Development of the MTpy software package for magnetotelluric data analysis

Alison Kirkby*
Geoscience Australia
GPO Box 378
Alison.Kirkby@ga.gov.au

Jared Peacock
United States Geological Survey
Reston, Virginia, United States
peacock.jared@gmail.com

Jingming Duan
Geoscience Australia
GPO Box 378
Jingming.Duan@ga.gov.au

Fei Zhang
Geoscience Australia
GPO Box 378
Fei.Zhang@ga.gov.au

**presenting author*

SUMMARY

The magnetotelluric (MT) method is becoming more widely used in the geoscience community as it becomes increasingly recognised as a useful exploration tool. However, while the analysis and inversion tools available to the MT community have increased over recent years, the software available to work with these tools is still somewhat limited and often costly in comparison to some of the more mature techniques like gravity, magnetics and seismic.

The MTpy python library is open source software that aims to assist MT practitioners in carrying out the processing and analysis steps that need to be carried out with MT data and in working with the various inversion codes that are available. However, MTpy still contains coding issues, bugs and gaps in functionality, which have limited its use to date. We are currently developing MTpy to rectify these problems and expand the functionality, and thus facilitate the use of MT as an exploration technique. Key improvements include adding new functions and modules, refactoring the code to give better quality and consistency, fixing bugs and adding new Graphic User Interfaces.

Key words: magnetotelluric, software, processing, visualisation

INTRODUCTION

The magnetotelluric (MT) method is increasingly being recognised as a valuable geophysical technique for exploration, as data analysis and inversion tools have become more sophisticated. However, the software available for MT data processing and analysis is still very limited in comparison to other geophysical methods such as gravity, magnetics and seismic. Using such software often requires specialist knowledge, and this has impeded the widespread use of this technique. In order to work with MT data, it is often necessary to either purchase expensive software, or write or modify code. There are few inexpensive options available to practitioners who lack the time or programming skills to work with source code.

MTpy is an open source software package that has recently been developed to assist with MT data processing, analysis, modelling, visualisation and interpretation. It was initiated at the University of Adelaide in 2013 as a means to store and share python code amongst the MT community, to enable key processing, analysis and modelling steps that need to be carried out on MT data (Krieger and Peacock, 2014).

Written in the Python programming language, MTpy contains modules to carry out much of the processing and analysis that needs to be applied to MT data, working with the industry standard EDI file format amongst other formats. It also contains modules to create inputs to, and visualise outputs from many of the inversion codes that are currently in use in the MT community. However, despite continued development of this software, much of the code lacks internal unit tests, and contains bugs and gaps. These factors, together with a lack of user interface, have limited its use in the wider MT community to date.

Geoscience Australia in collaboration with developers at the University of Adelaide and the United States Geological Survey continue to develop MTpy to improve its functionality and provide increased user support. The aim is to make the software toolkit easier to work with MT data and thus facilitate the use of MT in the wider geophysics community.

FUNCTIONALITY DEVELOPMENT

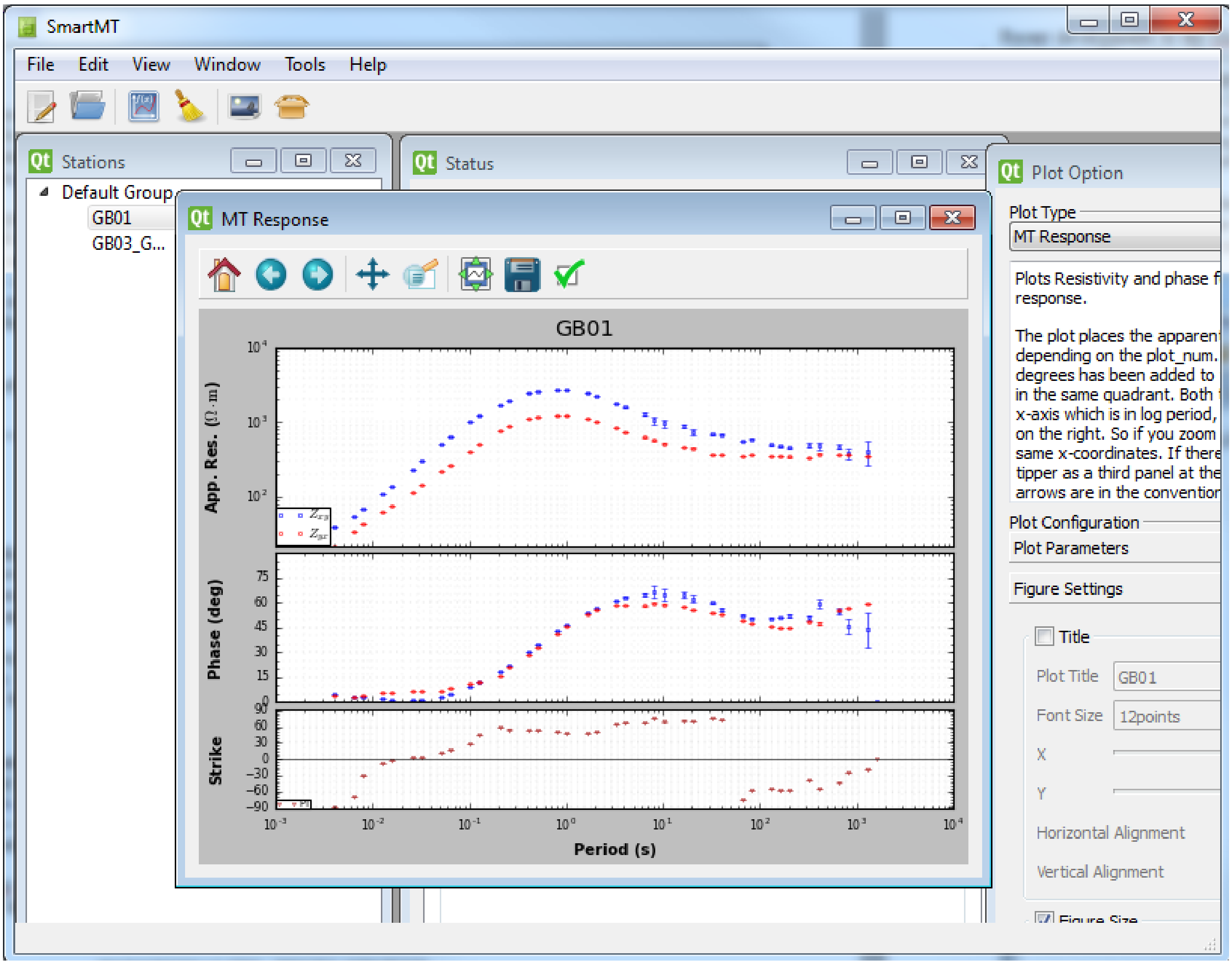
MTpy is structured around key requirements for working with MT data, namely processing, analysis, modelling and imaging. These modules are based upon core modules, which deal with reading and writing impedance tensor data from industry standard formats such as EDI (Krieger and Peacock, 2014).

The processing modules are designed to facilitate working with time series data and generating inputs for existing processing codes, for example, the robust algorithm BIRRP (Chave et al 1987, Chave and Thomson, 2004). A key challenge with time series data is working with and viewing large volumes of data. Planned developments to these modules will help with the managing of these large datasets, likely using high performance data formats such as hdf5.

The analysis modules deal with aspects of MT data analysis, including phase tensor and dimensionality analysis, calculation of impedance tensor invariants, calculation of and removal of distortion, and sensitivity analysis. Recent developments include addition of functions to calculate and visualise depth of investigation.

A number of changes have been added to the modelling modules. The modelling modules in MTpy cover creation of inputs and analysis of outputs for many of the widely used modelling codes used in the MT community. Improvements to the modelling modules over the last 12 months have focused mainly on working with the ModEM code. ModEM is a 3D inversion code developed at Oregon State University, which has been made available to the MT community for academic use (Egbert and Kelbert, 2012, Kelbert et al, 2014). MTpy contains modules to create input data files from EDI files and topographic data, automatically generate a mesh, and to write the necessary model files for running an inversion using ModEM.

Recent developments to the ModEM modules in MTpy by Geoscience Australia include the addition of modules to allow topography and bathymetry to be added to models, and to allow reading and writing of models to and from GOCAD™ sgrid format. GOCAD™ is a software package developed by Paradigm for 3D spatial analysis and geological modelling, and will allow for more detailed analysis of resistivity models produced by ModEM. We have also made developments to the handling of projections in ModEM, to facilitate the creation of large models that cross UTM zones. Finally, updates have been made to the visualisation modules, allowing visualisation of input data and modelled responses as plots showing resistivity and phase as a function of period, and as phase tensor maps. In addition, the modules allow plotting of resistivity models as depth slices and cross sections.



Plot Type: MT Response

Plots Resistivity and phase f
response.

The plot places the apparent
depending on the plot_num.
degrees has been added to
in the same quadrant. Both
x-axis which is in log period,
on the right. So if you zoom
same x-coordinates. If there
tipper as a third panel at the
arrows are in the convention

□ Title

Plot Title: GB01

Font Size: 12points

Figure 1. Example of a Graphical User Interface (GUI) being developed for the mtpy software. This window shows a plot of apparent resistivity and phase, as well as strike angle calculated from the Phase Tensor (Caldwell et al. 2004), for a magnetotelluric station.

As well as adding to the functionality contained in MTpy, we are working to apply software engineering best practices and techniques to improve software quality and usability. In addition, a GUI is being developed (Figure 1) to assist with the use of all

MTpy modules. Our aim is to make MTpy a comprehensive toolkit that can be easily installed and used by the wider geophysics community.

CONCLUSIONS

The MTpy Python library was created to assist with many aspects of MT data processing, analysis and modelling. Geoscience Australia in collaboration with developers at the University of Adelaide and the United States Geological Survey continues the development to improve functionality, remove duplication, and facilitate its use. Key developments include improvements to the processing modules, addition of functions to calculate and visualise penetration depth, and improvement to the modelling modules to assist with creation of input files, and visualisation and analysis of outputs. A GUI is also being developed to help with the easy use of the functions contained within MTpy. For more updated details please visit the open source code repository: https://github.com/MTgeophysics/mtpy.

REFERENCES

Caldwell, T.G., Bibby, H.M. and Brown, C., 2004. The magnetotelluric phase tensor. Geophysical Journal International, 158 (2), 457-469.

Chave A.D., Thompson DJ and Ander, M.E., 1987. On the robust estimation of power spectra, coherences and transfer functions. Journal of Geophysical Research 92, 633–648.

Chave, A.D., and Thomson, D.J., 2004. Bounded influence magnetotelluric response function estimation. Geophysical Journal International 157, 988–1006.

Egbert, G. D. and Kelbert, A., 2012. Computational recipes for electromagnetic inverse problems. Geophysical Journal International, 189 (1): 251-267.

Kelbert, A., Meqbel, N., Egbert, G. D., and Tandon, K. (2014). ModEM: a modular system for inversion of electromagnetic geophysical data. Computers & Geosciences, 66: 40-53.

Krieger, L. and Peacock, J., 2014. MTpy: A Python toolbox for magnetotellurics. Computers and Geosciences, v. 72, pp. 167-175.