

ECloud – Magnetotelluric Webapp

Andrew Pethick*
Curtin University
26 Dick Perry Ave, Kensington
Andrew.Pethick@curtin.edu.au

Brett Harris
Curtin University
26 Dick Perry Ave, Kensington
B.Harris@curtin.edu.au

Petr Lebedev
Sydney University
pleb0344@uni.sydney.edu.au

**presenting author asterisked*

SUMMARY

Geophysical web applications are highly underrated. There are many potential benefits for transitioning to a cloud based web applications including compatibility improvement, low on-going operating costs and better access to existing third-party code and tools for facilitating rapid development. For users, time spent on installation, general IT maintenance such license management and upgrading is minimized.

While web applications for business style applications are deeply established, scientific applications on the web are only just emerging. This research builds upon our previous feasibility study (Pethick and Harris 2015), which was the development of a 1D MT web application. This is to be redevelop into a commercial grade cloud hosted geophysical inversion web app that can cooperatively invert seismic and magnetotelluric (MT) data. Our software, ECloud, is designed to be user friendly where geoscientists and drillers can upload MT field data directly from their laptop, tablet or mobile to obtain subsurface geo-electrical distributions quickly with minimal input. This application is designed to be scalable, suiting cloud environments and is currently hosted on an Amazon EC2 instance. The preconceived notion that web applications are slow will be challenged. The purposefully designed MT algorithm and software structure will hopefully result in lower computation times while minimizing restrictions based on hardware requirements (i.e., primary memory).

Key words: Cloud, MT, Seismic, Cooperative Inversion, Web Application

INTRODUCTION

There are a huge number of web application programming interfaces (API) available at our fingertips such as Google Maps (Google 2016b), Google Charts API (Google 2016a), YouTube API (Google 2016c) and DropZone (DropzoneJS). This access to a large number of robust tools facilitates rapid development, enabling developers to access and integrate robust and easy to use libraries. Taking a web application paradigm, the dangers of piracy are also reduced as the user potentially does not have direct access to proprietary code. Users also benefit from such architectures, as computation can be performed at any location, on device with limited computational ability and with minimal set up.

Deploying on the web also has several advantages, primarily, high end tasks such as forward modelling and inversion can be performed from any HTML compatible device. Secondly, the computational limitations found on even the most power workstations are overcome through taking advantage of the massive computational scalability of cloud servers.

The pool of existing web based geophysical implementations are small but growing, notable applications include the Virtual Geophysical Laboratory (Fraser et al. 2013) and Modelr (Hall 2014). We intend to add to this growing pool of accessible applications by building upon our previous feasibility study (Pethick and Harris 2015), which was the development of a 1D MT forward modelling web application. This is to be redevelop into a commercial grade cloud hosted geophysical inversion web app that can cooperatively invert seismic and magnetotelluric (MT) data. Our software, ECloud, is designed to be user friendly where geoscientists and drillers can upload MT field data directly from their laptop, tablet or mobile to obtain subsurface geo-electrical distributions quickly with minimal input. This application is designed to be scalable, suiting cloud environments and is currently hosted on an Amazon EC2 instance. The preconceived notion that web applications are slow will be challenged. The purposefully designed MT algorithm and software structure will hopefully result in lower computation times while minimizing restrictions based on hardware requirements (i.e., primary memory).

METHOD AND RESULTS

Currently full 1D magnetotelluric inversion has been completed within the ECloud framework. Cooperative seismic inversion is still ongoing. The steps we have taken to develop the cooperative inversion web framework include the following:

1. Develop an MT forward code

We have developed 1D magnetotelluric forward modelling Python code based upon the 1D plane wave derivations of Pedersen and Hermance (1986) and Niwas, Gupta, and Gaur (2005). Overall the algorithm initially solves infinite halfspace impedance, iterating upwards solving the induction parameters, exponential factors and intrinsic impedances of each layer. The top layer impedance is then converted into an apparent resistivity and phase. A comprehensive solution is shown in Pethick and Harris (2015).

2. Develop the MT inverse solution

We have developed code that uses a regularized Marquardt-Levenberg method. The non-linear damped least squares inversion method minimizes the following objective function (1):

$$\phi = q_1 + \beta q_2 = e^T e + \beta (x^T x - L_0^2) \quad (1)$$

Where,

e – Misfit

β – Damping factor

L_0 – Lower bounds

To solve calculate the perturbations of the model parameters for each iteration, we need to differentiate the objective function (1), leading to the solution (2)

$$(A^T A + \beta I)x = A^T y \quad (2)$$

Where,

A – Jacobian

x – Model parameters

y – Data

The model perturbation updates for the geo-electrical parameters for subsequent iterations are shown in (3)

$$x_p = [A^T A + \beta I]^{-1} A^T y \quad (3)$$

The resulting program computes the magnetotelluric plane wave response over an isotropic 1D layered earth. We chose to have an object-oriented structure, breaking the program into several components including the global and inversion specific parameters, forward modelling operator, the inversion, data model and earth model. In this way, specific data import, export and data visualization regimes can be easily added. The overall structure of the Python script is show in Figure 1.

3. Develop an interface for the inversion input

At the core of our web application is the PHP session. PHP sessions contain all inversion parameters/options, earth model's definitions and imported files. The interface was built around the session variables using HTML, PHP and JavaScript. Several libraries and API's were implemented to aid its rapid development. The X-Editable library was used to retrieve user input and to update the PHP session variables while DropzoneJS was used to retrieve user file uploads. The Google Maps API was accessed to displayed each MT station location. Finally, the Omnium tools library was integrated into the application to manage the imported EDI MT data file formats. An overview of the web application framework is shown in Figure 2. Several pages were created including the dashboard showing an overview of the software status. MT & seismic data import, earth model definition, inversion and display settings, execution and analysis pages were developed. A sample screenshot displaying the execution pane of the web application is shown in Figure 3.

4. Seismic Integration

SEG Y support was included through the open source library “Seisplot” (Hall et al., 2015) which integrated the ObsPy python global seismology toolkit (Beyreuther et al., 2010). These were used to import, sort traces and export. Currently only 2D SEG Y models were created. A full overview of the development is seen in Lebedev (2016). The method of use would be to through uploading the SEG Y files using DropZone which would trigger the generation a seismic cross section image. Within earth model creation, users have the option of selecting main structural boundaries to insert into the geo-electrical model. These boundaries will form part of the starting model for inversion. The final results show a co-rendered Seismic and 1D MT inversion output for either the TE or TM modes. See Figure 5.

5. Link the interface with the algorithm

The web interface is linked via PHP to the inversion algorithm. PHP is powerful, it can write JavaScript and HTML dynamically, write and delete files on the server’s disk, execute code residing on the web server, access shell commands and create and delete directories. On the server side, once a user submits an inversion job (1) inversion input files are generated by PHP (2) A shell script command is formulated connecting the inversion code with the newly created files and (3) the command is executed using the shell_exec command. After the code is executed (4) the output files read dynamically when in the analysis page. An example of the generated images is shown in Figure 4.

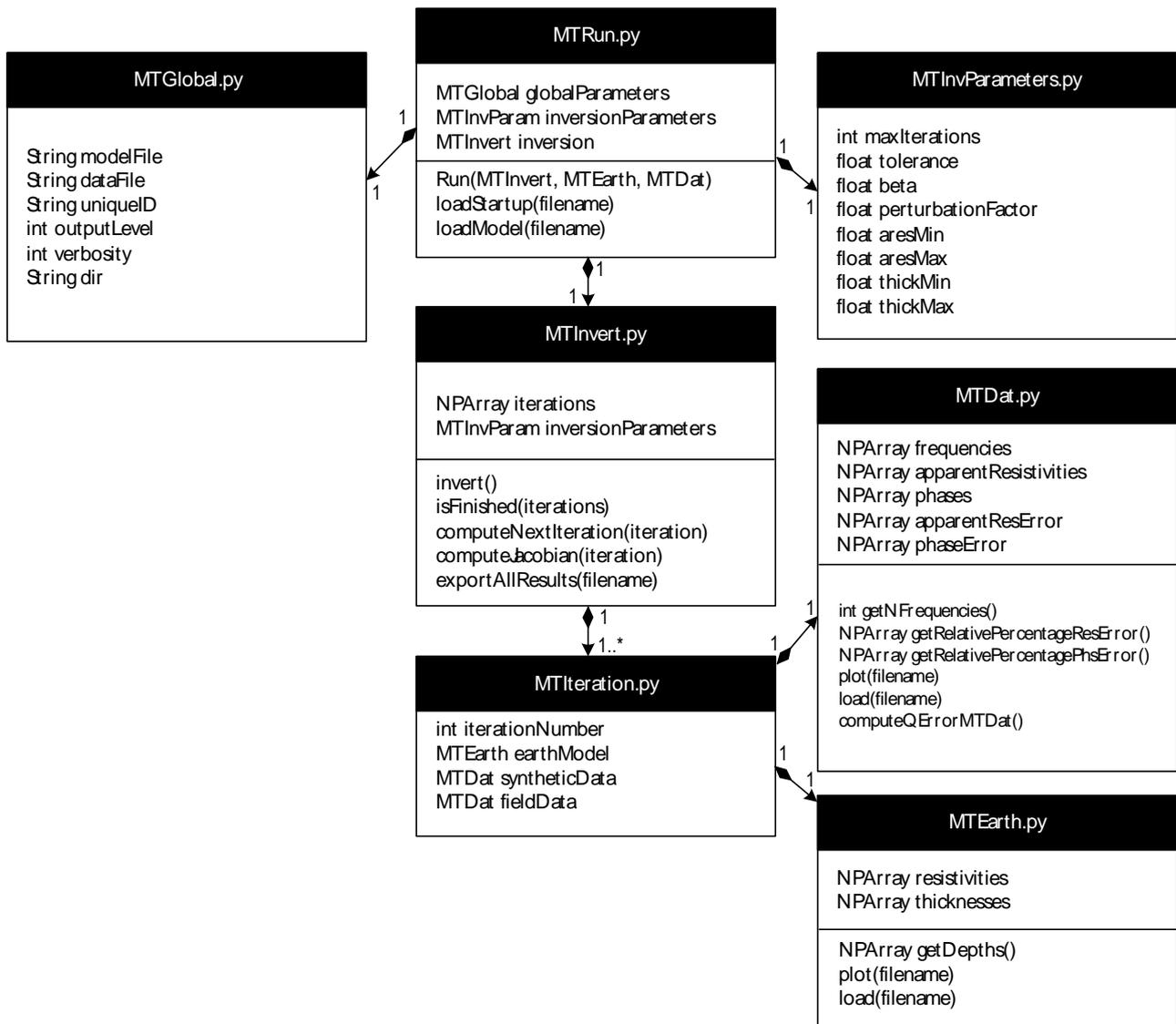


Figure 1 The UML structure of the magnetotelluric 1D inversion Python code

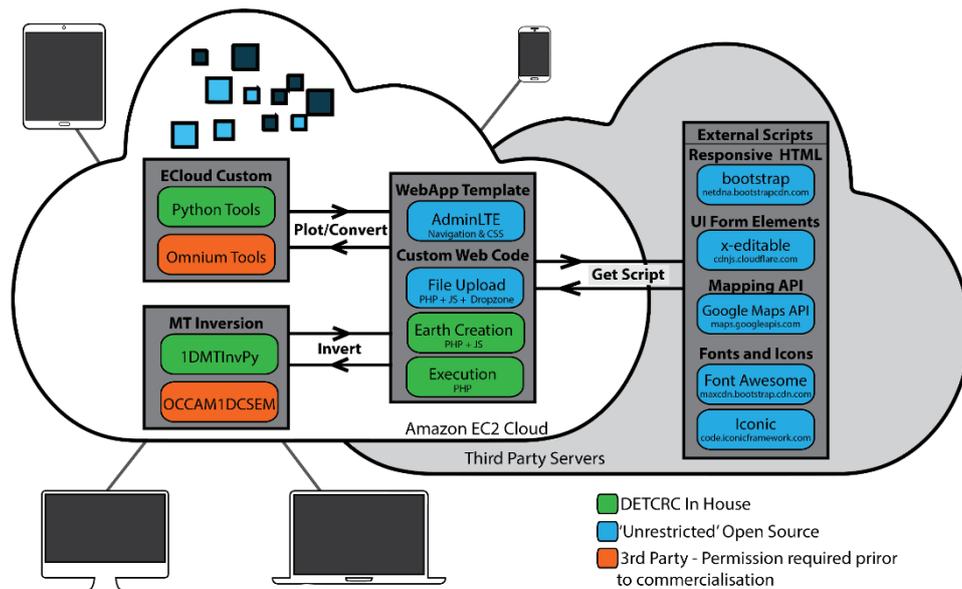


Figure 2 The ECloud framework. The software is sectioned into four groups. (1) The ECloud custom group contains specifically developed code that performs magnetotelluric and seismic data management. (2) The MT Inversion group handles the number crunching to compute the subsurface geoelectrical distribution. (3) The web app and custom web code group contains custom and third-party code that handles interactions between the data model and the front end. (4) External scripts are hosted on third party servers and are accessed when served by a user request on the ECloud web app. ECloud has been designed to be responsive, enabling its usage on multiple devices including mobile phones.

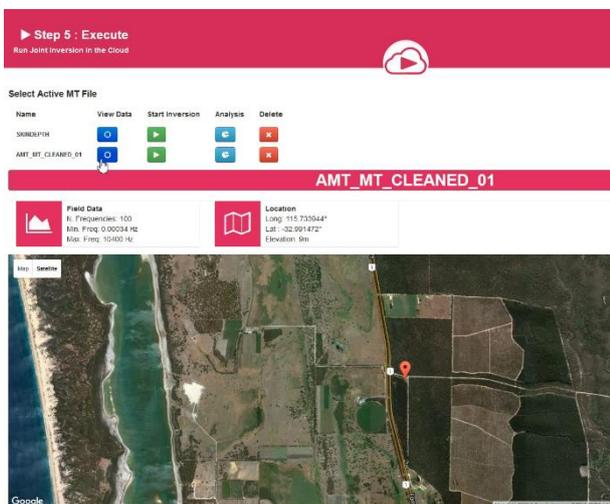


Figure 3 The web application interface showing data execution and display options. A mock location of the MT station is displayed using the Google Maps API.

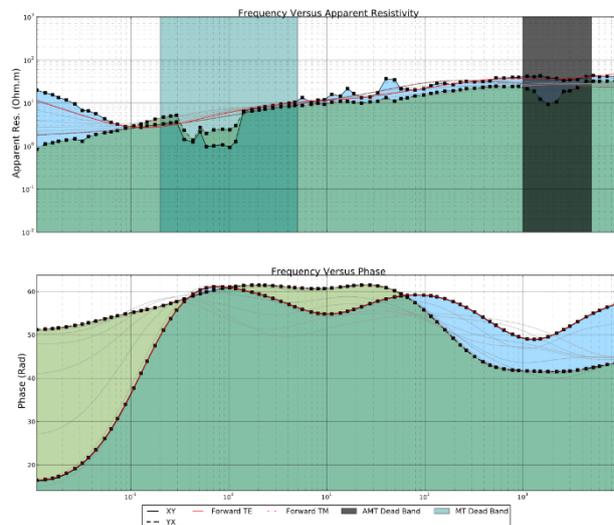


Figure 4 The MT apparent resistivity and phase data for the TE and TM modes with corresponding 1D inverted curves. All iterations are displayed, and as the iteration solution progresses, the curve become progressively opaque. The MT and AMT dead band locations are displayed to aid interpretation.

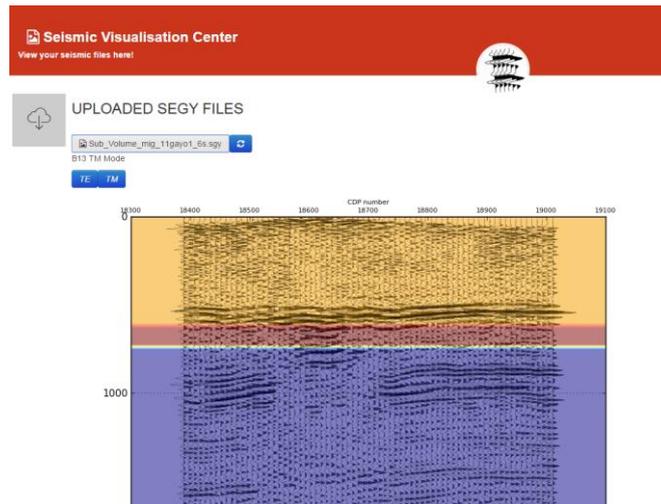


Figure 4 An example co-rendered seismic and MT inversion results (reproduced from Lebedev, 2016)

Conclusions

We have developed a joint/cooperative magnetotelluric and seismic inversion web framework and is currently deployed on an Amazon EC2 server instance. It has been developed with a range of web and scientific technologies including Python, PHP, HTML, JavaScript and Java. Our framework currently enables users to carry out 1D magnetotelluric inversion and 1D seismic forward modelling from any HTML 5 compliant device. It can also be extended for 2D and 3D magnetotelluric inversion and possesses the ability to access the immense computational power of the cloud.

Acknowledgements

The work has been supported by the Deep Exploration Technologies Cooperative Research Centre whose activities are funded by the Australian Government's Cooperative Research Centre Programme. This is DET CRC Document 2017/1043.

REFERENCES

- Du, B., 2000, Comparison of crosswell diffraction tomography to Kirchoff migration: *Exploration Geophysics*, 31, 366-371.
- Hall, M., 2016, Old Skool Plot Tool, retrieved from <https://agilescientific.com/blog/2016/2/2/old-skool-plot-tool>, Aug 2016.
- Hendrick, N., and Hearn, S., 1999, Polarisation analysis: What is it? Why do you need it? How do you do it?: *Exploration Geophysics*, 30, 177-190.
- Leaney, W.S., 1990, Parametric wavefield decomposition and applications: 60th Meeting, SEG, San Francisco, Expanded Abstracts, 1097-1100.
- Lebedev, P. 2016, Cloud Based Cooperative Magnetotelluric and Seismic Inversion, Department of Exploration Geophysics, Honours Thesis.
- Press, W.H., Flannery, B.P., Teukolsky, S.A., and Vetterling, W.T., 1992, *Numerical recipes in Fortran77 2nd edition - The art of scientific computing*: Cambridge University Press.
- Zhou, B., 1988, Crosshole resistivity and acoustic velocity imaging: 2.5-D Helmholtz equation modeling and inversion: Ph.D. Thesis, University of Adelaide.