submitted to *Geophys. J. Int.*

Direct 3D Gravity Inversion: a review of theory and methodology with examples

Joshua Leland, 1209978

SUMMARY

This paper presents the theory and methodology for direct, 3D inversion of gravity data. The problem is is modeled by discretizing cells with a constant density. Gravity suffers from non-uniqueness and relies on good parameter weighting to get a reasonable solution. The objective function makes use of depth weighting, which counteracts the decay of the kernel with depth, and helps overcome the fact gravity has no depth resolution. A logarithmic barrier is added to the cost function to impose positivity and gives more reasonable solutions. Synthetic data is generated in an example to show the inversion results and demonstrate its advantages and shortcomings.

Key words: Potential fields, 3D Gravity, Inversion.

1 INTRODUCTION

1.1 The role of gravity surveys

Gravity surveys are used over a wide array of disciplines of exploration including: oil and gas applications, mining, environmental, and tectonic studies. The low cost makes gravity a popular choice for a variety of applications. Gravity is traditionally thought of as being used mainly for reconnaissance, but with most basin locations being known, and many case histories and data bases available, gravity is taking a role in detailed prospecting, reservoir monitoring, and integration with seismic data to help overcome of the pitfalls that seismic data can suffer from. (LaFehr, 2012).

1.2 Inversion and gravity

Gravity inversion is used to find an underground density distribution that predicts gravity data similar to field data, but with a model that is also geologically reasonable. This paper follows the work of Li et al. 1999 closely, which relies heavily on parameter weighting. Gravity suffers from the problem of non-uniqueness, and this can be shown by the problem of Green's equivalent layer, where an infinite amount of mass distributions can give the same gravitational potential. This ambiguity combined with the use of the minimum norm solution causes inverted solutions to be concentrated on the surface. This depth ambiguity is addressed by Li et al. 1996 by introducing a depth weighting function which gives weight to deeper solutions by fitting a function to counteract the data kernel decay with depth. This paper will review the theory and methodolgy for preforming 3D gravity inversion to give geologically reasonable solutions that honor the data. The paper will start with formuating a forward model, construct two cost functions to be minimized, review solutions to the cost functions, show two examples, and finally end with a dissucssion.

2 THEORY AND METHODOLOGY

2.1 Formulation of the forward model

For direct gravity inversion the starting point is the typical gravity forward model found in many textbooks or papers (LaFehr, 2012,Telford, 1990, Li, 1998). The continuous forward model for the gravitational attraction in the z direction at the ith point is:

$$g_z(\mathbf{r}_i) = \gamma \int_V \rho(\mathbf{r}) \frac{z - z_i}{(\mathbf{r} - \mathbf{r}_i)^3} \mathrm{d}V \tag{1}$$

Where:

(i) $\mathbf{r}$ is the radial vector.
(ii) $\mathbf{r}_i$ is the radial vector at a point i.
(iii) z is the depth with positive being into the earth.
(iv) z_i is the depth of the point i.
(v) V is the volume of the mass.
(vi) $\rho(\mathbf{r})$ is the density distribution.
(vii) γ is the gravitational constant.

Equation (1) is too complicated to solve as analytically, so to simplify it we can discretize the ground into small cubes each with a constant density resulting in the new equation:

$$g_z(\mathbf{r}_i) = \gamma \sum_{j=1}^{M} \rho_j \int_{\Delta V_j} \frac{z - z_i}{(\mathbf{r} - \mathbf{r}_i)^3} \mathrm{d}V \tag{2}$$

Where:

(i) ΔV_j is the volume for the jth cube.
(ii) ρ_j is the density of the jth cube.
(iii) M is the number of cubes the earth has been divided into.

Equation (2) is the sum of the gravitational attraction of all of the discretized cells at a point i. Evaluating the integral in equation (2) can be done many different ways. Numerically integrating is perhaps the simplest option, but it can lead to very high computation times. For each ΔV_j you would need to sub-divide that cell into many point masses to get a solution close to analytical. The further the ΔV_j is away from the observation point: the less point masses are needed to get a good approximation. It is recommended that the number of point masses for each cell be inversely proportional to the radius from the observation point to the cube to avoid overly long computation times.

The recommended way to compute the integral in equation (2) would be to use an analytical solution of the gravitational attraction of a cube (Nagy, 1966, LaFehr, 2012, Blakey, 1996). Blakey (1996) gave the solution for the gravitational attraction of a singular cube with respect to the origin to be:

$$g_z = \gamma\rho\sum_{i=1}^{2}\sum_{j=1}^{2}\sum_{k=1}^{2}\mu_{ijk}\left[z_k arctan\left(\frac{x_i y_j}{z_k R_{ijk}}\right) - x_i ln\left(R_{ijk}+y_j\right) - y_j ln\left(R_{ijk}+x_i\right)\right] \tag{3}$$

Where:

(i) $\mu_{ijk} = (-1)^i(-1)^j(-1)^k$
(ii) x_i, y_j, z_k are the respective Cartesian coordinates
(iii) $x_1 < x < x_2$
(iv) $y_1 < y < y_2$
(v) $z_1 < z < z_2$
(vi) $R_{ijk} = \sqrt{x_i^2 + y_j^2 + z_k^2}$

Using the analytical solution to equation (2), a system of equations can be written to calculate the gravity felt at N observation points for M cubes in simple matrix notation:

$$\mathbf{d} = \mathbf{G}\boldsymbol{\rho} \tag{4}$$

Where:

(i) **d** is an Nx1 data vector of observations.
(ii) **G** is an NxM system matrix representing equation (3).
(iii) $\boldsymbol{\rho}$ is a Mx1 parameter vector of densities.

Equation (4) shows a linear relationship between the data kernel **G** at N observation points and the M densities that represent the earth. This linear relationship allows **G** to be computed one time for a given geometry. This paper will assume a 3D geometry, and the spatial dimensions are vectorized from 3D to 1D with increasing x, y, and z.

2.2 Construction of a cost function

With the forward model defined, a cost function needs to be developed that will give reasonable, geologically sound solutions, and will also honor the data points. The cost function needs to have two parts: the data part, and the parameter part. With the forward model defined, we follow the Li et al. (1998) scheme for the data part of the cost function.

$$\phi_d = ||\mathbf{W}_d(\mathbf{d}_{obs} - \mathbf{d}_{pred})||_2^2 \tag{5}$$

Where:

(i) $\mathbf{W}_d$ a square matrix representing the weight of each data point.

(ii) $\mathbf{d}_{obs}$ is an Nx1 data vector representing the observed data points.

(iii) $\mathbf{d}_{pred}$ is an Nx1 data vector representing our predicted data points from equation (4).

The data weighting matrix $\mathbf{W}_d$ is defined by being diag$\{1/\sigma_1^2, 1/\sigma_2^2, ..., 1/\sigma_N^2\}$ where σ_i is the standard deviation, or error in the ith data point. This is usually taken from repeated gravity measurements in the field and represents the confidence at each of the stations so points with higher errors will carry less weight than points with lower errors.

Formulation of the regularization, or the model weighting, of the cost function consists of four matrices. Following the same scheme as *Li et al. 1996* any of the four matrices in the form of:

$$\mathbf{W}_i = \alpha_i \mathbf{S}_i \mathbf{D}_i \mathbf{Z} \tag{6}$$

Where:

(i) i = s, x, y, z.

(ii) α_i is a constant that gives relative weighting to each matrix.

(iii) $\mathbf{S}_s$ is a diagonal matrix with weighting for each individual cell.

(iv) $\mathbf{S}_{x,y,z}$ is a diagonal matrix with weighting for each interface in the respective direction.

(v) $\mathbf{D}_s$ is a diagonal matrix with $\sqrt{\Delta x \Delta y \Delta z}$ on its diagonal where: $\Delta x, \Delta y, \Delta z$ are the dimensions of the cell.

(vi) $\mathbf{D}_{x,y,z}$ is a derivative operator for the respective direction. $\mathbf{D}_x$, for example, will have 2 elements: $\pm\sqrt{\Delta z \Delta y / \delta x}$ in each row for adjacent cells in the respective direction, with δx being the distance to the center of the adjacent cubes.

(vii) **Z** is a diagonal matrix that represents the depth weighting function, which will be talked about in the next section.

Gravity, naturally, has no physical depth resolution, and this can be proven by Green's equivalent layer (LaFehr, 2012), so if inversion is carried out using the minimum norm solution and no depth weighting, the solution be concentrated on the surface. Li et al. 1996 introduces a depth weighting function to their regularization (**Z** in equation (6)). The data kernel will decay with depth for gravity at $1/r^2$ so introducing a depth function that will decay at the same rate as the kernel will cause the density distribution to vary more uniformly with depth. The depth weighting function is introduced as:

$$w(z) = \frac{1}{(z_0 + z)^{\beta/2}} \tag{7}$$

Where:

(i) z_0 is picked empirically but depends on the cell height. A good starting point is half the cell height.

(ii) β depends on the data type and is typically 2 for gravity.

(iii) z is the distance to the centre of the cell.

z_0 and β are found empirically, but the suggestions above are a good place to start. They are found by plotting the decay of the data kernel for a single cube with depth, and plotting $w^2(z)$ on the same plot, and then adjusting the two parameters in equation (7) until $w^2(z)$ overlaps the data kernel. If a mesh with different block sizes is used this procedure will need to be repeated for all of the different block sizes. **Z** can then be computed by using plugging in the depths to the centre of a cell for a given $z0$ and β.

With the parameter matrices defined, the regularization of our cost function can now be defined. We again follow the methodology of Li et al. 1996 where we define our regularization term to be:

$$\phi_m(\boldsymbol{\rho}) = (\boldsymbol{\rho} - \boldsymbol{\rho}_0)^{\mathbf{T}}\Big(\sum_{i=s,x,y,z} \mathbf{W}_i^{\mathbf{T}}\mathbf{W}_i\Big)(\boldsymbol{\rho} - \boldsymbol{\rho}_0) \tag{8}$$

Where:

(i) $\boldsymbol{\rho}_0$ is a reference density.

Which can be rewritten as:

$$\phi_m(\boldsymbol{\rho}) = (\boldsymbol{\rho} - \boldsymbol{\rho}_0)^{\mathbf{T}}(\mathbf{W_m}^{\mathbf{T}}\mathbf{W_m})(\boldsymbol{\rho} - \boldsymbol{\rho}_0), \tag{9}$$

and finally:

$$\phi_m(\boldsymbol{\rho}) = ||\mathbf{W_m}(\boldsymbol{\rho} - \boldsymbol{\rho}_0)||_2^2. \tag{10}$$

With both parts of the cost function defined the full cost function can now be constructed:

$$\phi(\boldsymbol{\rho}) = \phi(\boldsymbol{\rho})_d + \mu\phi(\boldsymbol{\rho})_m \tag{11}$$

Where:

(i) μ is the trade-off parameter.

The objective is to minimize equation (11) for a given trade off parameter (μ). μ gives a relative weighting to model part of the objective function. A target μ needs to be found that lets the model fit the data and stay close to a desired model. Solutions to equation (11) for a good parameter μ tends to leave long tails in the model. Li et al. 1998 suggests imposing positivity to remove the tail effect. The most effective way to impose positivity is suggested by Li et al. 2003 using a logarithmic barrier method, and is done by adding one more term to the objective function:

$$\phi(\boldsymbol{\rho}) = \phi(\boldsymbol{\rho})_d + \mu\phi(\boldsymbol{\rho})_m - 2\lambda \sum_{j=1}^{M} ln\left(\frac{\rho_j}{\rho^+}\right) \tag{12}$$

Where:

(i) λ is the logarithmic constant
(ii) ρ^+ is a large number.

Equation (12) makes minimization much more complex because the problem now becomes non-linear. The ρ^+ term needs to be introduced because densities greater than one need to be allowed, so scaling the term inside of the logarithmic function becomes necessary to keep from subtracting from the objective function. λ starts out as a very large number, which will decrease as the solution iterates. This ensures the solution starts at a value above zero and then will stay positive until convergence is reached.

2.3 Minimization of the cost functions

Two cost functions have been presented: equation (11) and equation (12). Minimization of equation (11) is relativity straight forward, while equation (12) is more complex. The solution to equation (11) for a reference density of $\boldsymbol{\rho}_0$ is found by taking the derivative with respect to $\boldsymbol{\rho}$ and setting the equation to zero. The solution to equation (11) for a reference density of $\boldsymbol{\rho}_0$ is shown below:

$$\boldsymbol{\rho} = \boldsymbol{\rho}_0 + \left(\mathbf{G}^{\mathbf{T}}\mathbf{W}_e\mathbf{G} + \mu\mathbf{W}_m^{\mathbf{T}}\mathbf{W}_m\right)^{-1}\mathbf{G}^{\mathbf{T}}\mathbf{W}_e\left(\mathbf{d}_{obs} - \mathbf{G}\boldsymbol{\rho}_0\right) \tag{13}$$

Where

(i) $\mathbf{W}_e = \mathbf{W}_d^{\mathbf{T}}\mathbf{W}_d$, the same as $\mathbf{W}_d$ as in equation (5)

Equation (13) is written symbolically and the explicit inverse does not have to be computed. The most efficient way to solve (13) to use a CGLS algorithm to solve for $\boldsymbol{\rho}$, but using the slash command in matlab will work as well.

The solution to the non-linear equation (12) is taken from Li et al. (2003) and requires iteration. The solution is shown below, with $\mathbf{W}_e = \mathbf{I}$ and a change of variables where $\mathbf{m} = \frac{\rho_j}{\rho^+}$ and as a result $\gamma = \mu(\rho^+)^2$. $\mathbf{G}$ is also now scaled by a factor of ρ^+.

$$\left(\mathbf{G^T G} + \gamma \mathbf{W}_m^\mathbf{T} \mathbf{W}_m + \lambda^n \mathbf{X}^{-2}\right) \Delta\mathbf{m} = -\mathbf{G^T}\delta\mathbf{d} - \gamma \mathbf{W}_m^\mathbf{T} \mathbf{W}_m \delta\mathbf{m} + \lambda^n \mathbf{X}^{-1}\mathbf{e} \tag{14}$$

Where:

(i) $\mathbf{X} = diag\left(m_1^{n-1}, m_2^{n-1}, ..., m_M^{n-1}\right)$
(ii) $\mathbf{e} = (1, ..., 1)^\mathbf{T}$
(iii) $\delta\mathbf{d} = \mathbf{G}\mathbf{m}^{(n-1)} - \mathbf{d}_{obs}$
(iv) $\delta\mathbf{m} = \mathbf{m}^{(n-1)} - \mathbf{m}_o$
(v) λ is initally set to a large number.

$\mathbf{m}$ is then updated by:

$$\mathbf{m}^{(n)} = \mathbf{m}^{(n-1)} + \sigma\beta\Delta\mathbf{m} \tag{15}$$

If $\Delta\mathbf{m}$ is all positive $\beta = 1$ otherwise it becomes the minimum value of $\frac{\mathbf{m}_j}{|\Delta\mathbf{m}_j|}$ that has a $\Delta\mathbf{m}_j$ less than zero. σ is set to 0.925. λ is then updated by:

$$\lambda^{(n+1)} = [1 - min(\beta, \sigma)]\, \lambda^{(n)} \tag{16}$$

The solution then iterates though until two stopping criteria are met: λ is sufficiently small so the logarithmic term has almost no contribution to equation (12), and the change in the objective function from the previous iteration is less than one percent.

3 SYNTHETIC EXAMPLE

To demonstrate the gravitational inversion algorithm that is proposed here a synthetic example is produced. A 100m x 100m x 50m grid is generated with grid spacing of 5 meters. This produces a model with 4000 cells. A cube with dimensions of 20m on each side is put into the center of the model with its center at 50m depth. Data is collected on the surface with a 60m x 60m grid with a spacing of three meters centered over top of the cube. This produces 400 observation points and the forward matrix is then composed of 4000 columns and 400 rows. The cube is assigned a density of 1, and the background density is set to 0 to avoid edging effects. Once the data is generated Gaussian noise was added with a maximum being 2 percent of the maximum value recovered from the observed data. Figure 1 is a subplot showing the observed data on the top, and a vertical slice of the 3D model showing the initial density distribution on the bottom.

3.1 Parameter weighting

Before doing the inversion, the parameter weighting matrix is established. All of the individual parameter weighting is set to unity, and the derivative matrices are calculated in the same way explained earlier. With all of the cells being the same size the matrices are straight forward to evaluate. The depth weighting function is calculated as discussed previously. Figure 2 shows a plot of the depth weighting function plotted over top of the decay of the kernel, the values used were $\beta = 2$ and $z0 = 1$. With the depth weighting function calculated the data can now be inverted. The α_s is set to 0.0005, as recommended by Li et al. 1998 and $\alpha_x = \alpha_y = \alpha_z = 1$.

3.2 Inversion without imposing positivity

Inversion without imposing positivity is quite simple. The solution, equation (13), is computed multiple times, for different values of μ. μ is initially set to be a large number and slowly decreased until a desirable value fit of the observed data to predicted data is found. The data is considered to be a good fit when the data passes the χ^2 test. Figure 3 shows the recovered result for the inversion without logarithmic barrier, and the observed and initial model are shown in a subplot for reference. It would also be reasonable to lower the standard deviation if the predicted data doesn't match well with the observed data, but this should be done with caution as in practical applications trying to fit something too well can give a false structure. Notice how the inverted model has the long tails described above in the model. While for simple anomalies it may not harm interpretation, when there is more of a complicated structure all of the tails will make interpretation impossible.

3.3 Inversion with imposing positivity

Inversion with imposing positivity utilizes equation (14) to find a solution that is above zero, or a reference density. The main reason for setting background density to zero is to eliminate edging effects from the model. Once gravity processing is completed and the anomaly is isolated, it is recommended to have your anomaly isolated to give a positive gravity anomaly for this algorithm otherwise equation (14) will fail, and it will need to be generalized to constraining densities between a higher and lower range. Doing a proper regional/residual separation and data reduction with this in mind should be straight forward.

The inversion with imposing positivity is shown in figure 4. There is much less smear on the recovered model in figure 3. The model shown in figure 5 is a result of slightly lowering the dampening parameter from where it passes the χ^2 test. The reason for this is to get a better fit of the predicted data, as the peak shown in the observed data is missing from the predicted data in figure 4. The resulting model in figure 5 more closely resembles the initial model as a result, but this needs to be done carefully as trying to fit the model too closely will cause the model to fit noise and can create a false structure.

3.4 Inversion on real data

The last example is inversion on real data from a 2D line in Bergheim, Sk, with 50 meter spacing between stations. The line is Free Air and Bouguer corrected to 2.35 g/cm^3. The current model for the area shows a fault in the Judith River and Lea Park shales. It is also belived to have gravel channels showing up as minor gravity lows. Figure 6 shows the inversion of the 2D line. The model shows the contact at the 5km mark, and maps out the gravity lows that are belived to be gravel channels also show up shallow in the model. The geometry of the cells was set to be 50 meters in the east, and 20 meters in depth. The model requires a 3D geometry, so the northing direction was taken to be very long. The inversion in a technical success, but in order to get a better idea of the true structure more information is needed to refine the model, and this model is a good example of the limitations of gravity inversion.

4 DISCUSSION

The two examples shown above show the how the lack of depth resolution affects the recovered models even with depth weighting. This is because of the nature of the physics. Green's equivalent layer, as discussed earlier shows that a point mass at a distance away will give the same gravitational pull as a sphere with the same mass at the same distance away. The problem of Green's equivalent layer shows how little information there actually is in the gravity data. It is important to realize that a large amount of influence outside of the physics is put into the model constraints to get solutions. The importance of geology becomes very clear as the set of solutions that can fit the data has huge variance. When applying this inversion scheme it is necessary to keep in mind how much of the solution is being molded by the user and not the physics of the problem. As a result of the heavy constraints this inversion algorithm cannot be seen as a black box where data goes in and a solution comes out. It is recommended that multiple different scenarios are looked at and the local geology is understood. This inversion algorithm is a powerful tool if geology is properly integrated, and if the inversion can be integrated with data that has a depth resolution, for example a seismic volume or well log data.

5 CONCLUSION

The previous sections built a framework for generating an accurate forward model with an analytical solution, constructing an accurate cost function that will minimize the difference between the observed data and predicted data, while making sure that the model is geologically viable. This was done by making a parameter weighting matrix that gives a penalty to undesirable solutions, allows weighting of individual cells and cell interfaces, and implements a depth weighting function to allow for deeper solutions to be recovered. The framework was tested with a synthetic example of a buried cube for both solutions that were presented in this paper. The linear solution, equation (13), gives smooth solutions at an appropriate depth that fit the predicted data, but the resulting tails are not geologically viable, and to solve this problem equation (14) is presented which uses the logarithmic barrier to impose positivity and the resulting solutions remove the tailing effect. The results show that the solution is close to the initial model, and gives a satisfactory inversion, at the cost of heavy regularization. Gravity is a powerful, cost effective tool that when used correctly can give a large amount of information to help uncover the geology of an area of importance.

REFERENCES

Yaoguo, Li, & Douglas W. Oldenburg, 1998. 3-D inversion of gravity data, *Geophys. J Int.*, **63**, 109–119.

Yaoguo, Li, & Douglas W. Oldenburg, 1996. 3-D inversion of magnetic data, *Geophysics*, **61**, 394–408.

Yaoguo, Li, & Douglas W. Oldenburg, 2003. Fast inversion of large scale magnetic data using wavelet transforms and logarithmic barrier method, *Geophys. J Int.*, **152**, 251–265.

Tomas R. Lafehr, & Misac N. Nabighian, 2012. Fundimentals of gravity Exploration, *Geophysical Monograph Series* **(17)**, SEG publication.

Telford, William Murray, & Robert E. Sheriff, 1990. *Applied geophysics Vol.* **(1)**, Cambridge university press.

Blakely, Richard J., 1996 *Potential theory in gravity and magnetic applications* , Cambridge university press.