BIDO version 3.2

1. Outline

BIDO is a computer program for identifying the phase velocities of surface waves by using microtremor array data. Version 3.2 is an updated version from 3.0. The core of the spectral analysis is unchanged from 3.0, so please refer to the manuals of the previous versions for an explanation of the data processing and algorithms. The differences from 3.0 include the operating environment for Windows users, small changes to the preprocessing programs, and the addition of a GUI (pvreader) for phase velocity reading. The changes to the preprocessing programs include the use of irregular arrays and the ability to concurrently use array data sets acquired at different times.

The following describes the parts of BIDO that have changed between 3.0 and 3.2.

2. Installation

2-1 Operating Environment

BIDO runs on a combination of shell scripts, the Fortran program, and Python scripts. The file processing through shell scripts includes fundamental Unix commands combined with "pipes". It works well on Linux and OSX that have Unix-like environments, while on Windows the processing takes more time.

Therefore, BIDO3.2 is primarily designed for use on Linux and OSX, and the UI (e.g., MSYS terminal) and executables for Windows that were included in BIDO until 3.0 have been removed in 3.2. However, BIDO can still run on Windows through WSL (Windows Subsystem for Linux), which has been installed in Windows as default since version 10.

BIDO cannot handle 2-byte characters (e.g., Japanese). Therefore, login names to an operation system, data directories, etc. should be one-byte alphanumeric characters, and 2-byte characters should not be used. This also applies to the path names of BIDO data and programs.

To use BIDO3.2, please install the following external software, which can be downloaded for free. They are required for both the core part of BIDO3.2 (i.e., spectral analyses) and the GUI for phase velocity reading (pvreader).

xterm   https://en.wikipedia.org/wiki/Xterm↵
gnuplot http://www.gnuplot.info/↵

On Linux/WSL (ubuntu), these applications can be installed by the following commands:

sudo apt install xterm

sudo apt install gnuplot-x11

2-2 Window Users

   WSL can be easily installed by searching for "Ubuntu" in the Microsoft Store. We recommend you search for appropriate documentation on WSL for more information. At the time of writing, WSL versions 1 and 2 and Ubuntu versions 18, 20, and 22 are available. The operation of BIDO was confirmed with the combination of WSL 1 and Ubuntu 20.
   If you are using WSL for the first time, after installing WSL, open a Linux terminal and prepare your environment by typing the following commands:

sudo apt update
sudo apt upgrade
sudo apt install gnuplot-x11
sudo apt install xterm

This may take some time (approximately 20–30 minutes or more). Then, follow the process described in Section 2-3.

2-3 Compiling yourself (OSX/general Linux users)
   To run the program on a general version of either Linux or OSX, please follow the instructions in this section. First, as well as in the case of WSL on Windows, prepare your environment by typing the following commands:

sudo apt update
sudo apt upgrade
sudo apt install gnuplot-x11
sudo apt install xterm

   If you have a development environment with a "gfortran" compiler installed on your PC, you can compile the BIDO source programs yourself. Other Fortran compilers have not been tested for compilation or operation.
   For Ubuntu, including an instance running on WSL on Windows, a development environment can be installed through the following steps. First, execute

sudo apt install build-essential
sudo apt install gfortran

Next, install the Python development environment (Python3) to run the GUI pvreader. Please install the Python modules as follows:

sudo apt install pip
pip install --upgrade pip
pip install numpy
pip install matplotlib
sudo apt install python3-tk

The development environment is now ready. You can compile the BIDO program as follows:

```
cd  BIDO3.2
./Install.sh  [hit return key]
```

3. How to use

The procedure for analyzing phase velocity is the same as for BIDO3.0. Specifically, start a terminal (xterm) and type the following command:

```
run.sh [hit return key]
```

or

```
run.sh data_path/param.sh [hit return key]
```

Note that "data_path" should be replaced with the path to the directory containing param.sh (as well as seism.d and the relevant data files).
Please refer to the attached manual for instructions on the use of the GUI (pvreader) for reading the phase velocities.

4. Analysis

Only the parts changed from BIDO3.0, concerning array analysis, are described in the following.

4-1. Accepting irregular arrays

If a circular array is described in seism.d, the calculation is performed as before, but in addition, processing for two-point SPAC (1-pair SPAC) (Cho, 2020), ESAC (Ohori et al., 2002), and nc-SPAC (Cho, 2019) is also performed. If an irregular array is described in seism.d, only the processing of two-point SPAC (1-pair SPAC), ESAC, and nc-SPAC methods is applied.
For example, if seism.d is written as follows, it describes a circular array with a radius of 1.5 m, as shown below, so CCA, ESAC, nc-CCA, nc-SPAC, two-point SPAC, and ordinary SPAC methods will be applied automatically.

```
#COMP 3 [1 ud only/ 2 horizontal (ns & ew) only/ 3 three components]
#DT   0.01 [Sampling time interval of the waveform data (s)]
  0.000000     0.000000     B09.d  1
  0.000000     0.001500     B10.d  0
 -0.001299    -0.000750     B11.d  0
  0.001299    -0.000750      B12.d  0
```

Array Geometry with radius 1.5 [m]
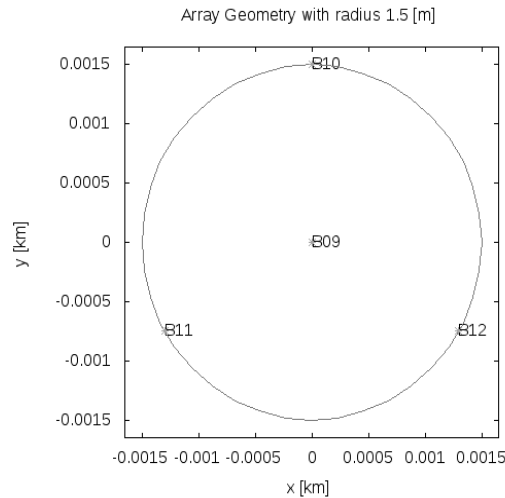
On the other hand, if seism.d is written as follows, it describes a linear array with sensors at 0, 3, 4, 5, 8, and 12 m along a straight line, not a circular array. Therefore, ESAC, nc-SPAC, and SPAC (1-pair SPAC) processing will be applied.

```
#COMP 1 [1 ud only/ 2 horizontal (ns & ew) only/ 3 three components]
#DT   0.005 [Sampling time interval of the waveform data (s)]
0.000000 0.000000 GS01.d 0
0.003000 0.000000 GS02.d 0
0.004000 0.000000 GS03.d 1
0.005000 0.000000 GS04.d 0
0.008000 0.000000 GS05.d 0
0.012000 0.000000 GS06.d 0
```


Array Geometry
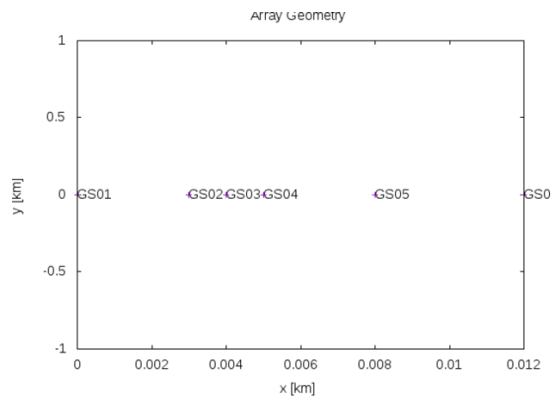
Note

The parameter "ID" is set to 1 to indicate to the BIDO program that the location is the "center point" of the array. For example, you can see that the ID of GS03.d is set to 1 in the above seism.d. The BIDO program will check whether all the sensors with ID=0 are on the circumference of a circle. If it is the case, BIDO regards the array as a "circular array". In contrast, if BIDO finds that any single sensor (with the ID of 0) is not on the circumference of a circle with the other sensors, it regards the array to be a "non-circular array". In other words, we can make the program regard an array to be "non-circular" by simply setting the ID of an appropriate sensor not at the center of a circle to 1 and the IDs of all the other sensors to 0.

4

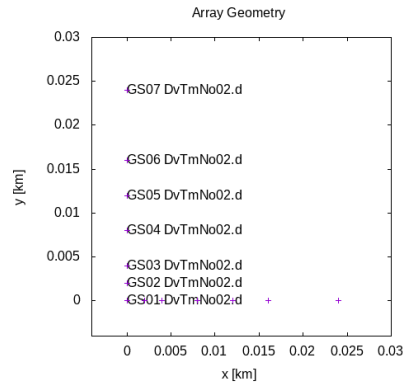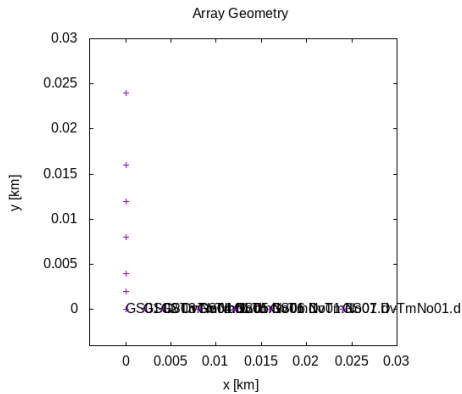## 4-2. Accepting array data sets recorded at different times

BIDO3.2 is designed to simultaneously process array data acquired at a single site but at different times (e.g., Morikawa et al., 2004; Cho 2023). For example, consider the following seism.d:

```
#COMP 1 [1 ud only/ 2 horizontal (ns & ew) only/ 3 three components]
#DT   0.005 [Sampling time interval of the waveform data (s)]
0.000000      0.000000      GS01GS11_DvTmNo01.d 1
0.002000      0.000000      GS02GS11_DvTmNo01.d 0
0.004000      0.000000      GS03GS11_DvTmNo01.d 0
0.008000      0.000000      GS04GS11_DvTmNo01.d 0
0.012000      0.000000      GS05GS11_DvTmNo01.d 0
0.016000      0.000000      GS06GS11_DvTmNo01.d 0
0.024000      0.000000      GS07GS11_DvTmNo01.d 0
0.000000      0.000000      GS01GS11_DvTmNo02.d 0
0.000000      0.002000      GS02GS11_DvTmNo02.d 0
0.000000      0.004000      GS03GS11_DvTmNo02.d 0
0.000000      0.008000      GS04GS11_DvTmNo02.d 0
0.000000      0.012000      GS05GS11_DvTmNo02.d 0
0.000000      0.016000      GS06GS11_DvTmNo02.d 0
0.000000      0.024000      GS07GS11_DvTmNo02.d 0
```

The corresponding locations of the geophones are shown in the following figure.
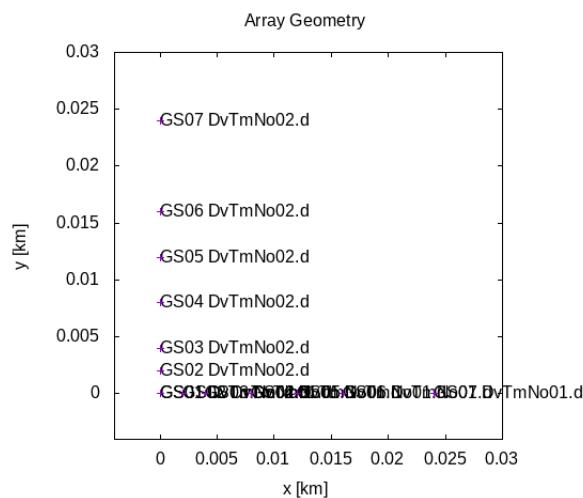
In this case, seism.d describes a microtremor array observation conducted by first installing a linear array of sensors at the locations 0, 2, 4, 8, 12, 16, and 24 m in the x-direction (see the left panel in the following figure) and then installing another linear array of the same specifications in the y-direction (see the right panel in the following figure) and conducting the observations again. (In these figures, an operating geophone is represented by a plus mark with the corresponding filename.)

Array Geometry

y [km]

0.03
0.025
0.02
0.015
0.01
0.005
0

+
+
+
+
+
+
+
GS0GS0GS0GS0GS06GS06_DvTmNo01GS07_DvTmNo01.d

0   0.005 0.01 0.015 0.02 0.025 0.03
x [km]

Array Geometry

y [km]

0.03
0.025
0.02
0.015
0.01
0.005
0

GS07 DvTmNo02.d

GS06 DvTmNo02.d

GS05 DvTmNo02.d

GS04 DvTmNo02.d

GS03 DvTmNo02.d
GS02 DvTmNo02.d
GS01 DvTmNo02.d   +   +

0   0.005 0.01 0.015 0.02 0.025 0.03
x [km]

In this way, each of the seven data files in the following two sets were recorded simultaneously:

GS01GS11_DvTmNo01.d
GS02GS11_DvTmNo01.d
GS03GS11_DvTmNo01.d
GS04GS11_DvTmNo01.d
GS05GS11_DvTmNo01.d
GS06GS11_DvTmNo01.d
GS07GS11_DvTmNo01.d

and

GS01GS11_DvTmNo02.d
GS02GS11_DvTmNo02.d
GS03GS11_DvTmNo02.d
GS04GS11_DvTmNo02.d
GS05GS11_DvTmNo02.d
GS06GS11_DvTmNo02.d
GS07GS11_DvTmNo02.d

You will note that the data file names with the same file ID, "_DvTmNoXX", are considered to be part of a set of array data recorded simultaneously. To concurrently process array data sets recorded at different times, add the following ID to the end of the filename:

_DvTmNoXX

BIDO will then automatically recognize whether they were simultaneously recorded. Note that this is case-sensitive. The horizontal bar "_" at the beginning is an underscore character. In BIDO3.2, when "_DvTmNo" is included in the data file name, the cross spectrum, coherence, and SPAC coefficients are evaluated only for data with the same "_DvTmNoXX" file name.
Here is another example seism.d, but rather more complicated:

#COMP 1 [1 ud only/ 2 horizontal (ns & ew) only/ 3 three components]
#DT   0.005 [Sampling time interval of the waveform data (s)]
 0.0000   -0.00750   GS01_DvTmNo23.d 0
 0.0000   -0.00625   GS02_DvTmNo22.d 0

6

```
 0.0000    -0.00500      GS03_DvTmNo21.d 0
 0.0000     0.00500       GS05_DvTmNo21.d 1
 0.0000     0.00625       GS06_DvTmNo22.d 0
 0.0000     0.00750       GS07_DvTmNo23.d 0
-0.00750    0.00000      GS01_DvTmNo43.d 0
-0.00625    0.00000      GS02_DvTmNo42.d 0
-0.00500    0.00000      GS03_DvTmNo41.d 0
 0.00500    0.00000      GS05_DvTmNo41.d 0
 0.00625    0.00000      GS06_DvTmNo42.d 0
 0.00750    0.000000      GS07_DvTmNo43.d 0
```
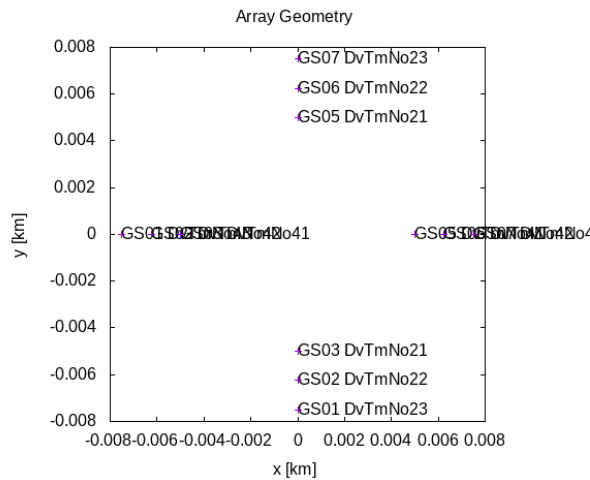

It should be apparent from the above seism.d what array observations were conducted. In this case, a measurement with a single two-point (1-pair) array of geophones was repeated six times for three distances and two orientations. The following six pairs of data files correspond to the data sets obtained by a single two-point array:

GS01_DvTmNo23.d and GS07_DvTmNo23.d
GS02_DvTmNo22.d and GS06_DvTmNo22.d
GS03_DvTmNo21.d and GS05_DvTmNo21.d
GS01_DvTmNo43.d and GS07_DvTmNo43.d
GS02_DvTmNo42.d and GS06_DvTmNo42.d
GS03_DvTmNo41.d and GS05_DvTmNo41.d

To explain this visually, the following panels show the deployment of the two-point arrays.

7
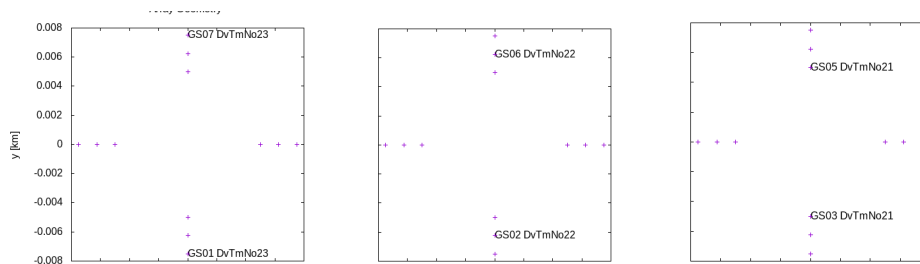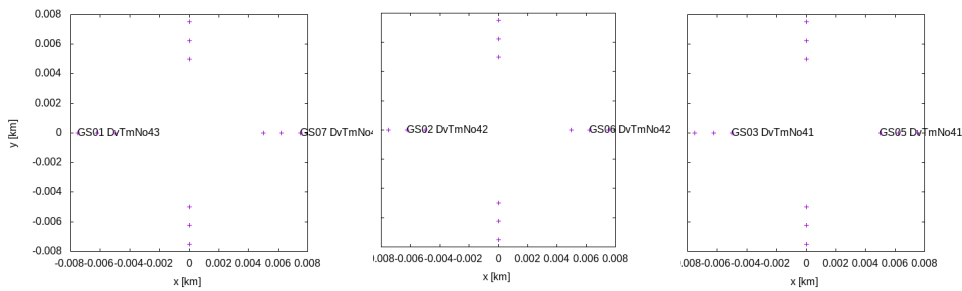
4-3. Clustering by array radius

As the number of sensors constituting a microtremor array increases, the variety of combinations of either two or three sensors within the array rapidly increases. For example, arrays of 5, 10, and 15 sensors have 10, 45, and 105 combinations of two sensors, respectively. As such, as the array gets large, it becomes impractical to manually check the processed results of either all individual two-point arrays or triangular arrays.

Therefore, in BIDO3.2, such subarrays extracted from an original large array are clustered based on array radius. For this purpose, all combinations of three pairs of two-sensor arrays (3-pair array, 3p array) that satisfy the following conditions are considered to be triangular arrays.

* The three inner angles of the triangle range from $40°$ to $140°$.
* The coefficient of variation of the side lengths is within 0.1. This is equivalent to being closer to an equilateral triangle than a right-angled isosceles triangle. Note that the side length is the distance between two sensors or the radius of a two-sensor array.

Next, the resulting 3p arrays are grouped using the agglomerative hierarchical clustering algorithm (Spath, 1980) based on the side length (i.e., radius). If the average radii differ by more than 10% within the same group, they are again subdivided into different groups.

The SPAC coefficients and the corresponding radii involved in a certain group are averaged to be the representative SPAC coefficients and radius of the group, which is assigned the file ID "3pAPX_n**CL**". "3p" means 3 pairs of two-point SPAC coefficients with the same sensor-to-sensor distance (array radius). "APX" stands for "approximation", meaning that the array radii are approximately the same. "n**" indicates the number of 3p arrays involved in the group, and "CL**" is the cluster (group) number.

A similar grouping scheme is taken for cross/L-shaped arrays. To be specific, all combinations of two pairs of two-sensor arrays (2-pair array, 2p array) that satisfy the following conditions are considered as a cross/L-shaped array.

* The angle between the two axes of a cross/L-shaped array ranges from $40°$ to $140°$.
* The coefficient of variation of the axis lengths of a cross/L-shaped array is within 0.1.

The resulting 2p arrays are grouped using an agglomerative hierarchical clustering algorithm based on the average axis length (i.e., radius). If the radii differ by more than 10% within the same group, they are again subdivided into different groups.

The SPAC coefficients and the corresponding radii for a group are averaged to give the representative SPAC coefficients and radius of the group. These results

are assigned the file ID "2pAPX_n**CL**", where the "2p" means two pairs of two-point SPAC coefficients with the same radius.

4-4. Applying a zero-crossing method

BIDO3.2 allows phase velocity identification based on the zero-crossing method (Aki, 1957; Ekström et al., 2009; Cho et al., 2021; Cho 2023). This method can be applied with the phase velocity reading GUI (pvreader), which has been added from this version. The algorithm for automatic reading of zero-crossing points of SPAC spectra is described in Cho (2023). Please refer to the separate attached manual for details.

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

Cho, I., 2019, Compensating for the impact of incoherent noise in the spatial autocorrelation microtremor array method, B*ull. Seismol. Soc*. Amer. **109**, 199-211, doi:10.1785/0120180153.

Cho, I., 2020, Two-sensor microtremor SPAC method: potential utility of imaginary spectrum components, *Geophys. J. Int*., 1735-1747, doi:10.1093/gji/ggz454.

Cho, I. 2023, Spatial autocorrelation method for a simple microtremor array survey at a rock/stiff-soil site, *J. Geophys. Int.*, in press.

Cho, I., S. Senna A. Wakai, K. Jin and H. Fujiwara, 2021, Basic performance of a spatial autocorrelation method for determining phase velocities of Rayleigh waves from microtremors, with special reference to the zero-crossing method for quick surveys with mobile seismic arrays, *Geophys. J. Int.,* **226**, 1676-1694*,* doi:10.1093/gji/ggab149.

Ekström, G., Abers, G. A. & Webb, S. C., 2009. Determination of surface-wave phase velocities across US Array from noise and Aki's spectral formulation, *Geophys. Res. Lett*., **36**, L18301. doi:10.1029/2009GL039131

Morikawa, H., Sawada, S. & Akamatsu, J., 2004. A method to estimate phase velocities of Rayleigh waves using microseisms simultaneously observed at two sites, *Bull. Seismol. Soc. Am*, **94**, 961–976. doi: 10.1785/0120030020.

Ohori, M., Nobata, A. & Wakamatsu, K., 2002. A comparison of ESAC and FK methods of estimating phase velocity using arbitrarily shaped microtremor arrays, *Bull. Seismol. Soc. Am.,* **92**, 2323–2332. doi:10.1785/0119980109

Spath, H., 1980, Cluster analysis algorithms: for data reduction and classification of objects, Chapter 4, John Willey & Sons, N.Y., 226p.

Contact:

Ikuo Cho (**ikuo-chou@aist.go.jp**)

GUI for phase-velocity reading: pvreader

1. Outline

Pvreader is a graphical user interface (GUI) for visual reading of phase velocities obtained by BIDO3.2.

2. How to use

Launching dialog buttons

Start a terminal (xterm) and execute the following command.

pvreader.sh

The following buttons will appear. The buttons may look different depending on your computer environment.
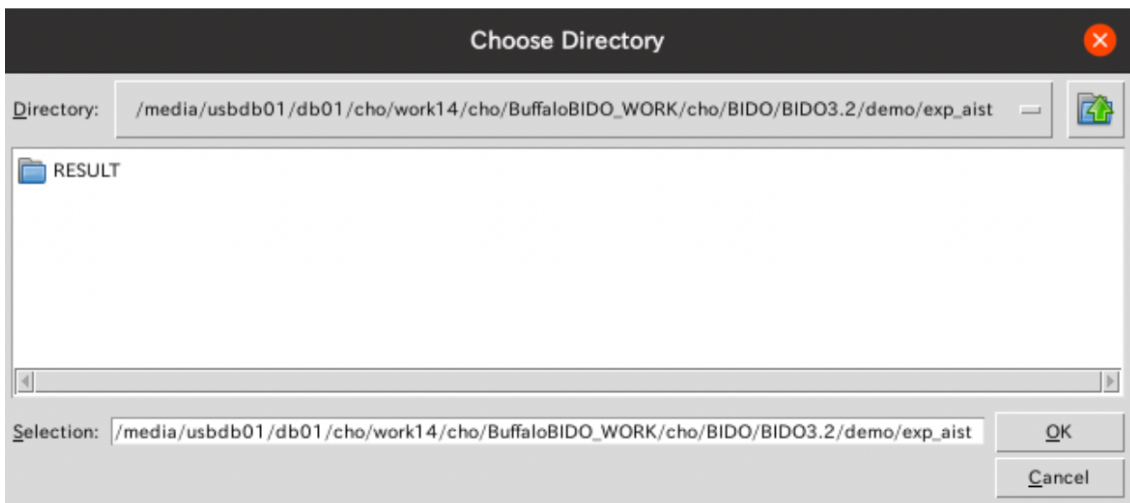

The meanings of the buttons are as follows.
DataDir: Specify the data directory where the calculation results by
        BIDO3.2 are stored.
RunShell: Execute pvreader.
Quit: Quit.

Selecting directories

First, click the "DataDir" button, select the data directory that contains the BIDO3.2 calculation results using the dialog box, and click "OK".

When you have multiple data directories, such as for the case where you have multiple array measurements performed at a single site that you want to evaluate together, select all the directories by repeating the above data directory selection process for each. That is, after one dialog box disappears, click the "DataDir" button again, select the next data directory, and click "OK", repeating this process until all data directories have been selected.

Launching pvreader GUI

Click "RunShell". A dialog such as
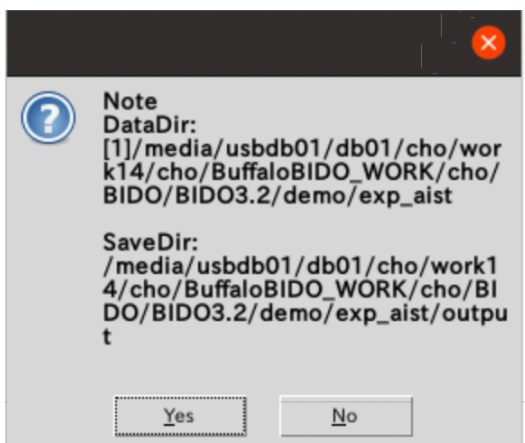
will appear to show the "DataDir" and a directory "SaveDir" that specify where the results of reading by pvreader are stored.

If you click "No", the program will terminate. If you click "Yes", xterm will then start the preparation of the phase velocity plots. Pre-

processing, such as automatic reading of the zero-crossing points, will also be performed in this step.

Adjusting the axis ranges

After a while, dispersion-curve graphs will appear, as shown below. You can adjust the visibility of the graph by selecting the axis ranges and the data to be plotted. After inserting values for the axis ranges, click "Refresh" button to update the plots.
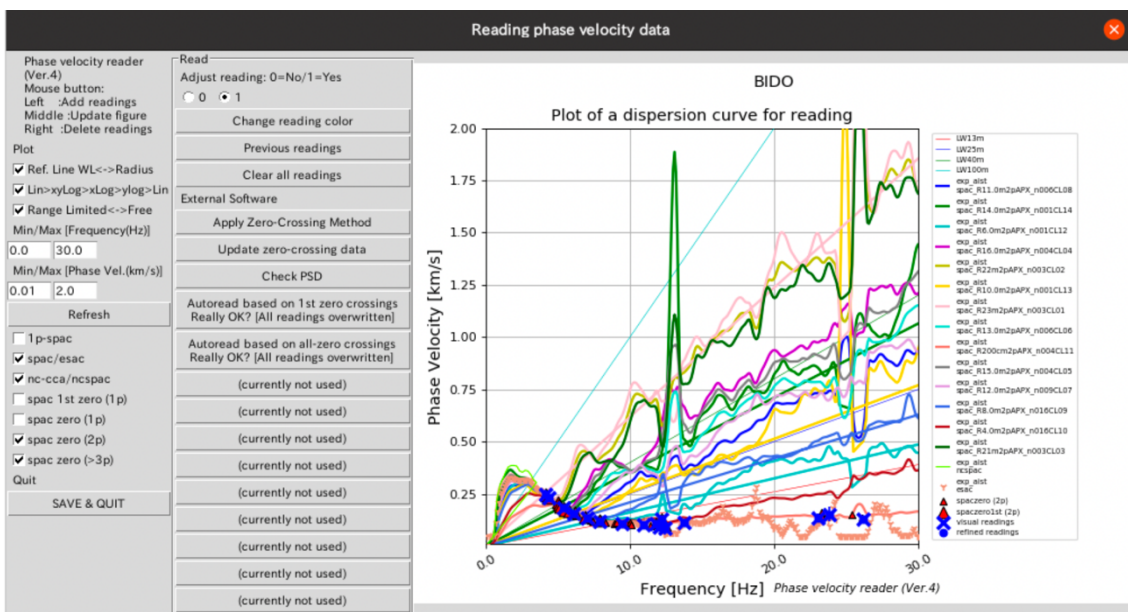
Selecting the data for the plot

   The following checkboxes allow you to select and deselect data to be plotted.

   "1p-spac" is 1-pair SPAC, representing a two-point array consisting of two sensors, or a single pair of sensors, the minimum array for the SPAC method. BIDO3.2 applies the 1-pair SPAC method to all pairs of sensors that constitute a microtremor array, and therefore generally has a large number of processing results (see figure below).
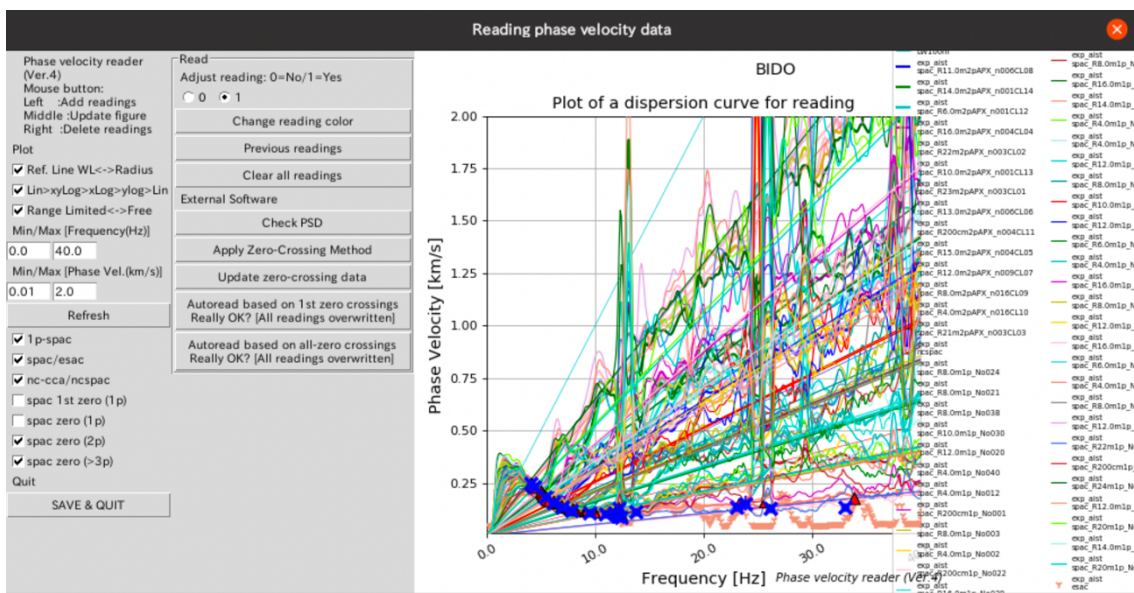
   "spac 1st zero (1p)" obtains phase velocities by applying the zero-crossing method to the first zero-crossing point of the 1p-spac coefficients.
   "spac zero (1p)" obtains phase velocities by applying the zero-crossing method to all zero-crossing points of the 1p-spac coefficients.
   "spac 1st zero (2p)" obtains phase velocities by applying the zero-crossing method to 2-pair arrays. This item corresponds to the analysis results of BIDO3.0 and 3.2 with a file ID of "spac_Rxx2p" (BIDO3.0 manual, Appendix 1-2).

13

"spac 1st zero (> 3p)" obtains phase velocities by applying the zero-crossing method to 3-pair arrays or arrays consisting of a larger number of pairs. This includes ordinary triangular arrays.

Initial automatic readings

Blue crosses appear on the initial panel when pvreader starts up. These are automatic phase-velocity readings, obtained based on the zero-crossing method and other phase-velocity determining methods. Each method has a slightly different reliable wavelength range and the characteristics of each method are taken into account in the automatic reading. These automatic readings are reasonably informative in general.

Note, however, that the automatic reading is an optional assistance function and might be incorrect due to noise, and the results should be verified visually by referring back to the SPAC coefficients (see "Applying zero-crossing method").

Visual (manual) readings

A reading mark can be added to the graph with a left-click and removed with a right-click.

pvreader will also perform "automatic adjustment of reading results". For example, in the figure below, a point added by a left-click is marked with an X in the red square, but the actual reading point, represented by the small blue circle, is somewhat different from the X point. The reading is automatically adjusted to a weighted average value of the multiple neighboring phase velocity results, based on the wavelength relative to the array size. Please be aware of this function and ensure you have accounted for all the automatically adjusted points in your visual analysis.

Plot of a dispersion curve for reading

Phase velocity reader (Ver.3.3)
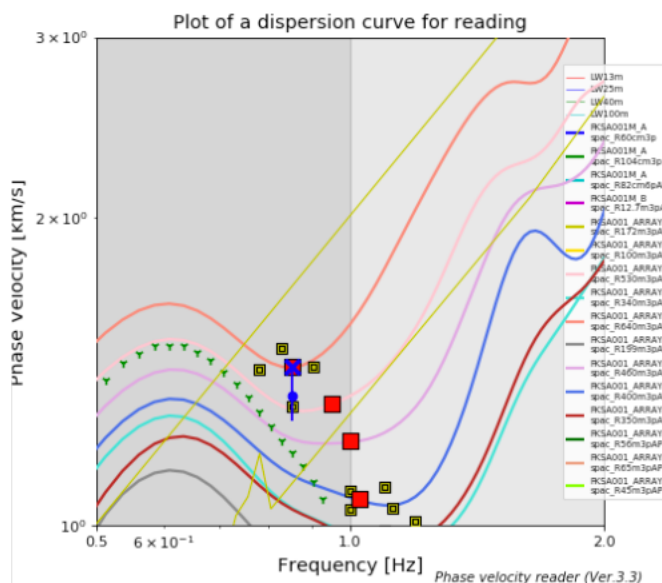
The "automatic adjustment of reading results" function can be turned off by setting the "Adjust reading" radio button in pvreader to 0. If this is set to 0, the auto-adjustment function will be turned off only for the next piece of data; if you want to read all the data with auto-adjustment off, you must select 0 with the radio button for each data point.
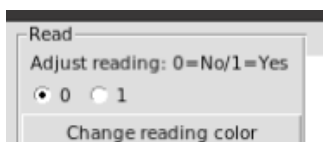

With the function turned off, the reading marked with the X above will be aligned with the red square, as shown below.

Phase Velocity [FKSA001_NIED]

Plot of a dispersion curve for reading
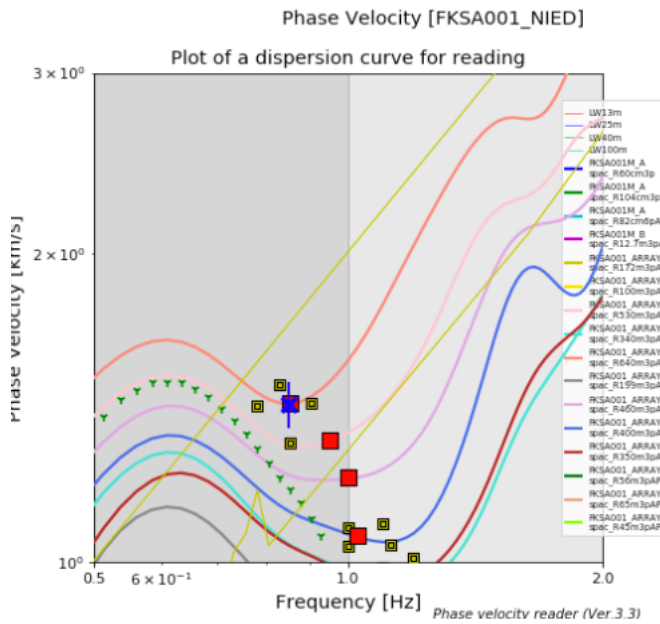
Function of each button

"Check PSD"
Press this button to plot the PSD. It may be helpful to estimate the SNR. The plot is displayed with gnuplot. You can exit the plotting screen by clicking the quit button on the toolbar or typing "q" on the graph.
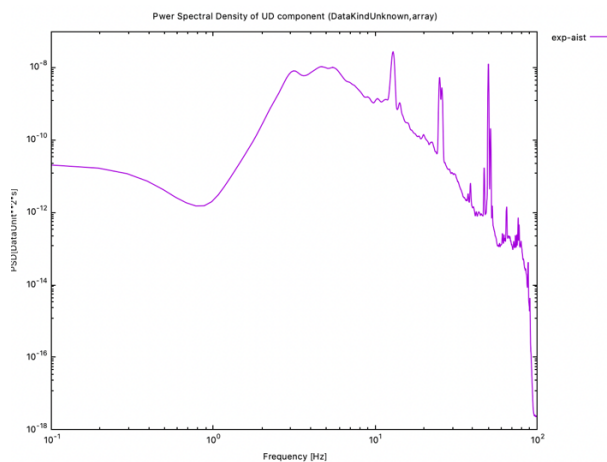
"Apply Zero-Crossing Method"
Press this button to launch a reader of the zero-crossing points of the SPAC coefficient spectra. The details are described later.

"Update zero-crossing data"

After/during a reading with the "Applying Zero-Crossing Method", press this button to update the plot of the phase velocity corresponding to the zero-crossing point (red squares, red triangles, yellow squares).

"Auto read based on 1st zero crossings"
Converts the first zero-crossing points in the SPAC coefficients of 2p (2-pair), 3p (3-pair) arrays to manual readings of phase velocity. If the readings are scattered over a small frequency range, the readings within the frequency range will be averaged. Note that all previous readings will be overwritten by pressing this button.

"Auto read based on all zero crossing"
As for "Auto read based on 1st zero crossing" button, but using both the first and later zero-crossing points as conversion information. The later zero-crossing points are weighted inversely proportional to the zero number so that the later they are, the smaller the weight. Note that all previous readings will be overwritten by pressing this button.

"SAVE & QUIT"
Exit pvreader. After saving and quitting, the reading results can be seen by opening the file index.html created in SaveDir with an appropriate browser (e.g., Chrome, Firefox). The phase velocity reading results are also downloadable from a link on this page.
  Individual reading results are stored in the following file in SaveDir (these have the same contents as seen in the browser).
*info.d Information on DataDir and SaveDir
*vr.d   Readings of phase velocities in the following format: frequency [Hz], phase velocity [km/s], standard deviation [km/s], no. of data, read_adjust_ID. The no. of data parameter matches nseg_segave in BIDO (if NROBUST4AVERAGE_INC is not specified in src/PARAM.h when compiling BIDO). The "standard deviation" in the file vr.d represents a value associated with the nearest phase-velocity data at the reading point. Interpolated values may be assigned to readings for which there are no data, in which case they are not strictly correct. In addition, since phase velocities obtained by several different methods are plotted here, the variances of the results obtained by different methods should also be evaluated. Therefore, the errors presented here should be understood only as a simple convenient indicator.
  *read_adjust_ID: 0/1. This parameter is written as 0 either for the initial readings or when the reading was executed by the user with the "automatic adjustment of reading results" function turned off.

*spaczeros.d Information on the zero-crossing points of the SPAC coefficients in the following format: frequency [Hz], phase velocity [km/s], zero number, directory file, array radius [km].

*image.png Snapshot of pvreader panel.

*png   png files of the SPAC coefficients and PSDs with readings.

## GUI for zero-crossing method: SPAC zero reader

Clicking on the "Applying zero-crossing method" button on pvreader launches a GUI for visually reading the zero-crossing points of the SPAC-coefficient curve, as in the following:
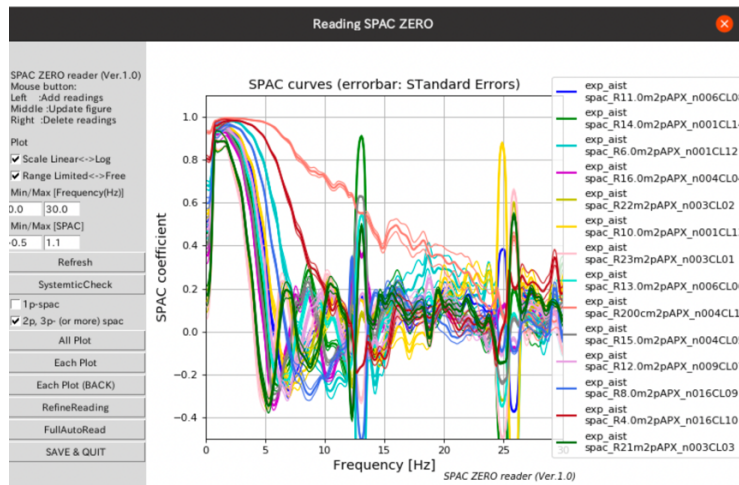
The initial plot shows all SPAC coefficients of the 1p-spac or 2p-, 3p- (or more) spac (i.e., selected according to the checkbox).

Visual readings

Click "Each Plot" and the individual SPAC coefficients will be plotted for reading.
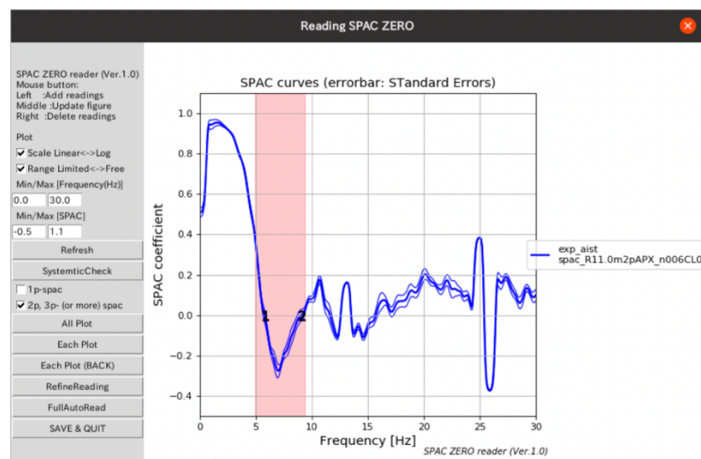
The SPAC coefficients have already been automatically read, but if the reading results are not correct, you may need to reread them visually. Left-click to read up to 9 SPAC coefficients (right-click to delete). The bold line and thin lines represent the average and

standard error, respectively. Use this error for reference in assessing the reliability (significance) that the SPAC coefficient curve crosses the zero line.
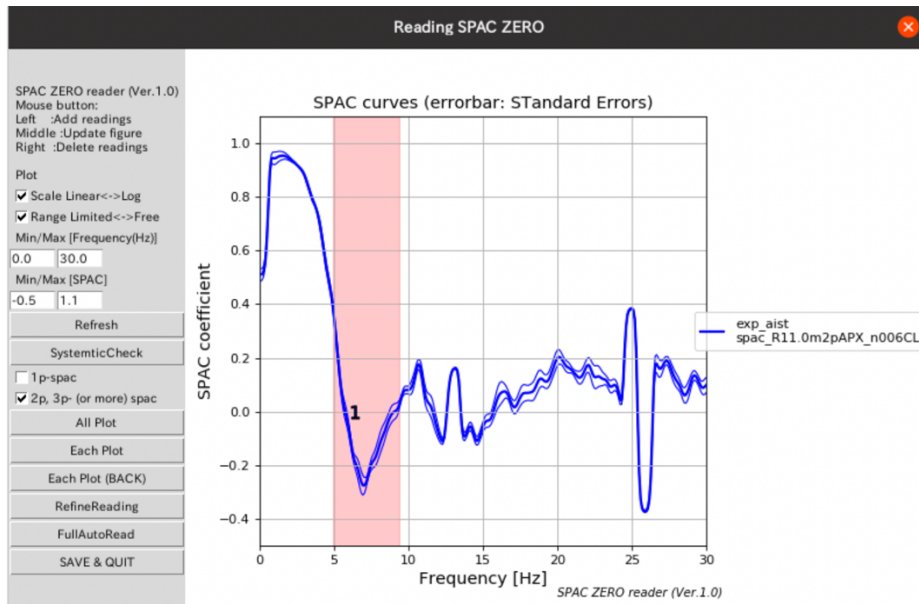
If a reading is off position, as shown in the above figure, press the "RefineReading" button to automatically adjust the readings. If the "FullAutoRead" button is pressed, the result of the automatic reading will be reproduced.

When the reading is finished, click "Each Plot" again to proceed to the next SPAC coefficient. If you want to start over, click "Each Plot (BACK)" to return to the previous plot. Some SPAC coefficients are labeled with "APX" and others with "CL (cluster)". The meanings of these labels are described in Section 4.3 of the BIDO3.2 manual.

Automatic reading

The zero-crossing points based on the automatic reading criterion of Cho (2023) are plotted along with a number representing the zero number of the SPAC coefficients. Please refer to these automatic reading results for visual checks/corrections.

Note that the automatic readings are based on data that is smoothed more strongly than the plotted SPAC coefficients, to stabilize the automatic analysis (i.e., reading). Therefore, the plot of the SPAC coefficients may deviate from the automatic reading results. If this is the case, please correct the reading results through a visual inspection. Since the SPAC coefficients plotted here are the original, greater confidence should be placed in these data.

Reading assistance function (red band)

   A vertical red band indicates the range where the first zero-crossing point is expected to lie. This red band is generated by the following procedure.
1) The SPAC coefficients are divided into two groups, $r > r_0$ and $r < r_0$, with r_0 being the radius of the SPAC coefficient to be read.
2) The frequencies corresponding to the first zero-crossing point of the $r > r_0$ group are averaged to obtain the minimum frequency of the red band. (Note that these frequencies are calculated using already-read data.) When averaging, a weight r_0/r is applied, such that the weights are smaller for larger radius deviations.
3) The frequency corresponding to the first zero-crossing point of the group with $r < r_0$ is also averaged to obtain the maximum frequency of the red band. When averaging, a weight r/r_0 is applied, such that the weights are smaller for larger radius deviations.
   Note that the above process is based on the assumption that the first zero-crossing point moves toward higher frequencies as r decreases. This is equivalent to assuming normal-dispersed phase velocities.

Reading assistance function（"SystematicCheck" button）

   Pressing the SystematicCheck button will plot all the SPAC coefficients (2-p, 3-p arrays) in descending order of r. Clicking the "x" button on the toolbar or typing "q" on the graph will exit a plot screen and proceed to the next plot. Like in an animation, you can see how the zero-crossing readings gradually shift to the right (high frequency side) as r decreases. On top of the SPAC coefficients panel, the PSD of the observation used to evaluate the SPAC coefficients is also plotted. This may help to detect unnatural zero-crossing points due to machine-induced tremors (monotonic peaks), etc., to identify misreadings.
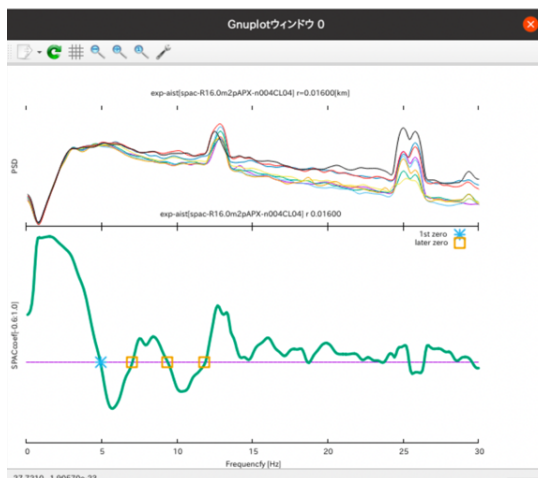   For example, in the following snapshot, the PSD in the upper panel has monotonic peaks around 12.5 Hz and 25 Hz. Correspondingly, the SPAC coefficients in the lower panel have unnatural pulses. The automatic reading reads the zero-crossing points associated with these peaks as the later zero-crossing points.
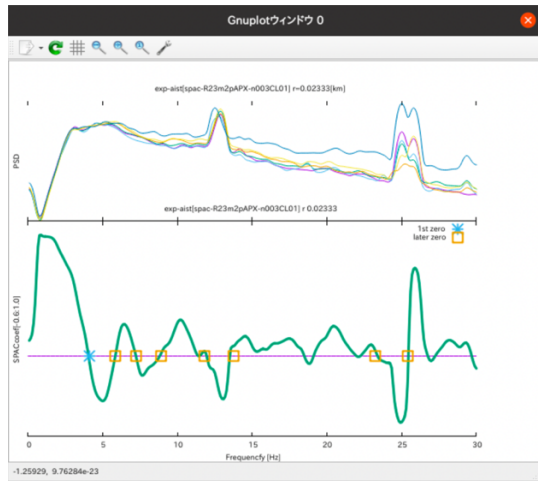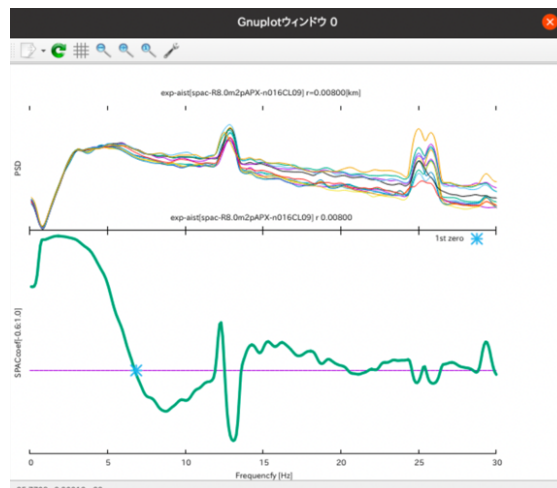
These readings can then be excluded by a visual inspection.
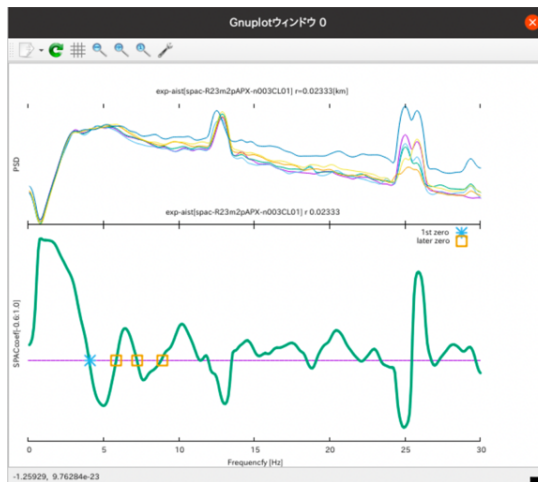
As such, proceeding with the reading so that the zero-crossing points change systematically would be generally successful.

Cho, I. 2023, Spatial autocorrelation method for a simple microtremor array survey at a rock/stiff-soil site, *J. Geophys. Int.*, in press.