



ELSEVIER

Contents lists available at ScienceDirect

Computers & Geosciences

journal homepage: www.elsevier.com/locate/cageo

Case Study

APASVO: A free software tool for automatic P-phase picking and event detection in seismic traces ☆, ☆ ☆



José Emilio Romero*, Manuel Titos, Ángel Bueno, Isaac Álvarez*, Luz García, Ángel de la Torre, M^a Carmen Benítez

Department of Signal Theory, Telematics and Communications, CITIC-UGR, University of Granada, ETS Ingenierías Informática y de Telecomunicación, Periodista Daniel Saucedo Aranda, s/n, 18071 Granada, Spain

ARTICLE INFO

Article history:

Received 15 May 2015

Received in revised form

4 February 2016

Accepted 5 February 2016

Available online 18 February 2016

Keywords:

Seismology

Software

Automatic

P-phase picking

Detection

STA-LTA

AMPA

Autoregressive models

ABSTRACT

The accurate estimation of the arrival time of seismic waves or picking is a problem of major interest in seismic research given its relevance in many seismological applications, such as earthquake source location and active seismic tomography. In the last decades, several automatic picking methods have been proposed with the ultimate goal of implementing picking algorithms whose results are comparable to those obtained by manual picking. In order to facilitate the use of these automated methods in the analysis of seismic traces, this paper presents a new free, open source, software graphical tool, named APASVO, which allows picking tasks in an easy and user-friendly way. The tool also provides event detection functionality, where a relatively imprecise estimation of the onset time is sufficient. The application implements the STA-LTA detection algorithm and the AMPA picking algorithm. An autoregressive AIC-based picking method can also be applied. Besides, this graphical tool is complemented with two additional command line tools, an event picking tool and a synthetic earthquake generator. APASVO is a multiplatform tool that works on Windows, Linux and OS X. The application can process data in a large variety of file formats. It is implemented in Python and relies on well-known scientific computing packages such as ObsPy, NumPy, SciPy and Matplotlib.

© 2016 Elsevier Ltd. All rights reserved.

1. Introduction

The precise determination of the arrival time of seismic waves or picking is a major problem in many seismological applications, such as tomographic inversion of active-source data (Ibáñez et al., 2008; García-Yeguas et al., 2012), hypocenter location or earthquake early warning systems (Horiuchi et al., 2005; Nakamura et al., 2009). Traditionally, this task has been performed manually by trained personnel. This manual approach has, however, several drawbacks. The analyst must often cope with the processing of a large amount of seismic data, which makes picking a tedious, time consuming task and therefore prone to human errors (Zhang et al., 2003). In addition, the results depend largely on the experience of the analyst (Küperkoch et al., 2011; Ross and Ben-Zion, 2014), and their quality can be obscured by several factors such as

background and non-stationary noise processes from diverse sources such as waves, tides, wind, microtremors or human activity (Havskov and Alguacil, 2004).

Another significant shortcoming of manual picking is the difficulty of dealing with the systematic processing of huge amounts of available seismic data incoming from large-scale monitoring, a quantity that grows as the number of seismometers installed worldwide increments every day. The occurrence of earthquakes grows exponentially as the considered magnitude decreases (Gutenberg and Richter, 1956). Thus, while approximately 150 000 earthquakes occur annually with magnitude sufficient to be perceived by the population (Gutenberg and Richter, 1965), a much larger number of small microearthquakes go unnoticed to people daily but are recorded by monitoring devices. Analysis of this microseismicity would be useful to address the study of the seismicity of a region in a more reliable way taking into account only the earthquakes of a greater magnitude, whose occurrence is much lower. In practice, however, addressing the study of such microseismicity using a manual approach becomes a difficult task due to the aforementioned problems. The volume of data to be examined can be quite large, and the small magnitude of the events makes them difficult to distinguish from background noise, even for an experienced analyst.

*Source code and binary releases are hosted on APASVO's GitHub repository: <https://github.com/cageo/Romero-2016>.

**APASVO is also available from Python PyPi repository: <https://pypi.python.org/pypi/APASVO>.

* Corresponding authors.

E-mail addresses: jemromerol@gmail.com (J.E. Romero), isamaru@ugr.es (I. Álvarez).

Limitations of the manual approach make picking a task whose automation is desirable, and have led to the development of many automatic picking algorithms using a variety of different approaches. We have to separate detection methods, where a relatively imprecise estimation of the onset time is sufficient, and picking algorithms, aimed at providing a precise determination of the phase arrival time (Baer and Kradolfer, 1987). The classical *short term average over long term average* method, or STA-LTA (Allen, 1978, 1982), based on estimating the ratio between the average energy of two windows of different length, has been used extensively as a detection algorithm. The naive version of this method can be easily implemented in an efficient way, yet it may have a poor performance as a picking method (Alvarez et al., 2013). Besides, this method provides a great flexibility, both in its parameter configuration (Trnkoczy, 2002) and accepting a significant number of possible modifications (Withers et al., 1998). Several other more sophisticated picking methods have been proposed, including direct use of an envelope function with a dynamic trigger level (Baer and Kradolfer, 1987), higher-order statistics (Saragiotis et al., 2002; Küperkoch et al., 2010; Nippres et al., 2010), polarization approaches (Montalbetti and Kanase-wich, 1970; Kurzon et al., 2014), local-maxima distribution (Panagiotakis et al., 2008), wavelet-based methods (Zhang et al., 2003) or even pattern recognition methods (Tong, 1995) such as neural networks (Dai and MacBeth, 1995; Gentili and Michelini, 2006) or computer vision inspired methods (Joswig, 1990).

Autoregressive models have been successfully applied for picking (Takanami and Kitagawa, 1988, 1991; Leonard and Kennett, 1999; Kitagawa et al., 2001; Kurz et al., 2005). The standard approach, commonly named AR-AIC, is based on estimating autoregressive models for both background noise and incoming seismic phase and then finding the phase arrival time by just looking for the time point that attains the minimum value of the Akaike Information Criterion (AIC) of the locally stationary AR model (Akaike, 1998). Autoregressive methods provide very precise picking results. However, they are computationally very intensive, which often limits their application to selected fragments of the input seismic trace, usually taken on the basis of prior estimate onset picking (Alvarez et al., 2013).

The ability to cope with poor signal to noise ratio conditions is another decisive factor when selecting a picking algorithm. The AMPA algorithm (Adaptive Multi-Band Picking Algorithm) (Alvarez et al., 2013) applies a series of filtering stages to the input signal in order to mitigate as far as possible the background noise as well as enhance the arrival of the seismic phase. The algorithm consists of two steps: Firstly, input signal is processed by using a filter bank in order to estimate the signal envelope and equalize noise for each subband. Secondly, the resulting envelope obtained from the previous step is filtered by using a set of filters, each one of a different length, designed to both enhance the phase arrival and lower emergent or impulsive noises. Results obtained by this algorithm achieve high accuracy even under very adverse signal to noise ratio conditions.

Nowadays, while none of the methods proposed so far has managed to completely replace the job of an expert analyst, they provide a valuable tool to support their work. A number of computer applications have been released for this purpose (Goldstein et al., 1998; Havskov and Ottemöller, 1999; Stockwell, 1999; Le-sage, 2009; Abdelwahed, 2012; Olivieri and Clinton, 2012). However, there is still demand for simple, easy to use analysis tools that manage to lower the need for training manuals, specially for users not familiar with complex computer environments. The APASVO software introduced in this paper is intended to ease the use of automatic picking methods using an user friendly graphical interface.

2. Program description

APASVO software tool consists of three different applications, named *apasvo-gui*, *apasvo-detector* and *apasvo-generator*. The application *apasvo-gui* is the graphical interface of APASVO, and allows the user to perform automatic onset detection and picking in an interactive way on one or more files simultaneously. The application implements the STA-LTA detection algorithm (Allen, 1982) and the AMPA picking algorithm (Alvarez et al., 2013). An autoregressive AIC-based picking method (Takanami and Kitagawa, 1988) can be applied either to a user selected signal fragment or in combination with the above algorithms. It is written using PySide and ObsPy. PySide is a python binding of the Qt cross-platform application framework, and its design resembles other digital signal processing and audio processing tools so that users can quickly feel comfortable when they start using the application. ObsPy is a python framework for processing seismological data (Beyreuther et al., 2010). Thus, *apasvo-gui* offers functionality to handle data in many seismological data formats. The application includes manual event relocation. P-wave arrival times can be manually manipulated and even new picks can be added by hand. Optionally S-waves can be manually picked.

apasvo-detector is an application that allows us to use APASVO's picking and detection capacities from command line. Like *apasvo-gui*, *apasvo-detector* allows us to process one or multiple files in a single run. The files can be processed either on batch or supervised mode. On supervised mode the application shows graphically on a separate window each one of the picked onset times and requests the user for feedback asking whether to accept the result shown or not, while on batch mode input files are processed without any interaction from the user about the results that are being generated.

Finally, *apasvo-generator* is a command line application, mainly designed for testing purposes, that generates seismic traces containing a synthetic seismic signal contaminated with background noise. It also allows us to add background noise to an existing seismic trace.

2.1. *apasvo-gui*'s main functionality

Fig. 1 shows the main components of *apasvo-gui*'s graphical interface. The application can display different traces of files with different seismic formats simultaneously. One trace from a sac data file and four traces from the same segy data file are opened. The file in sac format is selected. At the top, the application includes several toolbars that allow the use of most of the functionality. The *main toolbar* provides tools for opening or closing files, saving results, doing/undoing actions and configuring settings. The *navigation toolbar* comes with tools for zooming, panning, navigating between used views (e.g. returning to the previous view of the signal after zooming on it) or saving the current view of the signal to an image file. The *analysis toolbar* contains tools for applying the picking algorithms implemented by APASVO, as well as turning on/off the detection threshold and changing its value. It also includes filtering functionality. Finally, the *media toolbar* allows play backing the current signal. It also permits fine adjustment of current selection's boundaries.

Under the toolbar area the *event list* is displayed, showing a table with the data of the events picked so far. Each table row reports arrival time and characteristic function value at that time, plus additional information such as algorithm used, picking mode (i.e. manual or automatic), phase (P or S) confirmation status, an identifying name or label, and additional comments.

The central part of the interface is occupied by the *signal's main view*, which shows the portion of the signal being currently displayed using three different plots that share the same time axis.

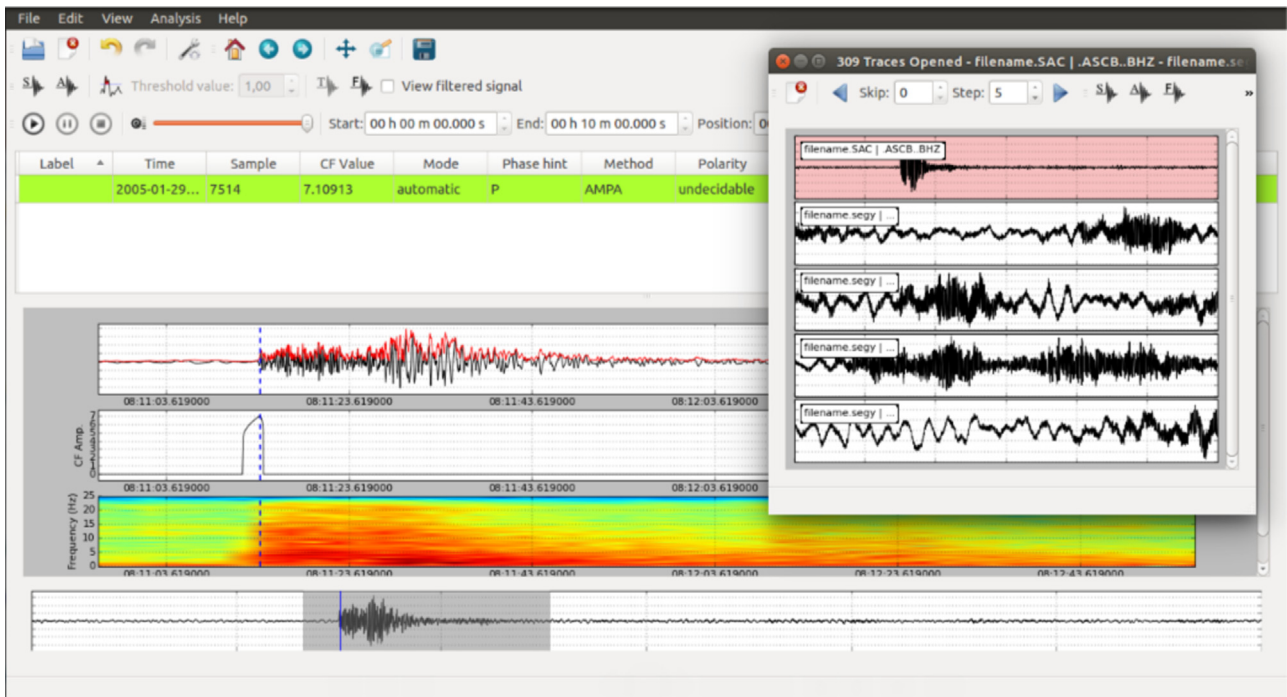


Fig. 1. *apasvo-gui*'s main interface. One trace from a sac data file and four traces from the same segy data file are opened. The file in sac format is selected. Below the toolbar area the list of picked events is displayed. The central part of the window shows the current visible portion of the signal using three plots: The first one displays the signal amplitude (in black) and the envelope (in red), the second plot shows the characteristic function obtained after applying AMPA and the third plot displays the spectrogram. Picked events are pointed out using a vertical dashed marker. At the bottom of the window, the *signal minimap* widget shows the entire signal where the current visible portion is shaded. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

The first plot shows signal amplitude and envelope (Bracewell, 1986). Users can select whether to display the envelope (by default) or not. The second plot is displayed after using STA-LTA or AMPA methods and shows the characteristic function returned as a result of the last algorithm applied. Lastly, the third plot shows the signal spectrogram. The spectrogram may assist in fitting the parameters of the automatic algorithms. Several aspects of the

spectrogram such as type, length and overlap of the window used can be configured with the *settings menu*.

Finally, under the *signal's main view*, the *signal minimap* widget is shown, which eases navigation along the signal. The *signal minimap* shows the whole seismic trace, where the shaded portion points out the signal currently displayed in the *signal's main view*. The size of the shaded portion can be manually modified. Users

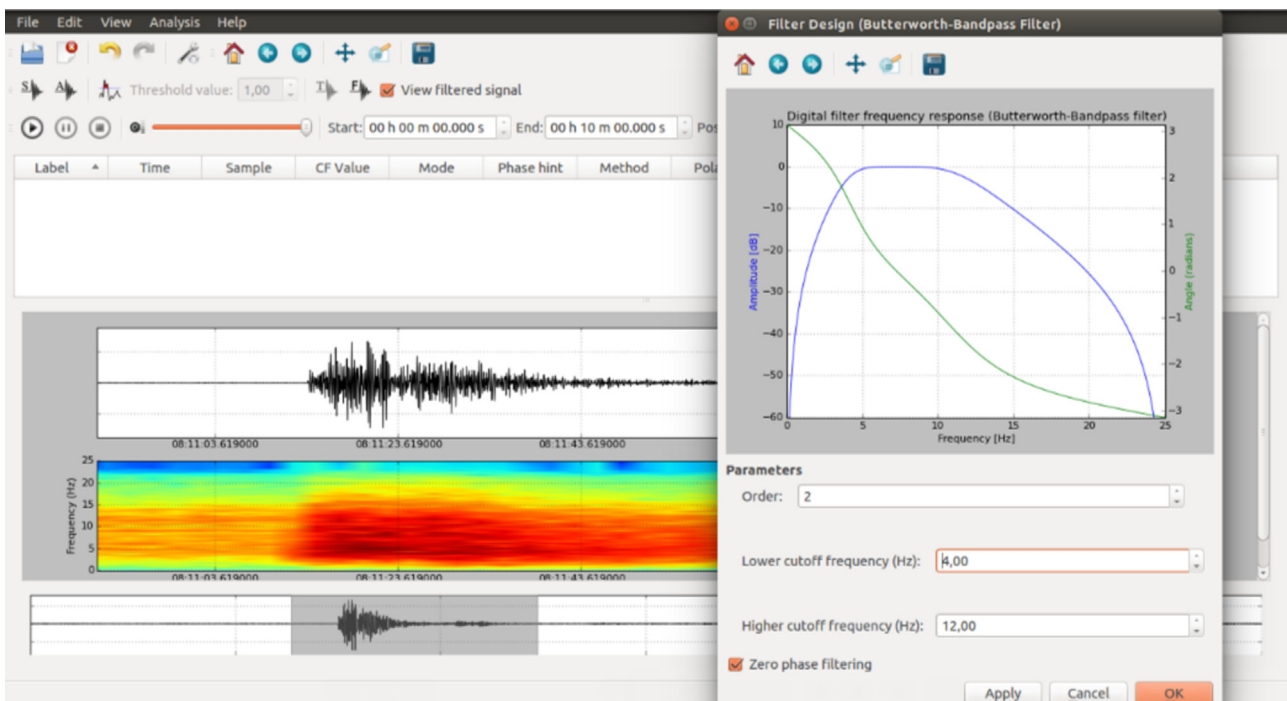


Fig. 2. Interface for designing a Butterworth bandpass filter. Both amplitude and phase responses are displayed.

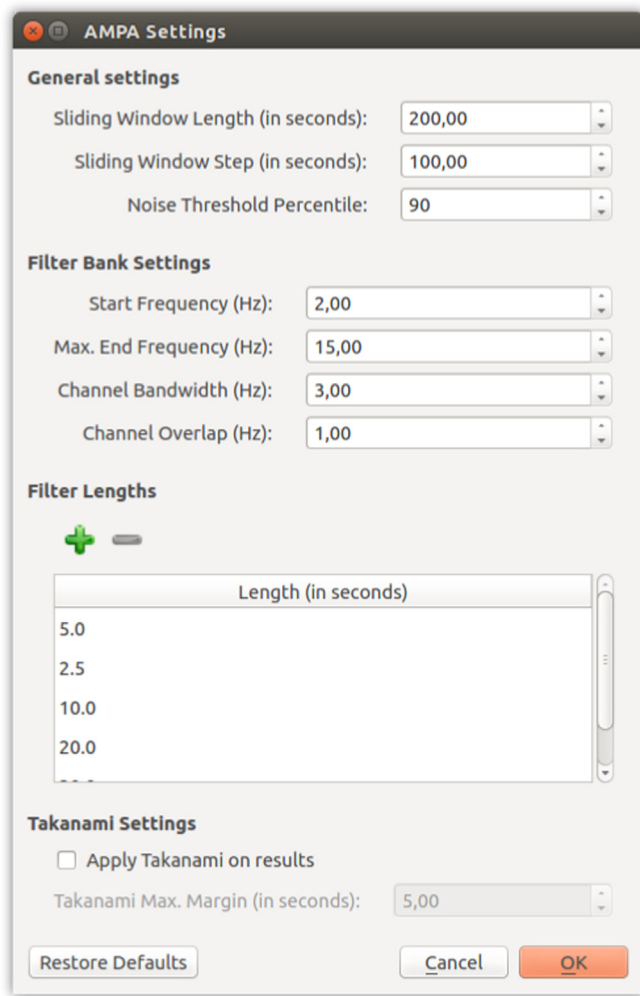


Fig. 3. Applying AMPA algorithm to a seismic trace with *apasvo-gui*. Used settings are shown on the AMPA settings dialog.

can drag the shaded portion to move along the signal or directly click on the part of the signal they want to be shown.

2.1.1. Opening a file

Users can open a data file either from *File* menu or from the *main toolbar*. APASVO can read files in common seismological data formats (such as *sac*, *mseed*, *seg*, etc.), text or binary format supporting different data types (16, 32 or 64 bits, integer or floating point) and endianness of data (little-endian or big-endian) and even audio files (*.wav*). After a file is selected, a dialog window is displayed. If the content of the file is not correctly read, the dialog window shows an error message. Some of the data characteristics can be automatically identified by APASVO, such as the format, the endianness or the sampling frequency in text files, added as a comment in the first line of the file.

2.1.2. Filtering a seismogram

APASVO offers functionality to filter the signals selected by the user using a Butterworth bandpass filter. Fig. 2 shows the interface for the design of the filter parameters (i.e. order, lower cutoff frequency and higher cutoff frequency). Both amplitude and phase responses of the filter designed can be displayed. In addition, users can select whether to apply causal filtering (by default) or zero-phase filtering.

2.1.3. Editing results

Each entry of the *event list* table contains a number of fields with information of a registered onset. Some of the fields, such as *label*, *status*, *comments* or *polarity* of the first arrival of the P-wave, can be modified by double clicking on them. However, manual modification of the P-wave arrival time requires to be done from *signal's main view* by left-clicking on its marker and then dragging it to desired time position. APASVO allows us to manually pick the S-wave arrival. The application also supports to manually create new picked events by clicking and dragging the interval on the main view of the signal where the new pick is going to be added and then right-clicking on the selected signal interval. If signal's characteristic function has been previously calculated, the new event will point out the maximum value of the function at the selected interval, otherwise it will be positioned at the midpoint of the interval.

2.1.4. Storing results

APASVO supports common data exchange formats, such as NonLinLoc, QuakeML or JSON. In addition, the content of the event list can be saved as tabular data using the text CSV (comma separated values) format, while the characteristic function can be stored either to text or raw binary format (16, 32 and 64-bits, integer or floating point, little-endian or big-endian). The content of the *signal's main view* can also be saved to an image file by using the *navigation toolbar*.

2.2. STA-LTA and AMPA algorithms with *apasvo-gui*

APASVO implements a naive version of STA-LTA algorithm based on taking the absolute value of the seismic trace as input (Allen, 1978, 1982). Implementation of AMPA algorithm is based on Alvarez et al. (2013). Implementation of AR-AIC method is based on Takanami et al. (Takanami and Kitagawa, 1988) using a 5th order model. Due to its computational complexity, Takanami's AR method is better suited for picking small localized signal fragments where the occurrence of a seismic phase arrival is suspected. Because of this, APASVO uses Takanami's AR algorithm as a complement to the results of STA-LTA and AMPA algorithms, which are applied to the entire trace.

Users can apply STA-LTA or AMPA algorithms by clicking on the *Apply STA-LTA* or *Apply AMPA* tools from the *analysis toolbar*. If a multichannel file is opened, users can select which traces they want to be picked by just clicking on them. If no trace is selected, then the algorithm is applied to all traces. After one of the picking method is selected, a dialog window is displayed allowing us to configure the settings of the method to be run. For STA-LTA, the dialog asks for the length of the STA and LTA windows and whether to apply Takanami's AR-AIC on an interval centered at the onset time estimated by the STA-LTA algorithm. If the user chooses to apply Takanami's algorithm to the results, the dialog allows us to select the radius of the interval in which it will be applied (*Takanami Max. Margin*).

Fig. 3 shows the dialog window for AMPA algorithm, which consists of four sections. *General Settings* section configures how AMPA is applied to a seismic trace. By default, APASVO implementation of AMPA is applied using overlapping windows. Thus, *general settings* allows us to choose the size of the sliding window and the step used in each successive application. Noise threshold percentile for AMPA's noise reduction stage can also be chosen. *Filter Bank Setting* section permits users to configure the filter bank used in AMPA's multi-band processing stage. Section *Filter Lengths* shows the list of filter lengths used by AMPA at the enhancement filter stage. The length of a filter is related to the duration of the detected events, so an enhancement filter for long duration events can negate short duration events and vice versa. *List add* and *list remove* buttons can be used to add/delete filters to

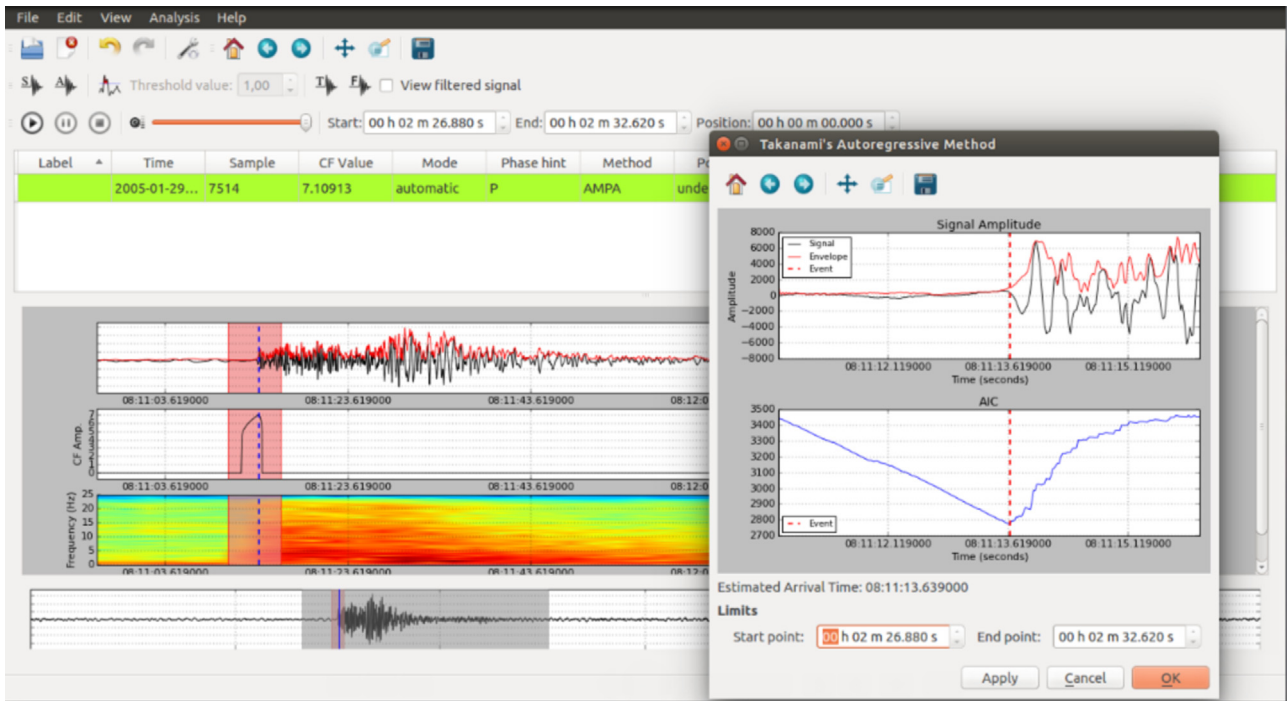


Fig. 4. Applying Takanami's AR-AIC method. The upper part of Takanami's AR dialog displays the signal amplitude (in black) and the envelope (in red), while the lower part displays the computed AIC values. A vertical dashed marker points out the estimated onset time. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

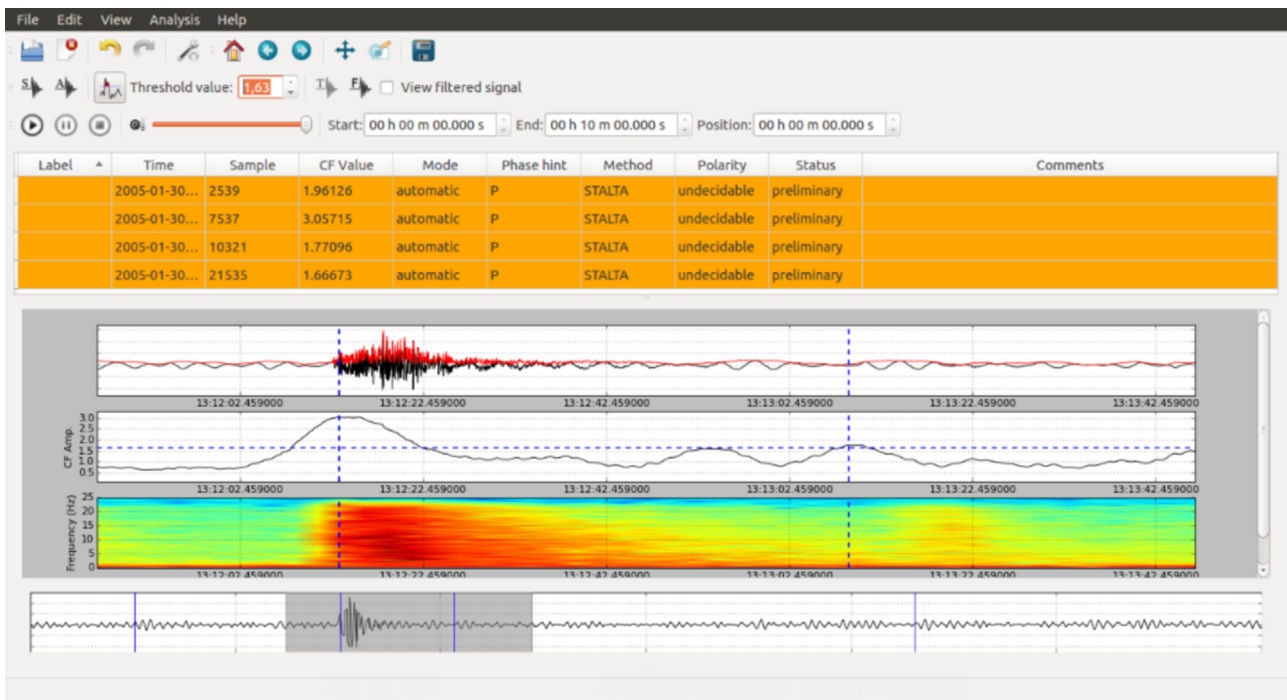


Fig. 5. Applying STA-LTA algorithm with detection threshold enabled. Characteristic function peaks over selected threshold are marked as possible events.

the list, while double-clicking on a filter allows us to edit its value. Lastly, *Takanami Settings* section allows us to choose whether to apply Takanami's AR algorithm on AMPA results as commented above for STA-LTA's settings dialog.

After applying a picking algorithm, the characteristic function is displayed in the main view of the signal and the list of estimated onsets are shown in the list of events. Each estimated event is indicated by a vertical marker in the main view and in the signal minimap widget.

2.3. AR-AIC picking with apasvo-gui

APASVO implementation of Takanami's AR-AIC algorithm can be used in two ways. Users can select a signal portion by clicking and dragging and then use the *Apply Takanami* tool from the *analysis toolbar*. The autoregressive method can also be subsequently applied to both STA-LTA and AMPA algorithms. After selecting the method a window is displayed with the results (Fig. 4). The results window shows the signal amplitude and its envelope

as well as the estimated Akaike Information Criterion (AIC) in the selected portion of the trace. The estimated onset time is indicated by a vertical marker. The results window also allows us to modify the boundaries of the selected interval and reapply the algorithm.

2.4. Event detection with *apasvo-gui*

APASVO includes the possibility of using a detection threshold that can be enabled from *analysis toolbar*. Fig. 5 shows an example of applying STA-LTA with detection threshold enabled. The characteristic function represents the STA-LTA ratio. Instead of returning only the maximum value of the characteristic function, the application return all *peaks* of the characteristic function whose values are greater than the threshold value and correspond to a maximum value of the characteristic function in an interval $\pm \delta$. The default value for δ is 1 s. Once enabled, the threshold is displayed as a dashed horizontal line in the characteristic function's plot of the signal's main view. Its value can be modified by clicking and dragging it or from the *analysis toolbar*.

2.5. Command line analysis with *apasvo-detector*

As commented in Section 2, the *apasvo-detector* tool allows serial processing of multiple files at once from the system console, including single- and multichannel files. Analysis can be performed in two ways: supervised and batch mode. On batch mode, which is the default mode, traces are processed and results are saved without user intervention. On supervised mode, the application plots graphically each of the picked/detected events on a separated window and asks a confirmation for saving or discarding it, continuing the analysis in batch mode or cancelling it. Despite being a command line application, plotting results requires supervised mode to be run in a system graphic mode, not pure text mode. E.g. the following command:

```
$ apasvo-detector *.sac -csv example.csv -f 100 -
  m stalta --lta 60 --sta 5 --takanami -s --
  show-all
```

performs onset picking on all seismic traces stored in sac format, using a sample rate of 100 Hz (`-f 100`). STA-LTA (`-m stalta`) algorithm with 60 s of LTA window length (`--lta 60`) and 5 s of STA window length (default value) is used together with

Takanami's AR method (`--takanami`). Processing is done in supervised mode (`-s`), showing characteristic function, envelope function and spectrogram for each possible onset time (`--show-all`). Finally, results are saved in CSV file format to `example.csv`.

2.6. Synthetic earthquake generation with *apasvo-generator*

The *apasvo-generator* tool allows us to generate sets of artificial seismic signals for testing purposes as well as to add background seismic noise to existing signals. Artificial earthquakes are generated at a desired start point from white noise band-filtered and modulated by using a set of envelope functions with different length and amplitude for each band. Similarly, seismic background noise is modeled from white noise filtered by a noise model consisting of an estimate of the noise power spectral density affecting a set of real seismic noise observations (Peterson, 1993). In `bfir1s` directory, a file named `IAGPDS-bfir1s.txt` is included containing the FIR filter coefficients obtained from a seismic noise model generated from a set of 400 natural earthquakes recorded in the Andalusian Institute of Geophysics (IAGPDS) seismic network. The contents of that file can be used to model the seismic background noise generated with *apasvo-generator*. If no noise model is specified, white Gaussian noise is used by default. E.g. Fig. 6 shows a synthetic earthquake generated using the following command:

```
$ apasvo-generator -o example.out -f 100 -l 600 -
  t 200 -ep 5 -np 0 --f-low 4.0 --f-high 15.0 -
  fir IAGPDS-bfir1s.txt
```

where the seismic trace generated has a length of 600 s (`-l 600`), sample rate is 100 Hz (`-f 100`) and earthquake starts at the second 200 (`-t 200`) having 5 dB of earthquake power (`-ep 5`), 0 dB of noise power (`-np 0`) and a frequency range between 4 Hz and 25 Hz (`-f-low 4.0 -h-high 25.0`). The signal is corrupted using the noise model stored in `IAGPDS-bfir1s.txt` file. Result is saved to `example.out` file.

3. Conclusion

The increasing amount of available seismic data is encouraging the interest in the development of computer programs to

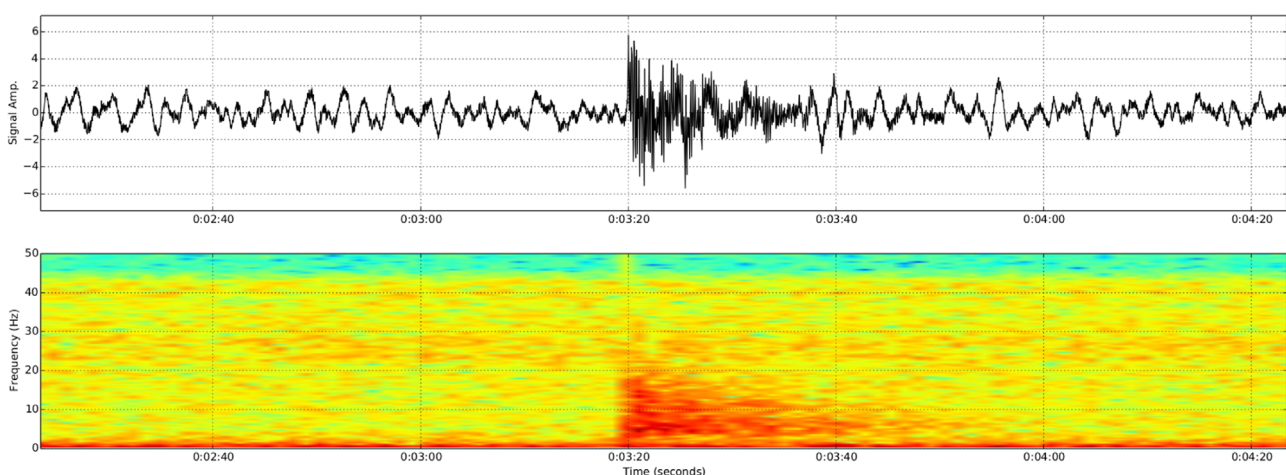


Fig. 6. A synthetic earthquake generated using *apasvo-generator* (upper panel) and its spectrogram (lower panel). Signal has been polluted using a seismic noise model generated from a set of 400 natural earthquakes provided by the IAGPDS seismic network.

automate the processing of such data, complementing the work of the expert operators in charge of analysis and achieving results to be more independent of the bias introduced by human factors. In this context, this paper introduces APASVO, a software tool that aims to become a useful utility for many seismological tasks involving the picking of seismic arrivals, such as earthquake source location or active seismic tomography. APASVO is a graphical tool that lets users to perform event detection and picking in seismic traces using a friendly, easy to use graphical interface. The application can process data in a large variety of file formats. To facilitate its use, APASVO includes common features of digital signal processing and audio processing tools such as filtering, scrolling, zooming, panning or play backing.

To perform the automatic determination of the onset time, APASVO implements STA-LTA, AMPA and AR-AIC algorithms. The STA-LTA detection algorithm is widely used for its simplicity and flexibility, but lacks in poor performance under noisy conditions, yielding an approximate arrival time. On the other hand, the AMPA picking algorithm is a novel method that gets good performance even under adverse noise conditions, based on applying a series of filtering stages to the input signal both reducing background noise and enhancing onset arrival. Besides, an implementation of Takamami's AR-AIC picking algorithm is included, which can be applied either to a user selected signal fragment or in combination with the above algorithms.

In addition to the graphical application *apasvo-gui*, APASVO is complemented with two additional command-line applications: *apasvo-generator*, a synthetic seismic trace generator, and *apasvo-detector*, that allows picking one or multiple files simultaneously in either batch or supervised mode.

Finally, APASVO is free, open source software, licensed under the GPL license. Both the source code and precompiled binaries for various operating systems are available from the application repositories.

Acknowledgments

This work was partially supported by the Spanish MINECO research projects TEC2012-31551 ("Advanced signal processing algorithms for recognition and characterization of seismic signals from volcanoes", APASVO) and TEC2015-68752-R (KNOWAVES), and by the European Union project MED-SUV (EC-FP7, Grant agreement no 308665). The authors would also like to thank Prof. J. M. Ibáñez and Prof. G. Alguacil (Andalusian Institute of Geophysics and Earthquake Disaster Prevention, Spain) for providing a data set of real seismograms as well as for the overall validation of the software tool.

Appendix A. Supplementary data

Supplementary data associated with this article can be found in the online version at <http://dx.doi.org/10.1016/j.cageo.2016.02.004>.

References

- Abdelwahed, M.F., 2012. SGRAPH (SeismoGRAPHer): seismic waveform analysis and integrated tools in seismology. *Comput. Geosci.* 40, 153–165. <http://dx.doi.org/10.1016/j.cageo.2011.06.019>.
- Akaike, H., 1998. Information theory and an extension of the maximum likelihood principle. In: *Selected Papers of Hirotugu Akaike*. Springer, New York, pp. 199–213.
- Allen, R., 1978. Automatic earthquake recognition and timing from single traces. *Bull. Seismol. Soc. Am.* 68 (5), 1521–1532.
- Allen, R., 1982. Automatic phase pickers: their present use and future prospects. *Bull. Seismol. Soc. Am.* 72 (6B), S225–S242.
- Alvarez, I., Garcia, L., Mota, S., Cortes, G., Benitez, C., De la Torre, A., 2013. An automatic P-phase picking algorithm based on adaptive multiband processing. *IEEE Geosci. Remote Sens. Lett.* 10, 1488–1492. <http://dx.doi.org/10.1109/LGRS.2013.22260720>.
- Baer, M., Kradolfer, U., 1987. An automatic phase picker for local and teleseismic events. *Bull. Seismol. Soc. Am.* 77 (4), 1437–1445.
- Beyreuther, M., Barsch, R., Krischer, L., Megies, T., Behr, Y., Wassermann, J., 2010. ObsPy: a python toolbox for seismology. *Seismol. Res. Lett.* 81 (3), 530–533. <http://dx.doi.org/10.1785/gssrl.81.3.530>.
- Bracewell, R.N., 1986. *The Fourier Transform and its Applications*. McGraw-Hill, New York.
- Dai, H., MacBeth, C., 1995. Automatic picking of seismic arrivals in local earthquake data using an artificial neural network. *Geophys. J. Int.* 120 (3), 758–774. <http://dx.doi.org/10.1111/j.1365-246X.1995.tb01851.x>.
- García-Yeguas, A., Koulakov, I., Ibáñez, J.M., Rietbrock, A., 2012. High resolution 3D P wave velocity structure beneath Tenerife Island (Canary Islands, Spain) based on tomographic inversion of active-source data. *J. Geophys. Res.* 117 (B09309). <http://dx.doi.org/10.1029/2011JB008970>.
- Gentili, S., Michelini, A., 2006. Automatic picking of P and S phases using a neural tree. *J. Seismol.* 10 (1), 39–63. <http://dx.doi.org/10.1007/s10950-006-2296-6>.
- Goldstein, P., Dodge, D., Firpo, M., Ruppert, S., 1998. What's new in SAC2000? enhanced processing and database access. *Seismol. Res. Lett.* 69 (3), 202–205. <http://dx.doi.org/10.1785/gssrl.69.3.202>.
- Gutenberg, B., Richter, C.F., 1956. Magnitude and energy of earthquakes. *Ann. Geophys.* 9 (1), 1–15. <http://dx.doi.org/10.4401/ag-5590>.
- Gutenberg, B., Richter, C.F., 1965. *Seismicity of the Earth and Associated Phenomena*. Hafner, New York.
- Havskov, J., Alguacil, G., 2004. *Instrumentation in Earthquake Seismology*. Springer-Verlag, New York, NY, USA.
- Havskov, J., Ottemöller, L., 1999. Seisan earthquake analysis software. *Seismol. Res. Lett.* 70 (5), 532–534. <http://dx.doi.org/10.1785/gssrl.70.5.532>.
- Horiuchi, S., Negishi, H., Abe, K., Kamimura, A., Fujinawa, Y., 2005. An automatic processing system for broadcasting earthquake alarms. *Bull. Seismol. Soc. Am.* 95 (2), 708–718. <http://dx.doi.org/10.1785/0120030133>.
- Ibáñez, J.M., Rietbrock, A., García-Yeguas, A., 2008. Imaging an active volcano edifice at Tenerife Island, Spain. *Eos, Trans. Am. Geophys. Union* 89 (32), 289–290. <http://dx.doi.org/10.1029/2008EO320001>.
- Joswig, M., 1990. Pattern recognition for earthquake detection. *Bull. Seismol. Soc. Am.* 80 (1), 170–186.
- Kitagawa, G., Takamami, T., Matsumoto, N., 2001. Signal extraction problems in seismology. *Int. Stat. Rev.* 69 (1), 129–152.
- Küperkoch, L., Meier, T., Diehl, T., 2011. Automated event and phase identification. In: Bormann, P. (Ed.), *New Manual of Seismological Observatory Practice 2 (NMSOP-2)*. Deutsches GeoForschungsZentrum GFZ, Potsdam, pp. 1–52. http://dx.doi.org/10.2312/GFZ.NMSOP-2_ch16.
- Küperkoch, L., Meier, T., Friederich, W., working group, E., 2010. Automated P-wave arrival time determination using higher order statistics. *Geophys. J. Int.* 181 (2), 1159–1170. <http://dx.doi.org/10.1111/j.1365-246X.2010.04570.x>.
- Kurz, J.H., Grosse, C.U., Reinhardt, H.-W., 2005. Strategies for reliable automatic onset time picking of acoustic emissions and of ultrasound signals in concrete. *Ultrasonics* 43 (7), 538–546. <http://dx.doi.org/10.1016/j.ultras.2004.12.005>.
- Kurzon, I., Vernon, F., Rosenberger, A., Ben-Zion, Y., 2014. Real-time automatic detectors of P and S waves using singular value decomposition. *Bull. Seismol. Soc. Am.* 104 (4), 1696–1708. <http://dx.doi.org/10.1785/0120130295>.
- Leonard, M., Kennett, B., 1999. Multi-component autoregressive techniques for the analysis of seismograms. *Phys. Earth Planet. Inter.* 113 (1), 247–263. [http://dx.doi.org/10.1016/S0031-9201\(99\)00054-0](http://dx.doi.org/10.1016/S0031-9201(99)00054-0).
- Lesage, P., 2009. Interactive Matlab software for the analysis of seismic volcanic signals. *Comput. Geosci.* 35 (10), 2137–2144. <http://dx.doi.org/10.1016/j.cageo.2009.01.010>.
- Montalbetti, J.F., Kanasewich, E.R., 1970. Enhancement of teleseismic body phases with a polarization filter. *Geophys. J. Int.* 21 (2), 119–129. <http://dx.doi.org/10.1111/j.1365-246X.1970.tb01771.x>.
- Nakamura, H., Horiuchi, S., Wu, C., Yamamoto, S., Rydelek, P., 2009. Evaluation of the real-time earthquake information system in Japan. *Geophys. Res. Lett.* 36 (L00B01). <http://dx.doi.org/10.1029/2008GL036470>.
- Nippres, S.E.J., Rietbrock, A., Heath, A.E., 2010. Optimized automatic pickers: application to the ANCORP data set. *Geophys. J. Int.* 181 (2), 911–925. <http://dx.doi.org/10.1111/j.1365-246X.2010.04531.x>.
- Olivieri, M., Clinton, J., 2012. An almost fair comparison between Earthworm and SeisComp3. *Seismol. Res. Lett.* 83 (4), 720–727. <http://dx.doi.org/10.1785/0220110111>.
- Panagiotakis, C., Kokinou, E., Vallianatos, F., 2008. Automatic P-phase picking based on local-maxima distribution. *IEEE Trans. Geosci. Remote Sens.* 46 (8), 2280–2287. <http://dx.doi.org/10.1109/TGRS.2008.917272>.
- Peterson, J., 1993. Observations and Modeling of Seismic Background Noise.
- Ross, Z.E., Ben-Zion, Y., 2014. Automatic picking of direct P, S seismic phases and fault zone head waves. *Geophys. J. Int.* 199 (1), 368–381. <http://dx.doi.org/10.1093/gji/ggu267>.
- Saragiotis, C.D., Hadjileontiadis, L.J., Panas, S.M., 2002. PAI-S/K: a robust automatic seismic P phase arrival identification scheme. *IEEE Trans. Geosci. Remote Sens.* 40 (6), 1395–1404. <http://dx.doi.org/10.1109/TGRS.2002.800438>.
- Stockwell Jr., J.W., 1999. The CWP/SU: seismic un*x package. *Comput. Geosci.* 25 (4), 415–419. [http://dx.doi.org/10.1016/S0098-3004\(98\)00145-9](http://dx.doi.org/10.1016/S0098-3004(98)00145-9).
- Takamami, T., Kitagawa, G., 1988. A new efficient procedure for the estimation of onset times of seismic waves. *J. Phys. Earth* 36 (6), 267–290. <http://dx.doi.org/>

- 10.4294/jpe1952.36.267.
- Takanami, T., Kitagawa, G., 1991. Estimation of the arrival times of seismic waves by multivariate time series model. *Annals of the Institute of Statistical Mathematics* 43 (3), 407–433.
- Tong, C., 1995. Characterization of seismic phases—an automatic analyser for seismograms. *Geophys. J. Int.* 123 (3), 937–947. <http://dx.doi.org/10.1111/j.1365-246X.1995.tb06900.x>.
- Trnkoczy, A., 2002. Understanding and parameter setting of STA/LTA trigger algorithm. IASPEI New Manual of Seismological Observatory Practice, vol. 2, pp. 1–19. http://dx.doi.org/10.2312/GFZ.NMSOP_r1_IS_8.1.
- Withers, M., Aster, R., Young, C., Beiriger, J., Harris, M., Moore, S., Trujillo, J., 1998. A comparison of select trigger algorithms for automated global seismic phase and event detection. *Bull. Seismol. Soc. Am.* 88 (1), 95–106.
- Zhang, H., Thurber, C., Rowe, C., 2003. Automatic P-wave arrival detection and picking with multiscale wavelet analysis for single-component recordings. *Bull. Seismol. Soc. Am.* 93 (5), 1904–1912. <http://dx.doi.org/10.1785/0120020241>.