

# Avoiding common pitfalls in Euler deconvolution of potential field data

Leonardo Uieda, Vanderlei C. Oliveira Jr., and Valéria C. F. Barbosa. Observatório Nacional, Rio de Janeiro, RJ, Brazil.

In this tutorial we'll talk about a widely use method of interpretation for potential field data called Euler deconvolution. Our goal is to demonstrate its usefulness and, most importantly, call attention to common pitfalls encountered in the interpretation of the results. The code and synthetic data required to reproduce our results and figures can be found in the accompanying IPython notebooks ([ipython.org/notebook](http://ipython.org/notebook)) at [github.com/pinga-lab/paper-tle-euler-tutorial](https://github.com/pinga-lab/paper-tle-euler-tutorial). The notebooks also expand the analysis presented here. We encourage you to download the data and try it on your software of choice. For this tutorial we'll use the implementation in the open-source Python package Fatiando a Terra ([fatiando.org](http://fatiando.org)).

## Brief overview

Euler deconvolution was first developed by Thompson (1982) under the name “EULDPH”. It was later extended and given its current name by Reid et al. (1990). Since then, it has been adapted and improved upon by Keating (1998), Mushayandebvu et al. (2004), and many others. This popularity is largely due to its the great simplicity of implementation and use, making it the tool of choice for a quick initial interpretation.

In many cases, maps of gravity and magnetic data (and transformations thereof) provide good constraints on the horizontal location of an anomaly source. Euler deconvolution is able to add an extra dimension to the interpretation. It estimates a set of (x, y, z) points that, ideally, fall inside the source of the anomaly. The deconvolution requires the x-, y-, and z-derivatives of the data and a parameter called the *structural index* (SI). The SI is a number (usually integer) that is related to the homogeneity of the potential field and varies for different fields and source types (*Stavrev and Reid, 2007*). For example, in the case of total field magnetic anomaly data, a sphere is represented by an SI of 3, whereas a dike is represented by an SI of 1. There are methods that can estimate the SI and we refer the reader to Barbosa et al. (1999) and Melo et al. (2013) for more information.

## An example with synthetic data

The best way to fully grasp the usefulness and limitations of any geophysical method is to apply it in a controlled setting using synthetic (or modeled) data. To make things more interesting, we'll apply the deconvolution on data generated by a model that differs slightly from ideal sources (Figure 1).

Let's see how we can use Fatiando a Terra to run the deconvolution on our data. The first we need is to calculate the 3 directional derivatives of the potential field. For gridded data, this can be done with the Fourier transform using module `fourier` from Fatiando. Assuming that the data has been loaded into Numpy arrays `x`, `y`, `z` with the respective x-, y-, z-coordinates of each data point and `data` with our magnetic data (see the accompanying notebook):

```
from fatiando.gravmag import fourier
from fatiando import utils
# Need data in SI units
data_si = utils.nt2si(data)
# shape = (ny, nx) number of points in y and x
dx = fourier.derivx(x, y, data_si, shape)
```

```
dy = fourier.derivy(x, y, data_si, shape)
dz = fourier.derivz(x, y, data_si, shape)
```

Now that we have the derivatives, we have to create a solver to run the deconvolution. Fatiando provides the `Classic` solver which uses the whole dataset to estimate a single point (assuming a single source). The usual strategy for multiple sources, introduced by Reid et al. (1990), is to use a *moving window* over the data. A point is estimated for each window using the data that fall inside it, producing a large number of solutions. The challenge then becomes filtering out the good solutions from the spurious ones. FitzGerald et al. (2004) provide an overview of different selection criteria. Fatiando provides the `MovingWindow` solver that can run a given Euler solver on a moving window:

```
from fatiando.gravmag.euler import Classic, MovingWindow
classic = Classic(x, y, z, data_si, dx, dy, dz, struct_index)
solver = MovingWindow(classic, windows=(50, 50), size=(1000, 1000))
solver.fit()
```

Calling `solver.fit()` runs the deconvolution using the specified  $50 \times 50$  windows of size 1 km. The estimated points are stored in the `solver.estimate_` variable. See the accompanying notebook for a more detailed explanation.

For comparison, we ran the deconvolution using various different combinations of structural index and window size. Figure 2 shows scatter plots of the solutions (the typical way Euler solutions are presented). Notice how changing the window size influences the spread of the solutions and can affect our interpretation. For example, for a 1 km window and structural index 3, we cannot tell apart the two model bodies in the left. In general, increasing the structural index will increase the depths of the solutions.

We can view these solutions in 3D using the Fatiando module `myv`, a wrapper for the Mayavi visualization software ([code.enthought.com/projects/mayavi](http://code.enthought.com/projects/mayavi)):

```
from fatiando.vis import myv
myv.figure()
myv.points(solver.estimate_)
myv.axes(myv.outline(model_bounds))
myv.show()
```

This will create something similar to Figure 3. It is curious how the sources make arch-like structures that have no relation to the actual forms of the sources.

### **Last few words of caution**

We repeat here for emphasis: Euler deconvolution solutions **do not** represent the three-dimensional outline of the sources. They indicate, at most, an approximate location. It is crucial to remember that Euler solutions are subject to non-uniqueness and bias just like any other geophysical inverse problem. Tests using synthetic data are indispensable to assert the plausibility of our interpretations.

*Corresponding author: leouieda@gmail.com*

## References

- Barbosa, V. C. F., J. B. C. Silva, and W. E. Medeiros (1999), Stability analysis and improvement of structural index estimation in Euler deconvolution, *Geophysics*, *64*, 48–60, doi:10.1190/1.1444529.
- FitzGerald, D., A. Reid, and P. McNerney (2004), New discrimination techniques for Euler deconvolution, *Computers & Geosciences*, *30*, 461–469, doi:10.1016/j.cageo.2004.03.006.
- Keating, P. B. (1998), Weighted Euler deconvolution of gravity data, *Geophysics*, *63*, 1595–1603.
- Melo, F. F., V. C. F. Barbosa, L. Uieda, V. C. Oliveira Jr, and J. B. C. Silva (2013), Estimating the nature and the horizontal and vertical positions of 3D magnetic sources using Euler deconvolution, *GEOPHYSICS*, *78*, 87, doi:10.1190/geo2012-0515.1.
- Mushayandebvu, M. F., V. Lesur, A. B. Reid, and J. D. Fairhead (2004), Grid Euler deconvolution with constraints for 2D structures, *Geophysics*, *69*, 489–496, doi:10.1190/1.1707069.
- Reid, A. B., J. M. Allsop, H. Granser, A. J. Millett, and I. W. Somerton (1990), Magnetic interpretation in three dimensions using Euler deconvolution, *Geophysics*, *55*, 80–91, doi:10.1190/1.1442774.
- Stavrev, P., and A. Reid (2007), Degrees of homogeneity of potential fields and structural indices of Euler deconvolution, *GEOPHYSICS*, *72*, 1, doi:10.1190/1.2400010.
- Thompson, D. T. (1982), EULDPH: A new technique for making computer-assisted depth estimates from magnetic data, *Geophysics*, *47*, 31–37, doi:10.1190/1.1441278.

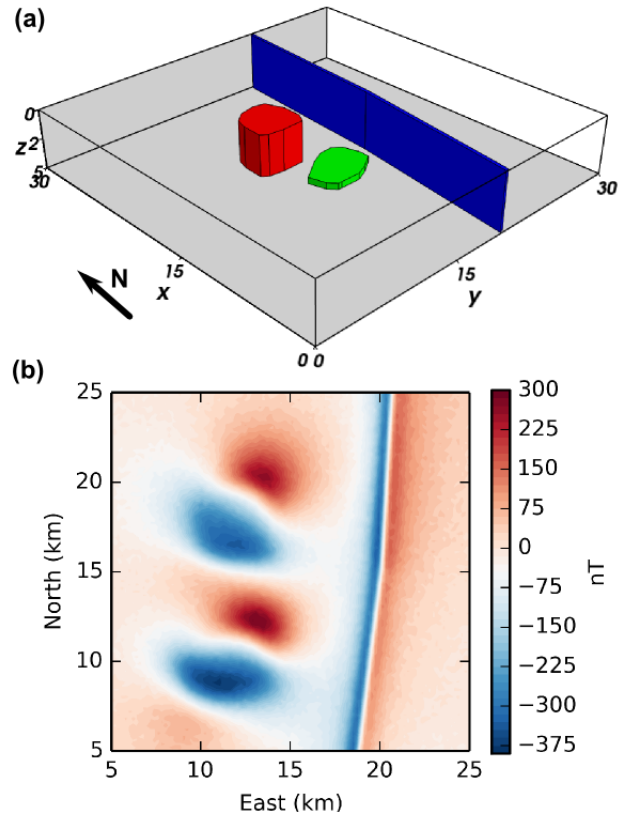


Figure 1: Our model (a) and synthetic total field anomaly data (b). The model simulates a dike (blue), a sill (green), and an intrusive body (red). The data is corrupted with 5 nT pseudo-random Gaussian noise.

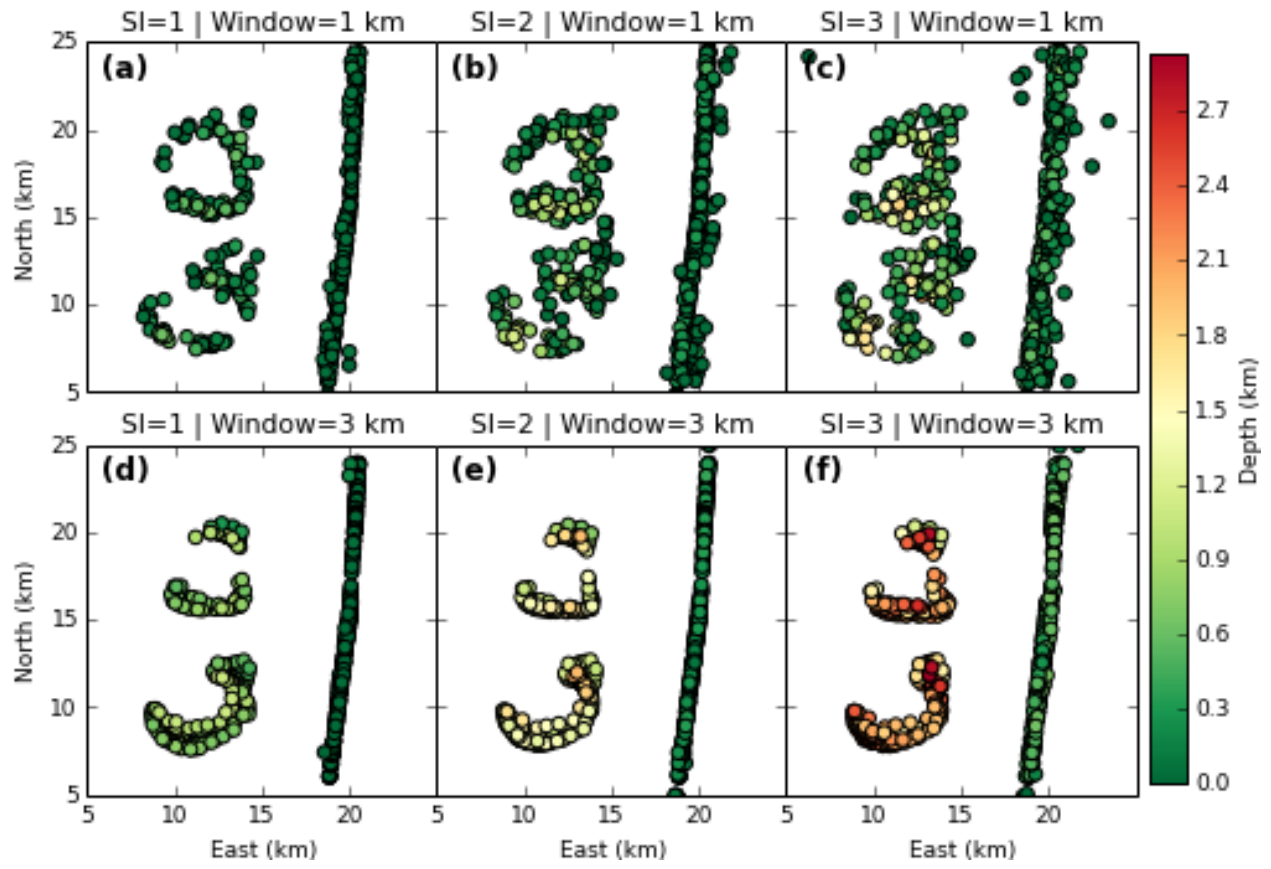


Figure 2: Euler deconvolution solutions for varying structural index (SI) and moving window size.

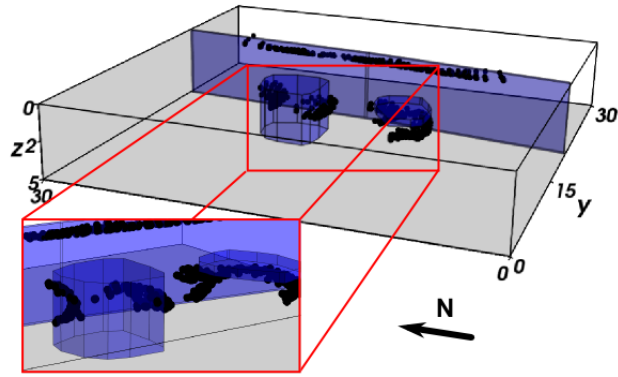


Figure 3: 3D view of the Euler deconvolution solutions (black dots) for a moving window of 3 km and structural index 3.