

Comparison of gravity forward calculations using prisms and point masses and application to a synthetically generated salt diapir

Sverre H.W. Hassing

June 2020

Contents

1 Abstract	2
2 Introduction	3
2.1 The theory of gravity	3
2.2 Gravitational effects on Earth	3
2.3 Gravity measurements in practice	4
2.4 Forward gravity modelling	4
2.5 Salt tectonics	6
2.5.1 Mechanical properties of salt	6
2.5.2 Salt rising	6
2.6 Scientific question	8
3 Theoretical derivations	9
3.1 Elementary equations of gravitation	9
3.2 The coordinate system	9
3.3 Point-mass	11
3.4 Prisms	12
3.4.1 Basic formulas	12
3.4.2 The gravitational potential	13
3.4.3 The gravity vector \vec{g}	17
3.4.4 The gravity gradient tensor (\mathbf{T})	18
3.4.5 Summary	21
3.4.6 Revisiting Poisson's equation	22
3.4.7 Better numerical stability	23
4 Computational methods	24
4.1 A basic function	24
4.2 Benchmark cases	28
4.2.1 Sphere	28
4.2.2 Buried prism	31
4.3 Salt diapir	33
4.3.1 Gradient magnitudes	33
5 Results	35
5.1 Benchmark cases	35
5.1.1 Timing	35
5.1.2 Relative error	36
5.1.3 Gravity plots	38
5.2 Salt diapir	39
5.2.1 Low resolution	39
5.2.2 High resolution	39
6 Discussion	43
7 Conclusion	44

Chapter 1

Abstract

Forward gravity modelling calculates the gravitational response based on a certain model of an object or terrain. This can be done with different discretised shapes. Perhaps the most simple is the rectangular cuboid, a three dimensional rectangle. This can be applied in two ways. For prisms, the gravitational response is integrated over the whole volume of the shape. For point masses, all of the mass is assumed to be concentrated in the middle of the rectangular cuboid. The first is more accurate, but takes long to compute. The second is faster, but less accurate. This research investigates which method is better to use under which circumstances by using two benchmark cases, a sphere and a buried prism, and a more realistic model of a synthetically generated salt diapir. The relative error and timing of both methods show that close to the model and regardless of its shape, the prism method makes up for its longer computation time with a much higher accuracy than the point mass method. At a larger distance, the two methods converge and the point mass method is better to use, because of the lesser computation time.

Chapter 2

Introduction

2.1 The theory of gravity

Gravity is an inescapable part of the human experience. Leaving aside modern inventions, humans have always been confined to the surface of Earth and everything that went up, came down again. Aristotle explained this phenomenon with one of the core principles of his physics. Every object has an intrinsic goal that it strives towards (teleology). Just like an acorn tries to become a tree, every object strives towards an equilibrium position towards the centre of the universe (Woodfield, 2010 [39]). An object could be set into motion by for example a push against it, but when the push stopped it would immediately start moving back to its 'goal'.

Throughout the ages this idea persisted, partly because it did describe human experiences with gravity well enough, but there was reoccurring critique as well. Perhaps most famously, Galileo showed that two objects of different mass gain the same acceleration, in contradiction to Aristotle's idea that the heavier object would fall faster. At the same time a slow shift was occurring in the way of thinking about the centre of the solar system in the European scientific world. Instead of Earth as centre of the universe, the planets would rotate around the sun (Darling, 2006 [8]).

These doubts and corrections on the Aristotelian model led to a new model of gravity. Isaac Newton formulated a universal law of gravitation describing gravity as a force that different objects with mass exert on each other. This theory would win support, until it became the dominant theory on gravity [8]. Newton's equations worked well to predict the motion of planets and objects on Earth.

Still there were some measurements that stubbornly refused to be explained with Newton's theory of gravity. The new model that would explain them was the theory of general relativity of Einstein. This got rid of gravity as a force, but instead explained it with curvature in space-time. Objects with mass will deform space-time. Other objects accelerate in a straight line through this deformed space-time [8].

Einstein's new model of gravity is more complex, both conceptually and mathematically, than Newton's gravity law. Newton's equations can still be used as a good approximation of gravity when the gravitational field is not exceptionally strong, which is the case on Earth, just like it had been used before Einstein (Unzicker, 2008 [37]). Newton's laws are used in the descriptions and derivations concerning gravity in this research.

As mentioned, this model works with different objects with mass that create an attractional force towards each other. It could be that Earth can be seen as a single, very large object for the complex models trying to explain the motion of the planets, but when studying Earth itself, this is not the case. There are a lot more effects that work on smaller scales that need to be considered.

2.2 Gravitational effects on Earth

Simple physical calculations often use a single value for the gravitational acceleration. The Standard System of Units (SI) states the standard gravitational acceleration as 9.80665 m/s^2 (Bureau International des Poides et Mesures, 2019 [9]). This value is taken as representative for the gravity at sea level. In reality the actual value changes over the surface due to various different influences.

Each object with a large enough mass influences this acceleration enough that it can be measured. Examples are the influence of the Sun and Moon that, although far away, have enough mass to cause the tides. Earth itself is also not the perfect sphere that would be needed to get a single gravitational acceleration over the whole surface. Earth is slightly flattened at its poles. As a consequence the radius at the equator is different than at the poles and the gravitational acceleration is also different. Even underneath the surface Earth is not perfectly homogeneous. Large differences in

density exist, not only between the different layers that make up the Earth (core, mantle, crust), but also on smaller scales. For example, an igneous intrusion underneath the surface has a higher density than salts deposited in a drying desert basin. All of these small changes create a complex pattern of the gravitation that changes over the surface of Earth.

This pattern and the strength of gravitation can give information on the structures that generated them. It can be used to locate and define masses with a different density or document changes in mass. This works on different scales. Some effects can only be measured locally, such as the gravitational response due to an intrusion, others are spread out over larger areas, such as large bodies of water. A measurement at an observation point take all effects of nearby sources into account.

During research into one of these effects, only the gravitation of that single cause is relevant. To get a data set that is useful, the other, irrelevant effects need to be filtered out. This can be done by creating a reference model of the gravity. This models the gravitational signal of various effects and sets it as the ‘expected’ pattern. Where the measured values diverge from this model, a so-called anomaly is found. If the model is built correctly, the gravity anomalies are caused by an effect that was not modelled.

The problem with the measurements is that they are dependent on the observation point. Measurements at sea-level will be different than those made in satellites or even at surface level in elevated areas. For the measurements to be of any use when combining the data from different sources, there needs to be some general format that everything can be converted to. This is the geoid.

The geoid is a hypothetical surface around the Earth whose height is determined by the strength of gravity. The geoid can be defined as the surface where the potential of the gravity field is equal (Mörner, 1976 [25]). This can be imagined as if the ocean covered the whole surface of the Earth, which means that the water is free to flow everywhere. The result is that more water would accumulate at places with strong gravity and less at places with weak gravity. The water surface would then represent the geoid.

2.3 Gravity measurements in practice

To create a model of the geoid, data is needed, which can be obtained from gravimeters at the surface or in the air. Relatively recent satellite missions have provided global models in increasing spatial resolution. The Gravity Recovery And Climate Experiment (GRACE) mission (Tapley et al., 2004 [35]) that was active from 2002 to 2017 provided measurements on anomalies of the gravitational acceleration. Data from this was combined with terrestrial data to come to a new global model of the geoid, EGM2008 (Pavlis et al., 2012 [29]).

Another satellite mission was Gravity Field and Steady-State Ocean Circulation Explorer (GOCE), launched in 2009 and retired in 2013. Instead of measuring the gravitational acceleration it measured the gravity gradient (Drinkwater et al., 2006 [10]). The information from the gravity gradient tensor is more difficult to interpret (Saad, 2006 [32]), but has the advantage that it can be measured directly, making it more reliable (Pail et al., 2011 [28]).

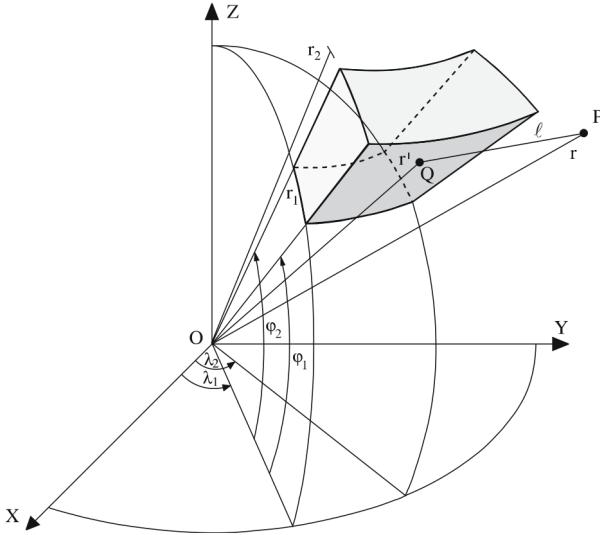
When the measurements are done, they can be used for research. Climate studies often use the data to investigate the dynamics of large bodies of water. Examples include seasonal variability of surface and ground water [34], variations in ice mass [16] and ocean circulation [22]. It can also be used for the detection of mass anomalies in the subsurface. In fact, gravity measurements are an often used method in exploration geology. In most cases the gravitational data is an addition to other methods of investigation. Geological and seismological research can be used to create a model of the subsurface. This can be enhanced with gravitational data.

The data can be used for a model in two ways. The first, inversion, is a method that tries to find the original mass distribution that led to the measured values of gravity. A model is created based on the measurements. Two examples of this method are the creation of a global model of the lithosphere (Afonso et al., 2019 [1]) and an inversion method tested on known data from Scandinavia and applied to the Rub’al-Khali desert (Bouman et al., 2015 [3]). Further discussion of this method lies outside the scope of this research.

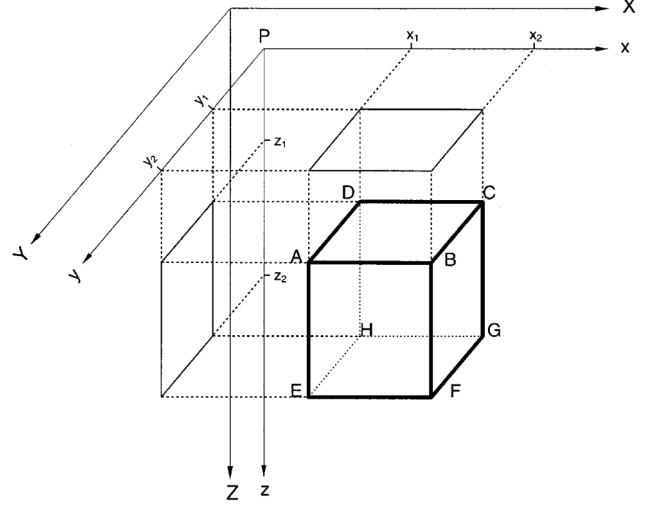
The second method of creating a model is forward gravity modelling. This uses a known model of the subsurface to calculate the resulting gravitational response, which can be compared with the measured data to provide constraints on the accuracy of the model. The method starts with a model and tries to calculate the measured values. Examples are a forward gravity model of the whole Earth (Kuhn and Featherstone, 2005 [23]), of the Eastern Alps (Ebbing et al., 2001 [12]) and a model of the sea floor that provides constraints on other models (Cadio and Korenaga, 2014 [4]).

2.4 Forward gravity modelling

Forward gravity modelling needs a model of the relevant area to be able to calculate the gravitational response. This means that the shape of the landscape, the underground structures and the density at every point need to be known. In



(2.1a) The classification of a tesseroid taken from Heck and Seitz, 2007 [15]



(2.1b) The classification of a rectangular cuboid taken from Nagy et al., 2000 [26]

reality this leads to a complex pattern of varying densities and irregular surfaces. It is not possible to derive an equation for the different components of gravity for each arbitrary shape with so much complexity.

The solution is to divide the area into smaller components whose gravitational signal is known. Because of the superposition principle (see section 3.1), the contribution of each component can be added together to get the full gravitation in an observation point. The primary assumption made is that each of these components has a constant density. Some of the real complexity is lost. This is now dependent on the resolution of the components. Using smaller components increases the resolution, thus giving an increase in accuracy. Another way to increase accuracy would be to change the shape of the components to one that fits with the structure.

There are various different shapes that can be used for the discretisation of the chosen area. The advantages of each depend on the geometry and the size of the area. When looking at the Earth on a larger scale, the curvature becomes relevant. Using spherical coordinates for this would seem obvious. A tesseroid is a shape bounded on each side by constant values in the spherical coordinate system as can be seen in figure 2.1a (Heck and Seitz, 2007 [15]). On a smaller scale the curvature is not as relevant and the normal, Cartesian coordinates can be used. Applying the same method as tesseroids gives a right rectangular parallelepiped or rectangular cuboid. This is simply the three dimensional version of a rectangle (see fig. 2.1b). Other examples include polyhedra (Tsoulis, 2012 [36]) and polygons (Plouff, 1976 [30]). The rectangular cuboid is of interest for this research and warrants extra discussion.

The most important aspect of using rectangular cuboids is that the gravitation can be integrated over the whole volume. This means that the equations describing the gravitational potential and the subsequent derivatives, the gravity vector and gradient tensor, can be derived (see section 3.4). If this is done, the shape is called a prism. The difficulty with using prisms for gravity forward calculations lies in the complexity of the equations that need to be used. The equation for the gravitational potential contains a triple summation which results in a formula containing 24 arctangents and 24 logarithms. This is computationally quite intensive. Another problem is mentioned in Heck and Seitz, 2007 [15]; at some point the numbers used in the calculations become too large for computers to handle without significant errors. An alternative method can be used to circumvent both of these problems.

This method uses the same rectangular cuboid, but instead of integrating over the whole volume, it assumes the mass is concentrated into a single point. This makes the equation a lot simpler. Instead of having to integrate the distance between observation point and cuboid, this distance is simply between the observation point and the point mass. The point mass method is less accurate due to this assumption, however. The resulting equation only contains a single term.

The advantage of using prisms is that the formulas used are the analytical solution of the gravitation of such a shape. The disadvantage is that the calculations will take longer to complete. Using the point mass method would take a lot less time. The accuracy could be increased by increasing the resolution of the discretisation. This means that there are more calculations to do, so more time is needed. The question is whether it is better to use prisms with a lower resolution or

point masses with a higher resolution.

This can be tested by applying both methods to a known test case. These benchmark cases should contain some simple geometry where the gravitational response is already known. The time it takes for both and the resulting values can be compared to find out which approximation delivers better results in shorter time. Then both methods can be applied to a more realistic scenario to see how the differences in accuracy would manifest. This scenario is a synthetically generated salt diapir. To give a background on the creation of such a structure, a brief overview of salt tectonics will be given with a focus on the rise of salt diapirs.

2.5 Salt tectonics

Salt layers play an important role in the development of various basins all over the world and the formation of several geologically interesting areas, such as the Zechstein layer in northwestern Europe (Harding and Huuse, 2015 [14], Zhang et al., 2013 [40]), various salt basins in the Gulf of Mexico (Hudec et al., 2013 [17]) or the eastern area of the Zagros Mountains in Iran where multiple salt diapirs have pierced the surface (Rahnama-Rad et al., 2009 [31], Jahani and Callot, 2009 [20]).

Salt bodies have also been extensively studied as their relative impermeability prevents fluids from passing through which makes them an important component in the formation of reservoirs for oil and gas. The same sealing properties also make salt an interesting area of study for the storage of (nuclear) waste (Hunsche and Hampel, 1999 [19]).

2.5.1 Mechanical properties of salt

In the strictest of geological terms, rock salt denotes an accumulation of the evaporitic mineral halite (NaCl). For salt tectonics, the term is more broadly used to refer to rocks that consist mostly of halite. Both other evaporites, such as gypsum, and non-evaporitic rocks can be captured in salt structures. These can come to be included into the structures in various ways, for example during deposition, igneous intrusions and trapped sediments (Hudec and Jackson, 2007 [18]).

Salt exhibits several unique characteristics that explain the special role it can play in tectonic evolutions:

- Salt can fracture under especially high strain rates, but under usual geological conditions it can realistically be described as purely viscous.
- Dry salt deforms through dislocation creep. If a little water is present, as little as 0.05% of its weight, the dominant deformation method is wet diffusion (Spiers et al., 1986 [33]).
- The density of pure halite is 2160 kg/m^3 , while that of impure halite is slightly higher, approximately 2200 kg/m^3 (Fossen, 2016 [13]).
- Salt is relatively incompressible [18].

The last two points mean that during burial, when the density of siliciclastic sedimentary rocks starts increasing, the density of salt does not change. The result is that it is less dense than most carbonates and moderately compacted rocks. The deformation mechanisms mentioned are easily activated in salt. As a result of the low yield strength, the viscous behaviour and the incompressibility, salt can be treated as a fluid (Fossen, 2016 [13]). When it is buried under a denser overburden, the salt becomes buoyant and the situation gravitationally unstable [18]. Under the right conditions, the salt starts moving upwards and may even start displacing the surrounding rocks.

2.5.2 Salt rising

After burial under a significant set of sedimentary layers, the gravitational conditions for salt are unstable. The layers on top, the overburden, have a higher density than the layer of salt below them. The burial also means that the sediments are compacted and stronger. The roof, the layer on top of the salt, is generally strengthened enough to contain the salt. A change in the situation is needed to start movement. Differential loading is seen as the dominant factor for this [18]. There are several ways that differences in load can form, but all will result in movement of salt away from the place with the highest load:

- For gravitational loading, the load consists of the weight of the overburden and the body forces inside the salt.
- Displacement loading is driven by the movement of rock boundaries relative to each other. If the layers on the sides start moving away from each other, space is created for the salt to flow into.

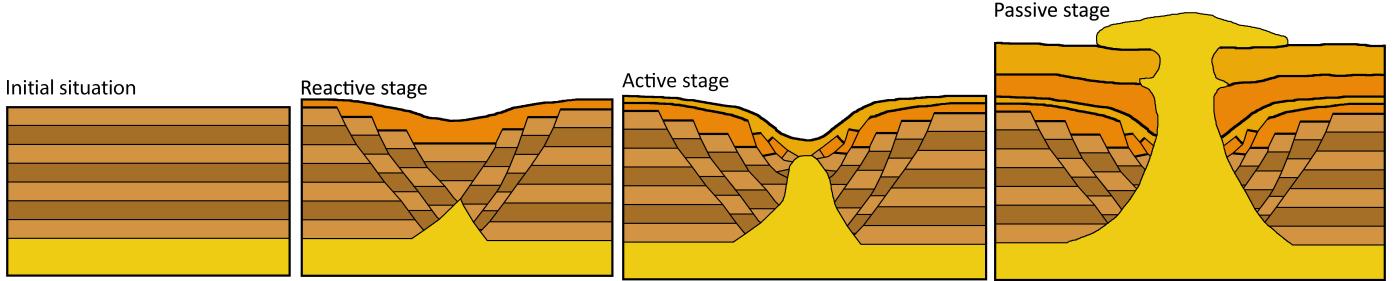


Figure 2.2: The different possible stages in the growth of a salt diapir in a thin-skinned extensional regime. The yellow layer depicts the salt, the brown, alternating layers are pre-rift sediments and the orange alternating layers are deposited during the diapirism. Adapted from Vendeville and Jackson, 1992 [38] and Hudec and Jackson, 2007 [18]

- Thermal loading is created by volume changes from differences in temperature. Hot salt expands and convection begins.

Movement can happen within the original confines under which the salt is buried. For significant movement of salt to happen, however, the salt must start piercing through layers. This piercing is called diapirism. In most cases the buoyant forces of the salt are not strong enough to overcome the strength of the roof. In these cases the strength of the roof must be weakened. The most common cause for this is tectonic stress, both extensional and contractional.

In an extensional regime, the roof will be thinned and fractured reducing its strength. The evolution can generally be divided into three distinct stages (Vendeville and Jackson, 1992 [38]), which are also illustrated in figure 2.2:

- During the first stage, the roof is strong enough to contain the salt. A graben is formed above the salt. The salt fills the space that opens between the diverging blocks, the movement is primarily driven by displacement loading. The fault blocks are not forced aside by the salt, but float in equilibrium with the salt. The speed of the diapirism is fully controlled by the speed of extension. Only reacting to the movement of the walls, hence the name reactive diapirism.
- The second stage can happen if the roof is weakened enough. Continued extension will assist this weakening, while sedimentation can thicken the overburden. Now the buoyant forces are strong enough for the salt to force aside its roof and rise up on its own. This is not driven by the extension, thus it will continue even if the extension would stop during the process. This stage is called active diapirism.
- When the diapir reaches the surface, growth can continue during what is called passive diapirism. The resulting shape depends on the relative rates of diapir growth and sedimentation. If the diapir grows faster than the sediments can build up, the diapir starts to spread laterally and will become wider. If the growth of the diapir is equal to the sedimentation rate, the diapir will start to build upwards, supported on its sides by the sediments. If the growth rate of the diapir is smaller than the sedimentation rate, the crest will be covered. The diapir will get covered again until its growth is suppressed.

This succession of stages assumes that the supply of salt continues. It is possible that at some point the source layer will not be able to provide more salt. This will stop the diapir from growing further. When the source layer becomes thinner, friction of the walls will start to play a relatively larger role, slowing the flow of salt. When the salt is exhausted at some part of the layer, the strata in the roof and bottom will become attached. This is called a salt weld [18]. Another important thing to note is that these stages can also occur by other causes. For example, passive diapirism happens when salt reaches the surface.

In a contractional regime, the tectonic stresses can buckle the roof. Salt will begin moving into the core of the anticline. This can also occur for pre-existing salt structures. Strictly speaking this is not diapirism, because the roof is not pierced. It is possible that active diapirism will start, breaking through the weakened roof. This is even possible without a density contrast when the salt is pressurised by tectonic stresses [18]. A pre-existing salt structure that is shortened can be pinched out in its middle section, even so much that the top crest is removed from its source layer. The middle part is pinched out and forms a salt weld. This kind of diapir is referred to as a teardrop diapir, while the remaining rise in the source layer is called a pedestal.

If the layer of salt is too thin, it will not be able to form a salt diapir. It can still play an important role in the local tectonics as a décollement. The overlying units can ‘glide’ over the structurally weak salt layer if it is extensive enough [13]. This changes the characteristics of deformation significantly (Costa and Vendeville, 2002 [6]).

2.6 Scientific question

The aim of this paper is to find out which of the two forward gravity modelling methods using rectangular cuboids is preferable to use. Prisms should give more accuracy, but also a larger computation time. The point mass approach could use a more detailed model to compensate the lack of accuracy, while still profiting off the smaller computation time. The method that is the most accurate after the same amount of computation time is the better one. The influence of other factors can also be taken into account, such as differences in the terrain that is modelled and the distance to the observation point.

Chapter 3

Theoretical derivations

3.1 Elementary equations of gravitation

Newton formulated a law of universal gravitation. This describes the force that two point masses exert on each other. In vector form it is formulated as:

$$\vec{F} = G \frac{m_1 m_2}{|\vec{r}_2 - \vec{r}_1|^2} \vec{e}_{12} \quad (3.1)$$

m_1 and m_2 are the masses. \vec{r}_1 and \vec{r}_2 are the location vectors of the point masses, making $|\vec{r}_2 - \vec{r}_1|$ the distance between the two point masses. G is the gravitational constant ($6.674 \text{ m}^3 \text{kg}^{-1} \text{s}^{-2}$). \vec{e}_{12} is the unit vector that indicates the direction from mass 1 to mass 2. This gravitation is often used as the acceleration due to this force. Formulated for the acceleration of point mass 1:

$$\vec{g} = G \frac{m_2}{|\vec{r}_2 - \vec{r}_1|^2} \vec{e}_{12} \quad (3.2)$$

As can be seen, the actual mass of point mass 1 is no longer relevant. This means that the gravitational acceleration can also be calculated at some observation point without any mass of its own. The acceleration vector can be defined everywhere in space. This means that there is a vector field for the gravitational acceleration. A scalar potential (U) can be defined underlying this. This potential field is defined as:

$$\vec{g} = -\vec{\nabla}U \quad (3.3)$$

This results in the following expression for the potential:

$$U = -G \frac{m_2}{|\vec{r}_2 - \vec{r}_1|} \quad (3.4)$$

Sometimes the change in acceleration is used. Mathematically this means differentiating each component for every direction. The result is the gravity gradient tensor \mathbf{T} . It is defined as:

$$\mathbf{T} = \vec{\nabla}\vec{g} \quad (3.5)$$

An interesting characteristic of gravitation is that the total potential (or one of its derivatives) can be calculated by adding the contribution of every object. This is called the principle of superposition. As an example, it can be used to calculate the gravitational response of both the Earth and the Moon on a point or the response of a model discretised into different prisms or point masses.

3.2 The coordinate system

No matter which method is used in the computation, the coordinate system used needs to be defined. When considering both prisms and point-mass based calculations these coordinate systems are conveniently the same. Two different coordinate reference frames are used. This was illustrated in figure 2.1b, but is shown again in figure 3.1.

On the larger scale every cell is defined in a single Cartesian coordinate system. This system is identified by using capitals. So a cell is delimited by coordinates X_1 to X_2 , Y_1 to Y_2 and Z_1 to Z_2 . The cell is free in its orientation relative to this system.

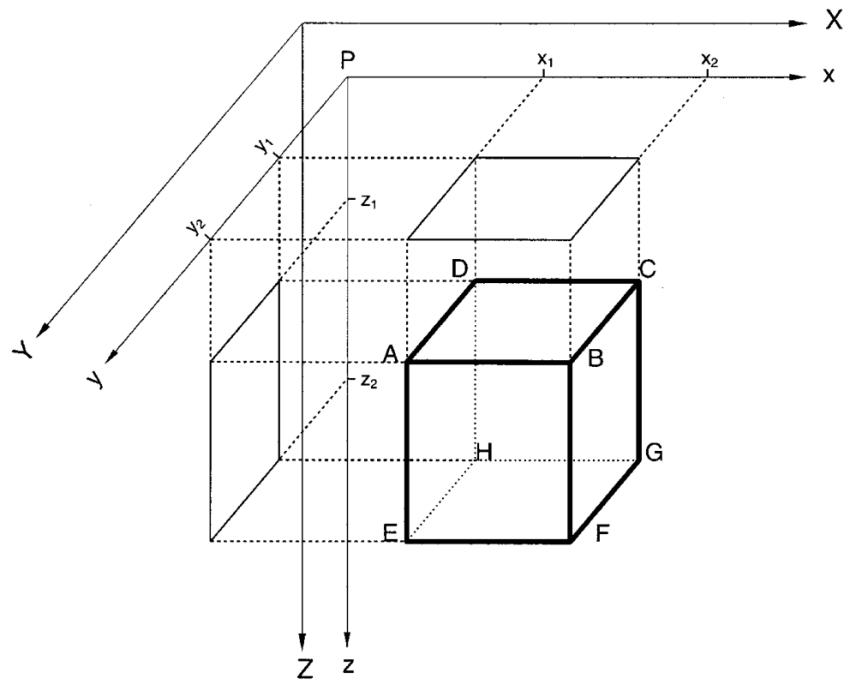


Figure 3.1: Classification of the prism in the standard coordinate system and from the observation point system. Taken from Nagy et al., 2000 [26]

When calculating the gravitational response a certain observation point ($(P(X_P, Y_P, Z_P))$) needs to be defined. The effect of gravity varies throughout space. The calculations are relative to this point. A second Cartesian coordinate system is defined with the observation point at its centre. The coordinates are identified as lower case letters. The axes must be parallel to the sides of the cell. For a rectangular cuboid, this is always possible. In the easiest case this is also parallel to the axes of the overall coordinate system. This would make the transformation from overall to observation coordinate system into a simple subtraction:

$$\begin{aligned} x_1 &= X_1 - X_P \\ x_2 &= X_2 - X_P \\ y_1 &= Y_1 - Y_P \\ y_2 &= Y_2 - Y_P \\ z_1 &= Z_1 - Z_P \\ z_2 &= Z_2 - Z_P \end{aligned} \tag{3.6}$$

The distance between the observation point and some point (x, y, z) in the cell is defined as:

$$r(x, y, z) = \sqrt{x^2 + y^2 + z^2} \tag{3.7}$$

3.3 Point-mass

This method assumes that the object can be discretised into rectangular cuboids, the same shape as the prisms. The density of a cell is homogeneous. The difference with prisms is that its mass is concentrated into a single point at the centre of the cell. This simplifies the integral for the gravitational potential (U). The original integral is gotten by taking equation 3.4 and integrating the density over the volume of the cuboid to get its mass. For an observation point P:

$$\begin{aligned} U(P) &= -G \iiint \frac{\rho(r)}{r} dV \\ &= -G \int_{x_1}^{x_2} \int_{y_1}^{y_2} \int_{z_1}^{z_2} \frac{\rho(r)}{r} dx dy dz \end{aligned} \tag{3.8}$$

The gravitational potential for a single point mass representing a rectangular cuboid of volume V and density ρ is:

$$U(P) = -\frac{G\rho V}{r} \tag{3.9}$$

The gravity vector \vec{g} is defined as $\vec{g} = -\vec{\nabla}U$. Each of the components can be calculated with the potential.

$$g_x = -\frac{\partial}{\partial x} G \frac{-\rho V}{\sqrt{x^2 + y^2 + z^2}^3} = -G \frac{\rho V x}{\sqrt{x^2 + y^2 + z^2}^3} = -G \frac{\rho V x}{r^3} \tag{3.10}$$

$$g_y = -\frac{\partial}{\partial y} G \frac{-\rho V}{\sqrt{x^2 + y^2 + z^2}^3} = -G \frac{\rho V y}{\sqrt{x^2 + y^2 + z^2}^3} = -G \frac{\rho V y}{r^3} \tag{3.11}$$

$$g_z = -\frac{\partial}{\partial z} G \frac{-\rho V}{\sqrt{x^2 + y^2 + z^2}^3} = -G \frac{\rho V z}{\sqrt{x^2 + y^2 + z^2}^3} = -G \frac{\rho V z}{r^3} \tag{3.12}$$

The norm of the gravity vector can also be computed:

$$\begin{aligned} |\vec{g}| &= \sqrt{\left(-G \frac{\rho V x}{r^3}\right)^2 + \left(-G \frac{\rho V y}{r^3}\right)^2 + \left(-G \frac{\rho V z}{r^3}\right)^2} \\ &= \sqrt{\left(\frac{-G\rho V}{r^3}\right)^2 (x^2 + y^2 + z^2)} \\ &= r \frac{G\rho V}{r^3} \\ &= \frac{G\rho V}{r^2} \end{aligned} \tag{3.13}$$

The gravity gradient tensor \mathbf{T} is calculated by repeating the derivative for each axis on each of the elements of the gravity vector. This tensor is symmetric, because the derivatives are commutative, $\frac{\partial^2}{\partial x \partial y} = \frac{\partial^2}{\partial y \partial x}$.

$$T_{xx} = \frac{\partial^2}{\partial x \partial x} \frac{-\mathcal{G}\rho V}{\sqrt{x^2 + y^2 + z^2}} = \mathcal{G}\rho V \frac{\sqrt{x^2 + y^2 + z^2}^2 - 3x^2}{\sqrt{x^2 + y^2 + z^2}^5} = \mathcal{G}\rho V \frac{r^2 - 3x^2}{r^5} \quad (3.14)$$

$$T_{yy} = \frac{\partial^2}{\partial y \partial y} \frac{-\mathcal{G}\rho V}{\sqrt{x^2 + y^2 + z^2}} = \mathcal{G}\rho V \frac{\sqrt{x^2 + y^2 + z^2}^2 - 3y^2}{\sqrt{x^2 + y^2 + z^2}^5} = \mathcal{G}\rho V \frac{r^2 - 3y^2}{r^5} \quad (3.15)$$

$$T_{zz} = \frac{\partial^2}{\partial z \partial z} \frac{-\mathcal{G}\rho V}{\sqrt{x^2 + y^2 + z^2}} = \mathcal{G}\rho V \frac{\sqrt{x^2 + y^2 + z^2}^2 - 3z^2}{\sqrt{x^2 + y^2 + z^2}^5} = \mathcal{G}\rho V \frac{r^2 - 3z^2}{r^5} \quad (3.16)$$

$$T_{xy} = T_{yx} = \frac{\partial^2}{\partial x \partial y} \frac{-\mathcal{G}\rho V}{\sqrt{x^2 + y^2 + z^2}} = \mathcal{G}\rho V \frac{\partial}{\partial y} \frac{x}{\sqrt{x^2 + y^2 + z^2}^3} = -\mathcal{G}\rho V \frac{3xy}{\sqrt{x^2 + y^2 + z^2}^5} = -\mathcal{G}\rho V \frac{3xy}{r^5} \quad (3.17)$$

$$T_{xz} = T_{zx} = \frac{\partial^2}{\partial x \partial z} \frac{-\mathcal{G}\rho V}{\sqrt{x^2 + y^2 + z^2}} = \mathcal{G}\rho V \frac{\partial}{\partial z} \frac{x}{\sqrt{x^2 + y^2 + z^2}^3} = -\mathcal{G}\rho V \frac{3xz}{\sqrt{x^2 + y^2 + z^2}^5} = -\mathcal{G}\rho V \frac{3xz}{r^5} \quad (3.18)$$

$$T_{yz} = T_{zy} = \frac{\partial^2}{\partial y \partial z} \frac{-\mathcal{G}\rho V}{\sqrt{x^2 + y^2 + z^2}} = \mathcal{G}\rho V \frac{\partial}{\partial z} \frac{y}{\sqrt{x^2 + y^2 + z^2}^3} = -\mathcal{G}\rho V \frac{3yz}{\sqrt{x^2 + y^2 + z^2}^5} = -\mathcal{G}\rho V \frac{3yz}{r^5} \quad (3.19)$$

3.4 Prisms

The derivations for prisms have been published in the early 50's [24]. However, to the best of my knowledge the full derivation has not been carried out in English with full detail, so this will be provided here. The derivations are based on those of Mader (1951) [24] and of Nagy et al. (2000) [26, 27]. Mader provided the derivations in some detail, while Nagy et al. interpreted the results in a modern style.

3.4.1 Basic formulas

The derivations for prisms are a lot more complicated than that for the point masses. There are two integral equations that are given as fundamental to the derivation:

$$\int \frac{x^2 dx}{x^2 + z^2} = x - z \arctan \frac{x}{z} \quad (3.20)$$

$$\int \frac{dx}{\sqrt{x^2 + y^2 + z^2}} = \ln \left(x + \sqrt{x^2 + y^2 + z^2} \right) \quad (3.21)$$

Another equation that will come back multiple times has the form:

$$\int \frac{du}{(v^2 + w^2)\sqrt{u^2 + v^2 + w^2}} \quad (3.22)$$

This can be solved with a trigonometric substitution, where $u = \sqrt{v^2 + w^2} \tan \phi$. This means that $du = \frac{\sqrt{v^2 + w^2}}{\cos^2 \phi} d\phi$.

$$\begin{aligned} \int \frac{du}{(v^2 + w^2)\sqrt{u^2 + v^2 + w^2}} &= \int \frac{\sqrt{v^2 + w^2}}{\cos^2 \phi} \frac{1}{(v^2 + w^2)\tan^2 \phi + v^2} \frac{d\phi}{\sqrt{v^2 + w^2 + (v^2 + w^2)\tan^2 \phi}} \\ &= \int \frac{\sqrt{v^2 + w^2}}{\cos \phi^2} \frac{1}{v^2(\tan^2 \phi + 1) + w^2 \tan^2 \phi} \frac{d\phi}{\sqrt{(v^2 + w^2)(\tan^2 \phi + 1)}} \\ &= \int \frac{\sqrt{v^2 + w^2}}{\cos \phi^2} \frac{1}{\frac{v^2 + w^2 \sin^2 \phi}{\cos^2 \phi}} \frac{d\phi}{\frac{\sqrt{v^2 + w^2}}{\cos \phi}} \\ &= \int \frac{\cos \phi d\phi}{v^2 + w^2 \sin^2 \phi} \end{aligned} \quad (3.23)$$

A second substitution is needed where $t = \frac{w}{v} \sin \phi$ and $dt = \frac{v}{w} \cos \phi d\phi$

$$\begin{aligned}
\int \frac{\cos \phi d\phi}{v^2 + w^2 \sin^2 \phi} &= \int \frac{v dt}{w(v^2 + v^2 t^2)} \\
&= \frac{1}{vw} \int \frac{dt}{1+t^2} \\
&= \frac{1}{vw} \arctan t \\
&= \frac{1}{vw} \arctan \frac{w \sin \phi}{v}
\end{aligned} \tag{3.24}$$

Now the $\sin \phi$ needs to be converted back to u, v, w . If it is known that $\tan \phi = \frac{u}{\sqrt{v^2+w^2}}$, then it follows that $\sin \phi = \frac{u}{\sqrt{u^2+v^2+w^2}}$

$$\arctan \frac{1}{vw} \frac{w \sin \phi}{v} = \frac{1}{vw} \arctan \frac{uw}{v\sqrt{u^2+v^2+w^2}} \tag{3.25}$$

So the result is:

$$\int \frac{du}{(v^2 + w^2)\sqrt{u^2 + v^2 + w^2}} = \frac{1}{vw} \arctan \frac{uw}{v\sqrt{u^2 + v^2 + w^2}} \tag{3.26}$$

3.4.2 The gravitational potential

Each prism has an assumed constant density ρ . The gravitational potential is integrated over the whole volume of the prism. With equation 3.4 this is:

$$\begin{aligned}
U(P) &= -u(P)\mathcal{G}\rho = -\mathcal{G}\rho \int_V \frac{dxdydz}{r} \\
&= -\mathcal{G}\rho \int_{x_1}^{x_2} \int_{y_1}^{y_2} \int_{z_1}^{z_2} \frac{dxdydz}{\sqrt{x^2 + y^2 + z^2}}
\end{aligned} \tag{3.27}$$

In what follows we work out the exact form for the $u(P)$ term. Elementary equation 3.21 can be applied to the integral for dx in equation 3.27.

$$\begin{aligned}
\iiint \frac{dxdydz}{\sqrt{x^2 + y^2 + z^2}} &= \iint \left(\int \frac{dx}{\sqrt{x^2 + y^2 + z^2}} \right) dydz \\
&= \iint \ln \left(x + \sqrt{x^2 + y^2 + z^2} \right) dydz
\end{aligned} \tag{3.28}$$

For a partial integration to y the parts are:

$$\begin{array}{c|c}
f = \int \ln \left(x + \sqrt{x^2 + y^2 + z^2} \right) dz & g' = dy \\
\hline
f' = \frac{y}{(x + \sqrt{x^2 + y^2 + z^2})\sqrt{x^2 + y^2 + z^2}} & g = y
\end{array}$$

$$\int fg' dy = fg - \int f g' dy \tag{3.29}$$

$$u(P) = \underbrace{y \int \ln \left(x + \sqrt{x^2 + y^2 + z^2} \right) dz}_A - \underbrace{\iint \frac{y^2 dz}{(x + \sqrt{x^2 + y^2 + z^2})\sqrt{x^2 + y^2 + z^2}} dy}_B \tag{3.30}$$

The calculation of $u(P)$ resolves into two large integrals denoted A and B , calculated in separate subsections.

The calculation of A

The first step in calculating A is carrying out a similar partial integration as seen before.

$$\begin{array}{c|c}
 f = \ln(x + \sqrt{x^2 + y^2 + z^2}) & g' = dz \\
 \hline
 f' = \frac{\partial f}{\partial z} = \frac{z}{(x + \sqrt{x^2 + y^2 + z^2})\sqrt{x^2 + y^2 + z^2}} & g = z
 \end{array}$$

$$\begin{aligned}
 A &= y \left(z \ln(x + \sqrt{x^2 + y^2 + z^2}) - \int \frac{z^2 dz}{(x + \sqrt{x^2 + y^2 + z^2})\sqrt{x^2 + y^2 + z^2}} \right) \\
 &= \underbrace{yz \ln(x + \sqrt{x^2 + y^2 + z^2})}_{A_0} - \underbrace{y \int \frac{z^2 dz}{(x + \sqrt{x^2 + y^2 + z^2})\sqrt{x^2 + y^2 + z^2}}}_{A_1}
 \end{aligned} \tag{3.31}$$

We now focus on the A_1 integral. We first multiply the numerator and denominator by $-x + \sqrt{x^2 + y^2 + z^2}$. The last step uses the equations (3.21), (3.26) and (3.20) respectively for each term.

$$\begin{aligned}
 A_1 &= \int \frac{z^2 dz}{(x + \sqrt{x^2 + y^2 + z^2})\sqrt{x^2 + y^2 + z^2}} \frac{-x + \sqrt{x^2 + y^2 + z^2}}{-x + \sqrt{x^2 + y^2 + z^2}} \\
 &= \int \frac{(-xz^2 + z^2\sqrt{x^2 + y^2 + z^2})dz}{(x^2 + y^2 + z^2 - x^2)\sqrt{x^2 + y^2 + z^2}} \\
 &= \int \frac{-xz^2 dz}{(y^2 + z^2)\sqrt{x^2 + y^2 + z^2}} + \int \frac{z^2\sqrt{x^2 + y^2 + z^2} dz}{(y^2 + z^2)\sqrt{x^2 + y^2 + z^2}} \\
 &= \int \frac{-x(z^2 + y^2 - y^2) dz}{(y^2 + z^2)\sqrt{x^2 + y^2 + z^2}} + \int \frac{z^2 dz}{y^2 + z^2} \\
 &= \int \frac{-xdz}{\sqrt{x^2 + y^2 + z^2}} + \int \frac{xy^2 dz}{(y^2 + z^2)\sqrt{x^2 + y^2 + z^2}} + \int \frac{z^2 dz}{y^2 + z^2} \\
 &= -x \int \frac{dz}{\sqrt{x^2 + y^2 + z^2}} + xy^2 \int \frac{dz}{(y^2 + z^2)\sqrt{x^2 + y^2 + z^2}} + \int \frac{z^2 dz}{y^2 + z^2} \\
 &= -x \ln(z + \sqrt{x^2 + y^2 + z^2}) + y \arctan \frac{xz}{y\sqrt{x^2 + y^2 + z^2}} + z - y \arctan \frac{z}{y}
 \end{aligned} \tag{3.32}$$

Now this can be combined to get the final expression for A:

$$A = y \left(z \ln(x + \sqrt{x^2 + y^2 + z^2}) + x \ln(z + \sqrt{x^2 + y^2 + z^2}) - y \arctan \frac{xz}{y\sqrt{x^2 + y^2 + z^2}} - z + y \arctan \frac{z}{y} \right) \tag{3.33}$$

The last two terms can be left out because they will cancel out when computing the integration boundaries from x_1 to x_2 , because these terms do not contain the variable x .

Finally we arrive at the following expression for A:

$$A = yz \ln(x + \sqrt{x^2 + y^2 + z^2}) + xy \ln(z + \sqrt{x^2 + y^2 + z^2}) - y^2 \arctan \frac{xz}{y\sqrt{x^2 + y^2 + z^2}} \tag{3.34}$$

The calculation of B

The inner integral can be simplified similarly to how A1 was simplified in (3.32), by multiplying both numerator and denominator with $-x + \sqrt{x^2 + y^2 + z^2}$. The last step uses equations (3.26) and (3.20).

$$\begin{aligned}
B &= - \int y^2 \int \frac{dz}{(x + \sqrt{x^2 + y^2 + z^2}) \sqrt{x^2 + y^2 + z^2}} \frac{-x + \sqrt{x^2 + y^2 + z^2}}{-x + \sqrt{x^2 + y^2 + z^2}} dy \\
&= - \int y^2 \int \frac{-x + \sqrt{x^2 + y^2 + z^2}}{(x^2 + y^2 + z^2 - x^2) \sqrt{x^2 + y^2 + z^2}} dz dy \\
&= - \int y^2 \left(- \int \frac{xdz}{(y^2 + z^2) \sqrt{x^2 + y^2 + z^2}} + \int \frac{dz}{y^2 + z^2} \right) dy \\
&= - \int y^2 \left(- \frac{1}{y} \arctan \frac{xz}{y\sqrt{x^2 + y^2 + z^2}} + \frac{1}{y} \arctan \frac{z}{y} \right) dy
\end{aligned} \tag{3.35}$$

Again the second term can be left out, because it does not contain the variable x . The next step is to apply a partial integration to B .

$$\begin{aligned}
B &= \int y \arctan \frac{xz}{y\sqrt{x^2 + y^2 + z^2}} dy \\
&\quad \begin{array}{c|c}
f = \arctan \frac{xz}{y\sqrt{x^2 + y^2 + z^2}} & g' = y \\
\hline f' = -xz \frac{\frac{1}{y^2\sqrt{x^2+y^2+z^2}} + \frac{1}{(x^2+y^2+z^2)^{\frac{3}{2}}}}{\frac{x^2z^2}{y^2(x^2+y^2+z^2)} + 1} & g = \frac{y^2}{2}
\end{array} \\
B &= \frac{y^2}{2} \arctan \frac{xz}{y\sqrt{x^2 + y^2 + z^2}} + \underbrace{\frac{xz}{2} \int y^2 \frac{\frac{1}{y^2\sqrt{x^2+y^2+z^2}} + \frac{1}{(x^2+y^2+z^2)^{\frac{3}{2}}}}{\frac{x^2z^2}{y^2(x^2+y^2+z^2)} + 1} dy}_{B_1}
\end{aligned} \tag{3.37}$$

Let us finish by calculating the integral B_1 :

$$\begin{aligned}
B_1 &= \frac{xz}{2} \int y^2 \frac{\frac{1}{y^2\sqrt{x^2+y^2+z^2}} + \frac{1}{(x^2+y^2+z^2)^{3/2}}}{\frac{x^2z^2}{y^2(x^2+y^2+z^2)} + 1} dy \\
&= \frac{xz}{2} \int y^2 \frac{\frac{x^2+y^2+z^2}{y^2(x^2+y^2+z^2)^{3/2}} + \frac{y^2}{y^2(x^2+y^2+z^2)^{3/2}}}{\frac{x^2z^2+y^2(x^2+y^2+z^2)}{y^2(x^2+y^2+z^2)}} dy \\
&= \frac{xz}{2} \int y^2 \frac{\frac{x^2+2y^2+z^2}{y^2(x^2+y^2+z^2)^{3/2}}}{\frac{x^2z^2+y^2(x^2+y^2+z^2)}{y^2(x^2+y^2+z^2)}} dy \\
&= \frac{xz}{2} \int y^2 \frac{x^2+2y^2+z^2}{\sqrt{x^2+y^2+z^2}(x^2z^2+y^2(x^2+y^2+z^2))} dy \\
&= \frac{xz}{2} \int y^2 \frac{x^2+2y^2+z^2}{\sqrt{x^2+y^2+z^2}(x^2+y^2)(z^2+y^2)} dy \\
&= \frac{xz}{2} \left(\int \frac{2dy}{\sqrt{x^2+y^2+z^2}} + \int \frac{-(x^2+z^2)y^2 - 2x^2z^2}{(x^2+y^2)(y^2+z^2)\sqrt{x^2+y^2+z^2}} dy \right) \\
&= xz \ln(y + \sqrt{x^2+y^2+z^2}) - \frac{xz}{2} \int \frac{(x^2+z^2)y^2 + 2x^2z^2}{(x^2+y^2)(y^2+z^2)\sqrt{x^2+y^2+z^2}} dy \\
&= xz \ln(y + \sqrt{x^2+y^2+z^2}) - \frac{xz}{2} \int \frac{x^2y^2 + y^2z^2 + 2x^2z^2}{(x^2+y^2)(y^2+z^2)\sqrt{x^2+y^2+z^2}} dy \\
&= xz \ln(y + \sqrt{x^2+y^2+z^2}) - \frac{xz}{2} \int \frac{x^2(y^2+z^2) + z^2(x^2+y^2)}{(x^2+y^2)(y^2+z^2)\sqrt{x^2+y^2+z^2}} dy \\
&= xz \ln(y + \sqrt{x^2+y^2+z^2}) - \frac{xz}{2} \int \frac{x^2}{(x^2+y^2)\sqrt{x^2+y^2+z^2}} dy - \frac{xz}{2} \int \frac{z^2}{(y^2+z^2)\sqrt{x^2+y^2+z^2}} dy \\
&= xz \ln(y + \sqrt{x^2+y^2+z^2}) - \frac{xz}{2} \frac{x^2 \arctan \frac{yz}{x\sqrt{x^2+y^2+z^2}}}{xz} - \frac{xz}{2} \frac{z^2 \arctan \frac{xy}{z\sqrt{x^2+y^2+z^2}}}{xz} \\
&= xz \ln(y + \sqrt{x^2+y^2+z^2}) - \frac{x^2}{2} \arctan \frac{yz}{x\sqrt{x^2+y^2+z^2}} - \frac{z^2}{2} \arctan \frac{xy}{z\sqrt{x^2+y^2+z^2}}
\end{aligned} \tag{3.38}$$

This can be combined to get the full expression for B :

$$B = xz \ln(\sqrt{x^2+y^2+z^2} + y) - \frac{x^2}{2} \arctan \frac{zy}{x\sqrt{x^2+y^2+z^2}} + \frac{y^2}{2} \arctan \frac{xz}{y\sqrt{x^2+y^2+z^2}} - \frac{z^2}{2} \arctan \frac{xy}{x\sqrt{z^2+y^2+z^2}}$$

Combining A and B

Now A and B can be combined to get the expression of $u(P)$ (and from that $U(P)$).

$$\begin{aligned}
u(P) &= A + B \\
&= yz \ln(x + \sqrt{x^2+y^2+z^2}) + xy \ln(z + \sqrt{x^2+y^2+z^2}) - y^2 \arctan \frac{xz}{y\sqrt{x^2+y^2+z^2}} \\
&\quad + xz \ln(\sqrt{x^2+y^2+z^2} + y) - \frac{x^2}{2} \arctan \frac{zy}{x\sqrt{x^2+y^2+z^2}} + \frac{y^2}{2} \arctan \frac{xz}{y\sqrt{x^2+y^2+z^2}} - \frac{z^2}{2} \arctan \frac{xy}{x\sqrt{z^2+y^2+z^2}} \\
&= yz \ln(x + \sqrt{x^2+y^2+z^2}) + xy \ln(z + \sqrt{x^2+y^2+z^2}) + xz \ln(y + \sqrt{x^2+y^2+z^2}) \\
&\quad - \frac{x^2}{2} \arctan \frac{zy}{x\sqrt{x^2+y^2+z^2}} - \frac{y^2}{2} \arctan \frac{xz}{y\sqrt{x^2+y^2+z^2}} - \frac{z^2}{2} \arctan \frac{xy}{x\sqrt{z^2+y^2+z^2}}
\end{aligned} \tag{3.39}$$

The boundaries for the volume from (3.6) need to be applied to the result of the integration and $U(P)$. The boundary conditions are computed by plugging the upper value into the equation and subtracting the equation with the lower value

plugged in. When the upper and lower values are respectively x_2 and x_1 for some function $f(x)$, this is $f(x_2) - f(x_1)$. This can be represented more efficiently with a summation over the subscript. Something needs to be added to still keep the subtraction in there. This can be done by adding a factor of -1^i , where i is the summation index. This will be positive when i is even and negative when i is odd. The new way of showing the result would be $\sum_{i=1}^2 -1^i f(x_i)$. This is especially useful when there are three different integration boundaries to resolve. r will be used instead of $\sqrt{x^2 + y^2 + z^2}$ as seen in equation (3.7).

$$\begin{aligned} u(P) &= \left| \left| \left| yz \ln(x+r) + xy \ln(z+r) + xz \ln(y+r) - \frac{x^2}{2} \arctan \frac{zy}{xr} - \frac{y^2}{2} \arctan \frac{xz}{yr} - \frac{z^2}{2} \arctan \frac{xy}{xr} \right|_{x_1}^{x_2} \right|_{y_1}^{y_2} \right|_{z_1}^{z_2} \\ &= \sum_{i=1}^2 \sum_{j=1}^2 \sum_{k=1}^2 (-1)^{i+j+k} \left(y_j z_k \ln(x_i + r_{ijk}) + x_i y_j \ln(z_k + r_{ijk}) + x_i z_k \ln(y_j + r_{ijk}) \right. \\ &\quad \left. - \frac{x_i^2}{2} \arctan \frac{z_k y_j}{x_i r_{ijk}} - \frac{y_j^2}{2} \arctan \frac{x_i z_k}{y_j r_{ijk}} - \frac{z_k^2}{2} \arctan \frac{x_i y_j}{x_i r_{ijk}} \right) \end{aligned} \quad (3.40)$$

$$U(P) = \mathcal{G} \rho \sum_{i=1}^2 \sum_{j=1}^2 \sum_{k=1}^2 (-1)^{i+j+k} \left(y_j z_k \ln(x_i + r_{ijk}) + x_i y_j \ln(z_k + r_{ijk}) + x_i z_k \ln(y_j + r_{ijk}) \right. \\ \left. - \frac{x_i^2}{2} \arctan \frac{z_k y_j}{x_i r_{ijk}} - \frac{y_j^2}{2} \arctan \frac{x_i z_k}{y_j r_{ijk}} - \frac{z_k^2}{2} \arctan \frac{x_i y_j}{x_i r_{ijk}} \right)$$

3.4.3 The gravity vector \vec{g}

In 3D Cartesian coordinates the gravity vector is expressed as

$$\vec{g} = -\vec{\nabla} U = \begin{pmatrix} -\frac{\partial U}{\partial x} \\ -\frac{\partial U}{\partial y} \\ -\frac{\partial U}{\partial z} \end{pmatrix} \quad (3.41)$$

The easiest way to calculate this is by including the partial derivatives in the original integral (3.27) and recalculating the result.

$$\begin{aligned} u_x(P) &= \iiint \frac{\partial}{\partial x} \frac{dxdydz}{\sqrt{x^2 + y^2 + z^2}} \\ &= - \iiint \frac{x dxdydz}{(\sqrt{x^2 + y^2 + z^2})^3} \\ &= \iint \frac{dydz}{\sqrt{x^2 + y^2 + z^2}} \end{aligned} \quad (3.42)$$

The integral (3.21) can be used, followed by the calculation of A as seen in Section 3.4.2 without the multiplication with y

$$\begin{aligned} u_x(P) &= \int \ln \left(x + \sqrt{x^2 + y^2 + z^2} \right) dz \\ &= z \ln \left(y + \sqrt{x^2 + y^2 + z^2} \right) + y \ln \left(z + \sqrt{x^2 + y^2 + z^2} \right) - x \arctan \frac{yz}{x\sqrt{x^2 + y^2 + z^2}} \end{aligned} \quad (3.43)$$

The integration boundaries can be applied. Multiplication with \mathcal{G} and ρ is the final step in deriving the element of the gravity vector (g_x).

$$g_x = \mathcal{G}\rho \sum_{i=1}^2 \sum_{j=1}^2 \sum_{k=1}^2 (-1)^{i+j+k} \left(z_k \ln \left(y_j + \sqrt{x_i^2 + y_j^2 + z_k^2} \right) + y_j \ln \left(z_j + \sqrt{x_i^2 + y_j^2 + z_k^2} \right) - x_i \arctan \frac{y_j z_k}{x_i \sqrt{x_i^2 + y_j^2 + z_k^2}} \right) \quad (3.44)$$

The same can be done for the y- and z-components and in the end we obtain

$$\boxed{\begin{aligned} g_x &= \mathcal{G}\rho \sum_{i=1}^2 \sum_{j=1}^2 \sum_{k=1}^2 (-1)^{i+j+k} \left(z_k \ln (y_j + r_{ijk}) + y_j \ln (z_j + r_{ijk}) - x_i \arctan \frac{y_j z_k}{x_i r_{ijk}} \right) \\ g_y &= \mathcal{G}\rho \sum_{i=1}^2 \sum_{j=1}^2 \sum_{k=1}^2 (-1)^{i+j+k} \left(z_k \ln (x_i + r_{ijk}) + x_i \ln (z_j + r_{ijk}) - y_j \arctan \frac{x_i z_k}{y_j r_{ijk}} \right) \\ g_z &= \mathcal{G}\rho \sum_{i=1}^2 \sum_{j=1}^2 \sum_{k=1}^2 (-1)^{i+j+k} \left(x_i \ln (y_j + r_{ijk}) + y_j \ln (x_i + r_{ijk}) - z_k \arctan \frac{x_i y_j}{z_k r_{ijk}} \right) \end{aligned}}$$

These equations can be verified in various other papers such as eq. (6) in Heck and Seitz, 2007 [15], eqs. (8,11,12) in Nagy et al., 2000 [26] (note that there is a mistake here that is fixed in [27]), appendix A in Couder-Castaneda et al., 2015 [7] and the derivation between (14) and (15) in Mader, 1951 [24].

3.4.4 The gravity gradient tensor (\mathbf{T})

The different elements of the gravity gradient tensor can be determined by partially differentiating each component of the gravity vector with respect to each axis in space. As will be shown later, \mathbf{T} should be a symmetric matrix and its trace should equal zero.

The diagonal terms

$$\begin{aligned} T_{xx} &= \frac{\partial}{\partial x} g_x = -\frac{\partial^2}{\partial x \partial x} U(P) = \mathcal{G}\rho \frac{\partial^2}{\partial x \partial x} (-u(P)) \\ u_{xx} &= \iiint \frac{\partial^2}{\partial x \partial x} \frac{dxdydz}{\sqrt{x^2 + y^2 + z^2}} \\ &= \iint \frac{\partial}{\partial x} \frac{dydz}{\sqrt{x^2 + y^2 + z^2}} \\ &= - \iint \frac{x dy dz}{\sqrt{x^2 + y^2 + z^2}^3} \end{aligned} \quad (3.45)$$

A trigonometric substitution is applied to solve this integral. This uses $y = \sqrt{x^2 + z^2} \tan \phi$ and $dy = \frac{\sqrt{x^2 + y^2}}{\cos^2 \phi} d\phi$.

$$\begin{aligned} u_{xx} &= -x \iint \frac{\sqrt{x^2 + y^2}}{\cos^2 \phi} \frac{d\phi dz}{\sqrt{(x^2 + z^2) \tan^2 \phi + x^2 + z^2}^3} \\ &= -x \iint \frac{\sqrt{x^2 + y^2}}{\cos^2 \phi} \frac{d\phi dz}{\sqrt{(x^2 + z^2)(\tan^2 \phi + 1)}^3} \\ &= -x \iint \frac{\sqrt{x^2 + y^2}}{\cos^2 \phi} \frac{d\phi dz}{\left(\frac{\sqrt{x^2 + z^2}}{\cos \phi} \right)^3} \\ &= -x \int \frac{1}{x^2 + z^2} \int \cos \phi \, d\phi dz \\ &= -x \int \frac{1}{x^2 + z^2} \sin \phi dz \end{aligned} \quad (3.46)$$

Now the substitution needs to be undone. If $\tan \phi = \frac{y}{\sqrt{x^2+z^2}}$, then $\sin \phi = \frac{y}{\sqrt{x^2+y^2+z^2}}$.

$$u_{xx} = -xy \int \frac{dz}{(x^2+z^2)\sqrt{x^2+y^2+z^2}} \quad (3.47)$$

This can be solved by applying equation (3.26).

$$\begin{aligned} u_{xx} &= -\frac{xy}{xy} \arctan \frac{yz}{x\sqrt{x^2+y^2+z^2}} \\ &= -\arctan \frac{yz}{x\sqrt{x^2+y^2+z^2}} \end{aligned} \quad (3.48)$$

The tensor element T_{xx} is then formulated as follows (the other elements of the diagonal are found by cyclic permutation of x , y and z):

$$\boxed{\begin{aligned} T_{xx} &= \mathcal{G}\rho \sum_{i=1}^2 \sum_{j=1}^2 \sum_{k=1}^2 (-1)^{i+j+k} \left(-\arctan \frac{y_j z_k}{x_i r_{ijk}} \right) \\ T_{yy} &= \mathcal{G}\rho \sum_{i=1}^2 \sum_{j=1}^2 \sum_{k=1}^2 (-1)^{i+j+k} \left(-\arctan \frac{x_i z_k}{y_j r_{ijk}} \right) \\ T_{zz} &= \mathcal{G}\rho \sum_{i=1}^2 \sum_{j=1}^2 \sum_{k=1}^2 (-1)^{i+j+k} \left(-\arctan \frac{x_i y_j}{z_k r_{ijk}} \right) \end{aligned}}$$

The off-diagonal terms of the tensor

The other elements are easier to calculate, because the partial derivatives cancel out the integrals:

$$\begin{aligned} u_{xy} &= \iiint \frac{\partial^2}{\partial x \partial y} \frac{dxdydz}{\sqrt{x^2+y^2+z^2}} \\ &= \iint \frac{\partial}{\partial y} \frac{dydz}{\sqrt{x^2+y^2+z^2}} \\ &= \int \frac{dz}{\sqrt{x^2+y^2+z^2}} \\ &= \ln \left(z + \sqrt{x^2+y^2+z^2} \right) \end{aligned} \quad (3.49)$$

$$\begin{aligned} u_{xz} &= \iiint \frac{\partial^2}{\partial x \partial z} \frac{dxdydz}{\sqrt{x^2+y^2+z^2}} \\ &= \iint \frac{\partial}{\partial z} \frac{dydz}{\sqrt{x^2+y^2+z^2}} \\ &= \int \frac{dy}{\sqrt{x^2+y^2+z^2}} \\ &= \ln \left(y + \sqrt{x^2+y^2+z^2} \right) \end{aligned} \quad (3.50)$$

$$\begin{aligned} u_{yz} &= \iiint \frac{\partial^2}{\partial y \partial z} \frac{dxdydz}{\sqrt{x^2+y^2+z^2}} \\ &= \iint \frac{\partial}{\partial z} \frac{dxdz}{\sqrt{x^2+y^2+z^2}} \\ &= \int \frac{dx}{\sqrt{x^2+y^2+z^2}} \\ &= \ln \left(x + \sqrt{x^2+y^2+z^2} \right) \end{aligned} \quad (3.51)$$

From these calculations it should be obvious why \mathbf{T} is a symmetric tensor. When applying the second partial derivatives, their order does not matter:

$$u_{xy} = \iiint \frac{\partial^2}{\partial x \partial y} \frac{dxdydz}{\sqrt{x^2 + y^2 + z^2}} = \iiint \frac{\partial^2}{\partial y \partial x} \frac{dxdydz}{\sqrt{x^2 + y^2 + z^2}} = u_{yx} \quad (3.52)$$

The tensor elements following from this are:

$$T_{xy} = T_{yx} = \mathcal{G}\rho \sum_{i=1}^2 \sum_{j=1}^2 \sum_{k=1}^2 (-1)^{i+j+k} (\ln(z_k + r_{ijk}))$$

$$T_{xz} = T_{zx} = \mathcal{G}\rho \sum_{i=1}^2 \sum_{j=1}^2 \sum_{k=1}^2 (-1)^{i+j+k} (\ln(y_j + r_{ijk}))$$

$$T_{yz} = T_{zy} = \mathcal{G}\rho \sum_{i=1}^2 \sum_{j=1}^2 \sum_{k=1}^2 (-1)^{i+j+k} (\ln(x_i + r_{ijk}))$$

3.4.5 Summary

- Gravitational potential U

$$U(P) = \mathcal{G}\rho \sum_{i=1}^2 \sum_{j=1}^2 \sum_{k=1}^2 (-1)^{i+j+k} \left(y_j z_k \ln(x_i + r_{ijk}) + x_i y_j \ln(z_k + r_{ijk}) + x_i z_k \ln(y_j + r_{ijk}) - \frac{x_i^2}{2} \arctan \frac{z_k y_j}{x_i r_{ijk}} - \frac{y_j^2}{2} \arctan \frac{x_i z_k}{y_j r_{ijk}} - \frac{z_k^2}{2} \arctan \frac{x_i y_j}{x_i r_{ijk}} \right) \quad (3.53)$$

- Gravity vector \vec{g}

$$g_x = \mathcal{G}\rho \sum_{i=1}^2 \sum_{j=1}^2 \sum_{k=1}^2 (-1)^{i+j+k} \left(z_k \ln(y_j + r_{ijk}) + y_j \ln(z_k + r_{ijk}) - x_i \arctan \frac{y_j z_k}{x_i r_{ijk}} \right) \quad (3.54)$$

$$g_y = \mathcal{G}\rho \sum_{i=1}^2 \sum_{j=1}^2 \sum_{k=1}^2 (-1)^{i+j+k} \left(z_k \ln(x_i + r_{ijk}) + x_i \ln(z_j + r_{ijk}) - y_j \arctan \frac{x_i z_k}{y_j r_{ijk}} \right) \quad (3.55)$$

$$g_z = \mathcal{G}\rho \sum_{i=1}^2 \sum_{j=1}^2 \sum_{k=1}^2 (-1)^{i+j+k} \left(x_i \ln(y_j + r_{ijk}) + y_j \ln(x_i + r_{ijk}) - z_k \arctan \frac{x_i y_j}{z_k r_{ijk}} \right) \quad (3.56)$$

- Gravity gradient tensor \mathbf{T}

$$T_{xx} = \mathcal{G}\rho \sum_{i=1}^2 \sum_{j=1}^2 \sum_{k=1}^2 (-1)^{i+j+k} \left(-\arctan \frac{y_j z_k}{x_i r_{ijk}} \right) \quad (3.57)$$

$$T_{yy} = \mathcal{G}\rho \sum_{i=1}^2 \sum_{j=1}^2 \sum_{k=1}^2 (-1)^{i+j+k} \left(-\arctan \frac{x_i z_k}{y_j r_{ijk}} \right) \quad (3.58)$$

$$T_{zz} = \mathcal{G}\rho \sum_{i=1}^2 \sum_{j=1}^2 \sum_{k=1}^2 (-1)^{i+j+k} \left(-\arctan \frac{x_i y_j}{z_k r_{ijk}} \right) \quad (3.59)$$

$$T_{xy} = T_{yx} = \mathcal{G}\rho \sum_{i=1}^2 \sum_{j=1}^2 \sum_{k=1}^2 (-1)^{i+j+k} (\ln(z_k + r_{ijk})) \quad (3.60)$$

$$T_{xz} = T_{zx} = \mathcal{G}\rho \sum_{i=1}^2 \sum_{j=1}^2 \sum_{k=1}^2 (-1)^{i+j+k} (\ln(y_j + r_{ijk})) \quad (3.61)$$

$$T_{yz} = T_{zy} = \mathcal{G}\rho \sum_{i=1}^2 \sum_{j=1}^2 \sum_{k=1}^2 (-1)^{i+j+k} (\ln(x_i + r_{ijk})) \quad (3.62)$$

3.4.6 Revisiting Poisson's equation

Poisson's equation applied to gravity is $\nabla^2 U = 4\pi G\rho$. This is true in general cases, but can be verified for the derived equations for prisms, which would provide extra support for them. The equation can be used in two cases. Inside the prism, the density has some assigned value. Outside of the prism, the density is zero, so the result is $\nabla^2 U = 0$. These will be treated separately.

Outside the prism

The nabla in Poisson's equation for gravitation can be rewritten. $\nabla^2 U = 0$ outside of the prism becomes $\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} + \frac{\partial^2 U}{\partial z^2} = 0$. These partial derivatives are the elements of the main axis of \mathbf{T} , the addition is the same as the trace of \mathbf{T} . This can be calculated. The first step is adding all of the elements on the main axis and then the boundary conditions are applied. The formula to add arctangents together is as follows:

$$\arctan a + \arctan b = \arctan \frac{a+b}{1-ab} \quad (3.63)$$

Now adding the terms T_{xx} and T_{yy}

$$\begin{aligned} T_{xx} + T_{yy} &= (-1)^{i+j+k} \arctan \frac{yz}{xr} + (-1)^{i+j+k} \arctan \frac{xz}{yr} \\ &= (-1)^{i+j+k} \arctan \frac{\frac{yz}{xr} + \frac{xz}{yr}}{1 - \frac{yz}{xr} \frac{xz}{yr}} \\ &= (-1)^{i+j+k} \arctan \frac{\frac{y^2 z}{xyr} + \frac{x^2 z}{xyr}}{1 - \frac{xyz^2}{xyr^2}} \\ &= (-1)^{i+j+k} \arctan \frac{\frac{z(x^2+y^2)}{xyr}}{\frac{xy(r^2-z^2)}{xyr^2}} \\ &= (-1)^{i+j+k} \arctan \frac{xyzr^2(x^2+y^2)}{x^2y^2r(r^2-z^2)} \\ &= (-1)^{i+j+k} \arctan \frac{zr(x^2+y^2)}{xy(x^2+y^2+z^2-z^2)} \\ &= (-1)^{i+j+k} \arctan \frac{zr}{xy} \end{aligned} \quad (3.64)$$

By considering a right triangle with sides 1 and x, it should be easy to prove that:

$$\arctan x + \arctan \frac{1}{x} = \frac{\pi}{2} \quad (3.65)$$

This can be used to transform the arctan to one that is similar to T_{zz} .

$$T_{xx} + T_{yy} = (-1)^{i+j+k} \arctan \frac{zr}{xy} = (-1)^{i+j+k} \left(\frac{\pi}{2} - \arctan \frac{xy}{zr} \right) \quad (3.66)$$

The last step is to add the last term (T_{zz}).

$$\begin{aligned} \nabla^2 U &= T_{xx} + T_{yy} + T_{zz} = (-1)^{i+j+k} \left(\frac{\pi}{2} - \arctan \frac{xy}{zr} + \arctan \frac{xy}{zr} \right) \\ &= (-1)^{i+j+k} \frac{\pi}{2} \end{aligned} \quad (3.67)$$

The end result is a single value. When the boundary conditions are applied this single value will be subtracted from itself resulting in zero, so:

$$\nabla^2 U = 0 \quad (3.68)$$

Inside the prism

Inside the prism $\nabla^2 U = 4\pi G\rho$. This can be shown by putting the observation point at the centre of the prism. The coordinates of the prism are now:

$$\begin{aligned} -x_1 &= x_2 \\ -y_1 &= y_2 \\ -z_1 &= z_2 \end{aligned} \tag{3.69}$$

All eight terms for these conditions results give $\frac{\pi}{2}$, so the result is:

$$\nabla^2 U = G\rho 8 \frac{\pi}{2} = 4\pi G\rho \tag{3.70}$$

3.4.7 Better numerical stability

Heck and Seitz (2007) [15] modify the standard formulae for the prism to get a better numerical stability in the logarithms. This is done by dividing the inside of the logs by an extra factor:

$$\ln(z_k + r_{ijk}) \rightarrow \ln \frac{z_k + r_{ijk}}{\sqrt{x_i^2 + y_j^2}} \tag{3.71}$$

This extra factor is removed when computing the boundary conditions for z , so that the result is still the same.

$$\begin{aligned} \left| \ln \frac{z_k + r_{ijk}}{\sqrt{x_i^2 + y_j^2}} \right|_{z_1}^{z_2} &= \left| \ln(z_k + r_{ijk}) - \ln \sqrt{x_i^2 + y_j^2} \right|_{z_1}^{z_2} \\ &= \ln(z_2 + r_{ijk}) - \ln \sqrt{x_i^2 + y_j^2} - \ln(z_1 + r_{ijk}) + \ln \sqrt{x_i^2 + y_j^2} \\ &= \ln(z_2 + r_{ijk}) - \ln(z_1 + r_{ijk}) \end{aligned} \tag{3.72}$$

Chapter 4

Computational methods

The first step in creating the gravity forward model is creating a function that will allow the calculation of the gravitational response, U , \vec{g} and \mathbf{T} , while the model is set up separately. The model is split up into discretised parts. The larger space will be called the box and the parts cells. Each cell is assumed to have a homogeneous density. The coordinates and densities will be different for every cell, so these should be able to be changed. The results are the gravitational response, comprising the potential, gravity vector and gravity gradient components (U , \vec{g} and \mathbf{T}).

The cells are cubic in the benchmark cases, because this form is more convenient to use. Other data sets may use other dimensions, so the option for either needs to be available. Because of the superposition principle, the gravitational contribution of each cell can be added together to get the gravitational signal of the whole box. This contribution can be calculated according to the point mass and prism methods as derived in chapter 3. The first, most basic function will be one that calculates U , \vec{g} and \mathbf{T} when the geometry of the box is already defined.

Then, the accuracy of this function will be tested. This requires some cases where the analytical solution of the gravity is known. These benchmark cases should show whether the increased accuracy of the prisms or the better speed of the point mass approach gives better results. The first benchmark case is a sphere. This is a shape that is hard to approximate with the rectangular cuboids. The second case is a prism buried in a larger prism. This shape fits well with the shape of the cells. These two cases will show if the shape of the model has an effect on which method of calculation performs better. After this, another case is used. This is the salt diapir, a more realistic shape, to see how the calculations perform.

All of the functions are created using Python 3.7. A Jupyter Notebook with all of the code used and described here can be found at:

https://github.com/shwhassing/BSc_thesis

4.1 A basic function

As a first step, the contribution of a single cell must be defined. This is the most fundamental function of the model. There are two options for using prisms or point masses.

- prism: The location of the cell is identified by the corner with the smallest coordinates.
- obs_point: The gravity calculations always need an observation point from where the calculations are done. This contains the coordinates of that point.
- method: The mode of calculations. The options are the prism or point mass methods.
- rho: Each cell is assumed to have a homogeneous density. This variable gives that density.
- h_x: The dimensions of the cell on the x-axis. The dimensions on the other axes can also be given. If they are left empty, the function assumes that the cell is a cube and these other lengths will be set to the same as the length on the x-axis.

The first step is to convert the coordinate system from the larger one that is used to identify every point to one with the observation point at its centre. This is done by simply subtracting the coordinates of the observation point from the coordinates of the cube.

After this the steps for point mass and prism diverge. The point mass option calculates the coordinates of the centre of the cell, where the point mass is located. Then it determines the distance from this to the observation point and calculates the volume of the cell. With the equations from section 3.3 U , \vec{g} and \mathbf{T} are calculated.

The prism option starts by calculating the outer coordinates of the cell. The triple summation in the formula is adapted as a triple loop. There are certain values where there is no solution to the logarithms and arctans. As in Nagy et al., 2007 [26] the limit values are chosen for these cases. This means that for example $\ln(0) \rightarrow 0$ and $\arctan \frac{1}{0} \rightarrow \frac{\pi}{2}$. Then, the inner parts of the sums are calculated and finally multiplied with G and ρ . Heck and Seitz, 2007 [15] offer an alternative to these calculations that should give extra numerical stability in the logarithms. This can be turned on and off.

The function returns U , \vec{g} and T .

```

1 def grav_calc(prism,obs_point,method,rho,h_x,h_y=None,h_z=None, use_num_stab = False):
2     # Converting the given corner of the cell to the observation coordinate system
3     coords = prism - obs_point
4
5     # If they are not provided, they are just the same as h_x, so if only h_x is given, the cell is a
6     # cube
7     if h_y == None:
8         h_y = h_x
9     if h_z == None:
10        h_z = h_x
11
12    G = 6.6738480e-11 # gravitational constant [m^3 s^-2 kg^-1]
13
14    U_calc = 0.
15    g_calc = np.zeros(3,dtype=np.float64)
16    T_calc = np.zeros([3,3],dtype=np.float64)
17
18    if method == "point":
19        # The coordinates needed are at the point mass in the centre of the cell
20        x = coords[0] + 0.5*h_x
21        y = coords[1] + 0.5*h_y
22        z = coords[2] + 0.5*h_z
23
24        # calculating the distance between the observation point and the point mass
25        r = np.sqrt(x**2+y**2+z**2)
26        # The volume of the cell
27        dV = h_x*h_y*h_z
28
29        # calculating the different values
30        U_calc = -G * rho * dV / r
31        g_calc = G * rho * dV * x / r**3, G * rho * dV * y / r**3, G * rho * dV * z / r**3
32
33        T_calc[0,0] = -G*rho*dV*(3*x**2-r**2)/r**5
34        T_calc[1,1] = -G*rho*dV*(3*y**2-r**2)/r**5
35        T_calc[2,2] = -G*rho*dV*(3*z**2-r**2)/r**5
36        T_calc[0,1] = -G*rho*dV*3*x*y/r**5
37        T_calc[1,0] = T_calc[0,1]
38        T_calc[0,2] = -G*rho*dV*3*x*z/r**5
39        T_calc[2,0] = T_calc[0,2]
40        T_calc[1,2] = -G*rho*dV*3*y*z/r**5
41        T_calc[2,1] = T_calc[1,2]
42
43    elif method == "prism":
44        # The coordinates needed are the boundary coordinates of the prism. If the prism would
45        # range from 1 to 5 on the x axis, 2 to 4 on the y axis and 0 to 6 on the z axis the two
46        # points would be (1,2,0) and (5,4,6) and the arrays x=[1,5], y=[2,4] and z=[0,6].
47        x = [coords[0],coords[0]+h_x]
48        y = [coords[1],coords[1]+h_y]
49        z = [coords[2],coords[2]+h_z]
```

```

50
51 # The triple summation is implemented as a triple loop
52 for i in range(2):
53     for j in range(2):
54         for k in range(2):
55             # The distance between the observation point and the integration point is
56             # calculated
57             r = np.sqrt(x[i]**2+y[j]**2+z[k]**2)

58
59             # There are cases where the calculations will fail, f.e. log(0) or arctan(1/0).
60             # To stop this, exception cases are defined. They are set to the limit values,
61             # so log(0) -> 0 and arctan(1/0) -> 1/2 pi
62             if x[i] == 0:
63                 arctan_x = - 0.5*np.pi
64             else:
65                 arctan_x = - np.arctan((y[j]*z[k]/(x[i]*r)))
66             if y[j] == 0:
67                 arctan_y = - 0.5*np.pi
68             else:
69                 arctan_y = - np.arctan((x[i]*z[k]/(y[j]*r)))
70             if z[k] == 0:
71                 arctan_z = - 0.5*np.pi
72             else:
73                 arctan_z = - np.arctan((x[i]*y[j]/(z[k]*r)))

74
75             # Along with the exceptions, there is a variant of the equations that should
76             # offer extra numerical stability. This can be turned on if wanted
77             if r+x[i] == 0:
78                 log_x = 0
79             else:
80                 if use_num_stab == False:
81                     log_x = np.log(x[i]+r)
82                 else:
83                     log_x = np.log((x[i]+r)/(np.sqrt(y[j]**2+z[k]**2)))
84             if r+y[j] == 0:
85                 log_y = 0
86             else:
87                 if use_num_stab == False:
88                     log_y = np.log(y[j]+r)
89                 else:
90                     log_y = np.log((y[j]+r)/(np.sqrt(x[i]**2+z[k]**2)))
91             if r+z[k] == 0:
92                 log_z = 0
93             else:
94                 if use_num_stab == False:
95                     log_z = np.log(z[k]+r)
96                 else:
97                     log_z = np.log((z[k]+r)/(np.sqrt(x[i]**2+y[j]**2)))

98             U_calc += (-1)**(i+j+k) * (y[j]*z[k]*log_x + x[i]*y[j]*log_z + x[i]*z[k]*log_y +
99                                         + x[i]**2/2*arctan_x + y[j]**2/2*arctan_y +
100                                         + z[k]**2/2*arctan_z)

101
102             g_calc[0] += (-1)**(i+j+k) * (z[k]*log_y + y[j]*log_z + x[i]*arctan_x)
103             g_calc[1] += (-1)**(i+j+k) * (z[k]*log_x + x[i]*log_z + y[j]*arctan_y)
104             g_calc[2] += (-1)**(i+j+k) * (x[i]*log_y + y[j]*log_x + z[k]*arctan_z)

```

```

106             T_calc[0,0] += (-1)**(i+j+k) * arctan_x
107             T_calc[1,1] += (-1)**(i+j+k) * arctan_y
108             T_calc[2,2] += (-1)**(i+j+k) * arctan_z
109             T_calc[0,1] += (-1)**(i+j+k) * log_z
110             T_calc[0,2] += (-1)**(i+j+k) * log_y
111             T_calc[1,2] += (-1)**(i+j+k) * log_x
112
113     # T is symmetric
114     T_calc[1,0] = T_calc[0,1]
115     T_calc[2,0] = T_calc[0,2]
116     T_calc[2,1] = T_calc[1,2]
117
118     # When the summations are complete, everything is multiplied with G and rho to get the
119     # final values
120     U_calc *= G*rho
121     g_calc *= G*rho
122     T_calc *= G*rho
123
124     # The results are returned
125     return U_calc, g_calc, T_calc

```

This function calculates the contribution of a single cell to U , \vec{g} and \mathbf{T} with a single observation point. The next step is to combine every cell in the box. When everything is set up, this can be repeated for multiple observation points. This is combined in a function that has the following input:

- N_x, N_y, N_z : The amount of cell on each axis. The total amount of cells in the box is $N_x N_y N_z$.
- corners: A list with the coordinates of every cell. The coordinates are the same corner as described in the previous function.
- obs_points: A list with the coordinates of every observation point. U , \vec{g} and \mathbf{T} will be calculated for each of these points.
- rho: A list with the densities of every cell.
- h_x: The length of a cell. The conditions are the same as in the previous function.
- num_stab: Whether or not to use the extra numerical stability possible with prisms.
- tracking: Whether or not to print the progress of the calculations for every observation point done.

The function creates a loop over each observation point and one inside over each cell. The method of calculation is passed through to the function `grav_calc`, allowing calculations of both prisms and point masses. The contributions of each cell are added together to get the total value of each observation point. The results are arrays of U , \vec{g} and \mathbf{T} with values for every observation point.

```

1 def full_calc(corners,obs_points,method,rho,h_x,h_y=None,h_z=None, num_stab = False,
2                                         tracking = False):
3
4     M = len(obs_points) # the amount of observation points
5
6     # The total amount of cells is interpreted from the length of corners
7     NP = len(corners)
8
9     # These will contain the values of U and g
10    U = np.zeros(M, dtype = np.float64)
11    g = np.zeros([M,3],dtype=np.float64)
12    T = np.zeros([M,3,3],dtype=np.float64)
13

```

```

14 # A loop for every observation point
15 for k in range(M):
16     # And one for every prism
17     for l in range(NP):
18         # Getting the contribution of this prism with the prism calculations
19         U_dum, g_dum, T_dum = grav_calc(corners[:, :], obs_points[:, k], method, rho[l], h_x,
20                                         h_y, h_z, use_num_stab = num_stab)
21
22         # Adding it to the total for this observation point
23         U[k] += U_dum
24         g[:, k] += g_dum
25         T[:, :, k] += T_dum
26
27     # This prints how many observation points are done, it prints over itself so as not to
28     # clutter the output
29     if tracking == True:
30         print("\r", "Finished with observation point", k+1, "of", M, end="", flush=True)
31     # Enter a new line after the progress line
32     if tracking == True:
33         print("")
34
35     # Return the results
36     return U, g, T

```

4.2 Benchmark cases

A good way to compare the two methods and the functions that are defined is by setting up an easy geometry. If the analytical solution of such a case is known and not too difficult to obtain, the results from the functions can be compared in accuracy. The general formula for the relative error of a variable x with theoretical solution x_{th} can be calculated with:

$$\text{Rel. error} = \left| \frac{x - x_{th}}{x_{th}} \right| \quad (4.1)$$

The cost in computation time can also be determined. The accuracy should improve with decreasing cell size, but the amount of cells also increases. N is the amount of cells on one axis, so the total amount of cells is N^3 . When N is twice as large, the amount of cells is eight times as large. So the amount of calculations is also eight times as large. Based on this the computation time is expected to be related to N as $t \sim N^3$.

4.2.1 Sphere

The first model is a homogeneous sphere. The discretisation into cubes means that the shape of the sphere must always be approximated. It also means that some arrangements fit better into the sphere than others. How this is possible is shown in figure 4.1. Despite decreasing the cell size from A. to B., the approximation of the circle's volume is worse. The error that is created could perhaps be predicted for a sphere, but in realistic cases this is not possible. As such this will not be taken into account during the actual calculations.

The model creates a cubic box with sides the length of the diameter of the sphere. This means that the sphere fits exactly into the box. In this case, the sphere in question will be a simplified Earth. The box is divided into the smaller cells, with N cells on each axis. The centre of the coordinate system will be at the centre of the sphere. This way, the norm of a place vector will also be the distance from the core.

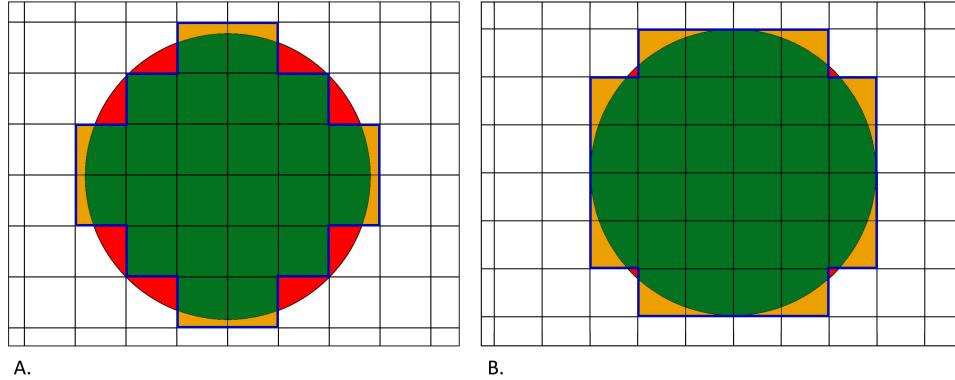


Figure 4.1: Two grids covering a sphere. The surface area of the same circle is approximated by using cells in both grids (blue line). Dark green shows the parts that belong in the circle and are used correctly, the light green parts show what is outside the circle, but is used by the approximation nonetheless and the red parts show the area that should belong to the circle, but does not because of the grid approximation. In A. the cell size is larger, but the approximation of the surface area of the circle is better, because the orange parts compensate for the red parts. In B. the cell size is smaller but the approximation is worse.

The sphere is identified when assigning the density to each cell. There are multiple possible density models to use. The easiest is a homogeneous sphere. In this case the density is the same everywhere and an analytical solution to the gravitational response is possible. There are also two density models for the real Earth, AK135 [21] and PREM [11]. The density is assigned according to the location of the centre of the cube. If the distance to the centre of the sphere is smaller than the radius of the sphere sphere, the cell gets a density value, otherwise the density is zero. The assignation is handled by a function. The input variables are:

- corner: The coordinates of the cube that needs a density assigned.
- h: The dimensions of the cube or the length of one side.
- mode: Which density distribution is used.
- rho0: The density of the homogeneous sphere.

```

1 def density_sphere(corner,h_x, mode, h_y=None,h_z=None,rho0 = 3000):
2
3     # If these are not provided, they are just the same as h_x, so if only h_x is given, the cell is
4     # a cube
5     if h_y == None:
6         h_y = h_x
7     if h_z == None:
8         h_z = h_x
9
10    # The density is assigned according to the density of the sphere at the centre of the cell, so the
11    # location of this centre is determined
12    radius = np.sqrt((corner[0]+0.5*h)**2+(corner[1]+0.5*h)**2+(corner[2]+0.5*h)**2)
13
14    # The density at some location of the sphere is determined by the density structure of the sphere.
15    # The options are homogeneous, PREM and AK135
16    if mode == "hom":
17        # If the sphere has a homogeneous density, the density should give rho0 if the centre is inside
18        # the sphere, otherwise it should be zero
19        if radius < 6371e3:
20            return rho0
21        else:

```

```

22         return 0
23     if mode == "PREM":
24         # The radius is converted to a fraction of the radius and that is used in formulas
25         # describing the PREM model as curves
26         x=radius/6371.0e3
27         if radius>6371.0e3:
28             densprem=0
29         elif radius<=1221.5e3:
30             densprem=13.0885-8.8381*x**2
31         # etc. the rest of the PREM function is cut off. The full version can be found at:
32         # https://github.com/cedrict/fieldstone/blob/master/images/prem/prem.py
33     if mode == "AK135":
34         # This method interpolates between data that needs to be imported
35         return np.interp(radius,depths,densities)

```

Setting up the sphere is also handled in a function. This combines creating the geometry of a sphere, assigning the density and the basic function that calculates U , \vec{g} and \mathbf{T} . These variables are also the results of this function. The input variables are:

- N: How many cells each axis contains.
- obs_points: The coordinates of the observation points that will be used.
- density_struc: Which density distribution is used for the sphere.
- rho0: The density of the homogeneous sphere.
- num_stab: Whether or not the alternative calculations for prisms that give extra numerical stability are used.
- tracking: Whether or not the progress per observation point is printed.

```

1 def full_sphere_calc(N,obs_points,method,density_struc = "hom",rho0 = 3000, tracking = False):
2
3     R = 6.371e6 # radius of the sphere [m]
4
5     volth=4/3*np.pi*R**3 # analytical volume of the sphere [m^3]
6
7     N = int(N) # making sure that N is an integer, otherwise it won't work
8
9     NP = N**3 # total amount of prisms
10
11    h = 2*R/N # the length of one cell
12
13    corners = np.empty([NP,3],dtype=np.float64) # the coordinates of the prism at the corner with the
14                                              # smallest coordinates
15
16    counter = 0
17    # Looping over every axis
18    for i in range(N):
19        for j in range(N):
20            for k in range(N):
21                # The loops start at coordinate -R,-R,-R and take steps of h slowly filling a grid
22                # to R-h,R-h,R-h
23                corners[counter,0] = i*h - R
24                corners[counter,1] = j*h - R
25                corners[counter,2] = k*h - R
26                counter += 1

```

```

27
28     rho = np.empty([NP], dtype=np.float64)
29
30     # Creating the density structure, depending on what mode is selected
31     for i in range(NP):
32         # The density function gives the density at a certain distance from the sphere according
33         # to the method of choice
34         rho[i] = density_sphere(corners[i,:], h, density_struc, rho0)
35
36     # Calculating the gravitational response with this model
37     U,g,T = full_calc(N,N,N,corners,obs_points,method,rho,h, tracking = tracking)
38
39     return U,g,T

```

4.2.2 Buried prism

The second model is a buried prism with a different density compared to its surroundings. This is a more advanced version of an example used in Arroyo et al., 2015 [2] (figure 7). The prism is a cube with sides of length R_{buried} in a larger box with sides R . It is buried to a depth of buried_depth . The prism has density ρ_0 , while the rest has density ρ_1 . This is shown in figure 4.2. The centre of the coordinate system is placed on the left bottom corner of the box. Every point inside the box has positive coordinates.

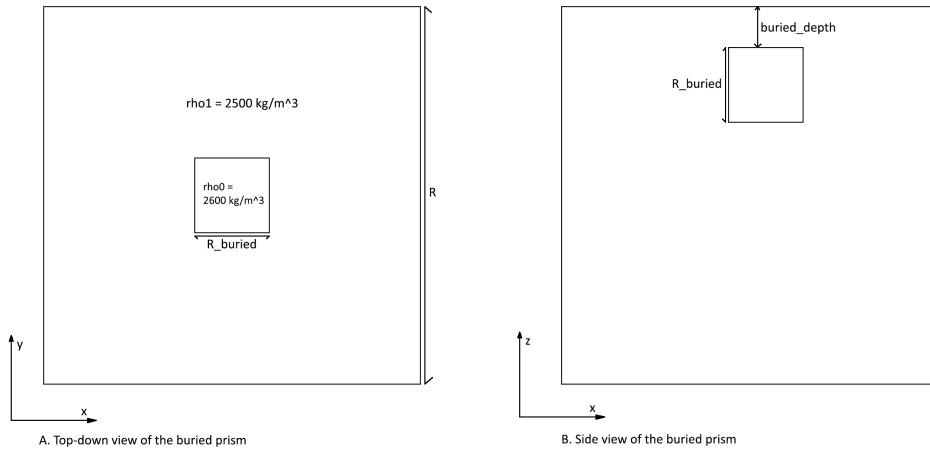


Figure 4.2: The different dimensions of the buried prism.

The original case from Arroyo et al, 2015 [2] is much simpler. The 'buried' prism floats 100 m above the observation mesh. The prism is a cube with sides of 200 m, while the observation grid is 1 km x 1 km. This floating cube represents a cube buried in a less dense material. The density of this material is subtracted from every cell. The density of the cube is the difference between its own density and the subtracted amount.

The reason for doing this is because in an ideal case the surrounding material would extend a long distance to all sides. This is not possible, so it is cut off at some point. The edges of the material would give boundary effects that interfere with the effects of the buried prism that are of interest. When the density of the surrounding material is zero, it is effectively not there and no boundary effects will show up. This method gives the following parameters for the program: $R = 200 \text{ m}$, $R_{\text{buried}} = 200 \text{ m}$, $\text{buried_depth} = 100 \text{ m}$, $\rho_0 = 100 \text{ kg/m}^3$ and $\rho_1 = 0 \text{ kg/m}^3$.

This benchmark case is interesting, because the cubes can fit perfectly inside the specified geometry. There are a few conditions that must be met for this to be the case:

- The cells must fit perfectly into the buried prism. This is the case if R_{buried} divided by N results in an integer.
- The cells must fit perfectly in the surroundings of the prism. This is the case if both $\text{buried_depth}/N$ and $\frac{1}{2}(R - R_{\text{buried}})/N$ result in an integer.

- The size of the cells is small enough to differentiate between the buried prism and its surroundings. This is the case if $R/N \leq \text{buried_depth}$.

The new geometry needs a new way to assign the density. In the interval of x and y with length R_{buried} around the middle of the box and of z with the same length ending at distance buried_depth from the top. This is done with parabolic equations for each interval.

```

1  def density_buried(corner,R,R_buried,buried_depth,rho0,rho1,h_x,h_y=None,h_z=None):
2
3      # If they are not provided, they are just the same as h_x, so if only h_x is given, the cell is
4      # a cube
5      if h_y == None:
6          h_y = h_x
7      if h_z == None:
8          h_z = h_x
9
10     # The x,y and z coordinates of the centre of the cell
11     x = corner[0] + 0.5*h_x
12     y = corner[1] + 0.5*h_y
13     z = corner[2] + 0.5*h_z
14
15     # These give intervals of length R_buried around R/2 on the x and y axis and around R -
16     # buried_depth - R_buried on the z-axis
17     if -(x-R/2)**2 + (R_buried/2)**2 >= 0 and -(y-R/2)**2 + (R_buried/2)**2 >= 0 and
18                     -(z-R+buried_depth+R_buried/2)**2 + (R_buried/2)**2 >= 0:
19         return rho1
20     else:
21         return rho0

```

This density function and the new geometry lead to a new function that gives the gravitational response. The input variables are:

- N_x, N_y, N_z: Amount of cubes on each axis.
- obs_points: The coordinates of the observation points.
- R: The size of the (cubic) box.
- R_buried: The size of the (cubic) buried prism.
- buried_depth: The distance between the buried prism and the surface.
- rho0: The density of the buried prism.
- rho1: The density of the surrounding material.

```

1  def full_buried_calc(N,obs_points,method,R,R_buried,buried_depth,rho0 = 2500,rho1=2600,
2                                         tracking = False):
3
4      N = int(N) # making sure N is an integer, otherwise it will cause problems
5
6      NP = N**3 # total amount of prisms
7
8      h = R/N # the length of one prism
9
10     corners = np.empty([NP,3],dtype=np.float64) # the coordinates of the prism at the corner with
11                                         # the smallest coordinates

```

```

12
13     counter = 0
14     # Looping over every axis
15     for i in range(N):
16         for j in range(N):
17             for k in range(N):
18                 # The loops start at coordinate -R,-R,-R and take steps of h slowly filling a grid
19                 # to R-h,R-h,R-h
20                 corners[counter,0] = i*h
21                 corners[counter,1] = j*h
22                 corners[counter,2] = k*h
23                 counter += 1
24
25     rho = np.empty([NP], dtype=np.float64)
26
27     # Creating the density structure, depending on what mode is selected
28     for i in range(NP):
29         # The density function gives the density at a certain distance from the sphere according to
30         # the method of choice
31         rho[i] = density_buried(corners[i,:],R,R_buried,buried_depth,rho0,rho1,h)
32
33     # Calculating the gravitational response with this model
34     U,g,T = full_calc(N,N,N,corners,obs_points,method,rho,h, tracking = tracking)
35
36     return U,g,T

```

4.3 Salt diapir

The benchmark cases are a good way to test the numerical values. They do not give a realistic representation of the gravitational response that could be encountered with real models of the subsurface. To be able to study the differences between the two methods of calculation even in these scenarios, the model of a salt diapir is introduced. The model is synthetically generated using methods described in Clausolles et al., 2019 [5]. The method takes seismic and geological data to constrain boundaries that are definitely part of the diapir, definitely not and an uncertainty envelope in between. The boundary of the diapir is generated in this envelope, with the possibility of taking geological constraints into account.

The model consists of an area of 2940x2100x3060 (xyz) metres. The source layer of the salt covers the bottom and is cut off at every side except its top by the boundaries of the box. The model only differentiates between the sediments surrounding the diapir and the salt. Every cell is assigned as either salt or not salt. There are no differences recorded within the sediments.

Two versions of the model are used. A ‘low’ resolution version with 98x70x153 cells and a ‘high’ resolution version with 306x195x135 cells. A visualisation of the low resolution version of the model can be seen in figure 4.3. Even the low resolution version of the salt diapir contains more than a million cells. The same method to reduce boundary effects that is described for the buried prism in section 4.2.2 can be applied to the sediments. The density of the sediments is subtracted from the density of every cell. Salt is less dense and will have a negative relative density. All of the sedimentary cells will have a density of zero and have no effect on the gravitation. To reduce the computation time, these cells can be ignored, which reduces the amount of cells by a factor of ten.

4.3.1 Gradient magnitudes

All calculations that have been discussed thus far, have resulted in one of the different variables describing the gravitational response (U , \vec{g} , T). When these values are mapped on the surface, they show complicated patterns that are not easily interpreted, especially the gravity gradient tensor. In Saad, 2006 [32] combinations of the tensor elements are used that correlate to the underground structure and are easier to interpret. The horizontal gradient magnitude of g_z shows the outline of the structure. This magnitude is defined as:

$$\text{Horizontal gradient magnitude of } g_z = \sqrt{T_{zx}^2 + T_{zy}^2}$$

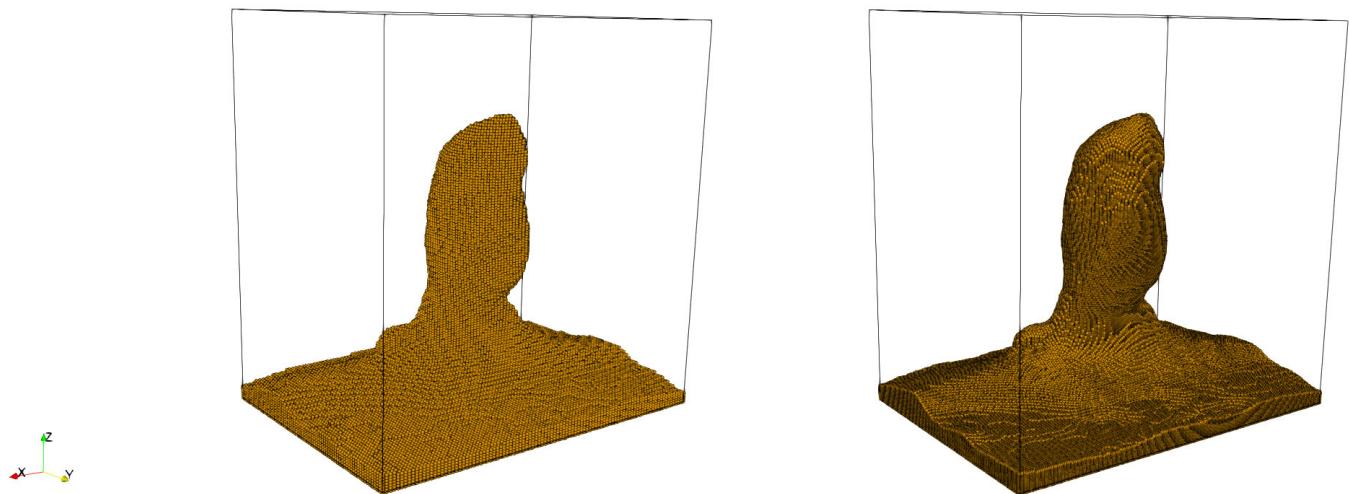


Figure 4.3: A visualisation of both the low and high resolution models of the salt diapir. The black outline gives the size of the box. Each orange circle denotes a cell that is identified as salt.

Chapter 5

Results

5.1 Benchmark cases

5.1.1 Timing

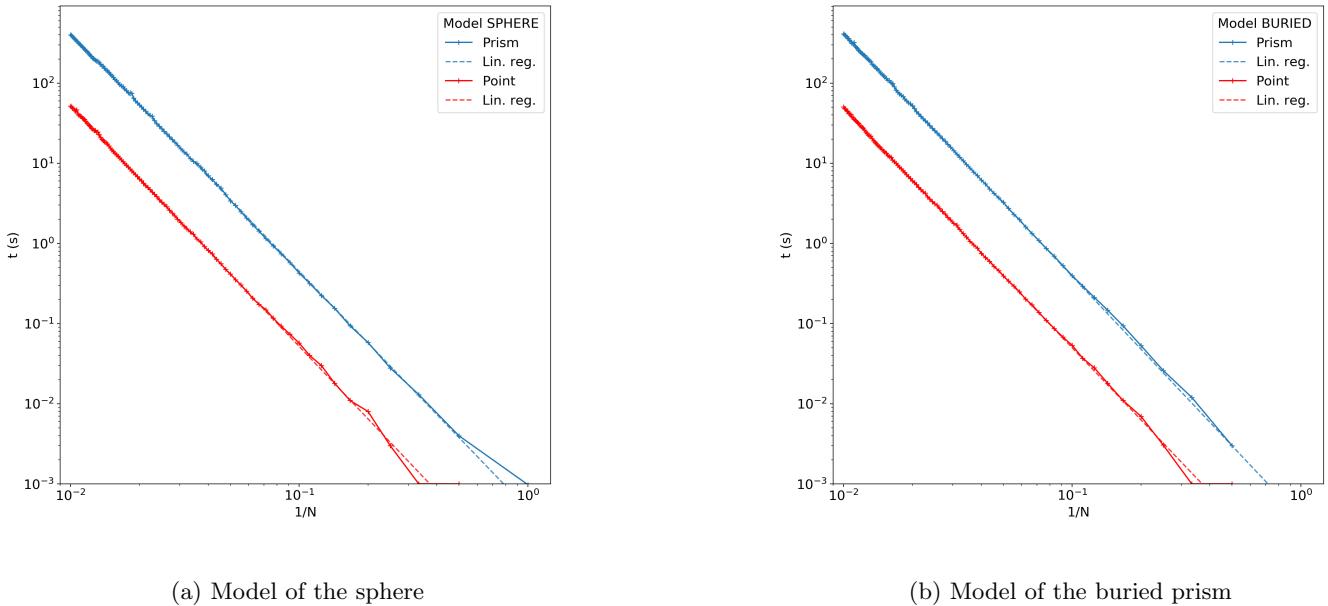


Figure 5.1: The computation time of the gravity calculations. The models are divided into $N \times N \times N$ cells, so that the cell size is $1/N$ of the total size of the box. The calculations use every cell to calculate the gravitational signal on a single observation point. The two different methods of calculations are used in the case of a sphere in space and a buried prism.

The duration of the gravitational calculations in both benchmark cases can be seen in figure 5.1. The most interesting observation that can be made is that the decrease in computation time with increase of the cell size is linear in log-log scale. The slope of this line is the same for prisms and point masses and for both models. A difference between the models would be unexpected. After all, the amount of cells is the same, only the densities differ, which should not give a significant difference in computation time.

Large cell sizes cause the computation time to diverge from the line. The calculations are executed at such a speed that there is some trouble with accurately recording the time. When N is one or two, it is even possible that the calculations are done so quickly that the computation time is recorded as zero seconds. For this reason the ten largest cell sizes are not included when the slope of the line is investigated.

The slope of the lines can be researched by applying a linear regression to the data. Because the line is linear in log-log space the coefficients have a different meaning than in normal space. Where normally the formula $y = a * x + b$ would be used, with a - slope and b - intersect, now the formula $\log y = a \log x + b \rightarrow y = 10^b x^a$ is used. The slope of the line can be interpreted as the exponent of the variable x .

A linear regression based on every set is shown as a dotted line in figure 5.1. The average slope of these regressions is -2.995 . The relation resulting from this is $t \sim (\frac{1}{N})^{-2.995} \rightarrow t \sim N^{2.995}$. This is very close to the expected value of 3 that was formulated in section 4.2.

The graphs (fig. 5.1) also show a consistent difference in computation time between the prism and point mass methods. The prisms take nine times as long to complete the calculations as the point mass method when the cell size is the same. That the calculations for the prism method would take longer is as expected considering the difference in complexity of the equations underlying each method.

5.1.2 Relative error

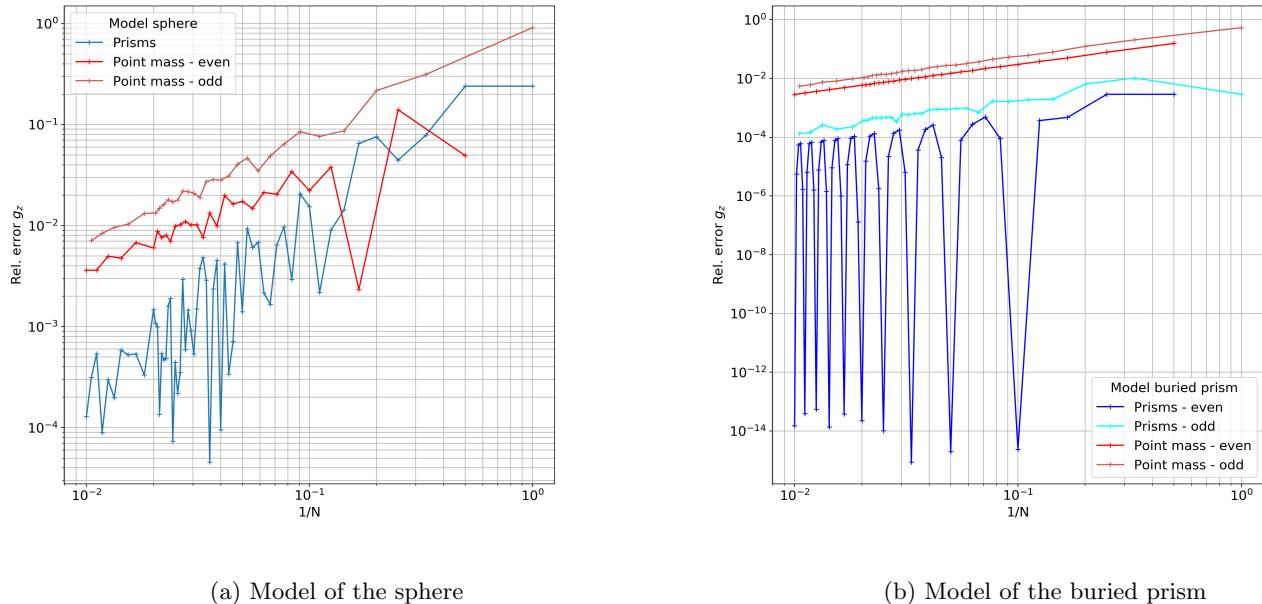


Figure 5.2: The relative error of the gravity calculations shown in the same way as in figure 5.1. The relative error recorded is of element g_z of the gravity vector. The two different methods of calculations are used in the case of a sphere in space and a buried prism. Even and odd values of N are separated so that the lines cross less and the figure is less cluttered.

The analytical solution for both models is available, so the relative error of the calculations can be computed. This error for different cell sizes is shown in figure 5.2. In some cases the errors for even and odd cases are separated. These represent different symmetries of the cells. When N is even, the centre of the box lies on the edge of a cell. When N is odd, the centre of the box lies at the centre of a cell. These two patterns of discretisation produce slightly different values.

The model of the sphere (fig. 5.2a) shows an irregular, but clearly decreasing error when the cell size is decreased. The irregular pattern is due to the approximation of a sphere as discussed in section 4.2.1. Except for two outliers when the cell size is still large, the prisms consistently have a lower relative error than the point mass method. The difference between the two methods even increases with smaller cell size. The actual difference is hard to calculate due to the irregular patterns. A rough estimation would be that the prisms are a factor of 50 more accurate compared to the point mass method.

The model of the buried prism (fig. 5.2b) shows multiple interesting characteristics. The first is that the relative error decreases in a regular pattern with a decreasing cell size, especially for the point mass method. The prisms show a more irregular pattern. If the large spikes are ignored, they will be investigated in the next paragraph, the prisms still show a far more regular pattern in this models than in the model of the sphere.

The second interesting characteristic is a regular pattern of dips in the relative error of the odd values of N of the prism method. The graph shows that this is the case when $1/N = \{0.1, 0.05, 0.033, .025, \dots\}$. These cases with high accuracy are the versions where the model suddenly lines up perfectly with the underlying geometry. This can be verified with the conditions described in section 4.2.2. The different dimensions of the model in this case are:

$$R = 1000 \text{ m}$$

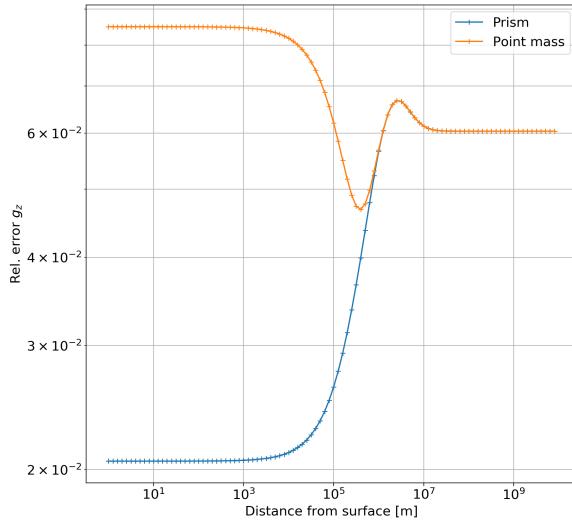
$$R_{\text{buried}} = 200 \text{ m}$$

$$\text{buried_depth} = 100 \text{ m}$$

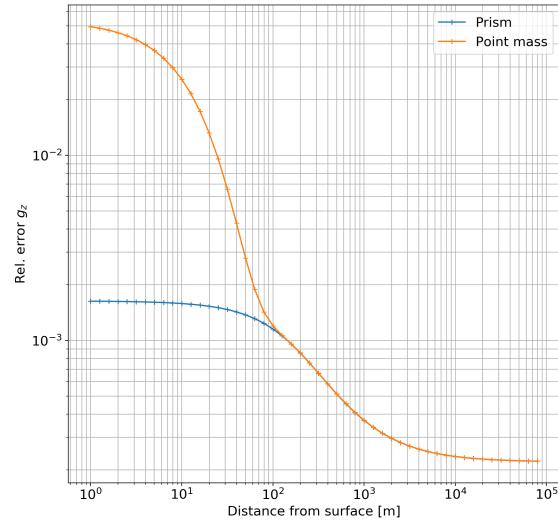
Based on these values, the first two conditions are met when N is five, ten or a multiple of ten. The last condition, however, is not met when N is five. So the result is that the model lines up perfectly when $N = 10k \{k \in \mathbb{Z}^+\}$. This fits with the values of $1/N$ that were gotten from the graph.

The relative error of the prism method for this model is consistently lower than that of the point mass method. The more smooth lines make it easier to make a comparison. The prisms are about a factor of 100 more accurate than the point masses.

Distance



(a) Model of the sphere



(b) Model of the buried prism

Figure 5.3: The relative error of the gravity calculations with increasing distance from the object. Both models show that the relative error converges at a distance of about $\frac{1}{10}$ of the size of the model. Both were created with $N = 11$.

Another factor that influences the relative error is the distance of the observation point from the object. The changing relative error with distance from the object can be seen for both models in figure 5.3. What can be observed in every case is that at a large distance the relative errors for both methods converge and can be treated as practically the same. For both the sphere and prism models this happens at a distance that is slightly less than the size of the modelled object. It is interesting to note that the relative error of the model of the sphere increases with larger distance and then stabilises, while the relative error in the buried prism model decreases and stabilises. This could be caused by the different approximations that the models make. The sphere needs an approximation that does not fully represent the surface of the sphere as seen in section 4.2.1. The buried prism has the advantage that the shape of both methods fits exactly. While the distance increases, the relatively small inaccuracies of the internal structure of the models decrease in importance, while the external shape becomes more important. This means that the approximation of the point masses is less and less relevant, while the imperfect shape of the sphere becomes more important.

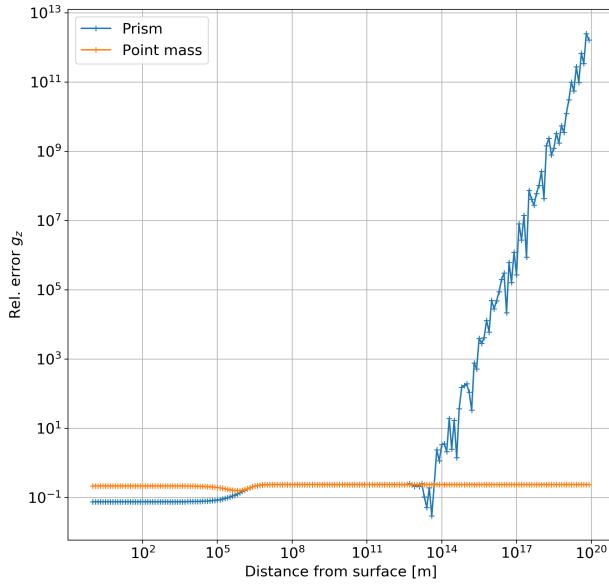
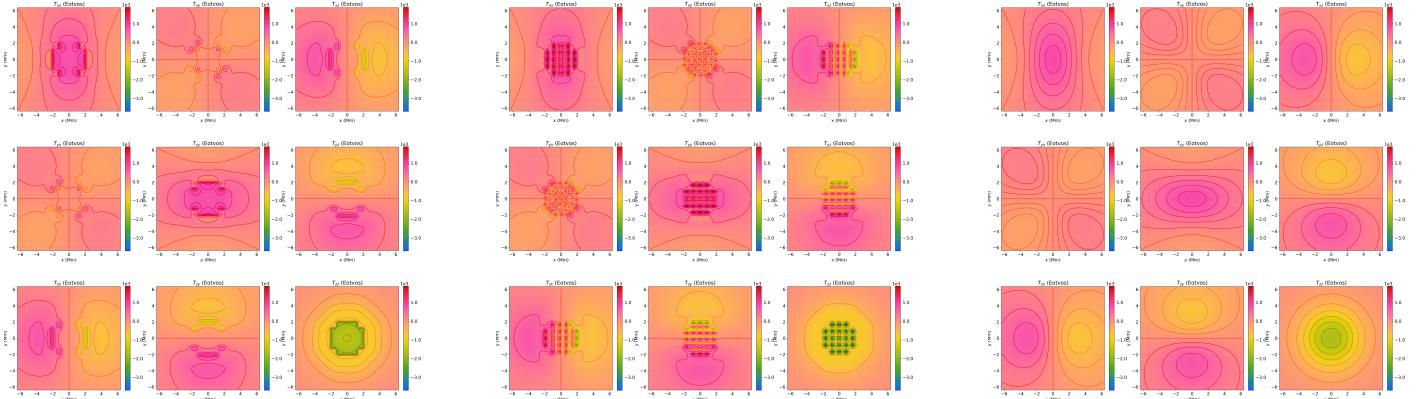


Figure 5.4: The relative error at different distances from the model of a sphere with $N = 5$. The prism method suddenly shows very large errors at a large distance from the sphere.

It was also mentioned that at some large distance from the object, the values for the prism method 'fall apart'. The values used in the complex equations are too large and computers can no longer provide the amount of detail required for accurate calculations. The effect can be observed in figure 5.4. It can only be tested using the model of a sphere, because the analytical solution for the buried prism model uses the same calculations as the prism method. The effect can be improved by increasing the amount of bits that can be used to store every number, which also means that the effect will happen sooner if the starting variables take more bits to store.

5.1.3 Gravity plots



(a) The gravity gradient response recorded with prisms.

(b) The gravity gradient response recorded with point masses.

(c) The analytical gravity gradient response.

Figure 5.5: The gravity gradient recorded using the sphere model with a homogeneous sphere of density 3000 kg/m^3 and N is 15. The observation grid is in a horizontal plane, 1000 m above the surface of the sphere.

The gravity measurements made with the different models can be visualised to study the resulting patterns. In figure 5.5 the results from the sphere model are shown together with the analytical solution. The first prominent observation

is that the approximation of the sphere shows up in both methods. The layer of cells that is closest to the observation grid can be seen clearly. The individual point masses in this closest layer show up prominently in figure 5.5b. The prisms show the edges of the closest layer. Comparing both graphs to the analytical solution shows that the general patterns are correct, especially further from the centre. The prisms show more accuracy at the centre, but not at the edges. The point mass approach shows significant error at the centre. This can also be compared by looking at the absolute error of both methods.

The absolute error resulting from figure 5.5 can be seen in figure 5.6. These graphs confirm the prior observations. The absolute error for the point masses is larger in the closest parts of the sphere, especially around the point masses themselves. The prisms only show a significant relative error around the edges of this closest slice.

Comparison with Arroyo et al., 2015

In section 4.2.2, the experiment of Arroyo et al., 2015 [2] was mentioned. Their conditions were replicated in the buried prism model and a grid of observation points was set up beneath the prism. The gravity gradient recorded in the observation grid can be seen in figure 5.7. A quick comparison with their results show that the recorded responses are very similar, except the values are opposite. This is caused by differing definitions of gravity. The x, y and z used in the formulas are defined the opposite way. Below their formula (9) it is shown that:

$$x_i = x_p - \xi_i, y_j = y_p - \eta_j, z_k = z_p - \zeta_k \quad (5.1)$$

Transformed into the definitions from section 3.2:

$$x_i = X_p - X_i, y_j = Y_p - Y_j, z_k = Z_p - Z_k \quad (5.2)$$

As can be seen when comparing this to equation 3.6, this definition is the opposite of the one that is used for the buried prism model.

5.2 Salt diapir

5.2.1 Low resolution

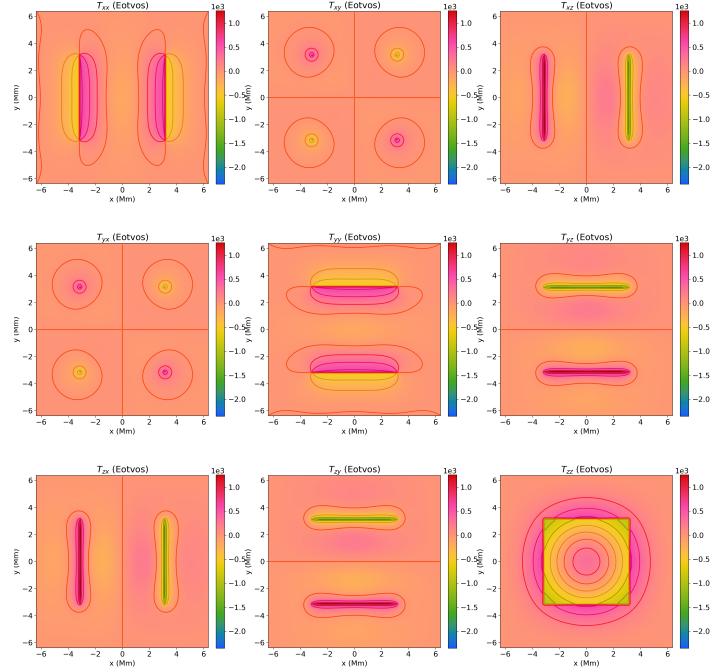
The gravitational response as observed at the surface of the model can be seen in figure 5.8. As can be observed, both plots are visually identical. The relative error was discussed in the benchmark cases. When the resolution of the model is high, there still is a large difference between the relative errors of the prism and point mass methods. The absolute values however, are more and more similar. The difference between the two methods is five orders of magnitude smaller than the actual, recorded values of g. This is measured in 10^{-10} m/s^2 . This shows that the differences are very slight.

The horizontal gradient magnitude can be applied to find the outline of the salt diapir and is shown in figure 5.9. Again the figures for both methods of calculation are very similar. Grey lines show the outline of the salt diapir at different depths. Comparison with figure 4.3 shows the same pattern of outlines; a broad base, a thin bottom and a round top. The gradient magnitude corresponds to the outline of the top part of the diapir. It is important to note that without the outline of the salt diapir, the circle with maximum values around it would seem the most obvious candidate for the outline. Another choice is the centre of the circle with the minimum values. Neither of these represent the actual outline, it is found between them, at roughly the same values that are also found far away from the diapir.

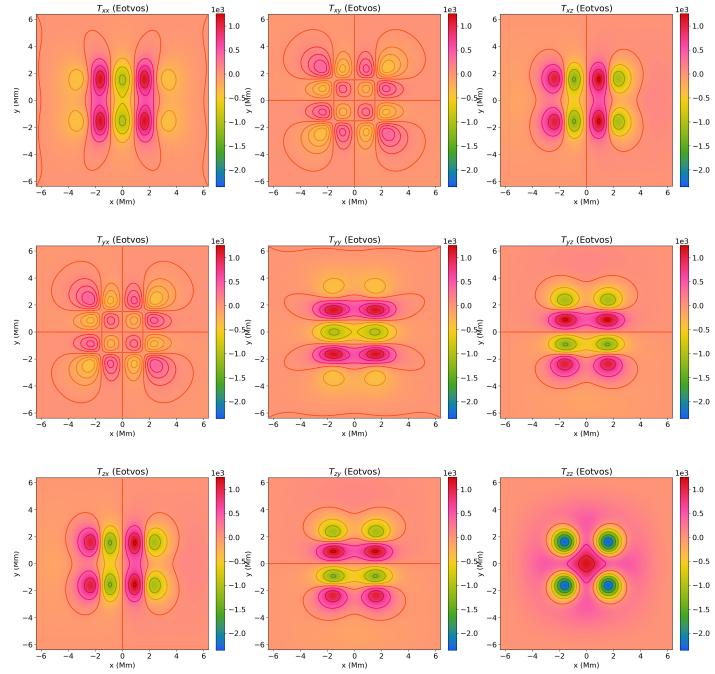
5.2.2 High resolution

The calculations on the low resolution model of the salt diapir took long. The two methods shown in figure 5.8 took 6 and 20 hours for the point mass and prism methods respectively. To repeat this with the high resolution model would take about eight times as long. It was not possible to dedicate more than a week of full time computation to generate these images. Decreasing the amount of observation points to an amount that would allow the calculations to be done, would decrease the resolution of the resulting plots too much. So much that it would not be possible to make any kind of comparison between them.

Calculations on single points are able to be computed. This showed that the same problem as described for the low resolution images would occur. The differences between the two methods of computation are very slight. The difference is of the same magnitude as reported for the low resolution model.

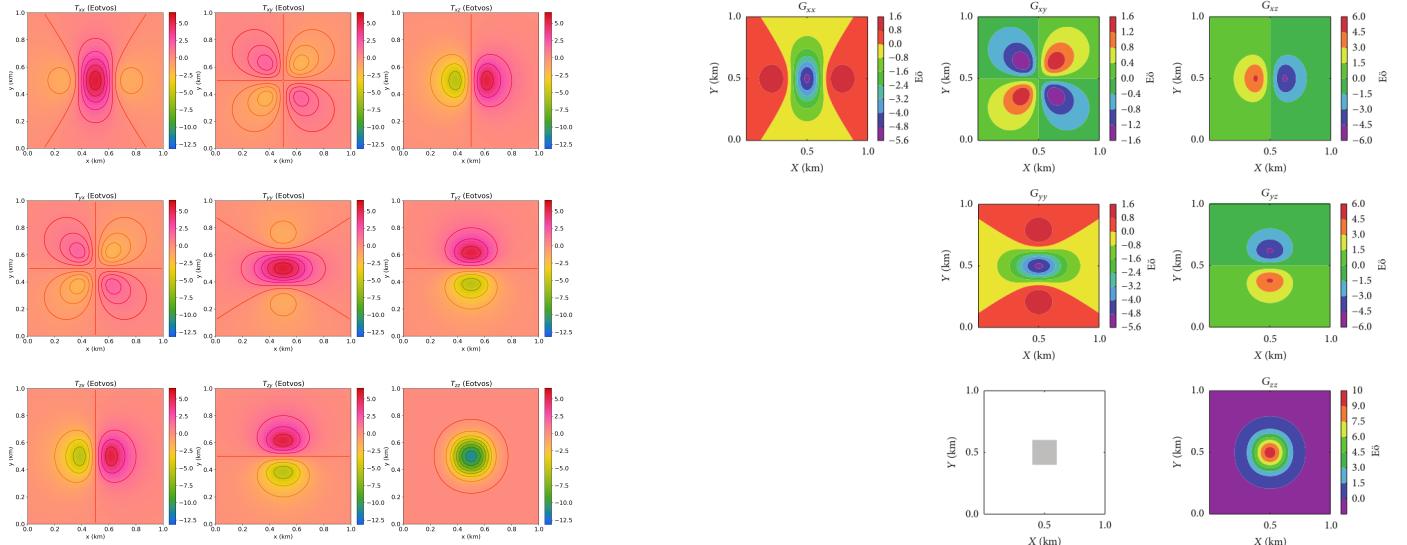


(a) The absolute error in gravity gradient response as recorded by the model with prisms.



(b) The absolute error in gravity gradient response as recorded by the model with point masses.

Figure 5.6: The absolute error of gravity gradient recorded using the model of a sphere with $N = 4$. The rest of the conditions are the same as in figure 5.5.



(a) The gravity gradient response as recorded by the new model.

(b) The response as recorded by Arroyo et al., 2015, taken from [2].

Figure 5.7: The gravity gradient recorded in a grid of 1 km x 1 km that lies 100 m below a floating cubic prism with sides of 200 m. Its density is 100 kg/m³. These conditions are taken from figure 7 in [2].

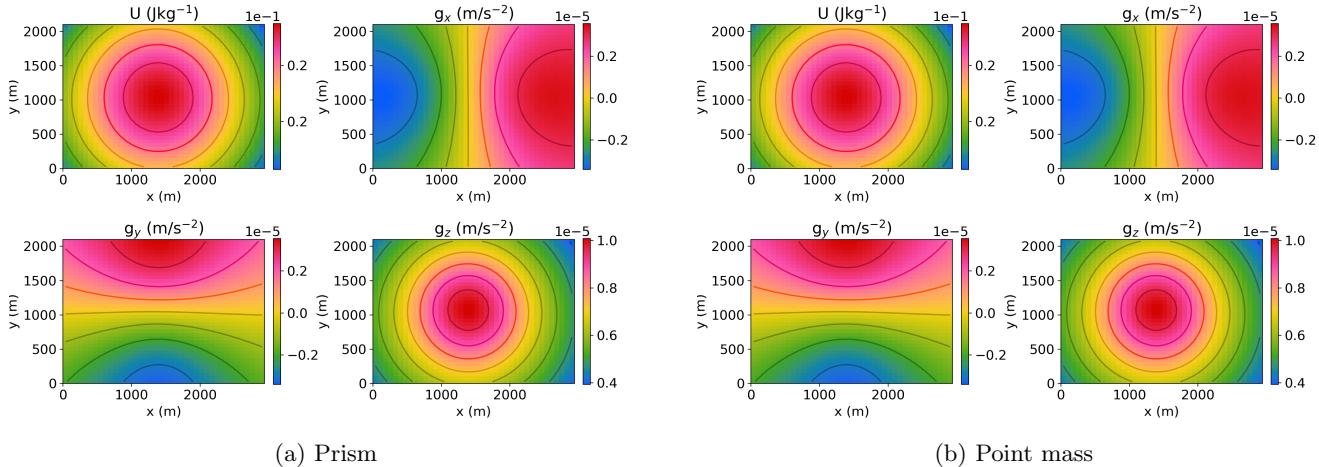


Figure 5.8: The gravitational potential and gravity vector components as recorded on the surface of the low resolution salt diapir model using the prism and point mass methods.

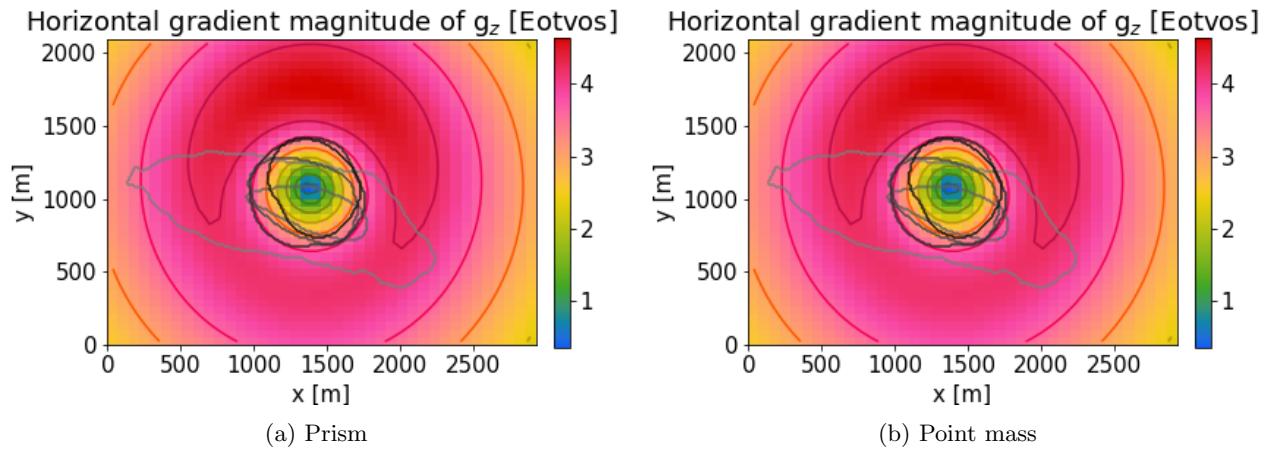


Figure 5.9: The horizontal gradient magnitude of g_z created under the same conditions as figure 5.8. The grey lines give the outline of the salt diapir at various depths (every 400m from 300m upwards). The colour gets progressively darker.

Chapter 6

Discussion

To complete the comparison between the prism and point mass methods, all of the results need to be taken into account. This shows that there is no single method that is to be preferred in every case. The timing has shown that the prisms take nine times longer to complete the calculations done on the same model as the point mass method. At the same time, the accuracy is much higher. Although the relative error can show irregular patterns, the error of the prism method is a factor of 50-90 lower than the point mass method. This shows that prisms compensate more than is sufficient for the higher computation time with higher accuracy.

At a larger distance from the modelled object, the difference in relative error starts to decrease. At a distance that depends on the size of the modelled object, the difference in relative error starts to decrease rapidly, until it is practically non-existent. In these cases, there is no bonus to the accuracy for the prism method. Then the preference goes to the point mass method, because of the decreased computation time.

This preference for the point mass method at large distances is also supported by the fact that the prism method cannot handle very large numbers. The same effect will also happen with the point mass method, but, because the equations are simpler, only at very large distances. Most uses of forward gravity modelling will not need to consider this effect. Even when modelling Earth as a sphere it only appeared at a distance of 10^{12} metres from the surface.

Two different benchmark cases were used. The buried prism had the same general shape as a rectangular cuboid, while the sphere has a different shape. In both cases the prism method was significantly more accurate. A more realistic model will have an irregular shape that does not fit with the shape of the prisms and point masses. If the prisms are more accurate in both cases, it means that it will probably generally provide more accurate results, so no analysis of the modelled area is necessary to decide on which method is used.

Both methods show a different pattern of absolute errors when observed in a surface grid. The point mass method shows small patterns of errors around every point mass. The prism method shows errors along the edges and corners of the object. The patterns show up clearly in the benchmark cases, especially when the resolution of the model is still low. When the resolution is increased and the modelled shape is more irregular, these patterns will be obscured. They were not visible in the plots that were created based on the salt diapir. The most important aspect to look out for is if the model contains layers that are cut off, these can give boundary effects. These boundary effects can be recognised by the specific patterns that result from prisms or point masses.

The plots from the salt diapir also showed that at a high resolution of the model, the difference between the two prism and point mass methods of calculation were small. If the goal of the calculations warrants this amount of accuracy, this is not a problem. It is possible that the accuracy of the gravitational response does not need to be this high. Then it is better to reduce the resolution of the model to decrease computation time or if this is too much of a problem, the point mass method could be used.

A last recommendation for creating models is to make smart decisions concerning what to model. As was seen with the salt diapir, it is much easier to leave out some parts of the calculation when it is known that they will not influence the results. This can be done by choosing the density relative to the density of one type of cell or by eliminating extra parts of the model that are not of interest.

Chapter 7

Conclusion

The main aim of this research was to find out whether the prism or point mass methods are to be preferred in gravity forward calculations. The prism method has a definite advantage in accuracy for objects close to the observation point, regardless of the shape of the object. For objects further away, this advantage decreases and the point mass method is to be preferred because of its shorter computation times. In the case of a salt diapir computation times were long, while the difference between the two methods was small. In these more realistic cases, the resolution of the model can be decreased to improve on computation times, while not impacting the error of the calculation too much.

The high resolution model of the salt diapir could not be used, because the computation time was too long. Future research could try to improve on the functions provided here to reduce the computation time. This would allow the functions to be used more easily and in more cases. An example could be using a real model and seeing how the results compare to actual measurements done in the area. Another possible improvement for the accuracy is described in Heck and Seitz, 2007 [15], where the prisms have different orientations compared to the overall coordinate system. This should improve the accuracy when modelling inclined and curved surfaces. Further theoretical work could go into expanding the comparisons with other shapes, as mentioned in the introduction.

Bibliography

- [1] Juan Carlos Afonso, Farshad Salajegheh, Wolfgang Szwilus, Jorg Ebbing, and Carmen Gaina. A global reference model of the lithosphere and upper mantle from joint inversion and analysis of multiple data sets. *Geophysical Journal International*, 217(3):1602–1628, 2019.
- [2] M. Arroyo, C. Couder-Castañeda, A. Trujillo-Alcantara, I.-E. Herrera-Diaz, and N. Vera-Chavez. A Performance Study of a Dual Xeon-Phi Cluster for the Forward Modelling of Gravitational Fields. *Scientific Programming*, 2015.
- [3] Johannes Bouman, Jörg Ebbing, Sjef Meekes, Rader Abdul Fattah, Martin Fuchs, Sofie Gradmann, Roger Haagmans, Verena Lieb, Michael Schmidt, Denise Dettmering, et al. Goce gravity gradient data for lithospheric modeling. *International Journal of Applied Earth Observation and Geoinformation*, 35:16–30, 2015.
- [4] C. Cadio and J. Korenaga. Resolving the fine-scale density structure of shallow oceanic mantle by bayesian inversion of localized geoid anomalies. *Journal of Geophysical Research: Solid Earth*, 119:3627–3645, 2014.
- [5] Nicolas Clausolles, Pauline Collon, and Guillaume Caumon. Generating variable shapes of salt geobodies from seismic images and prior geological knowledge. *Interpretation*, 7(4):T829–T841, 2019.
- [6] E. Costa and B.C. Vendeville. Experimental insights on the geometry and kinematics of fold-and-thrust belts above weak, viscous evaporitic décollement. *Journal of Structural Geology*, 24:1729–1739, 2002.
- [7] C. Couder-Castaneda, J.C. Ortiz-Aleman, M.G. Orozco del Castillo, and M. Nava-Flores. Forward modeling of gravitational fields on hybrid multi-threaded cluster. *Geofisica Internacional*, 54(1):31–48, 2015.
- [8] David Darling. *Gravity's arc : the story of gravity from Aristotle to Einstein and beyond*. Wiley, 2006.
- [9] Bureau International des Poids et Mesures. The international system of units (si). *SI Brochure*, 9th ed.:159, 2019.
- [10] Mark R. Drinkwater, R. Haagmans, D. Muzi, A. Popescu, R. Floberghagen, M. Kern, and M. Fehringer. The GOCE Gravity Mission: ESA’s First Core Earth Explorer. *Proceedings of the third International GOCE User Workshop*, SP-267:1–8, 2006.
- [11] A.M. Dziewonski and D.L. Anderson. Preliminary reference Earth model. *Phys. Earth. Planet. Inter.*, 25:297–356, 1981.
- [12] J. Ebbing, C. Braitenberg, and H.-J. Götze. Forward and inverse modelling of gravity revealing insight into crustal structures of the Eastern Alps. *Tectonophysics*, 337:191–208, 2001.
- [13] Haakon Fossen. *Salt tectonics*, pages 418–438. Cambridge University Press, Cambridge, 2016.
- [14] Rachel Harding and Mads Huuse. Salt on the move: Multi stage evolution of salt diapirs in the Netherlands North Sea. *Marine and Petroleum Geology*, 61:39–55, 2015.
- [15] B. Heck and K. Seitz. A comparison of the tesseroid, prism and point-mass approaches for mass reductions in gravity field modelling. *J. Geodesy*, 81:121–136, 2007.
- [16] M. Herceg, C. C. Tscherning, and J.F. Levinsen. Sensitivity of GOCE gradients on Greenland mass variation and changes in ice topography. *Journal of Geodetic Science*, 4:8–18, 2014.
- [17] Michael R. Hudé, Ian O. Norton, Martin P.A. Jackson, and Frank J. Peel. Jurassic evolution of the Gulf of Mexico salt basin. *AAPG Bulletin*, 97(10):1683–1710, 2013.
- [18] M.R. Hudé and M.P.A. Jackson. Terra infirma: Understanding salt tectonics. *Earth-Science Reviews*, 82:1–28, 2007.

- [19] Udo Hunsche and Andreas Hampel. Rock salt - the mechanical properties of the host rock material for a radioactive waste repository. *Engineering Geology*, 52(3-4):271–291, 1999.
- [20] Salman Jahani, Jean-Paul Callot, Jean Letouzey, and Dominique Frizon de Lamotte. The eastern termination of the Zagros fold-and-thrust belt, Iran: Structures, evolution and relationships between salt plugs, folding and faulting. *Tectonics*, 28:1–22, 2009.
- [21] B.L.N. Kennett, E.R. Engdahl, and R. Buland. Travel times for global earthquake location and phase association. *Geophys. J. Int.*, 122:108–124, 1995.
- [22] P. Knudsen, R. Bingham, O. Andersen, and Marie-Helene Rio. A global mean dynamic topography and ocean circulation estimation using a preliminary GOCE gravity model. *Journal of Geodesy*, 85:861–879, 2011.
- [23] M. Kuhn and W. Featherstone. Construction of a synthetic Earth gravity model by forward gravity modelling. *A Window on the Future of Geodesy*, 128:350–355, 2005.
- [24] K Mader. Das newtonsche raumpotential prismatischer körper und seine ableitungen bis zur dritten ordnung, osterr. *Z. Vermess. Sonderheft*, 11, 1951.
- [25] Nils-Axel Mörner. Eustasy and geoid changes. *The Journal of Geology*, 84(2):123–151, 1976.
- [26] D. Nagy, G. Papp, and J. Benedek. The gravitational potential and its derivatives for the prism. *Journal of Geodesy*, 74(7–8):552–560, 2000.
- [27] D. Nagy, G. Papp, and J. Benedek. Corrections to "The gravitational potential and its derivatives for the prism". *Journal of Geodesy*, 76(8):475, 2002.
- [28] Roland Pail, Sean Bruinsma, Federica Migliaccio, Christoph Förste, Helmut Goiginger, Wolf-Dieter Schuh, Eduard Höck, Mirko Reguzzoni, Jan Martin Brockmann, Oleg Abrikosov, Martin Veicherts, Thomas Fecher, Reinhard Mayrhofer, Ina Krasbutter, Fernando Sansò, and Carl Christian Tscherning. First GOCE gravity field models derived by three different approaches. *Journal of Geodesy*, 85:819–843, 2011.
- [29] Nikolaos K Pavlis, Simon A Holmes, Steve C Kenyon, and John K Factor. The development and evaluation of the earth gravitational model 2008 (egm2008). *Journal of geophysical research: solid earth*, 117(B4), 2012.
- [30] Donald Plouff. Gravity and magnetic fields of polygonal prisms and application to magnetic terrain corrections. *Geophysics*, 41(4):727–741, 1976.
- [31] J. Ranama-Rad, G. Farhoudi, H. Ghorbani, Sh. Habibi Mood, and R. Derakhshani. Pierced salt domes in the Persian Gulf and in the Zagros mountain ranges in southern Iran and their relationship to hydrocarbon and basement tectonics. *Iranian Journal of Earth Sciences*, 1:57–72, 2009.
- [32] A.H. Saad. Understanding gravity gradients - a tutorial. *The leading edge*, 25(8):942, 2006.
- [33] Chris J. Spiers, J.L. Urai, G.S. Lister, J.N. Boland, and H.J. Zwart. The influence of fluid-rock interaction on the rheology of salt rock. *Nuclear science and technology*, EUR 10399 EN, 1986.
- [34] Byron D. Tapley, Srinivas Bettadpur, John C. Ries, Paul F. Thompson, and Michael M. Watkins. GRACE Measurements of Mass Variability in the Earth System. *Science*, 305(5683):503–505, 2004.
- [35] Byron D. Tapley, Srinivas Bettadpur, Michael M. Watkins, and C. Reigber. The gravity recovery and climate experiment: Mission overview and early results. *Geophys. Res. Lett.*, 31(9), 2004.
- [36] Dimitrios Tsoulis. Analytical computation of the full gravity tensor of a homogeneous arbitrarily shaped polyhedral source using line integrals. *Geophysics*, 77(2):F1–F11, 2012.
- [37] Alexander Unzicker. Why do we still believe in newton's law? facts, myths and methods in gravitational physics. *arXiv preprint arXiv:gr-qc/0702009v8*, pages 1–38, 2008.
- [38] Bruno C Vendeville and Martin PA Jackson. The rise of diapirs during thin-skinned extension. *Marine and Petroleum Geology*, 9(4):331–354, 1992.
- [39] Andrew Woodfield. *Teleology*. Cambridge University Press, 2010.
- [40] Yanqiu Zhang, Michael Krause, and Maria Mutti. The formation and structure evolution of Zechstein (Upper Permian) salt in Northeast German Basin: a review. *Open Journal of Geology*, 3:411–426, 2013.