ASAT

Ver8.0

“ASATARTE Signal Analysis Tool”

A Tsunami Warning and Inversion Tool

- User Manual

Prepared by	**Filipe Bernardo Lisboa**
Reference	Error! Unknown document property name.
Issue	Error! Unknown document property name.
Revision	**Maria Ana Viana-Baptista**
Date of Issue	**February 2015**
Status	**Draft**
Document Type	Error! Unknown document property name.
Distribution	Error! Unknown document property name.

- Specifications document
- Developer’s guide

Filipe Lisboa, IPMA

Table of Contents

1 PURPOSE AND SCOPE

The purpose of this note is to enhance understanding of the tools developed in the framework of the author's research grant within the ASTARTE project. ASTARTE is a research and development project organized to foster tsunami resilience in Europe, through innovative research on scientific problems critical to enhancing forecast skills in terms of sources, propagation and impact. ASTARTE is a collaborative project within the European Union's FP7 programme.

The tool described here and all related routines were mainly conceived in MATLAB on an end-to-end basis. The routines provide fast ability to load data, manipulate extensive data sets, analyse variables and perform analysis routines for tsunami-related studies, as well as warnings and inversion.

In a standalone philosophy, a devoted Graphical User Interface (GUI) is present. In this way, it is expectable that the non-MATLAB expert is able to load data and perform basic analysis. Nonetheless, we strongly encourage the use of the MATLAB integrated version in order to have more options for data analysis. The reader will find two main sections of this document, one in the user's perspective where the general procedures are presented in a how-to guide for the standalone version and the second on the programmer's point-of-view where the code is explained in greater detail. Modifications to the code can be done, although we call for the reader's attention and thoughtful conscience acknowledging the author's work. The tool can be used freely although a citation to the article XYZ is mandatory. (Maria Ana, project coordinator, will give the final judgement of the copyrights of this work.)

2 ASSISTANCE

This tool was developed in Mac OS X 10.10 Yosemite. Therefore, assistance to all Unix systems will be immediate. The tool is also available for Windows Vista and above. In this case, due to cross compiling needs, assistance might take longer. Relevant questions and suggestions will have a direct answer from the author (Filipe – mezzo.lisboa@gmail.com) in less than a month.

3 INSTALLATION

The end user is provided with two options for installation. If MATLAB 2012b (and above) is already installed on the machine we strongly encourage the use of the package containing the non-compiled code. Recent versions of MATLAB allow the fast installation of MATLAB-Apps. A MATLAB-App version is also available.

For the non-MATLAB user there is no need to install MATLAB. Nevertheless, users are requested to have installed MATLAB Runtime (freely available at www.mathworks.com) before installing the ASAT tool. The standalone ASAT application is provided with the compiled code. In this case, specific packages are available:

1. Unix Systems (Mac OS X and Linux):
   Download the full .zip package entitled "ASAT.zip"
2. Windows:
   Download the devoted package and run the .exe file.

The following paragraphs contain step-by-step procedures for the installation of the Standalone Versions.

3.1 Installation on Mac OS X

3.1.1 Prerequisites for Deployment

Verify the MATLAB Compiler Runtime (MCR) is installed and ensure you have installed version 8.0 (R2012b). If the MCR is not installed, do the following:

(1) enter

>>mcrinstaller

at MATLAB prompt. The MCRINSTALLER command displays the location of the MCR Installer.

(2) run the MCR Installer.

Or download the Macintosh version of the MCR for R2012b from the MathWorks Web site by navigating to http://www.mathworks.com/products/compiler/mcr/index.html
For more information about the MCR and the MCR Installer, see Distribution to End Users in the MATLAB Compiler documentation in the MathWorks Documentation Center.
NOTE: You will need administrator rights to run MCRInstaller.

3.1.2 Files to Deploy and Package

Files of package for Standalone

1. run_ASTARTE_MAcOsX.sh (shell script for temporarily setting environment variables and
   executing the application)
2. to run the shell script, type

./run_ASTARTE_MAcOsX.sh <mcr_directory> <argument_list>

at Linux or Mac command prompt. <mcr_directory> is the directory where version 8.0 of MCR is installed or the directory where MATLAB is installed on the machine. <argument_list> is all the arguments you want to pass to your application. For example, if you have version 8.0 of the MCR installed in
filipebernardolisboa/mathworks/home/application/v80, run the shell script as:

./run_ASTARTE_MAcOsX.sh /mathworks/home/application/v80

If you have MATLAB installed in /mathworks/devel/application/matlab, run the shell script as:

```
./run_ASTARTE_MAcOsX.sh /mathworks/devel/application/matlab
```

-MCRInstaller.zip
  -if end users are unable to download the MCR using the above
   link, include it when building your component by clicking
   the "Add MCR" link in the Deployment Tool
-The Macintosh bundle directory structure ASAT.app
  -this can be gathered up using the zip command
   zip -r ASAT.zip ASAT.app
   or the tar command
   tar -cvf ASAT.tar ASAT.app
-This readme file

For information on deployment terminology, go to http://www.mathworks.com/help. Select MATLAB Compiler > Getting Started > About Application Deployment > Application Deployment Terms in the MathWorks Documentation Center.

3.1.3 Notes

A. Mac systems:
  On the target machine, add the MCR directory to the environment variable
  DYLD_LIBRARY_PATH by issuing the following commands:
     NOTE: <mcr_root> is the directory where MCR is installed on the target machine.

```
setenv DYLD_LIBRARY_PATH
   $DYLD_LIBRARY_PATH:
   <mcr_root>/v80/runtime/maci64:
   <mcr_root>/v80/sys/os/maci64:
   <mcr_root>/v80/bin/maci64:
   /System/Library/Frameworks/JavaVM.framework/JavaVM:
   /System/Library/Frameworks/JavaVM.framework/Libraries
setenv XAPPLRESDIR <mcr_root>/v80/X11/app-defaults
```

  For more detail information about setting MCR paths, see Distribution to End Users in
  the MATLAB Compiler documentation in the MathWorks Documentation Center.

     NOTE: To make these changes persistent after logout on Linux
         or Mac machines, modify the .cshrc file to include this
         setenv command.
     NOTE: The environment variable syntax utilizes forward
         slashes (/), delimited by colons (:).

NOTE: When deploying standalone applications, it is possible to run the shell script file run_ASAT.sh instead of setting environment variables. See section 2 "Files to Deploy and Package".

3.1.4 Launch application from Macintosh finder.

The application is not purely graphical, that is, it doesn't read from standard in or
write to standard out or standard error. It may be launched in the finder just like any
other Macintosh application but the user will loose access to data, errors, warnings and state of the processing.

3.2 Installation on Windows 7

3.2.1 Prerequisites for Deployment

Verify the MATLAB Compiler Runtime (MCR) is installed and ensure you have installed version 8.0 (R2012b). If the MCR is not installed, do the following:
 (1) enter
 >>mcrinstaller
 at MATLAB prompt. The MCRINSTALLER command displays the location of the MCR Installer.
 (2) run the MCR Installer.
Or download the Windows 64-bit version of the MCR for R2012b
from the MathWorks Web site by navigating to
 http://www.mathworks.com/products/compiler/mcr/index.html
For more information about the MCR and the MCR Installer, see Distribution to End Users in the MATLAB Compiler documentation in the MathWorks Documentation Center.
NOTE: You will need administrator rights to run MCRInstaller.

3.2.2 Files to Deploy and Package

Files in the standalone package

- ASAT.exe
- MCRInstaller.exe
- if end users are unable to download the MCR using the above link, there is a version with the MCR within the package.
- This readme file

3.2.3 Definitions

For information on deployment terminology, go to http://www.mathworks.com/help. Select MATLAB Compiler > Getting Started > About Application Deployment > Application Deployment Terms in the MathWorks Documentation Center.

4 USER MANUAL (AN HOW-TO FOR THE STANDALONE VERSION)

ASAT stands for ASTARTE Signal Analysis Tool. A how-to-use guide is present in this section. For the non-MATLAB user, this section will be sufficient to download and select data from either DART buoys or port gauges.

4.1 Loading Data

Loading water column height data can be done by the standalone version of ASAT tool. The interface is shown in the main window of the tool (for non-standalone versions run the function: START.m). This graphical user interface (GUI) is depicted in Figure 4-1. The users can then introduce a station or buoy identifier provided by IOC site, historical DART, real-time DART or a specific file with data. For IOC, historical and real-time options Internet access is mandatory and this application mustn't be blocked be a firewall. The tool will automatically download data from either IOC UNESCO's Sea Level Monitoring or US-NOAA's DART stations.

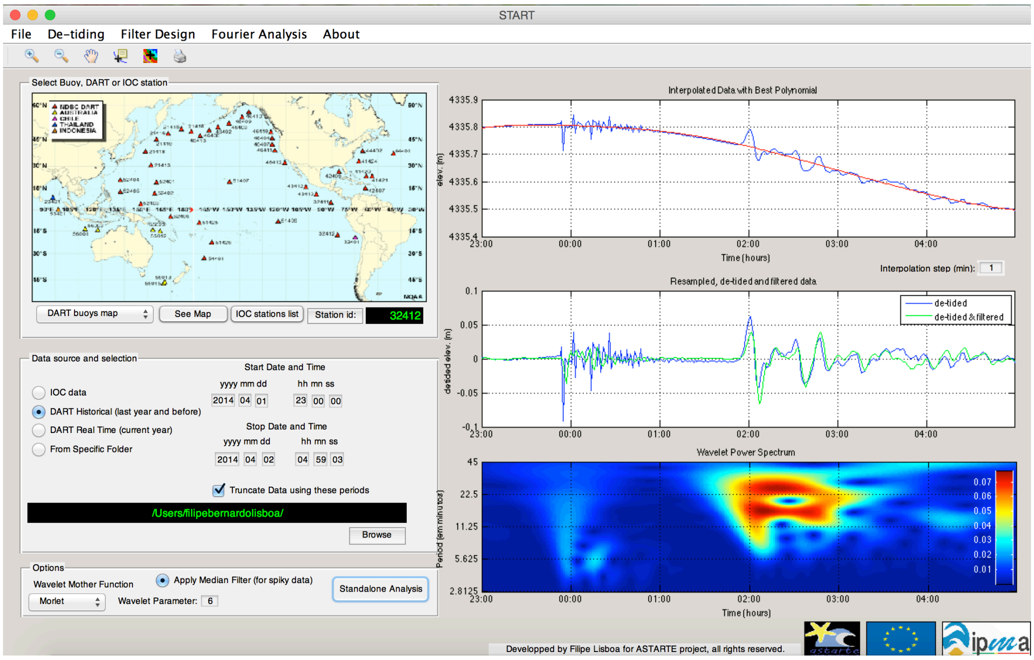


Figure 4-1 Main window of the ASAT 8.0 tool.

Upon loading, the overview of the original signal, with no data manipulation, is displayed on the right top box and a workspace of the variables is permanently saved with the name: tsunami_wkspace_xyz.mat where xyz is the buoy id or other identifier for the input file. The date in which this workspace was created is also part of its name.

An interactive map is displayed displaying the DART buoys or the IOC stations.
Plots presented in this manual are given as examples. They normally relate to the DART buoy 32412 with data from the Iquique, Chile earthquake that occurred on the evening of April, the 1st 2014.

4.2 Station id

The station id is either a 5-digit code for DART stations or a 4-character code for IOC data. This application provides maps with the location of these stations. Nevertheless, the NDBC site might contain updated information such as new stations that were not available by the time of this delivery. The same goes to the IOC data. In those cases, it is advisable to look for the station’s id online if it cannot be found in the ASAT maps.

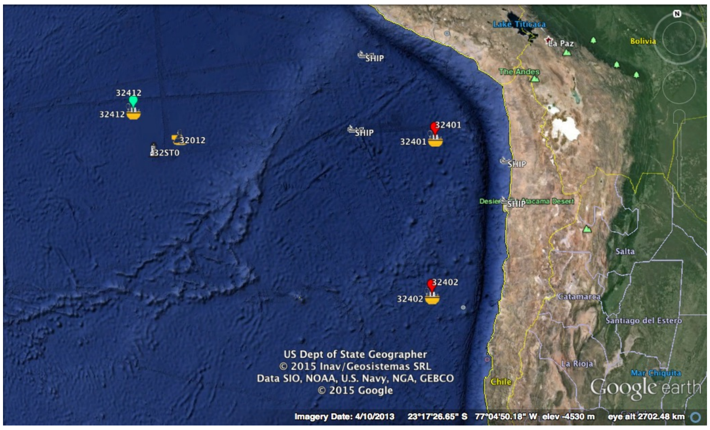


Figure 2 DART stations 32412, 12401 and 12402 off the coast of Chile.

The user must check for data availability. If data is not available for the given station the program will give an error.

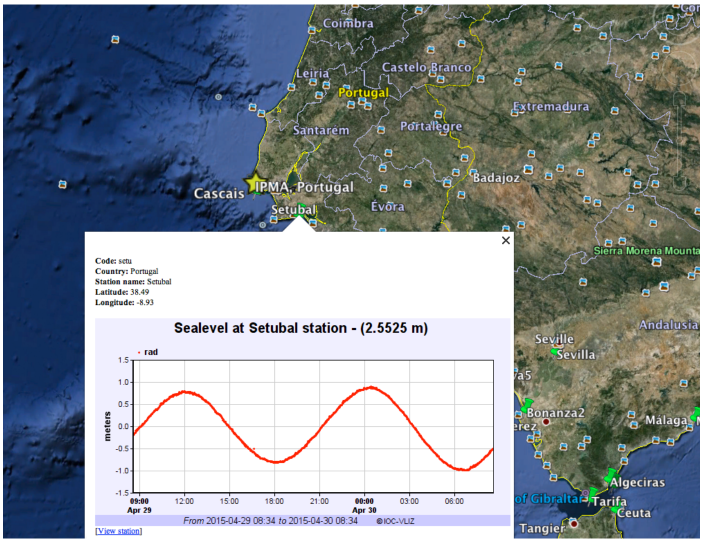


Figure 3 Sea level station in Setúbal, Portugal, as given by IOC site. Station id: "setu".

4.3 Data Source and Selection

Four options are available for data sources. The user can select data from sea level stations (gathered and distributed by the IOC site), from DART buoys or from a specific file. IOC sea level stations are normally located in the coast near or inside ports and therefore provide noisy data that might contain resonance effects. These resonance effects might be interesting to study how the site retains energetic waves. On the other hand, DART buoys are located in the open ocean and provide cleaner data where the transient aspect of Tsunami signal is evident. Due to the different historical and real-time data distribution of DART stations, the selection of this data is split into “historical” and “real-time” (DART data availability is described in section 4.3.1).

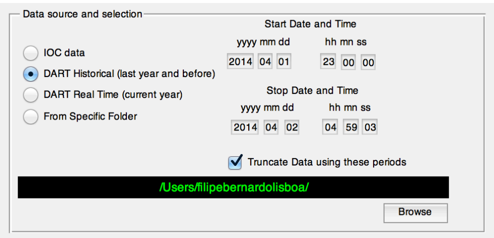


Figure 4 Data source and selection of data period.

All data from each of the four options can be truncated to a specific time-period, by introducing start and stop dates and times and checking the truncation button has depicted in Figure 4 (example in the figure for DART 32412. A Tsunami was generated by the Iquique, Chile earthquake).

4.3.1 DART data availability

When it comes to automatically load data from DART stations it is important to remember that, from research to operations, to data archive and dissemination, data availability is conducted collaboratively between the National Data Buoy Center, the Pacific Marine Environmental Laboratory, and the National Geophysical Data Center, with oversight by the [US] National Weather Service. The National Data Buoy Center manages and conducts all operational network activities and distributes real-time data to the public. The Pacific Marine Environmental Laboratory provides the research component in support of modeling and network enhancements for improved forecasting capability. The National Geophysical Data Center is responsible for the processing, archiving, and distribution of all retrospective data and integrates DART® tsunameter data with the National Geophysical Data Center global historical tsunami database. (George Mungov)

In sum, to load DART® tsunameter data, ASAT saves you the time to go to either NDBC or NGDC’s website by automatically download the files and create a devoted workspace. ASAT does so by accessing their websites. Historical data (from the last year and before) and current data are available in different servers. Historical data is available through NDBC’s FTP server whereas current year data is available, through HTML, at NOAA’s site. The user is then requested to specify whether is historical or from the current year.

4.4 Input file format

It is important to stress that data can be given from DART® buoys as well as other sources. The code was specifically developed to read files in DART® format but the user can still convert the input file into this standard format.
The standard format is composed of 8 (eight) columns: four digit year, two digit month, two digit day, two digit hour, two digit minute, two digit second, one digit for acquisition mode identification and a double variable for the water column height (S.I. metres are advisable):

```
format: #YY   MM DD hh mm ss T    HEIGHT
        2014   4  1  23  0  0  9     2428.7
        2014   4  1  23  1  0  9     1613.9
        2014   4  1  23  2  0  9     2837.8
        2014   4  1  23  3  0  9     1099.0
```

There are specified MATLAB functions to convert data into this ASAT format. We have found that it is a very good practice to maintain this common format in order to avoid loosing track of how data is displayed. Even without a header, this column/height information is clearly associated with the time stamp. Furthermore, data coming from DART buoys and pressure gauges have different acquisition samplings, specified by the T acquisition code.

4.4.1 Acquisition code

The T code indicates the mode in which the elevation height was acquired. When equal to "9" the data is the output of the ASAT tool.

T code	Acquisition mode
0	Non-DART data converted into DART/ASAT format
1	DART data in 15-minute "normal" measurement mode
2	DART data at 1-minute measurement. Tsunami event mode.
3	DART data at 15-second measurement. Seismic event mode.
9	This file is an output from ASAT tool. Data can be exported to Excel.

4.4.2 *Conversion to DART-like format*

Under 'File' -> 'Convert a File into DART-like' there is the possibility to convert a two-row file into this ASAT/DART format thus making it easier to uniformly organize data files. The user selects the specific file and is then asked about the time information within the file. Information about the starting date and time of the time series must be given as well as the time unit of the original file.

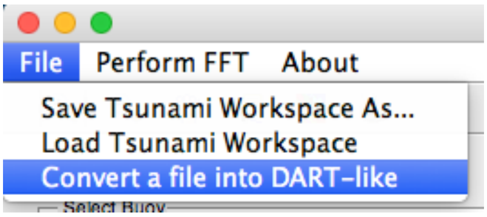


Figure 5 DART-like format conversion

After this you will be able to save the converted file in whichever directory you like.

Note: conversion into this file format was conceived if your original file has two columns: one with the time information and another with the water height elevation. For other files it is easily possible to rearrange data using a spreadsheet. Unfortunately, it is not possible for the tool to embrace all the different possibilities of data arrangements.

4.5 Time series patching

After loading data, four time-series are normally present in the workspace.

Time series name	Type of data
ts	Raw data, with no filtering or interpolation.
ts1	Resampled data with constant time step, provided by the user.
ts_detided	Result of applying polynomial fitting as the de-tiding method.
ts2	Truncated version of ts_detided used for FFT analysis
ts_filtered	Result of applying two de-tiding methods: polynomial fitting and band-pass filtering to ts2.

In all of these time series the time array is given in serial date number. Serial date number arrays represent each point in time as the number of days from January 0, 0000. The numeric values also represent elapsed time in units of days.

4.6 Output text files

The output files are automatically available in the ./output files directory. There are named with the combination of the input file name and the date and time in which the algorithm ran these data.

Their contents are displayed in the same fashion as the input files which is the same as saying they are in ASAT/DART-like format. There is one file for each of the time series specified in section 4.5 (above): interpolated, de-tided and filtered data. The exception to this is the L1 results, which contains extra columns combining all the time series into one file and has an extra column with the sequential time starting from the staring date & time.

On the L1 results text files the presence of an extra sequential time column makes it easier to import data, for example, to Origin or Excel platforms. Example:

```
year  month  day  hour  minute  second seq.time(hours) interpolated  detided   detided&filtered
2013   10    25    17     8       0          0           5125.123    -0.0014642   -0.00012601
2013   10    25    17     9       0      0.016667         5125.134     0.0031989   -0.0001686
```

Furthermore, as for the input files, the Tab Delimited Text (.txt) files are the standard for this algorithm.

4.7 Polynomial Fitting

Upon loading, data is fitted into a polynomial. The best polynomial is automatically generated by the devoted code. This is done by recursively calculate the error contained in delta (see MATLAB polyfit function specifications) and averaging it under a predefined threshold. This threshold was chosen by the team to be 1.1%.

S is the error estimation structure. This optional output structure is primarily used as an input to the polyval function to obtain error estimates. S contains the following fields:

Field	Description
R	Triangular factor from a QR decomposition of the Vandermonde matrix of x
df	Degrees of freedom
normr	Norm of the residuals

If the data in y is random, then an estimate of the covariance matrix of p is $(R_{inv}*R_{inv}')*r^2/df$, where R_{inv} is the inverse of R.

If the errors in the data in y are independent and normal with constant variance, then [y,delta] = polyval(...) produces error bounds that contain at least 50% of the predictions. That is, y ± delta contains at least 50% of the predictions of future observations at x.

What we did to automate the quality of this fitting is to average the delta parameter along the series to be bellow the predefined threshold.

The user can also pre-select a fixed degree for the polynomial fitting under ‘De-tiding’ -> ‘Set Polynom Degree’ -> ‘Degree 16’.

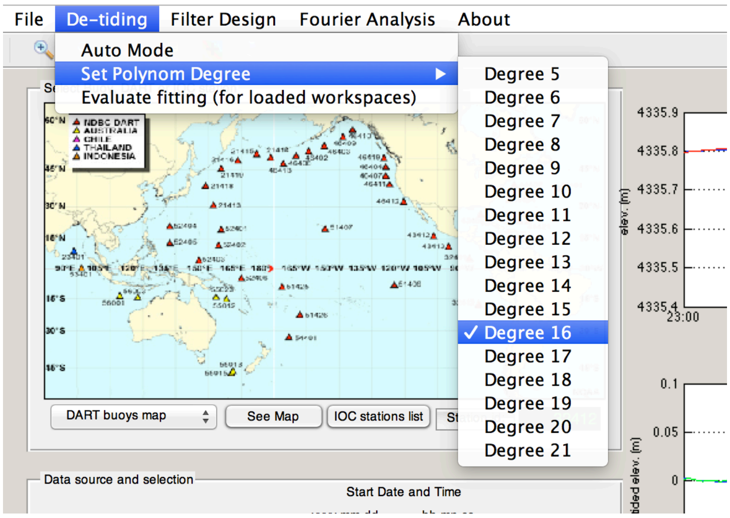


Figure 6 Selection of fixed polynomial degree.

4.8 Polynomial Fitting Interface

To access the quality of fitting of a created workspace there is a devoted interface. Note: This is for loaded workspaces only (in non-standalone versions the workspace must be loaded).

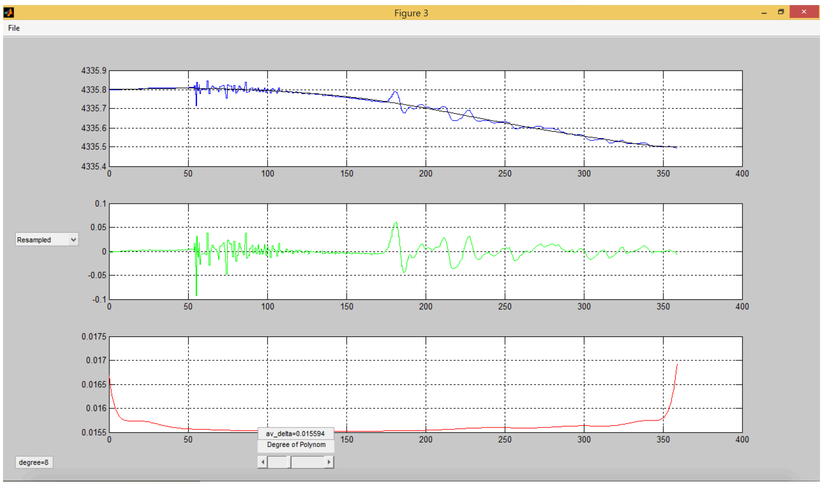


Figure 7 Polynomial Fitting Evaluation Interface.

This interface, given from the numerical variables in the workspace, displays the polynomial de-tiding process. Taking from the input data (either resampled or raw), it extracts the tide according to the given polynomial and displays the delta variable. This delta variable is a good measure of the overall de-tinding process as previously described (section 4.7).

4.9 Signal Analysis

The ASAT tool provides two fundamental types of signal analysis processes, the Fast Fourier Transform (FFT) and Wavelets. FFTs are very powerful in the precise identification of frequencies within a static signal whereas Wavelets are better suited for non-static, i.e. transient phenomena. Given that the de-tided signals of water elevation contain many non-static signals, the Wavelet is the first signal analysis the user is provided with. Because of this, standalone procedures automatically include the analysis through Wavelets. On a second phase the user can select a shorter period within the plotted time-series where the signal is somewhat static and perform an FFT to calculate, better than Wavelets, precise frequencies in the signal.
In this section we take the logical procedure to present the Fast Fourier Transform first and the Wavelets, since the last is a modification of the first.

4.9.1 Fast Fourier Transform

The fundamental idea behind Fourier Transform is that any kind of data that varies in time can be transformed into a different domain called the frequency space. This is very useful allowing observing periodic phenomena. It has been proven that, for any time-varying function f(t) can be written as a summation of sine and cosine terms of increasing frequency.

Short theory background

Although the MATLAB algorithm is well documented in MathWorks page, we found necessary to have a detailed look into the FFT routine since it is black-boxed by the interface. It might be useful for the reader to revisit the mathematical approach and save time whilst performing the basic functions of the routine: analyse its results; perform data visualization and re-organize outputs.
The FFT is "fast" with respect to the Discrete Fourier Transform, so that it uses some optimized algorithms to do exactly the same as the DTF but in much less computational time. As a parenthesis, the DFT is the equivalent of the Fourier Transform of a continuous signal that changes over time, f(t):

$$\boldsymbol{F(i\omega)} = \int_{-\infty}^{\infty} \boldsymbol{f(t)e^{-i\omega t}dt}$$

eq. 4-1

For a discrete, sampled version of the signal: $f[0], f[1], f[2], \dots, f[k], \dots, f[N-1]$;

$$\begin{aligned} F(i\omega) &= \int_{0}^{(N-1)T} f(t)e^{-i\omega t}dt \\ &= f[0]e^{-i0} + f[1]e^{-i\omega T} + \cdots + f[k]e^{-i\omega kT} + \cdots + f[N-1]e^{-i\omega(N-1)T} \end{aligned}$$

Thus, the general expression for the Discrete Fourier Transform is:

$$F(j\omega) = \sum_{k=0}^{N-1} f[k]e^{-i\omega kT}$$

eq. 4-2

We can evaluate this for any ω, but with only N data points to start with, only N final outputs will be significant.

The time taken to evaluate a DFT on a digital computer depends on the number of multiplications involved, since these are the slowest of operations. With the DFT this convergence is dominated by N^2.

More efficient algorithms have been developed more than fifty years ago. As mentioned before, these do exactly the same thing but in much less time, typically with a shorter, $O(n \log n)$, convergence.

If we rewrite

$$F(n) = \sum_{k=0}^{N-1} f[k]e^{-i\omega kT}$$

(the common form of eq. 4-2 with $n = i\omega$) as,

$$F(n) = \sum_{k=0}^{N-1} f[k]\, W_N^{nk}$$

Were W_N^{nk} are periodic functions calculated many times as the computation proceeds. Firstly the integer product nk repeats for different combinations of k and n; secondly, W_N^{nk} is a periodic function with only N distinct values.

4.9.1.1 Implementation

The FFT function is a built-in function of MATLAB. This means that the general proprieties found in the created routine come from the delivery with some minor modifications. Therefore, the FFT algorithm in this tool uses a set of variables:

- Input:
  The input must be a set of data, ts.Data, at least of one dimension.
- Output:
  The output of the FFT module gives three variables: Y, FFT_abs and period_freq_min.

The FFT ASAT function is named tsunamiFFT.m and can be called on the non-standalone versions by:

```
[Y,FFT_abs,period_freq_min]=tsunamiFFT(signal,step)
```

Variable Y is the complex 1 dimensional array of the FFT result, FF_abs is the module of Y and period_freq_min is the variable with the analised periods in minutes (positive only). The input

`signal` can be one of the time series described in section 4.5 and `step` is the time-step of the data series in minutes.
The variables are displayed in Figure 4-1

```
>> whos Y
  Name             Size                 Bytes  Class     Attributes

  Y            16384x1                 262144  double    complex

>> whos FFT_abs
  Name               Size                Bytes  Class     Attributes

  FFT_abs        8193x1                  65544  double

>> whos period_freq_min
  Name                    Size                Bytes  Class     Attributes

  period_freq_min         1x4                    32  double
```

Figure 4-8 The tool outputs a structure containing the result of the FFT as well as an array of the frequency domain, period_freq_min.

A known signal was used to test the algorithm:

$$y(x) = \cos(2\pi.10.x) + \cos(2\pi.25.x) + \cos(2\pi.50.x) + \cos(2\pi.100.x)$$

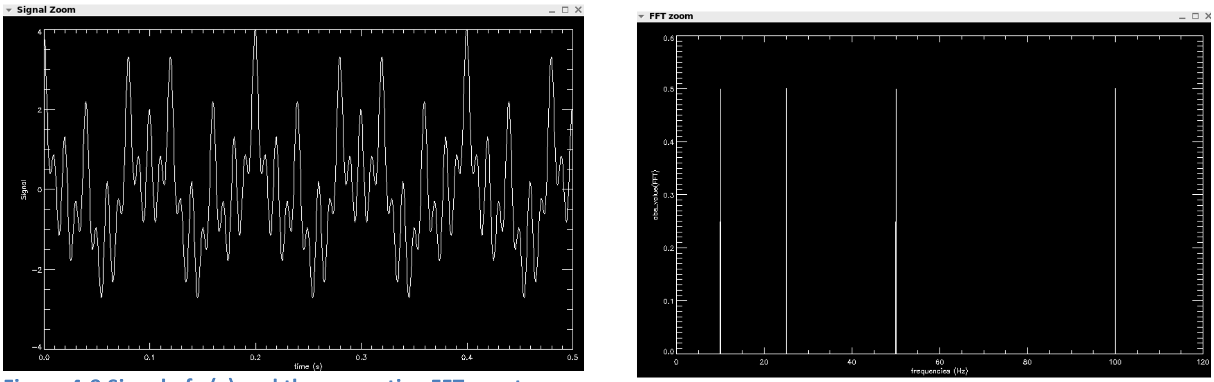


Figure 4-9 Signal of y(x) and the respective FFT spectrum

To the signal y we added a white noise generator creating the data in y2.

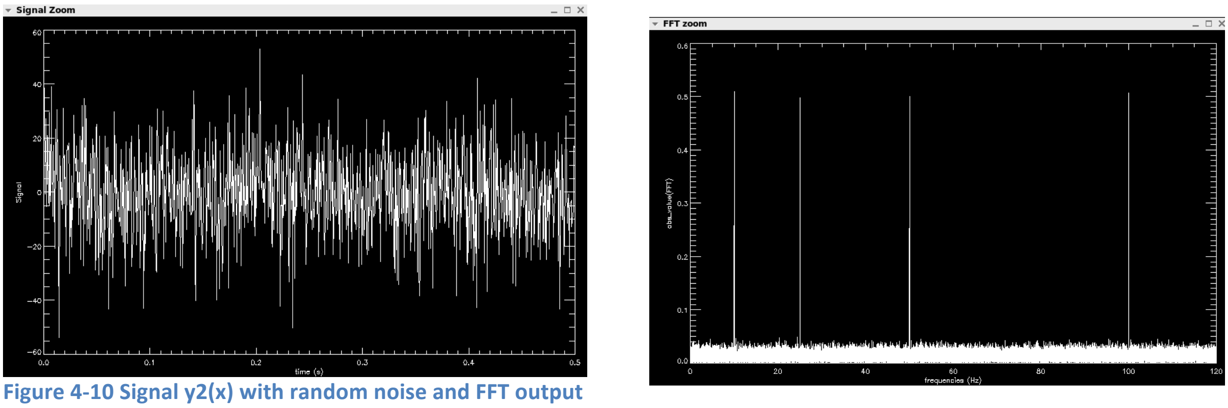


Figure 4-10 Signal y2(x) with random noise and FFT output

Figure 4-9 and Figure 4-10 represent the signal and FFT output from the created tool, respectively with and without noise. In many tasks it is important not only to calculate peak frequencies hidden in the signals but also the strength of the background noise. Our tool was found useful to assess the signal/noise ratios.

4.9.1.2 FFT handling

`step` is an important variable in the ASAT environment describing the sampling frequency in minutes after resampling. From this the set of variables is defined for the FFT analysis:

```
Fs=step/60.;             % Sampling frequency in Hz
L=length(signal);        % Length of the signal
NFFT = 2^nextpow2(L);    % Next power of 2 from length of L
Y = fft(signal,NFFT)/L; % Calculation of the FFT
f = Fs/2*linspace(0,1,NFFT/2+1); % Creation of the frequency array
```

4.9.1.3 FFT output file

An output file is generated from the FFT routine. This output file is tab delimited and can be imported into MS Excel or equivalent. Into this file, three columns contain the result of the analysis. The first column has the array of frequencies in Hz, the second the absolute single side amplitude of the analysis, and the third column has the complex array of the FFT result.

	A	B	C
1	f(Hz)	ABS	FFTy
2	0	0,00038673	-0,00038673
3	6,51E-05	0,00053332	0,0004103-0,0003407i
4	0,00013021	0,0009443	-0,00075629+0,00056545i
5	0,00019531	0,0008066	0,00080649-1,3485e-05i
6	0,00026042	0,00015345	-0,00015314-9,6755e-06i
7	0,00032552	0,00011042	-3,809e-05-0,00010364i
8	0,00039063	0,00044711	0,00019088+0,00040432i
9	0,00045573	0,00028026	0,00027086-7,2001e-05i
10	0,00052083	0,00046684	0,00039103-0,00025502i
11	0,00058594	0,00036258	-0,00021562-0,00029151i
12	0,00065104	0,00022009	-0,00022005-3,7517e-06i
13	0,00071615	0,00050536	0,00010557+0,00049421i
14	0,00078125	0,00061025	0,00058837-0,00016194i
15	0,00084635	0,0005842	0,00013142-0,00056922i
16	0,00091146	0,00056092	-0,00049172-0,0002699i
17	0,00097656	0,00045267	-0,00032529+0,0003148i
18	0,0010417	0,00059128	0,00031173+0,00050243i
19	0,0011068	0,00065476	0,00059102-0,00028179i
20	0,0011719	0,00066376	-6,363e-05-0,0006607i
21	0,001237	0,00068062	-0,00068062-3,8179e-07i
22	0,0013021	0,00057516	-6,8117e-05+0,00057111i
23	0,0013672	0,00071122	0,00063428+0,00032175i
24	0,0014323	0,00080755	0,00041918-0,00069023i
25	0,0014974	0,00060031	-0,00041028-0,00043823i
26	0,0015625	0,00059703	-0,00056641+0,00018873i

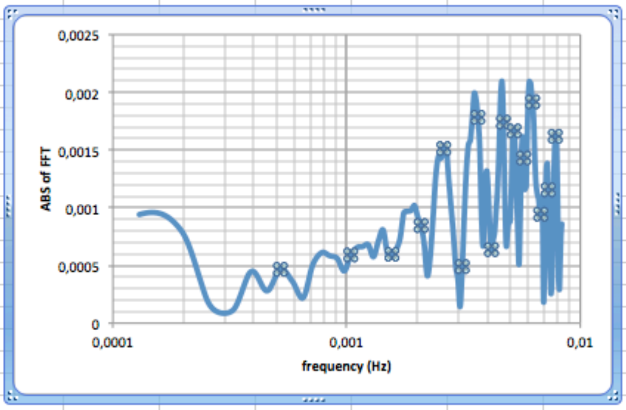


Figure 11 Importation of the FFT results file into Excel.

The name of this file is FFT_results.txt. It can be found into the ./output_files directory.

4.9.2 *Wavelet Transform*

Both Fourier analysis and Wavelets are known for their powerful ability to study many different signals. Fourier analysis efficiently converts the time series into frequency-domain spectra, allowing the identification of peak-frequencies. But this analysis has a serious drawback: when applying the transform function, the time information is lost, such as the information about when particular frequencies occur in time. On the other hand, wavelets have the advantage of providing information of when a particular wave component is found, even if it appears only once in the time series. Hence, it can be better suited to the analysis of transient signals like tsunami waves. In this study, we present the results of wavelet analysis using a Morlet mother function.

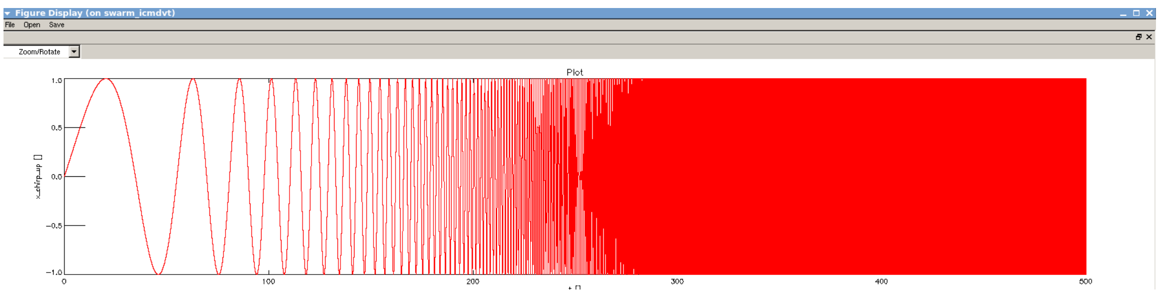
Example:

FFT is not useful for non-stationary signals such as the one in the following example: a so-called "chirp function" that linearly changes frequency in time.

In signals like this one, the FFT will not give useful results as the time analysis takes an all-spectrum look.

$$x(t) = \sin\left(2\pi f_0 \frac{k^t - 1}{\ln(k)}\right)$$

eq. 4-3



$$f(t) = f_0 + kt$$

eq. 4-4

In this case it's not possible to detect single-frequencies of the signal because they rather change continuously in time.

The continuous Wavelet Transform is a mathematical tool that uses the integral of the signal-function f(t) at a scale $a \in R^n$ and a translational value $b \in R^n$ expressed in the following manner:

$$W_{\Psi|f}(a,b) = \int_{-\infty}^{+\infty} \frac{1}{\sqrt{a}} \Psi^{*}\left(\frac{t-b}{a}\right) f(t)\, dt \quad a, b \in R, a > 0$$

eq. 4-5

Where $f(t)$ is a continuous function in both time and frequency domains. $\psi(t)$ is called the mother function wavelet and the asterisk represents the operation of complex conjugate. The purpose of this mother wavelet is to provide a source function to generate the daughter wavelets, which are the translated and scaled versions of the mother wavelet. In this way, unlike Fourier transforms, Wavelet transforms have the power to construct a time-frequency of a signal.

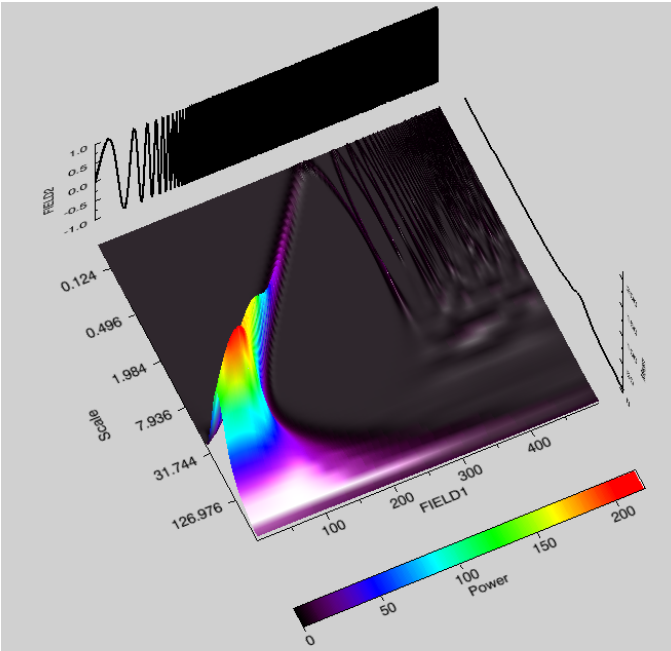


Figure 4-12 Example of the analysis interface with magnitude, period-scale and time-dependence of the signal

In this example, the variables relative to expression eq. 4-3 have been created into MATLAB.
The routines for the Wavelet transform used in the Standalone version result from adaption of code from Torrence and Compo (Compo).

There are different types of wavelets that can be used depending on the research goals. Wavelet functions are subdivided into discrete and continuous, and the latter may be real or complex functions. Examples of discrete wavelets include the Daubechies, Haar, ..., and continuous wavelets are, for example, the Mexican hat (real) or the Morlet (complex) amongst many others. In the vast majority of tsunami signals we recommend the use the Morlet wavelet function, consisting of a modulated Gaussian, or in other words, a complex exponential function multiplied by a Gaussian window. It was first introduced in Goupillaud et al. (1984). In the case of the Morlet 'mother wavelet' the definition is given by the following equation:

$$\Psi_0(t) = \pi^{-1/4} e^{i\omega_0 t} e^{-t^2/2}$$

The real part of a Morlet wavelet is shown in Figure 4-1, given by the product of a cosine function by a Gaussian:

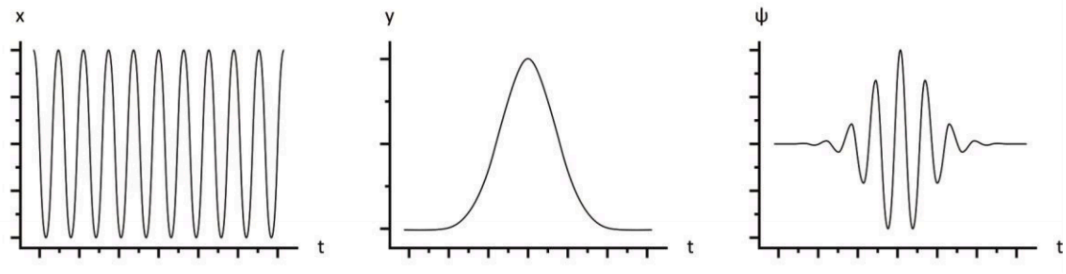


Figure 13 Real part of a Morlet wavelet: the function on the represents the real part of a Morlet wavelet, and results from the product of a cosine function

In most of tsunami research cases, when the phase of a signal ought to be studied, the use of a complex wavelet is crucial, as it yields a complex transform, with information on both the amplitude and phase. With real functions, there is no access to the phase information, available in the imaginary component. It is also important to note an inverse relationship between f (which is a unit easier to visualize) and ω, *i.e.* when it a function is expanded in the time domain it is also contracting in the frequencies space. This relationship is general and comes from the transform properties. For the particular choice ω_0= 6 we have f = $\omega_0/(2\pi s)$~1/s. With this ω_0, there is a very simple one-to-one relation between scale and frequency and we can use both terms interchangeably. This makes the interpretation of results more perceptible. The Morlet wavelet has an important property: it has optimal joint time-frequency concentration. The Heisenberg principle says that one cannot be simultaneously precise in the time and the frequency domain. Theoretically, the time-frequency resolution of the continuous wavelet transform is bounded by the so-called Heisenberg box. A "Heisenberg box" is located in the timefrequency plane: a rectangle with a time width and a frequency height. It represents a timefrequency localization. The area of the Heisenberg box describes the trade-off relationship between time and frequency and is minimized with the choice of the Morlet wavelet, since it is, in its essence, a Gaussian.

4.9.3 Wavelet Spectrum Plots

The most immediate result from the Standalone analysis is the one given by our Wavelet colour plot. In this Wavelet window the de-tided and/or filtered timeseries is presented along with the Wavelet result and the global wavelet spectrum.

Wavelet results can be presented in amplitude, Energy Density or logarithmic scales of the Energy Density.

While the amplitude generally leads to the interpretation of maximal wave heights (both positive and negative) the Eenrgy Density represents the squared version:

$$ED\left(f,t\right) = \left\|W_{\Psi|f}\left(f,t\right)\right\|^2$$

Representing the Energy Density might allow to better distinguish signal from noise whereas the logarithmic scale plots give better details of the Wavelet analyses.

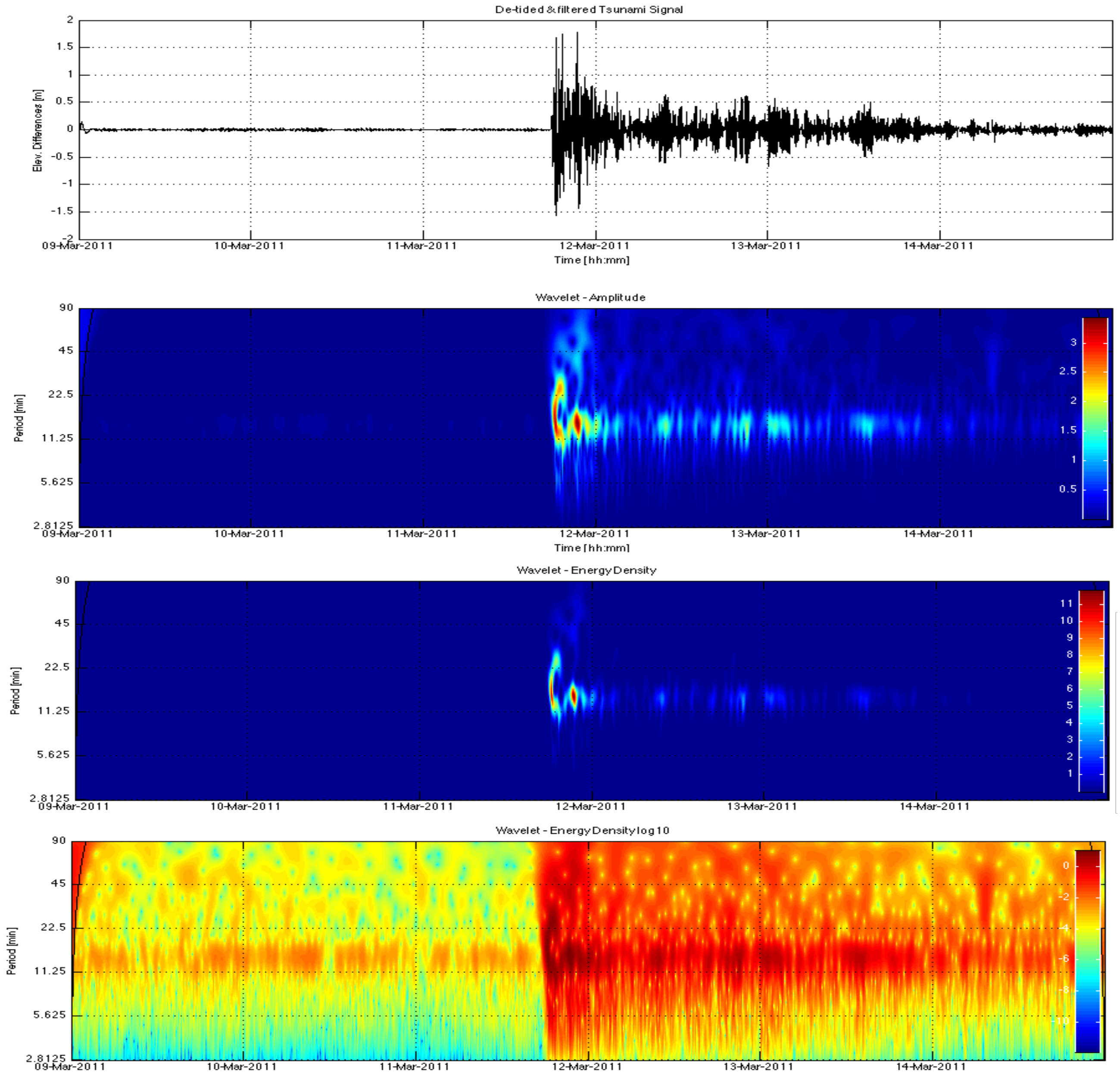


Figure 14 The same wavelet results display differently according to the different options: absolute part of the transform, energy density or logarithm of energy density. In these plots real data and ASAT results are displayed for station Nuku Hai (IOC code: nuku) during Tohoku event of 2011.

NOTE: To improve the display of the excitation mode on the wavelet EDS, the 95 per cent confidence level is highlighted. To estimate this confidence level, the hypothesis of a red noise is made (Allen & Greenslade 2009). The red noise model that is used is the univariate lag-1 autoregressive (AR-1 or Markov) process

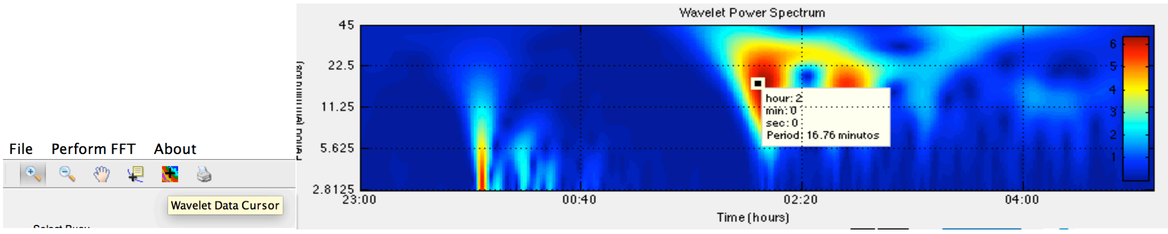
$x_n = \alpha * x_{n-1} + z_n$, (5) where α is assumed lag-1 autocorrelation. We assume that the red

noise follows a χ^2 law with two degrees of freedom.

4.10 **Data Display**

4.10.1 Data Cursors

Data cursors were added to the Tool Bar. These data cursors are specially configured to the time domain. Within the code, data series are always patched with Julian date but, upon plotting, these values are converted into hours, minutes and seconds for easier viewing. For wavelets, periods are indicated in minutes and for the data series the height is metres.



In the case of the FFT plots, whilst frequencies in Hz are displayed on the x axis, data cursor converts to the specific period in minutes, thus saving the need of conversion.

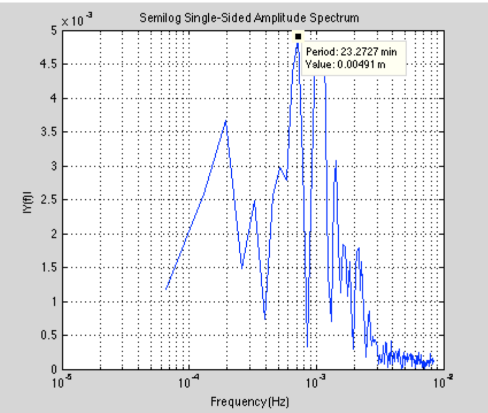


Figure 15 FFT plot with Data Cursor

5 DEVELOPPER'S INFORMATION

Unlike the User's Guide Section, this section is intended to give overall information about the code of ASAT tool. We strongly encourage all authorized intentions of code manipulation that can result in an enhancement of the tool.

5.1 General Architecture of the Code

The ASAT package is neatly organized. Whenever the tool should launched from within MATLAB the full compiling authorization must be given to all the files. This can be done by running the file START.m or by changing the current directory to ASAT folder and writing the command: `addpath(genpath(pwd))`.

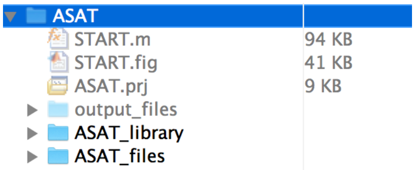
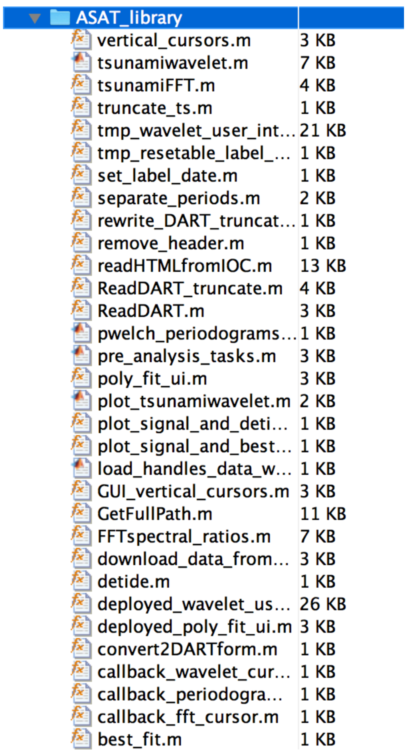


Figure 16 ASAT package and its contents.

START.m file is the main file for the package. By running it, the interface will be displayed. START.fig is the figure interface file for visual organisation of the interface. As in the Standalone Version, the 'output_files' folder is automatically created by if it does not exist in the ASAT package.

The ASAT_library folder contains the main routines of the tool:



Each of the code files in this directory contains vital functions for the functioning of the tool. Changing these files requires caution.

Please consider a careful read of the following table.

name	type	What does it do
vertical_cursors.m	function	that creates the vertical red lines in the wavelet plots to relate them with the de-tided data
tsunamiwavelet.m	routine	routine that results from the adaptation of the code from (Compo) and provides the wavelet analysis variables
tsunamiFFT.m	function	function that provides the FFT results
truncate_ts.m	function	that restricts the original data file to the desired time-span
tmp_wavelet_user_interface.m	function	will be discontinued and replaced by wavelet_user_interface.m
tmp_resetable_label_dates.m	function	will be discontinued
set_label_date.m	function	the system uses sequential date timing. This function converts those that numbers into date arrays time-labelling when plotting.
separate_periods.m	function	This function identifies and separates different sampling times. For example,
rewrite_DART_truncate.m	function	Rewrites a DART file with the truncated time span
remove_header.m	function	Removes header from a DART files before reading it, if it exists.
readHTMLfromIOC.m	function	Reads IOC data from the HTML data delivery content and automatically creates the tool's input variables.
read_DART_truncate.m	function	Reads and creates the variables from a DART variable for a specific time-span without creating a file.
readDART.m	function	Reads and creates the variables from a DART file with no time truncation and no file creation.
pwelch_periodograms.m	routine	Performs an analysis of the data using windowing through a Welch algorithm. This procedure also creates the associated plots.
pre_analysis_tasks.m	procedure	routine that gathers the de-tiding processes in one: polynomial fitting and filtering
poly_fit_ui.m	function	Polynomial fitting interface function for evaluation of the de-tiding processes in the workspaces

plot_tsunamiwavelet.m	procedure	plot tsunami wavelet results
plot_signal_and_detide.m	function	to plot signal and de-tide results
plot_signal_and_best_detide.m	function	plot signal and best de-tide solution from threshold definition
load_handles_data_workspaces.m	procedure	to (re)create all the variables from the handles structures
GUI_vertical_cursors.m	function	function responsible for the creation of the (three) red vertical lines in the interface. These lines relate the resampled data, with the de-tided data and wavelet results.
GetFullPath.m	function	Function that retrieves the full path of a file (normally used for the text data files)
FFT_spectral_ratios.m	function	Function that creates the spectral ratios from the signal and background
download_data_from_NOAA.m	function	function that downloads, unpacks and deliveries data files from NOAA's site. Either for historical or real time data: `http://www.ndbc.noaa.gov/data/`
detide.m	function	Simple de-tiding from polynomial
deployed_wavelet_user_interface.m	function	function that launches the Wavelet visual interface in the deployed (standalone) version
deployed_poly_fit_ui.m	function	deployed version of poly_fit_ui.m
convert2DARTform.m	function	Function that converts a two-column (time and data) file into a DART/ASAT format
callback_wavelet_cursor.m	function	function that creates appropriate data tips for the wavelet plots
callback_periodogram_cursor.m	function	creates appropriate data tips for the periodogram plots' cursors
callback_fft_cursor.m	function	creates appropriate data tips for the FFT plots' cursors
best_fit.m	function	function that calculates the best de-tiding polynomial degree solution according to the end result threshold

The folder ASAT_files contains important files such as images used by the interface, the FILTER.mat file with the bandpass filter structure and the User Manual PDF file.

Filipe Lisboa, IPMA

5.2 Some practical changes

We present here some of the most probable functions that might be changed for a customised used of the ASAT tool.

5.2.1 Change the Wavelet parameterization and period range

Functions tsunamiwavelet.m and plot_tsunamiwavelet.m work together in order to provide fast-made Wavelet plots. The wavelet parameters are the following:

```
pad = 1;                 % pad the time series with zeroes (recommended)
dj = 15./(60.*24.);      % the spacing between discrete scales i.e. the spacing
                         of the time domain in days. Default is 15 seconds.
                         % A smaller # will give better scale resolution, but be
                         slower to plot.
s0 = 2.5*dt;             % This si the period scale staring point. dt is the
                         sampling of the timeseries
j1 = 7/dj;               % this says do 7 powers-of-two with dj sub-octaves each
                         (Wavelet will stop at 360 s with dj=15s)
lag1 = 0.72;             % lag-1 autocorrelation for red noise background
```

The Wavelet mother function is given by the interface and given the structure field name variable 'mother'.

The way Wavelets are displayed is independent from this parameterization and can be found in plot_tsunamiwavelet.m . If Wavelet plots appear with horizontal white areas, the wavelet parametrization is such that there are fewer points than those allowed by the ploting code in plot_tsunamiwavelet.m

By default we use MATLAB's imagesc function to 3d-plot the result from Wavelets. The setting can be found between lines 21 and 29:

```
xlabel('Time [hh:mm]')
ylabel('Period [min]')
title('Wavelet Power Spectrum')
set(gca,'XLim',xlim(:))
set(gca,'YLim',log2([2.8/24./60.,90/24./60.]),    ...%[min(period),max(period)]),
...
    'YDir','normal', ... %before 'YDir','reverse', ...
    'YTick',log2(Yticks(:)), ...
    'YTickLabel',60.*24.*Yticks)   % converter ticks em dias para minutos
colorbar('east','Ycolor','white');
```

5.2.2 Change the time scale within the automated plots:

Adress to the function set_label_date.m in /library folder.

```
% set label for x date series
% You should edit this function if you need to change the scale in the
```

```
% stacked plots (signal + Wavelet)
      function set_label_date(dn)

            dt=dn(end)-dn(1);
            hours_long=24.*dt;

            labels = datestr(dn(1:60:end), 15);
            set(gca, 'XTick', dn(1:60:end));
            set(gca, 'XTickLabel', labels);
      end
```

this function is normally set to deal with one minute data. With this sampling every 60 entries of dn array is giving one-hour timelapse. The sampling should be change if the interval is different or the timeseries is so long that the grid is too closely spaced.

From version 8.0 a newly Wavelet User Guide is incorporated were function `deployed_wavelet_user_interface(wave,time,sst,xlim,ts1,period,Yticks,coi,global_ws,global_signif,step)` were function is responsible for giving customised plots. In this function, the user can select the way the wavelet results are seen and the time-labels ticks can be customized.

6 RELEASE HISTORY

```
All code created within ASTARTE project activities by Filipe Lisboa at IPMA,
I.P. - Portuguese Sea and Atmosphere Institute. In the very few cases of other
contributors, appropriate copyright was mentioned.
      - RELEASES HISTORY:
          ver 1.1:
              - fixed bug on loading successive workspaces
          ver 1.2:
              - Algorithm is prepared to import files where timeseries is
              in either ascending or descending order.
          ver 2.0:
              - Corrected mismatch between plot of Wavelet Spectrum and
              Global Wavelet Spectrum
              - Corrected reading of seconds from DART-formatted files. On
              date arrays seconds are the only entry that is not an
              integer. It's float.
          ver 2.1:
              - On GUI, cursor was changed from zooming to display
              time series values.
          ver 2.2:
              - Additional output files:
                      - original imported time series
                      - resampled time series
                      - de-tided time series
          ver 2.3:
```

```
    - Added Tool Bar to print, zoom and drag
    - Added Data cursors to the Tool Bar. These data cursors
    are specially configured to the time domain.
    Within the code, the data series are always patched with
    Julian date, upon plotting these values are converted into
    hours, minutes and seconds for easier viewing.
    For wavelets,  periods are indicated and for the data
    series the height in metres. In the case of the FFT plots,
    whilst frequencies in Hz are displayed, data cursor
    converts to the specific period in minutes,
    thus saving the need of conversion.
ver 2.3.1:
    - bug fix for loading PDF manual
ver 2.3.2:
    - added button for spikes removal
ver 3.0:
    - de-tided and de-tided & filtered plots are now present and
     labeled
ver 3.1:
    - Added the option to customize Wavelet parameters and mother
    function
    - Color bar labels are now in white for better contrast
ver 4.0:
    - Complete the outputting of .txt with the time series
    - IOC data can be imported given the 4-letter code of the
    station in "Station id" and select IOC data (make sure station
    is in operation for the time-period selected)
ver 4.1.1:
    -  Fixed  the  problem  updating  the  selection  table  when  loading
       reloading a workspace
ver 4.1.2:
    - Fixed the hyperlinks opening on Windows standalone
    version
ver 4.1.3:
    - Google Earth stations maps integrated for Standalone
    version only
ver 5.0:
    - First of Not-So-Standalone versions
    - Added a tool for polynomial fitting quality assessment
ver 5.0.1:
    - Minor bug fixes;
ver 5.1:
    - Added filter design menu
ver 5.1.1:
    - Corrected bug that created complications on loading
    previous ASAT made workspaces
ver 5.1.2:
    - On loading new plots are done
    - Performance modifications for the standalone compiled
    versions
ver 5.1.3:
    - The need for the Image Processing Toolbox was removed,
    making the tool a roughly 20% faster
ver 6.0:
    - Some output files are changed.
    - The Raw Data is no longer displayed but rather the
    interpolated (if applicable)
ver 7.0:
    - Added Rabinovich (1997) method for spectral ratios that
```

```
     yields the enhancement of Tsunami periods isolation
ver 7.1:
    - Data is passed between callbacks through the structure
    handles.data If it exists, there is no longer need to
    constantly load workspaces
    -  Fixed a problem when saving workspaces
ver 8.0:
    - Instead of the final Wavelet plot, a Wavelet User
    Interface is now available
```

7 BIBLIOGRAPHY

Compo, C. T. *A Practical Guide to Wavelet Analysis.* University of Colorado. Boulder, Colorado: Program in Atmospheric and Oceanic Sciences.

George Mungov, M. E. *DART® Tsunameter Retrospective and Real-Time Data: A Reflection on 10 Years of Processing in Support of Tsunami Research and Operations .* pegeoph: Pure Appl. Geophys. (2013) .

Singleton, R. (September 1968). *NIST Guide to Available Math Software.* Stanford Research Institute.