine:~/Desktop/CCA2020$
```

resamplec.sh: Execution

resamplec.prm:

```
 7 0.002 10   0.0  1  1.0   :nch,dt,nskip,ph0,ncenter,radius
9.0 4.0                     :ajudge,a_sgm
0.0 16.383                  :tst,tdur
YOSIMA.dat                  :output file name (A10)
 512                        :number of data in one time block after resampling
```





resampled file in ./cca_wkf/data/resampled_files: Yoshima.dat

```
7   20  512 1    0.00    1.00 (i8,f16.4, 7e15.7) mkine
      1          0.0000  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00
      2          0.0200  0.1886124E-06  0.1951861E-06  0.1940966E-06  0.1863137E-06  0.1889530E-06  0.1898792E-06  0.1919477E-06
      3          0.0400  0.2108473E-03  0.2167535E-03  0.2173618E-03  0.2086327E-03  0.2086433E-03  0.2078045E-03  0.2122828E-03
      …
```

7 channels
Resampling interval dt = 0.02 sec (50Hz)
512 samples in each time block➔ 10.24 sec/block
Sensor at the center: ON
ϕ_0=0.0 rad.
Radius:=1.0 m
Unit: mkine=1.0e-3 cm/sec

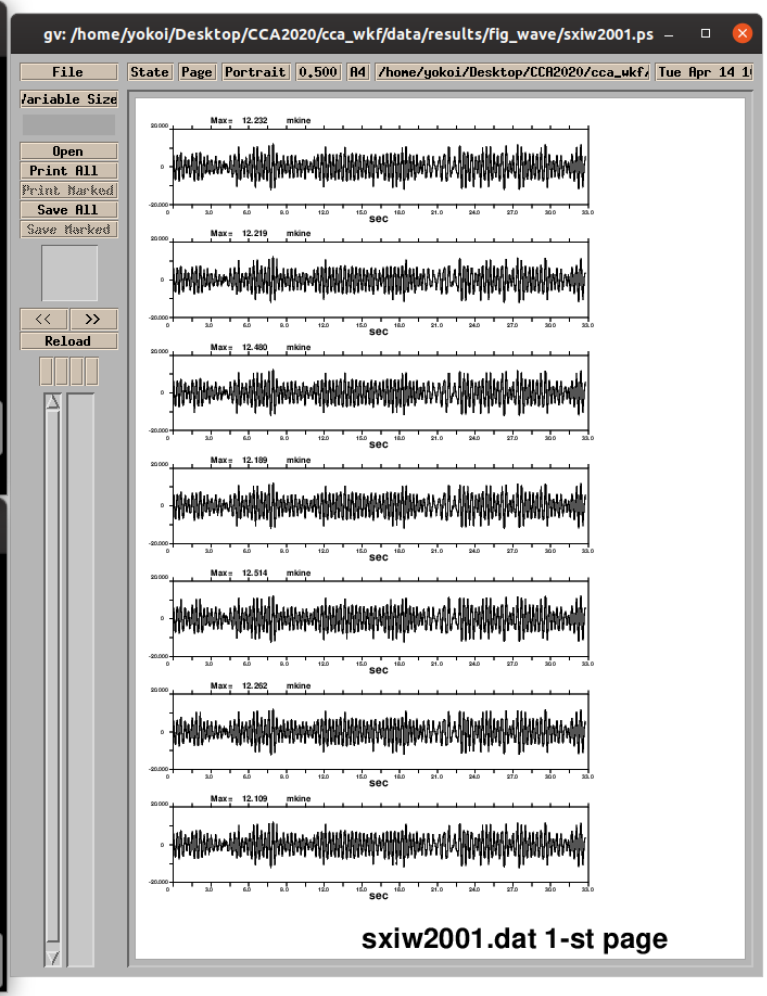
seewavc.sh: Execution

seewavc.prm:

```
7                          :nch
0 0.1   1.0 1.5 3          :nfilter,fl,fh,fs,nchara(=2:lowpass, =3:bandpass)
 1.5                       :dtl(sec/cm),25,50==>10,20 min/page
```

```
yokoi@eoan-ermine: ~/Desktop/CCA2020
sxiw2010.dat 2-nd page
yokoi@eoan-ermine:~/Desktop/CCA2020$ sh seewavc.sh
sxiw2001.dat
sxiw2002.dat
sxiw2003.dat
sxiw2004.dat
sxiw2005.dat
sxiw2006.dat
sxiw2007.dat
sxiw2008.dat
sxiw2009.dat
sxiw2010.dat
 10 measurement
./cca_wkf/data/results/fig_wave/sxiw2001.ps

sxiw2001.dat 1-st page
```

```
yokoi@eoan-ermine: ~/Desktop/CCA2020
./cca_wkf/data/results/fig_wave/sxiw2006.ps

sxiw2006.dat 1-st page
./cca_wkf/data/results/fig_wave/sxiw2007.ps

sxiw2007.dat 1-st page
./cca_wkf/data/results/fig_wave/sxiw2008.ps

sxiw2008.dat 1-st page
./cca_wkf/data/results/fig_wave/sxiw2009.ps

sxiw2009.dat 1-st page
./cca_wkf/data/results/fig_wave/sxiw2010.ps

sxiw2010.dat 1-st page
yokoi@eoan-ermine:~/Desktop/CCA2020$ gv &
```

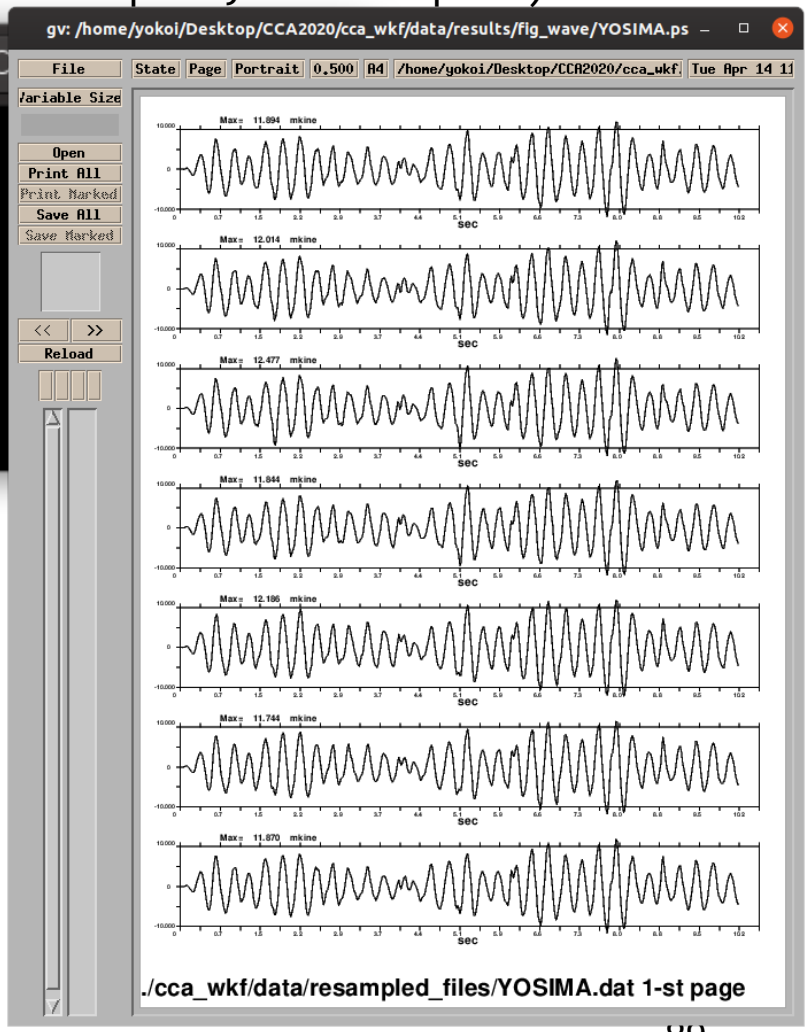


seeblkc.sh: Execution

seeblkc.prm:

```
0 0.1 1.0 1.5 3    :nfilter,fl,fh,fs,nchara(=2:lowpass, =3:bandpass)
```

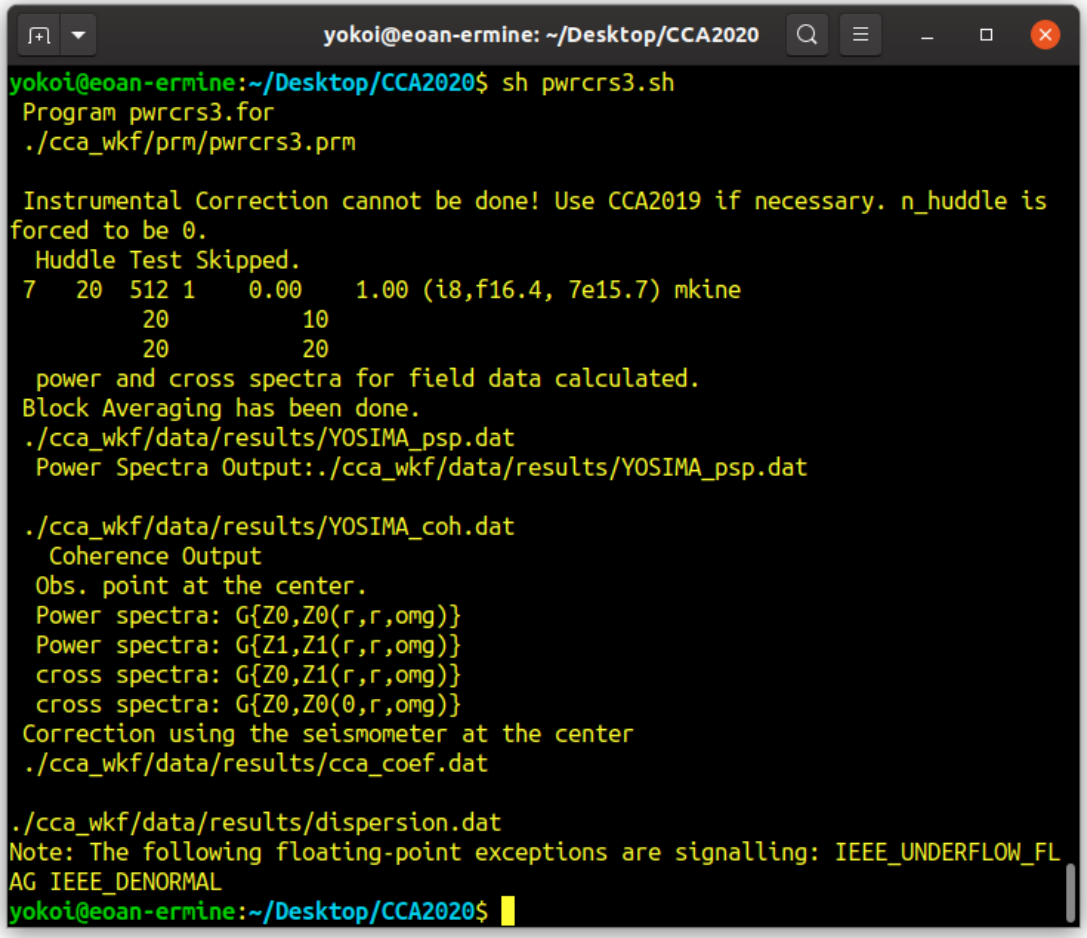
```
yokoi@eoan-ermine: ~/Desktop/CCA2020
yokoi@eoan-ermine:~/Desktop/CCA2020$ sh seeblkc.sh
 ./cca_wkf/data/resampled_files/rsmfile.lst

./cca_wkf/prm/seeblkc.prm
./cca_wkf/data/resampled_files/YOSIMA.dat
 7   20  512 1    0.00    1.00 (i8,f16.4, 7e15.7) mkine
 ./cca_wkf/data/results/fig_wave/YOSIMA.ps

yokoi@eoan-ermine:~/Desktop/CCA2020$
```



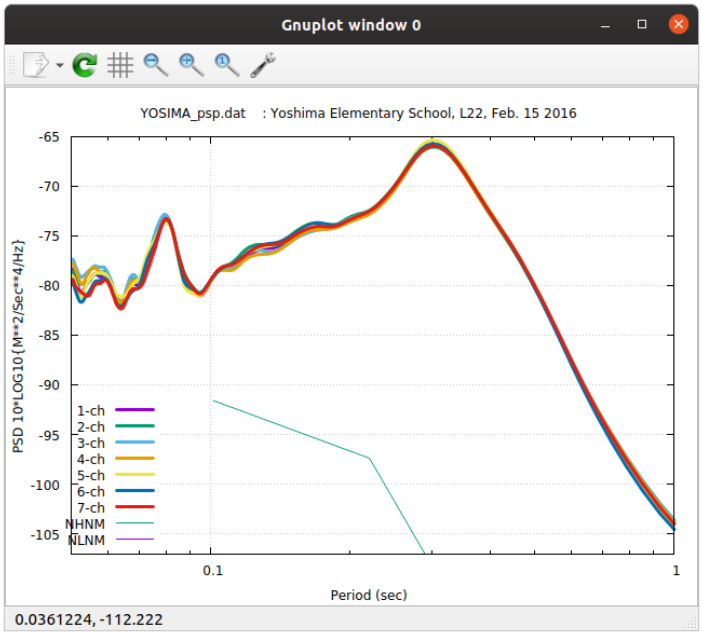
pwrcrs3.sh: Execution

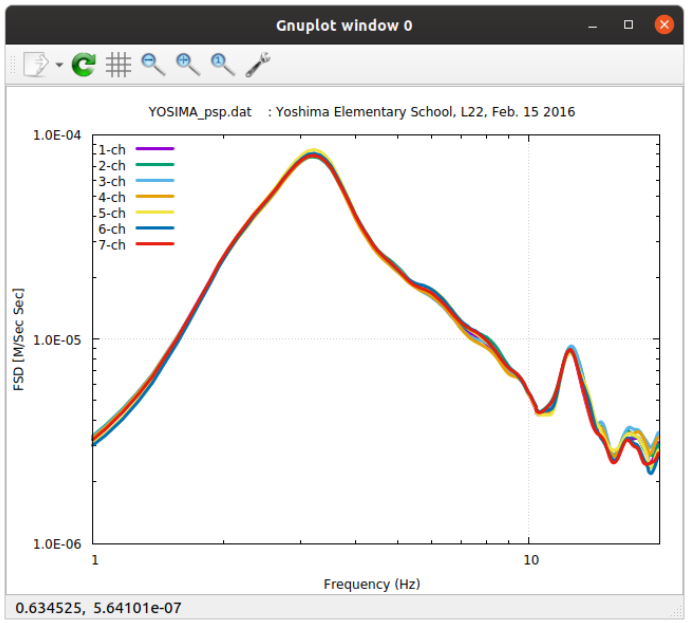
pwrcrs3.prm:

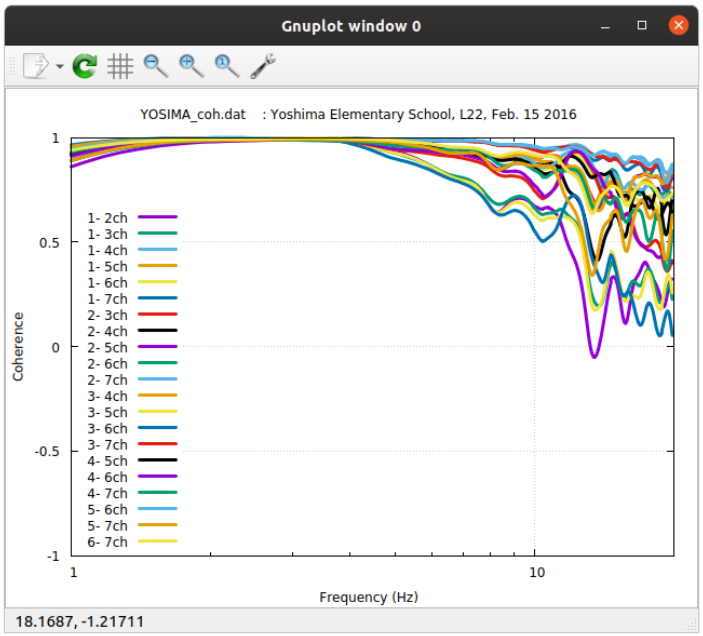
```
 1.0 20.0 0.02 0.4   .3    :fmin,fmax,dt,bw, smthf
YOSIMA.dat      1 1         :Field data File name(A12) ,coherence, power pectra,output flag
1   0                       :n_cor_center, n_mod
```

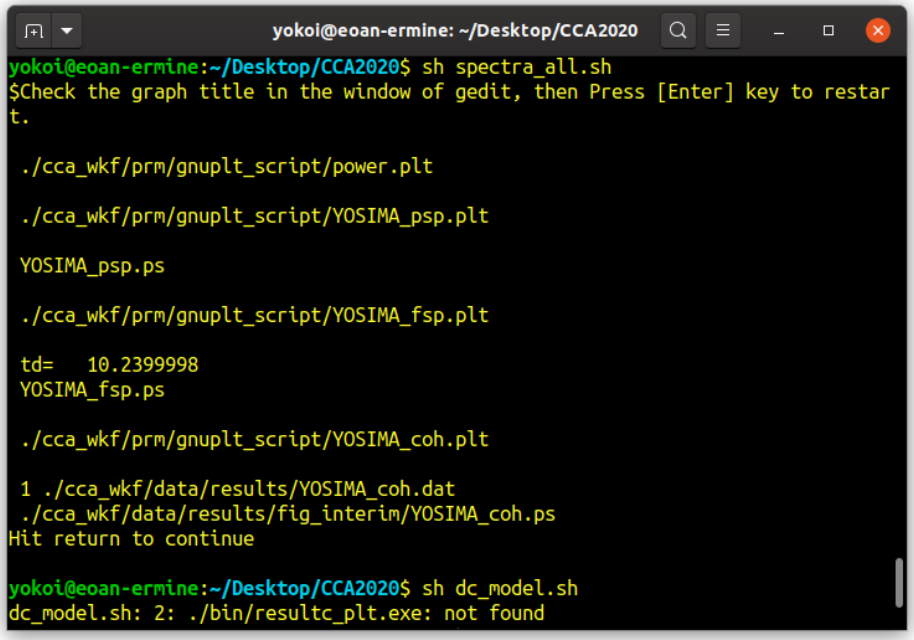


spectra_all.sh: Execution

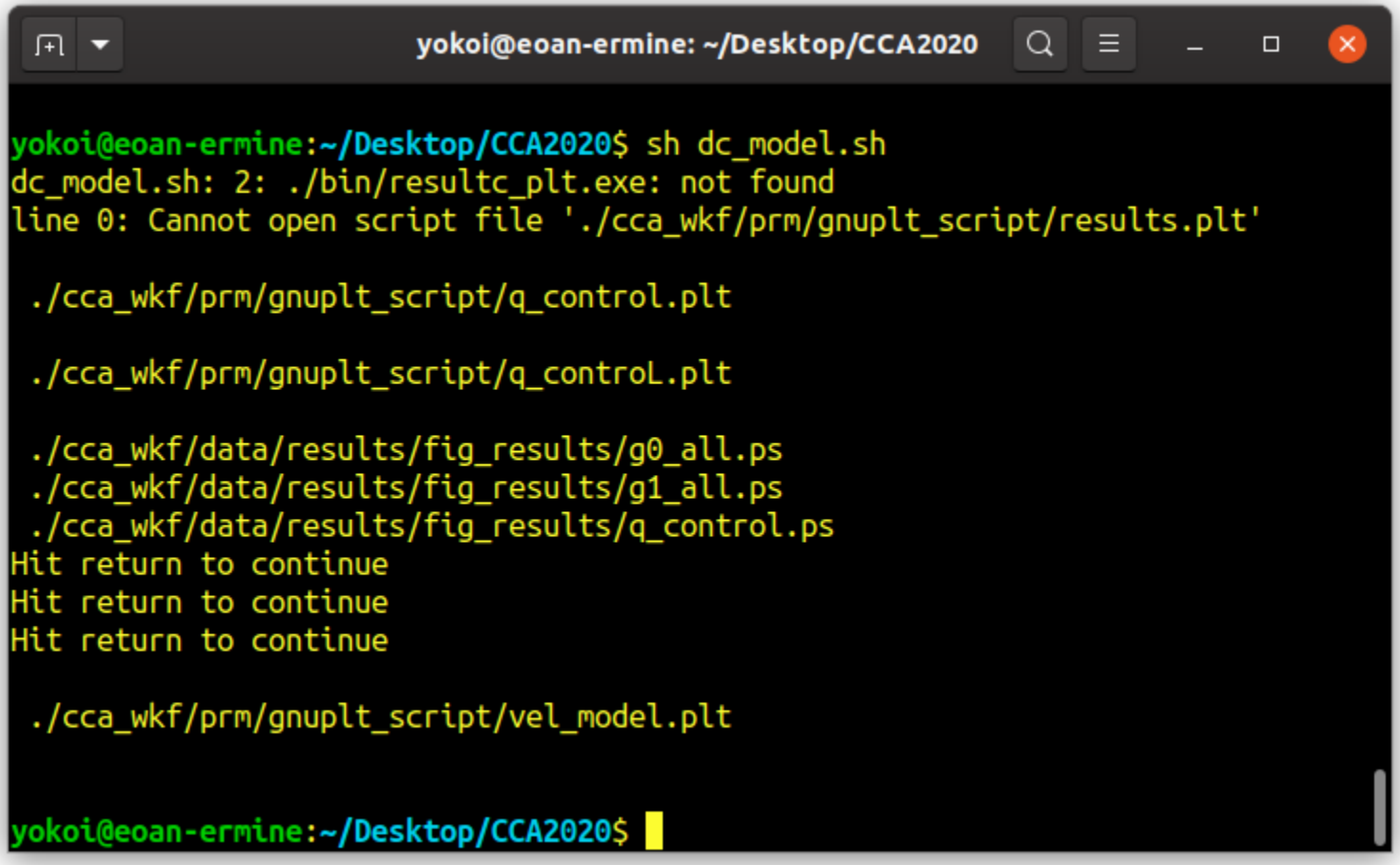


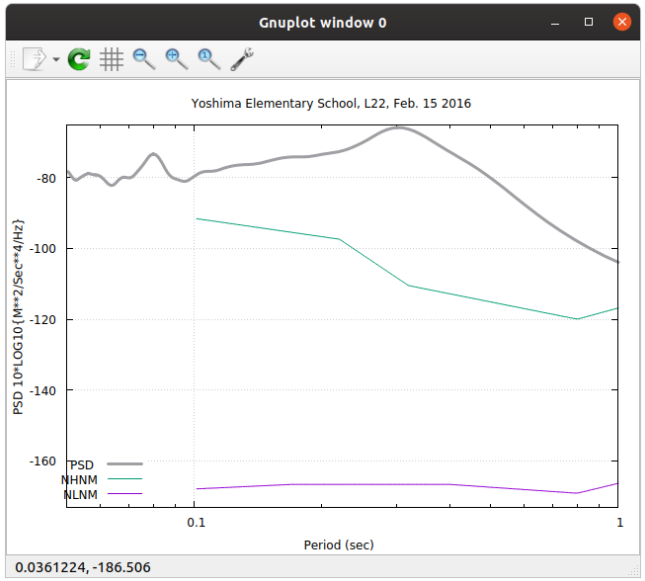


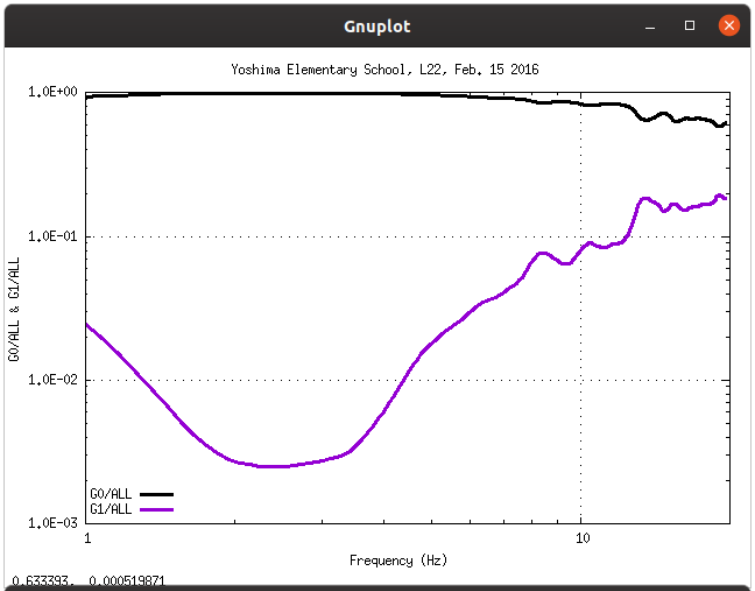


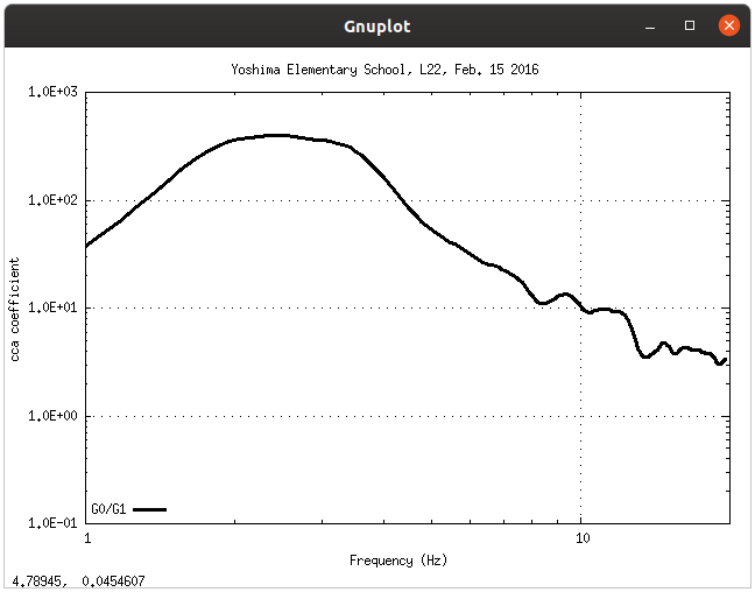


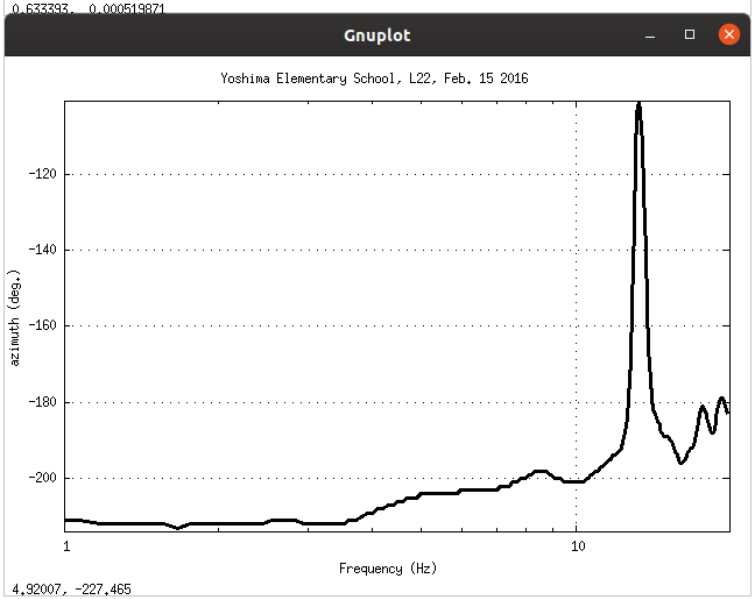
dc_model.sh: Execution

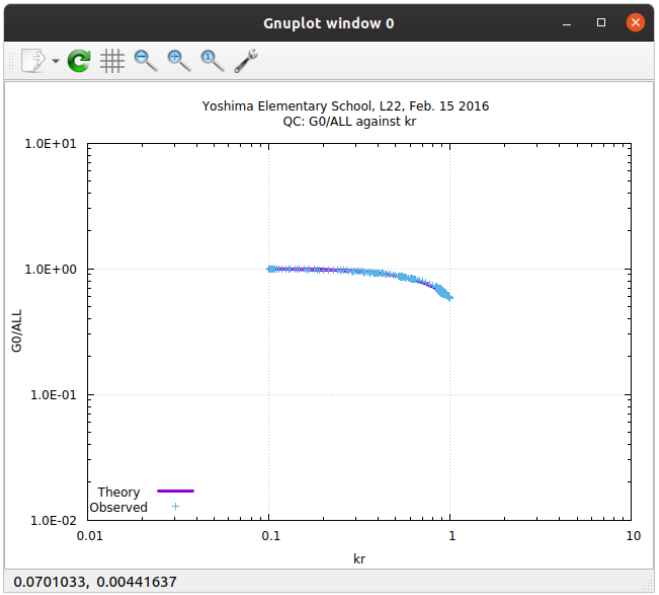


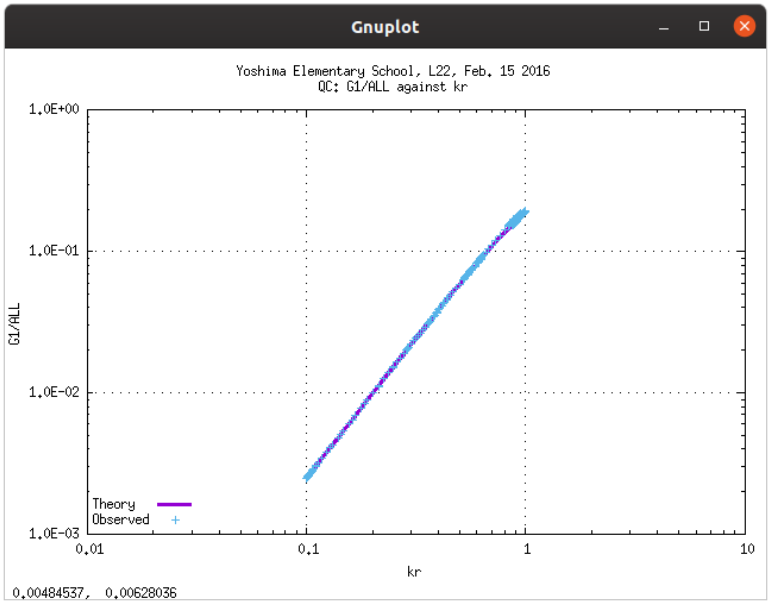

Gnuplot window 0
Yoshima Elementary School, L22, Feb. 15 2016
PSD 10*LOG10(M**2/Sec**4/Hz)
-80
-100
-120
-140
-160
PSD
NHNM
NLNM
0.1
1
Period (sec)
0.0361224, -186.506

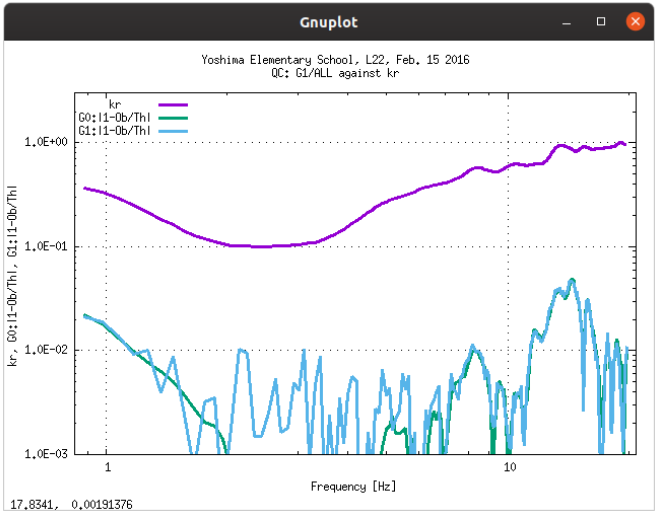

Gnuplot
Yoshima Elementary School, L22, Feb. 15 2016
G0/ALL & G1/ALL
1.0E+00
1.0E-01
1.0E-02
1.0E-03
G0/ALL
G1/ALL
1
10
Frequency (Hz)
0.633393, 0.000519871

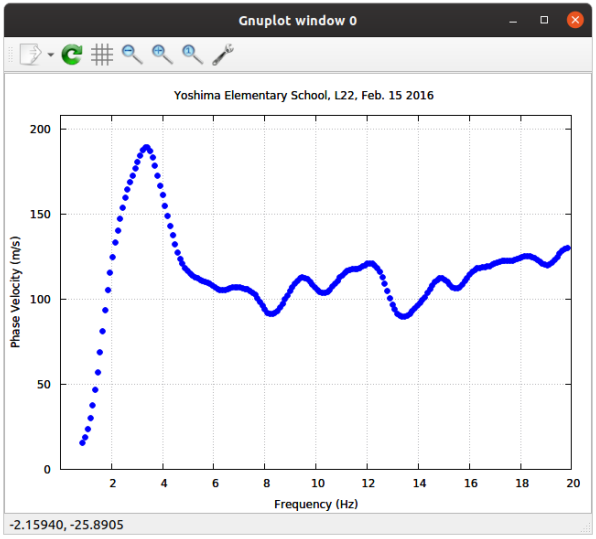

Gnuplot
Yoshima Elementary School, L22, Feb. 15 2016
cca coefficient
1.0E+03
1.0E+02
1.0E+01
1.0E+00
1.0E-01
G0/G1
1
10
Frequency (Hz)
4.78945, 0.0454607


Gnuplot
Yoshima Elementary School, L22, Feb. 15 2016
azimuth (deg.)
-120
-140
-160
-180
-200
1
10
Frequency (Hz)
4.92007, -227.465

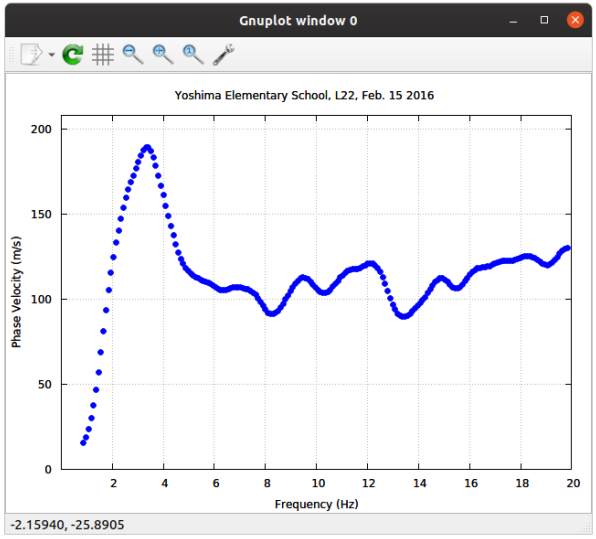

Gnuplot window 0
Yoshima Elementary School, L22, Feb. 15 2016
QC: G0/ALL against kr
1.0E+01
1.0E+00
1.0E-01
1.0E-02
G0/ALL
Theory
Observed
0.01
0.1
1
10
kr
0.0701033, 0.00441637


Gnuplot
Yoshima Elementary School, L22, Feb. 15 2016
QC: G1/ALL against kr
1.0E+00
1.0E-01
1.0E-02
1.0E-03
G1/ALL
Theory
Observed
0.01
0.1
1
10
kr
0.00484537, 0.00628036

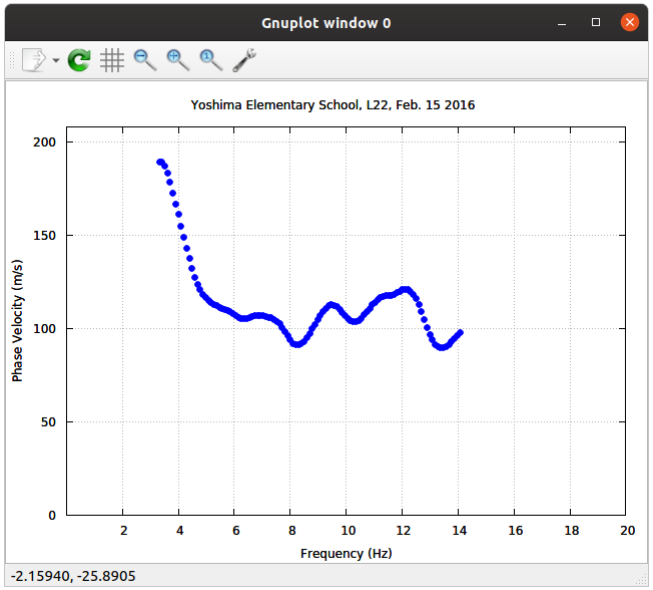

Gnuplot
Yoshima Elementary School, L22, Feb. 15 2016
QC: G1/ALL against kr
kr
G0:I1-Ob/Thl
G1:I1-Ob/Thl
1.0E+00
1.0E-01
1.0E-02
1.0E-03
kr, G0:I1-Ob/Thl, G1:I1-Ob/Thl
1
10
Frequency [Hz]
17.8341, 0.00191376


Gnuplot window 0
Yoshima Elementary School, L22, Feb. 15 2016
200
150
100
50
0
Phase Velocity (m/s)
2
4
6
8
10
12
14
16
18
20
Frequency (Hz)
-2.15940, -25.8905

Edit “dispersion.dat” using “gedit” or othrt text editor



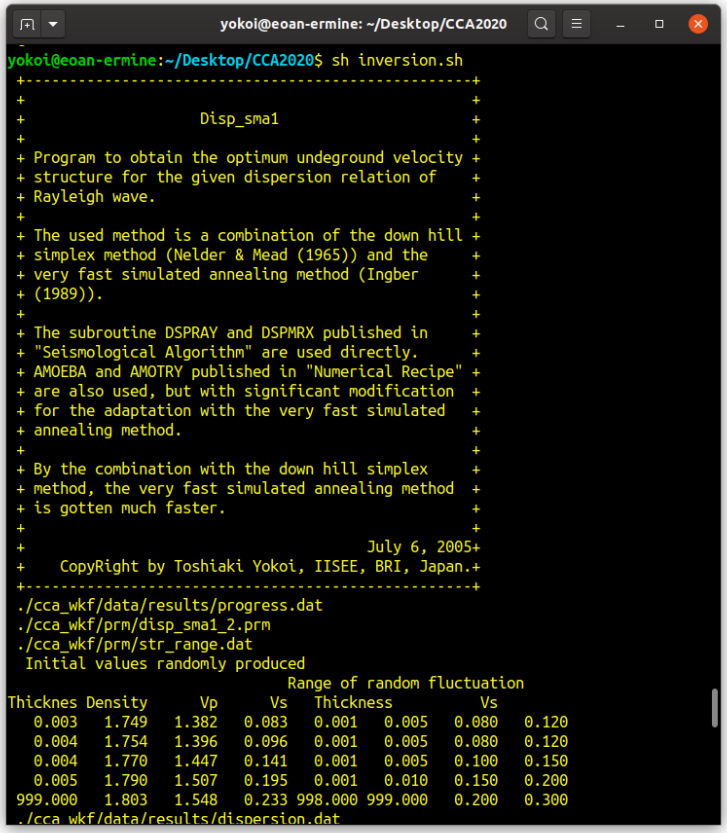
Then, redraw it using:
sh ./vel_model.sh



inversion.sh: Execution

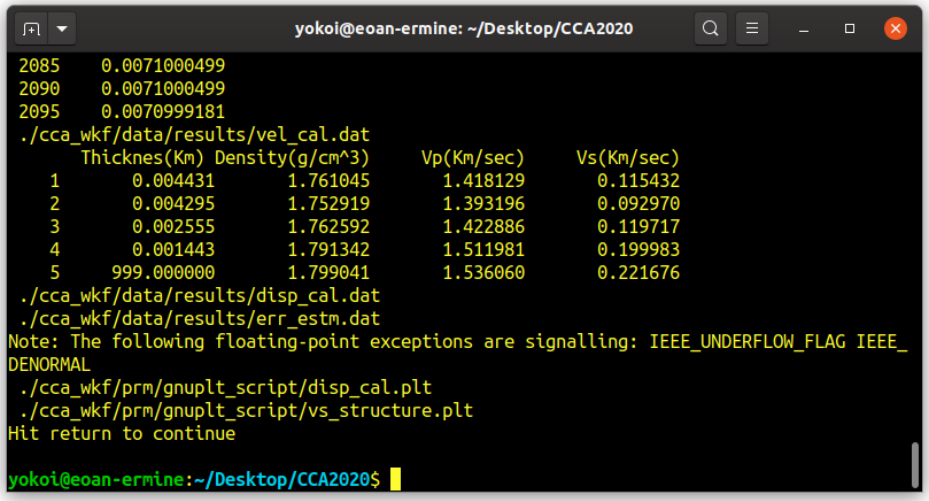
```
Yoshima Elementary School, L22, Feb. 15 2016
          5                          :IL(I5),Layer Number
1.9    1.5     0.001   0.005   0.08 0.12 :density,Vp,hmin,hmax,vmin,vmax
1.9    1.5     0.001   0.005   0.08 0.12 :density,Vp,hmin,hmax,vmin,vmax
1.9    1.5     0.001   0.005   0.10 0.15 :density,Vp,hmin,hmax,vmin,vmax
1.9    1.5     0.001   0.010   0.15 0.20 :density,Vp,hmin,hmax,vmin,vmax
2.0    1.70    998.0   999.0   0.20 0.30 :density,Vp,hmin,hmax,vmin,vmax
```

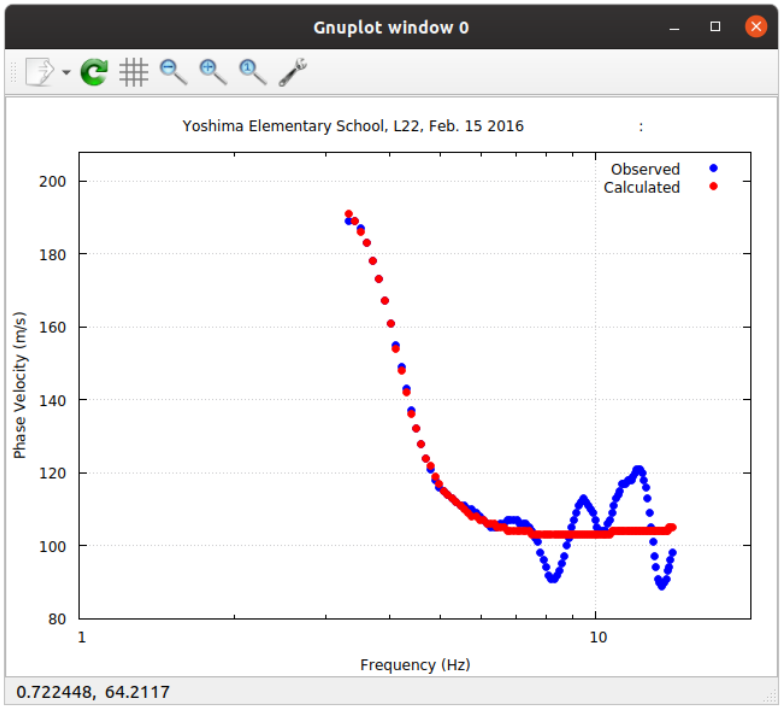
str_range.dat

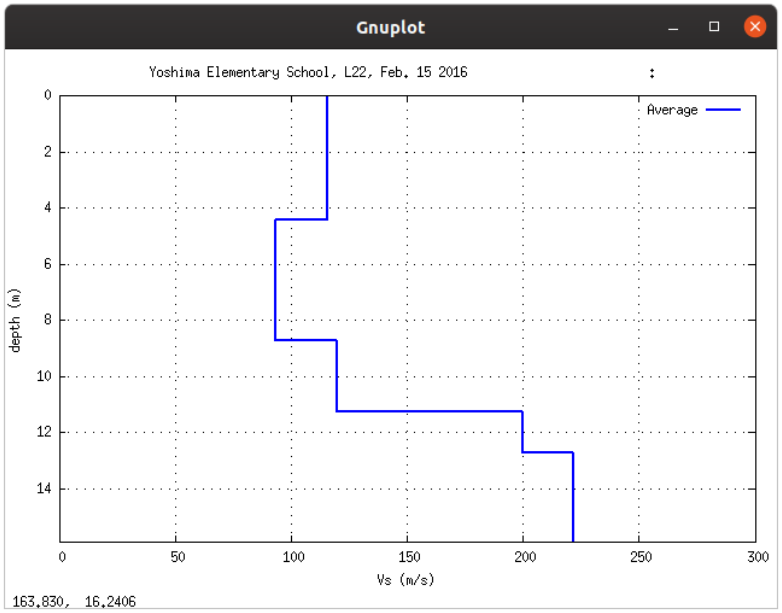



```
 1 1. 0.6 1.3 10000  5    :idum,t0,a,c,ntemp,j0
0.0071                    :eps0
   1    1                 :n_roh,n_vp
   1    0    1            :ini_flg,ndsp_flg,n_err
   0    1                 :kflg,jflg
   0    0                 :n_vs,n_th
```

disp_sma1_2.prm







Linear horizontal axis can be used by editing ./source/inv_plt.for .
Comment out the line 94 and 127:
c write(12,*)'set logscale x'

It is necessary to compile it again.

gfortran ./source/inv_plt.for –o ./bin/inv_plt.exe

Green circles can be plotted by setting n_mod=1 in pwrcrs3.prm.