Proceedings of the ASME 2015 34th International Conference on Ocean, Offshore and Arctic Engineering
OMAE2015
May 31-June 5, 2015, St. John's, Newfoundland, Canada

OMAE2015-41852

SOFTWARE ANALYSIS TOOLS FOR WAVE SENSORS

James Morrison
Lews Castle College, University of the Highlands and Islands
Stornoway, Isle of Lewis, Scotland

David Christie
Lews Castle College, University of the Highlands and Islands
Stornoway, Isle of Lewis, Scotland

Charles Greenwood
University of the Highlands and Islands
Stornoway, Isle of Lewis, Scotland

Ruairi Maciver
University of the Highlands and Islands
Stornoway, Isle of Lewis, Scotland

Arne Vogler
University of the Highlands and Islands
Stornoway, Isle of Lewis, Scotland

ABSTRACT

This paper presents a set of software tools for interrogating and processing time series data. The functionality of this toolset will be demonstrated using data from a specific deployment involving multiple sensors deployed for a specific time period.

The approach was developed initially for Datawell Waverider MKII/MKII buoys [1] and expanded to include data from acoustic devices in this case Nortek AWACs. Tools of this nature are important to address a specific lack of features in the sensor manufacturers own tools. It also helps to develop standard approaches for dealing with anomalous data from sensors.

These software tools build upon an effective modern interpreted programming language in this case Python which has access to high performance low level libraries. This paper demonstrates the use of these tools applied to a sensor network based on the North West coast of Scotland as described in [2,3].

Examples can be seen of computationally complex data being easily calculated for monthly averages. Analysis down to a wave by wave basis will also be demonstrated form the same source dataset. The tools make use of a flexible data structure called a DataFrame which supports mixed data types, hierarchical and time indexing and is also integrated with modern plotting libraries. This allows sub second querying and the ability for dynamic plotting of large datasets.

By using modern compression techniques and file formats it is possible to process datasets which are larger than memory datasets without the need for a traditional relational database.

The software library shall be of use to a wide variety of industry involved in offshore engineering along with any scientists interested in the coastal environment.

INTRODUCTION

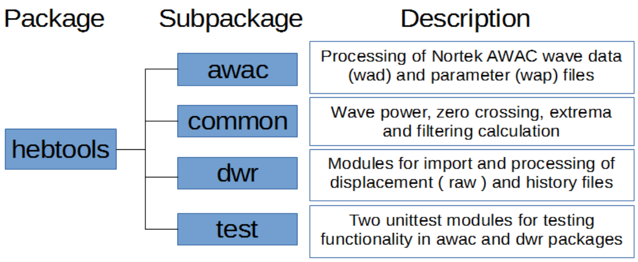
Once data has been recovered from deployed sensors, it can be challenging to transform the data into a common format which allows easy comparative time based analysis. The toolbox discussed here provides a means to import that data into a time indexed data format that offers power option for further processing.

The development of the toolbox hebtools[4] emerged initially from the need to process live Waverider buoy data in near real time and to provide accurate statistics from that data. Using the wave statistics file[5] produced by the RFBuoy receiving software it was clear that there were occasional spikes that were statistically unlikely. During investigating this issue further a method for filtering individual waves by going back to the raw data began to be developed in consultation with engineers from Datawell.

To enable accurate analysis of the buoy data it was necessary to timestamp incoming records as accurately as possible by making use of the starting timestamp from the filename, frequency of output and the number of records written to the text file. Through the use of the DataFrame data format which is part of pandas[6] a powerful data analysis library a time based index was created which is used to time stamp individual records to sub second accuracy.

ARCHITECTURE

The toolbox was developed on both Windows XP/7 and Ubuntu 12.04. The dependencies of the hebtools can be provided by the Anaconda Python distribution from Continuum Analytics[7] a full list of dependencies with versions is available on the repository homepage[5].



Package	Subpackage	Description
hebtools	awac	Processing of Nortek AWAC wave data (wad) and parameter (wap) files
	common	Wave power, zero crossing, extrema and filtering calculation
	dwr	Modules for import and processing of displacement (raw) and history files
	test	Two unittest modules for testing functionality in awac and dwr packages

Figure 1: High level package structure of hebtools

The code has been structured into sensor specific packages as can be seen in Figure 1. There is a certain amount of commonality between the data processing for the different sensors and this has been extracted out into the **common** package.

ANALYSIS

hebtools provides a number of functions for calculating statistics from the raw datasets which facilitates comparison with the established statistics available from the processed outputs of the sensors manufacturers' software.

The recommended approach for analyzing the resulting datasets from hebtools is in IPython[8] which is an enhanced interface to the Python language and one of its most useful features is the ability to create web browser based notebooks. These notebooks enable the combination of code and output in one document, they are a good method of demonstrating the capabilities of a toolbox and if you wish to share the output of your analysis it can be shared in a variety of text formats via the nbconvert library. The method of analysis described above has been used in the production of the datasets for a number of papers including [9,10].

Providing the underlying code, data and library dependencies are available these notebooks add the ability to reproduce the outputs shown relatively easily. As the notebook is a web application this allows powerful inline interactive plotting via JavaScript through libraries like mpld3 which provides an interactive graphical backend to the powerful matplotlib library which offers a MATLAB like plotting experience.

One demonstration of output made possible through hebtools is show in Figure 2 below where a comparison of directional spectra based on vertical displacement data for one buoy across 12 months is shown. This approach gives an indication of how power spectral density is spread across frequencies during various months of the year. These spectra were produced using Welch's method as implemented in the scipy signal package.

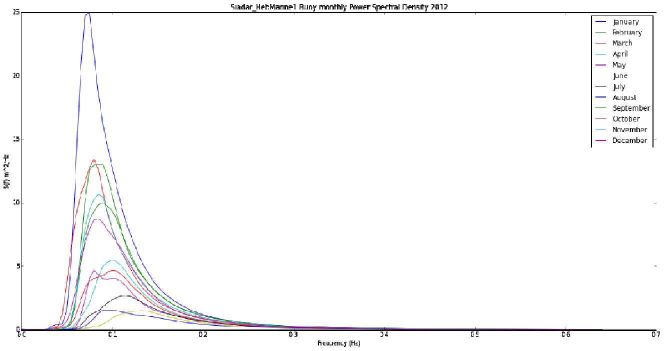


Figure 2: Monthly Averaged Spectra from buoy 3 in 2012

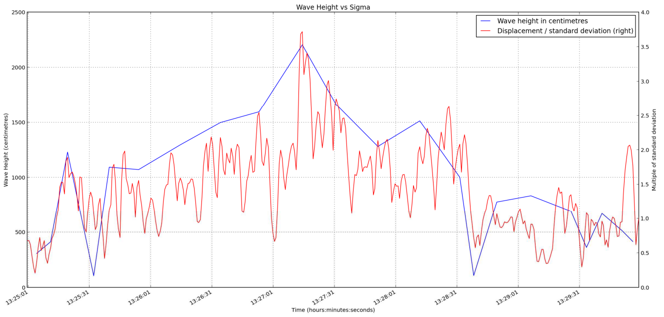


Figure 3: Wave height during an extreme storm

In Figure 3 above we can see data from the same set as was shown in Figure 2 but in this case we have concentrated on a five minute period during February 2013. The blue line shows a wave height approaching 22.5 metres during the peak of a storm, the red line show a comparison between the heave (vertical displacement) signal and the standard deviation of that signal over a 30 minute period. During this period the heave signal never exceeds 4 multiples of its standard deviation which can be considered a statistical upper bound in relation to linear wave theory.

The two different plots in Figures 2 and 3 highlight the versatility of the toolbox and its surrounding libraries. An example of another repository which is complementary to hebtools that has been recently developed is Waverider-to-DIWASP which provides an easy interface to the process of bulk importing the raw buoy data into the MATLAB toolbox DIWASP, this toolbox has proved useful for analyzing the spectral output over time and comparing it with the spectral output from the wave buoy itself and Datawell's own W@ves21 software. The hebtools package has been used as a mean of pre-filtering datasets for this new package.

STORAGE

The toolbox makes use of hdf5 a hierarchical data format which is widely used in domains dealing with very large data sets. This functionality is available through the PyTables[11] library and since version 2.2 of PyTables there is ability to use the high performance compression library which allows efficient use of disk storage while providing very fast access to data.

The pandas library also has the ability to export to MATLAB's mat format although as the DataFrame format doesn't exist in MATLAB some of the querying advantages and metadata of the source format are lost.

CONCLUSIONS

The hebtools package provides a number of tools for working with data from the sensors specified above and has been developed and openly licensed in a way which allows it to be extensible to other sensor data formats.

Through the use of the pandas library and its DataFrame format it is possible to utilise the powerful Scientific Python software stack. The DataFrame format has been inspired by the equivalent data frame structure in R and with that should be relatively intuitive to users that are coming from that background while also providing some basic interoperability with MATLAB's mat format.

hebtools has already been used in coordination with existing spectral toolbox libraries [12,13] and as described above provides a means of pre-filtering datasets and reproducing modified and time shifted raw files.

Further work required in this package is related to improving the filtering process to ensure all records relating to a suspicious wave can be removed simply and improving the documentation to enable the package to be more accessible to potential users.

ACKNOWLEDGMENTS

The authors wish to acknowledge the funding received from the primarily Scottish Funding Council funded Hebridean Marine Energy Futures Project. The author is grateful for the data and support from the Hebridean Marine Energy Futures project partners Aquamarine Power, Lews Castle College and the European Union funded FP7 project MERIKA led by the University of the Highlands and Islands

REFERENCES

[1] Datawell BV 2009 Datawell Waverider Reference Manual, Available at cdip.ucsd.edu/documents/index/gauge_docs/mk3.pdf

[2] Vögler, A., & Venugopal, V. (2012). Hebridean Marine Energy Resources: Wave-power characterisation using a buoy network. OMAE2012-83658 Rio de Janeiro.

[3] Greenwood, C.E., Venugopal, V., Christie, D., Morrison, J., and Vögler, A. (2013) Wave modeling for potential wave energy sites around the Outer Hebrides. In: Proceedings of the 32th International Conference on Ocean, Offshore and Arctic Engineering, OMAE2013-11356, Nantes.

[4] Morrison, J. (2014) hebtools - Time series analysis tools for wave data, Available at https://github.com/seumasmorrison/hebtools

[5] Datawell VB 2013 Datawell File Formats, Available at http://www.datawell.nl/Portals/0/Documents/TechnicalNotes/datawell_technicalnote_fileformats_2013-03-14.pdf

[6] McKinney, W. 2012. Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython, O'Reilly

[7] Continuum Analytics, 2014, Anaconda, Available at https://store.continuum.io/cshop/anaconda/

[8] Perez, F. (2007) "IPython: a system for interactive scientific computing" Computing in Science & Engineering

[9] Vögler, A., Morrison, J., & Venugopal, V. (2013). An Empirical Analysis of Coastal Shoaling Induced Modifications to Wave Climate and its Impact on Wave Power. In The Twenty-third International Offshore and Polar Engineering Conference. International Society of Offshore and Polar Engineers. Anchorage, Alaska

[10] Vögler, A. & Morrison, J. (2013). "Assessment of the Grid Capacity Sharing Potential for Wave and Wind Energy Conversion Systems in the Outer Hebrides of Scotland." European Wave and Tidal Conference, Aalborg, Denmark

[11] Alted, F. (2003) "PyTables: Processing And Analyzing Extremely Large Amounts Of Data In Python" PyCon 2003, Available at http://www.pytables.org/docs/pycon2003-paper.pdf

[12] Brodtkorb, P.A. et al., 2000. WAFO—a Matlab toolbox for analysis of random waves and loads. In Proceedings of the 10th international offshore and polar engineering conference. pp. 343–350.

[13] Johnson, D. (2011). DIWASP a directional wave spectra toolbox Version 1.4 for MATLAB: User Manual. MetOcean Solutions Ltd