



Technische Universiteit Delft
Faculteit Elektrotechniek, Wiskunde en Informatica
Delft Institute of Applied Mathematics

**Designing a search-algorithm that uses an
autonomously moving swarm of drones to efficiently
detect drifting steel shipping containers, using
Magnetic Anomaly Detection and Simulated
Annealing.**

Verslag ten behoeve van het
Delft Institute of Applied Mathematics
als onderdeel ter verkrijging

van de graad van

**BACHELOR OF SCIENCE
in
TECHNISCHE WISKUNDE**

door

T.P.N.M. VERTREGT

**Delft, Nederland
Juli 2020**



BSc verslag TECHNISCHE WISKUNDE

“Designing a search-algorithm that uses an autonomously moving swarm of drones to detect drifting steel shipping containers, using Magnetic Anomaly Detection and Simulated Annealing.”

T.P.N.M. VERTREGT

Technische Universiteit Delft

Begeleider

Ir. A.R.P.J. Vijn

Dr. ir. E.S.A.M. Lepelaars

Overige commissieleden

Dr. ir. M.B. van Gijzen

Prof. dr. ir. A.W. Heemink

June, 2020

Delft

Acknowledgements

I would like to thank Aad Vijn and Eugène Lepelaars. They did not only supervise me, but guided me towards something which I can be very proud of. I really like the way the Bachelor's thesis turned out, and this is for a large part because of their efforts in guiding me. Especially during these different times Eugène and Aad were always willing to answer my questions. They made me see all the possibilities of mathematics and that really fun things can be done with it.

Contents

1	Introduction	9
1.1	Goal	9
1.2	Strategy	9
1.3	Structure	9
2	Magnetostatics	11
2.1	Gradient	12
2.2	Divergence	12
2.3	Curl	13
2.4	Magnetisation	14
2.5	Magnetic Anomaly Detection (MAD)	14
3	Forward model	17
3.1	Forward problem	17
3.2	Magnetic Dipole model	17
3.2.1	The three-dimensional rod	19
3.2.2	Shrinking the rod to a line	20
3.2.3	Back to the potential	21
3.2.4	Calculating the H-field	23
3.3	Symmetry	24
4	Inverse model	27
4.1	Inverse Problem	27
4.2	Problem formulation	27
4.3	Simulated Annealing	29
5	Numerical results	33
5.1	Introduction	33
5.2	Implementation	36
5.3	Sensitivity analysis of alpha	37
5.4	Statistical analysis based on Monte Carlo simulations	41
6	Swarm Implementation	43
6.1	Introduction	43
6.2	Method 1: best guess method	46
6.3	Method 2: average guess method	50
6.4	Method 3: total signal method	53

6.4.1	Method 3.1: without best guess	53
6.4.2	Method 3.2: with best guess	55
6.5	Twin Experiment	57
6.6	Comparison of the localisation methods	58
7	Assessment	59
7.1	Conclusion	59
7.2	Discussion	59
7.3	Recommendation	60
A	Derivation of equation (3.61) to (3.62)	63

1. Introduction

In the night of the 1st of January 2019, the MSC Zoe set sail across the North Sea, when a heavy storm caused 342 containers to go overboard.[1] The shipping containers set adrift in the North Sea, and 19 of them resurfaced somewhere in the Wadden Sea, a nature reserve. The Dutch coastguard had great difficulty in recovering these shipping containers from the Wadden Sea, since it was a protected nature reserve. The environmental damage done by these containers is yet to be determined, but it may well be much more than the lost goods inside of them. This is one of the many examples in which shipping containers get lost and do more damage than just the cost of the lost goods they carry. If they were to be located more timely, their content, and more importantly our environment and society could be spared much damage and money.

1.1 Goal

The ultimate goal of this defence would be to construct a search algorithm designed to make a swarm of drones move autonomously towards a ferromagnetic target in littoral waters. The drones would be locating the target using Magnetic Anomaly Detection (MAD), which is a method already widely used. In the optimal scenario, information would be exchanged between the drones to optimally benefit from the fact that so much sensory data is available.

1.2 Strategy

The strategy of this defence is to use a combination of forward and inverse modelling to try and retrieve a useful search algorithm, based on the paradigm of Simulated Annealing. In short, at first a magnetic dipole model will be presented, which serves as the forward model. This is done since shipping containers resemble magnetic dipoles from a distance, something which will be discussed more elaborately in the next chapter. The inverse model assumes that the initial dipole model and its parameters are unknown, as in the real world. It attempts to find a dipole model which best fits the magnetic induction field signal, as detected by the surveillance drones. With this calculated dipole model, the position of the dipole can be easily deduced with iterative increasing accuracy. This so the drones converge to the position of the dipole (i.e. the shipping container).

1.3 Structure

First this thesis will cover the basics of magnetostatics necessary to understand some of the techniques and models used. Next, forward modelling is discussed, which serves as an introduction to the derivation of the forward model. After the forward model, the inverse problem is discussed, after which the method for finding a solution will be explained. The method is

Simulated Annealing, and it is the topic of the fourth chapter. This algorithm is then used to carry out the main focus of this thesis, the swarm of autonomous drones. Multiple methods for information sharing among those drones are presented, and compared to each other. The comparison of these methods is the main body of the final conclusion. The surveillance described in this thesis –with a swarm of drones flying above sea level– is a quite untouched subject, which means there are many possibilities for more elaborate research. Some recommendations for further research are mentioned last.

2. Magnetostatics

This section discusses some of the fundamental physics concerning magnetostatics. This knowledge will be used later on to deduce the magnetic dipole model in Section 2.1.

Magnetostatics is the field of physics which studies steady-state.¹ magnetic fields. These fields are present because of magnets. Three kinds of magnets exist: permanent magnets, temporary magnets and electromagnets. These three types have one thing in common: all of them can be modelled as magnetic dipoles by approximation. A magnetic dipole, as the name suggests, has two poles: a north and a south pole. The Earth can be modelled as a dipole as well, due to the large mass of liquid iron flowing in its core.

A very well-known person to study magnetostatics has been Michael Faraday in 1831, when he discovered what is now known as Faraday's law of induction (see equation (2.3)). A number of fundamental contributions to magnetostatics have been made by Maxwell and Gauss as well. The best way to look at magnetostatics is as a special case of electromagnetism. The four fundamental equations of electromagnetism, are called *Maxwell's equations*, and are stated in differential form below:

$$\nabla \cdot \mathbf{D} = \rho_v \quad (\text{Gauss's law}) \quad (2.1)$$

$$\nabla \cdot \mathbf{B} = 0 \quad (\text{Gauss's law for magnetism}) \quad (2.2)$$

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \quad (\text{Faraday's law}) \quad (2.3)$$

$$\nabla \times \mathbf{H} = \frac{\partial \mathbf{D}}{\partial t} + \mathbf{J}. \quad (\text{Ampère's law}) \quad (2.4)$$

The units of the quantities used in above equations are:

Variable	Unit		Name
$\mathbf{D}$	[C · m ⁻²]	Coulomb per square meter	Electric displacement field
ρ_v	[C · m ⁻³]	Coulomb per cubic meter	Charge density constant
$\mathbf{B}$	[T]	Tesla	Magnetic induction field
$\mathbf{E}$	[N · C ⁻¹]	Newton per Coulomb	Electric field
$\mathbf{H}$	[A · m ⁻¹]	Ampères per meter	Magnetic field strength
$\mathbf{J}$	[A · m ⁻²]	Ampères per square meter	Current density

Table 2.1: Declaration of variables and their units.

¹ Steady-state implies that all situations which are being considered are independent of time (i.e. all time-derivatives are equal to zero).

Only two of these equations are of use when studying the field of magnetostatics, namely equations (2.2) and (2.4) (and the constitutive relation (2.11)). Since only steady-state scenarios are considered in magnetostatics, the time derivatives are equal to zero.¹ The two fundamental laws governing magnetostatics follow from these premises:

$$\nabla \cdot \mathbf{B} = 0 \quad (\text{Gauss's law for magnetism}) \quad (2.5)$$

$$\nabla \times \mathbf{H} = \mathbf{0}. \quad (\text{Ampère's law}) \quad (2.6)$$

$\nabla \cdot \mathbf{B}$ implies the divergence of the $\mathbf{B}$ -field, and $\nabla \times \mathbf{H}$ denotes the curl of the $\mathbf{H}$ -field. Apart from curl and divergence there are more concepts important to discuss, which will be done below, starting with the gradient.²

2.1 Gradient

The gradient of a function f could be interpreted as the direction and rate of the fastest increase in value of the function. Suppose $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is a function, which assigns a value to each vector in a vector field (a vector field assigns a vector to each point in a given subset of space). The gradient $\nabla f : \mathbb{R}^n \rightarrow \mathbb{R}^n$, assigns a vector to each vector in the vector field. Suppose f has partial derivatives for all n variables, then the gradient in a of f is defined as :

$$\nabla f(a) = \begin{bmatrix} \frac{\partial f}{\partial x_1}(a) \\ \vdots \\ \frac{\partial f}{\partial x_n}(a) \end{bmatrix} \quad (2.7)$$

An important remark to make is that the gradient is often mistaken for the divergence of a vector field, which will be explained in the next section. The difference is obvious however, since the gradient is a property of a scalar valued function and the divergence of a vector valued function.

2.2 Divergence

The divergence of a magnetic induction field $\mathbf{B}$ is denoted by $\nabla \cdot \mathbf{B}$. An alternative notation for the divergence of a vector field is

$$\text{div } \mathbf{F}(x, y) = \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{F}}{\partial y} + \frac{\partial \mathbf{F}}{\partial z}. \quad (2.8)$$

This alternative notation more clearly shows that the divergence function takes in a point (x, y) of the vector field, and returns a scalar value concerned with this point.

²A great video about the concepts of curl and divergence can be found at <https://www.youtube.com/watch?v=rB83DpBJQsE>.

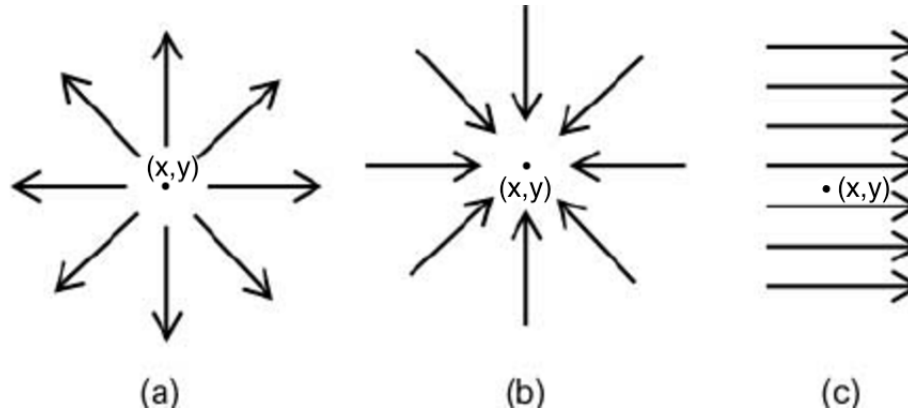


Figure 2.1: The different divergence scenarios for a vector field. [2]

In the figure above multiple scenarios (i.e. values for $\text{div } \mathbf{F}$) are shown. If $\text{div } \mathbf{F}(x, y) > 0$, then figure (a) applies. This means there is more 'flow' coming out of the point (x, y) , than that there is going in. This is why, if scenario (a) is the case, (x, y) is called a *source node* (much like a real-world (water)source). On the contrary, if $\text{div } \mathbf{F}(x, y) < 0$, there is going more flow into (x, y) , than there is flowing out of. In this case (x, y) is called a *sink node*, and scenario (b) is the case. In the case that $\text{div } \mathbf{F} = 0$, the flow in is equal to the flow out of the point (x, y) , depicted in scenario (c) in Figure 2.1.

2.3 Curl

The curl of the magnetic field strength is indicated by $\nabla \times \mathbf{H}$. The curl is a function which takes in a point z_0 of a vector field as well, but in contrast to the divergence, it returns a vector, directed according to the so-called *right-hand rule*.

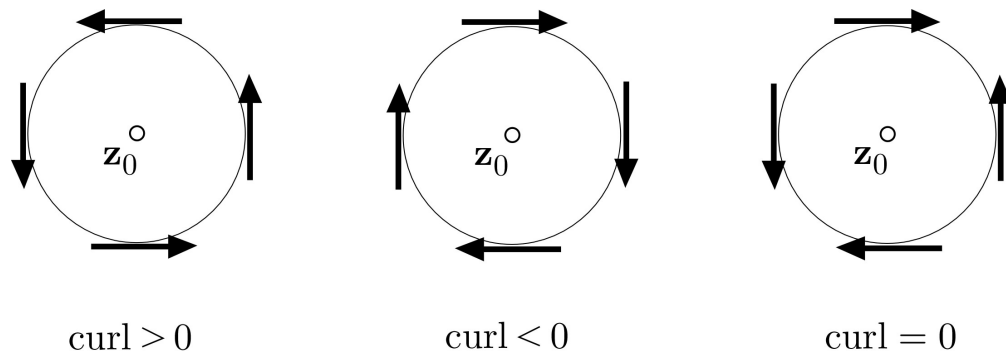


Figure 2.2: The different curl scenarios in a vector field. [3]

If the projection of the curl onto the z -axis is greater than zero, the vectors around the point z_0 rotate counterclockwise, and the direction of the curl points towards the reader. In the figure, the term 'curl' refers to the z component of the curl vector. On the contrary, if the z component of the curl is less than zero, the vectors around the point z_0 rotate clockwise and the curl points away from the reader. The case where the z component of the curl is equal to zero is depicted on the rightmost illustration in Figure 2.2, in this case the vectors do not point in a single (counter)clockwise direction. The right-hand rule states that if you wrap the fingers of your hand around the point in the direction of the flow, the curl will point in the direction of

your thumb. Let us now relate these properties to the two fundamental laws of magnetostatics which were stated in (2.2) and (2.4):

$$\nabla \cdot \mathbf{B} = 0 \quad (\text{Gauss's law for magnetism}) \quad (2.9)$$

$$\nabla \times \mathbf{H} = \mathbf{0} \quad (\text{Ampère's law}). \quad (2.10)$$

The physical corollary of equation (2.9) is that the magnetic induction field does not have any divergence. This statement implies that the total magnetisation of an object is always equal to zero (the integral of the magnetisation over a ferromagnetic object). Intuitively this means that no such thing as magnetic monopoles exist. It is not possible that magnetic induction fields spring from somewhere (in a source node, where the divergence would be > 0), or that it would disappear somewhere (in a sink node, where the divergence would be < 0). A magnetic induction field is rather a continuous flowing quantity.

Equation (2.10) implies that the curl of the magnetic field strength is equal to the null-vector. This is a direct consequence of the fact that only steady-state scenarios are considered. On top of that it is assumed that all currents are equal to zero as well ($\mathbf{J} = 0$), which causes the curl of $\mathbf{H}$ to be equal to zero. The curl of a magnetic field strength springs from an electrical current running in a loop.

2.4 Magnetisation

Another important aspect to discuss is the magnetisation field $\mathbf{M}$ [Am^{-1}]. This is the vector field which explains the relation between $\mathbf{B}$ and $\mathbf{H}$. The magnetisation $\mathbf{M}$ describes how strong a region of material is magnetised. Formally it is the distribution of magnetic moment $\mathbf{m}$ [Am^2] (a vector representing the orientation and magnetic strength of the ferromagnetic object) over the volume of a magnet. $\mathbf{M}$ only exists inside of a ferromagnetic object or magnet, everywhere outside of a magnet it is equal to zero. The magnetisation $\mathbf{M}$ has an important role in defining the relation between $\mathbf{H}$ and $\mathbf{B}$. The relation is called the constitutive relation:

$$\mathbf{H} = \frac{\mathbf{B}}{\mu_0} - \mathbf{M}, \quad (2.11)$$

where μ_0 equals the magnetic permeability through empty space, which is a scalar constant (and equal to $4\pi \times 10^{-7} \text{ N A}^{-2}$). The magnetisation $\mathbf{M}$ will be playing a role later on.

2.5 Magnetic Anomaly Detection (MAD)

Now that the basics of magnetostatics are covered, it is time to expand on the detection of shipping containers, in other words the collection of data from which the position and magnetic moment of the ferromagnetic object will be derived. To understand the idea of magnetic anomaly detection, a better understanding of a magnetic dipole is required. As mentioned earlier, a magnetic dipole has two poles: a north and a south pole. Consider a classical permanent magnet:

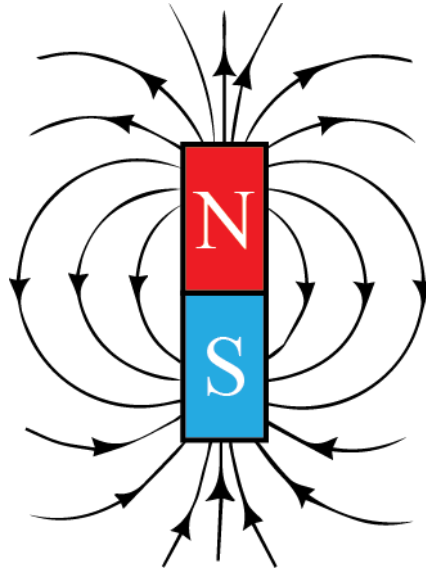


Figure 2.3: Schematic representation of a classic permanent magnet. [4]

Now, it may look like the magnetic field lines are disappearing at the blue south pole, and appearing at the red north pole. However, this cannot be true, since otherwise the divergence of this field would be not equal to zero. The magnetic field lines (the $\mathbf{H}$ -field) themselves are indeed discontinuous, however the magnetic induction field (the $\mathbf{B}$ -field) is not, thus preserving Gauss's law. The best way to explain this is by looking at equation (2.11), the magnetisation $\mathbf{M}$ is discontinuous by definition, since it is only defined inside of the ferromagnetic object. This implies that either the $\mathbf{B}$ -field or the $\mathbf{H}$ -field is discontinuous. By Gauss's law $\mathbf{B}$ is definitely not discontinuous, so $\mathbf{H}$ is.

Every ferromagnetic object can be modelled as a magnetic dipole from a certain distance. As mentioned before, the Earth has a magnetic field as well, and from space this field can be approximated by a magnetic dipole field. The consequence of this is that on the Earth's surface an almost uniform magnetic field is measurable.

From up close a large shipping container is a magnetic multipole, but from a distance large enough it is possible to approximate it with a magnetic dipole, just like the Earth. The same holds for shipping containers. These ferromagnetic objects exhibit a small magnetic field themselves. This so-called *magnetic distortion field* (or magnetic signature) can be detected against the Earth's (uniform) magnetic background field. The magnetic signature of a shipping container is an *anomaly* relative to the Earth's magnetic field, and this anomaly can be detected. This is the fundamental concept of magnetic anomaly detection.

3. Forward model

3.1 Forward problem

In a forward model one takes a given state of a system, and estimates with this state how a system behaves. In essence it is about using the known mathematical and physical laws of nature to describe a certain situation. The given state of the system in this thesis is the magnetic moment and position of a dipole, and the forward problem is to mathematically describe the $\mathbf{B}$ -field around the dipole.

3.2 Magnetic Dipole model

As mentioned earlier, the basic principle of this model is the fact that the magnetic field of any ferromagnetic object can be modelled from a distance by a magnetic dipole. This section will digress on the deduction of the forward model.

Consider a magnetic dipole Ω (i.e. the shipping container). The following variables are of interest: the position $\mathbf{d}$ [m] of Ω , the magnetic moment $\mathbf{m}$ [Am^2] of Ω and the position where the magnetic field is detected, $\mathbf{r}$ [m]. The magnetic moment of Ω is the orientation and magnetic strength of the magnetic object Ω . Furthermore, the vector $\hat{\mathbf{r}}$ [m] is of interest, this is the difference vector of $\mathbf{r}$ and $\mathbf{d}$. The spatial clarification is shown in Figure 3.1. All definitions, variables and their units are stated in Table 3.1. The magnetic induction field $\mathbf{B}$ will be deduced from the magnetic field strength $\mathbf{H}$, so it is mentioned as well in the table.

INPUT VARIABLES		
$\mathbf{d}$	[m]	position of Ω
$\mathbf{m}$	[Am^2]	magnetic moment of Ω
$\mathbf{r}$	[m]	position of detection
$\hat{\mathbf{r}}$	[m]	difference vector of $\mathbf{r}$ and $\mathbf{d}$
OUTPUT VALUES		
$\mathbf{B}(\mathbf{r})$	[T]	Detected magnetic field of Ω in $\mathbf{r}$
$\mathbf{H}(\mathbf{r})$	[Am^{-1}]	Detected magnetic field strength of Ω in $\mathbf{r}$

Table 3.1: Variables defining the dipole model.

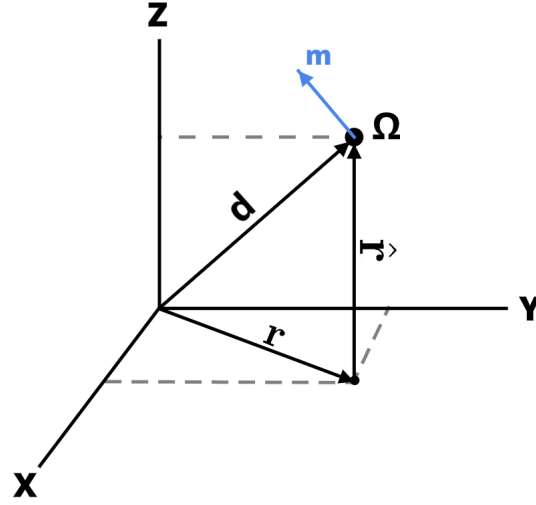


Figure 3.1: Spatial clarification of the vectors.

When a sensor detects the magnetic field at location $\mathbf{r}$, it obtains the magnetic induction field $\mathbf{B}$. The magnetic field strength follows from the fact that $\mathbf{M} = \mathbf{0}$ at each sensor location (since the sensor locations are outside of the ferromagnetic object):

$$\mathbf{B} = \mu_0 \mathbf{H} \quad (3.1)$$

This definition has been introduced to ease calculating the magnetic field, and sometimes may seem rather artificial. Nevertheless, it will prove very important, since $\mathbf{H}$ has a very useful property which will be used to calculate the model for a dipole.

To deduce the model, the magnetic scalar potential is defined using the magnetisation of a rod. Then the magnetic moment of this steel rod is calculated as a starting point. Next, the rod is shrunk in a certain way to a point. The potential is then used again to obtain the magnetic field strength $\mathbf{H}$ in a point dipole, from which the magnetic induction field $\mathbf{B}$ directly follows. Eventually, the model will be:

$$\mathbf{B}(\hat{\mathbf{r}}) = \frac{\mu_0}{4\pi} \left[\frac{(\mathbf{m} \cdot \hat{\mathbf{r}}) \cdot \hat{\mathbf{r}}}{\|\hat{\mathbf{r}}\|^5} - \frac{\mathbf{m}}{\|\hat{\mathbf{r}}\|^3} \right].$$

The model takes only the sensor location of the dipole in (since the magnetic moment of the object is known), and returns the magnetic induction field $\mathbf{B}$.

Let us start by deducing an expression for the magnetic scalar potential ψ of $\mathbf{B}$, this potential exists since $\nabla \times \mathbf{H} = 0$. $\nabla \times \mathbf{H}$ implies:

$$\nabla \times (\nabla f) = 0 \quad (3.2)$$

f being an arbitrary scalar-valued function, which proves the existence of the potential. The magnetic potential is a useful quantity to determine a magnetic field, in particular for permanent magnets such as a magnetic point dipole. The magnetic potential is defined by the following equation:

$$\mathbf{H} = -\nabla \psi. \quad (3.3)$$

In this case $\nabla \psi$ denotes the gradient of the potential. The equality defined in (2.11), is equal to:

$$\mathbf{B} = \mu_0(\mathbf{H} + \mathbf{M}). \quad (3.4)$$

Plug this expression for the $\mathbf{B}$ -field into equations (2.9) and (2.10) to obtain:

$$\nabla \cdot \mu_0(\mathbf{H} + \mathbf{M}) = 0 \quad (3.5)$$

$$\mu_0(\nabla \cdot \mathbf{H} + \nabla \cdot \mathbf{M}) = 0 \quad (3.6)$$

$$\nabla \cdot \mathbf{H} = -\nabla \cdot \mathbf{M}. \quad (3.7)$$

The equivalence from (3.5) to (3.6) follows because of the linearity of the nabla-operator ∇ . Substituting (3.3) in (3.7) will result in the following poisson equation:

$$\Delta\psi = \nabla \cdot \mathbf{M}. \quad (3.8)$$

with Δ being $(\nabla \cdot \nabla)$. From this one may conclude that $\nabla \cdot \mathbf{M}$ is the source of the magnetic field $\mathbf{B}$. (3.8) is of the form:

$$\Delta u = f. \quad (3.9)$$

Equation (3.9) is a non-homogenous poisson equation. Since the operator Δ is a linear operator, (3.9) has a solution which is the convolution of Green's function and $\nabla \cdot \mathbf{M}$. Green's function is the solution of:

$$\Delta G = \delta_0 \quad (3.10)$$

where G is Green's function and δ_0 the Dirac-delta function defined in 0. To put this method into practice now suppose Ψ exists such that $\Delta\Psi = \delta_0$. The potential ψ is given by the convolution of Green's function for the Laplace-operator in $\mathbb{R}^3$ and $(\nabla \cdot \mathbf{M})$. In general form, applied to equation (3.9):

$$\psi(\mathbf{r}) = \iiint_{\mathbb{R}^3} \Psi(\mathbf{r} - \mathbf{r}') f(\mathbf{r}') d\mathbf{r}' \quad , \text{ with } \Psi(\mathbf{x}) = \frac{1}{4\pi} \frac{1}{\|\mathbf{x}\|} \quad (3.11)$$

$\Psi(\mathbf{x})$ being Green's function for the Laplace operator in $\mathbb{R}^3$. Applying (3.11) to problem (3.8), the following result is obtained:

$$\psi(\mathbf{r}) = -\frac{1}{4\pi} \iiint_{\mathbb{R}^3} \frac{1}{\|\mathbf{r} - \mathbf{r}'\|} \nabla' \cdot \mathbf{M}(\mathbf{r}') d\mathbf{r}'. \quad (3.12)$$

This expression for the potential is not yet to use right now, but it will prove important later on. Together with equation (3.3) the $\mathbf{H}$ -field can be derived from the potential ψ . With the $\mathbf{H}$ -field the $\mathbf{B}$ -field is found by multiplying with μ_0 . The next logical step is to calculate the magnetic moment $\mathbf{m}$ of a magnetic point dipole. This shall be done by shrinking a three-dimensional rod to a point.

3.2.1 The three-dimensional rod

To calculate the magnetic moment of a magnetic dipole, first consider a cylinder Ω , defined using cylindrical coordinates: (x, ρ, ϕ) . The relation between Cartesian coordinates (x, y, z) and cylindrical coordinates is described below:

$$x = x \quad (3.13)$$

$$y = \rho \sin \phi \quad (3.14)$$

$$z = \rho \cos \phi. \quad (3.15)$$

Suppose the rod has length 2ℓ [m], radius a [m], and a total magnetisation in the x -direction of magnitude C [Am^{-1}], a constant which will be defined later. As mentioned before, magnetisation

only exists inside a magnetised material, so it makes sense to define the magnetisation of the rod using a Heaviside-function:

$$\mathbf{M} = C \mathbf{u}_x [U(x + \ell) - U(x - \ell)] U(a - \rho). \quad (3.16)$$

The magnetic moment of the rod is obtained by integrating the magnetisation of the rod over its volume:

$$\mathbf{m} = \iiint_{\mathbb{R}^3} \mathbf{M}(\mathbf{r}) dV \quad (3.17)$$

$$= \int_{x=-\infty}^{\infty} \int_{\rho=0}^{\infty} \int_{\phi=0}^{2\pi} C \mathbf{u}_x [U(x + \ell) - U(x - \ell)] U(a - \rho) dx \cdot d\rho \cdot d\phi \quad (3.18)$$

$$= C \int_{x=-\infty}^{\infty} \mathbf{u}_x [U(x + \ell) - U(x - \ell)] dx \int_{\rho=0}^{\infty} U(a - \rho) \rho d\rho \int_{\phi=0}^{2\pi} d\phi \quad (3.19)$$

$$= C \mathbf{u}_x \cdot 2\ell \cdot \int_{\rho=0}^a \rho d\rho \cdot 2\pi \quad (3.20)$$

$$= C \mathbf{u}_x \cdot 2\pi \ell a^2. \quad (3.21)$$

Define

$$C = \frac{m_x}{2\pi \ell a^2}, \quad (3.22)$$

with m_x the x -component of the magnetic moment of the rod. The magnetic moment of the rod now becomes:

$$\mathbf{m} = m_x \mathbf{u}_x \quad (3.23)$$

To shrink the rod to a single point in space, each dimension will be shrunk step-by-step.

3.2.2 Shrinking the rod to a line

By letting the radius a of the rod converge to zero, the rod will become a line. The magnetic moment of the rod, the orientation and magnetic strength of the object, should stay the same. This means that the magnetisation inside of the rod should increase (since the volume is decreasing), this shall be done by redetermining the constant C to accommodate for this difference. Suppose the magnetisation of this line is:

$$\mathbf{M} = C \cdot \mathbf{u}_x \cdot [U(x + \ell) - U(x - \ell)] \delta(y) \delta(z) \quad (3.24)$$

δ being the Dirac delta function. The constant C will be such that the magnetic moment of the line will be the same as the magnetic moment of the rod. Calculating the magnetic moment of the line in the same way as for the rod we obtain:

$$\mathbf{m} = \iiint_{V_\infty} \mathbf{M} dV \quad (3.25)$$

$$= \int_{x=-\infty}^{\infty} \int_{y=-\infty}^{\infty} \int_{z=-\infty}^{\infty} C \cdot \mathbf{u}_x \cdot [U(x + \ell) - U(x - \ell)] \delta(y) \delta(z) \cdot dx \, dy \, dz \quad (3.26)$$

Note that integrating along a line is essentially the same as integrating in only one direction, in this case the x -direction. Because in the y and z direction the magnetisation is defined as a Dirac delta function, they will contribute nothing to the magnetic moment (since $\int_{-\infty}^{\infty} \delta(x) \, dx = 1$, by definition). Note however that although the integration for the magnetic moment runs from $-\infty$ to ∞ , the line still has length 2ℓ .

The freedom in choosing C now allows one to choose it in such a way that the magnetisation of the rod and the line are equal. This result is obtained by setting $C = \frac{m_x}{2\ell}$, thus finding the following magnetic moment $\mathbf{m}$:

$$\mathbf{m} = C \cdot \mathbf{u}_x \cdot 2\ell \quad (3.27)$$

$$= \frac{m_x}{2\ell} \cdot 2\ell \mathbf{u}_x \quad (3.28)$$

$$= m_x \mathbf{u}_x. \quad (3.29)$$

Note that this is the same magnetic moment as the magnetic moment of the three-dimensional rod. The value of C is chosen such that the increase of C is inversely equivalent to the decrease of the volume of the rod. C is now equal to $\frac{m_x}{2\ell}$, since the rod has become a line. A cylinder is basically the surface of a circle πa^2 multiplied by the height of the cylinder, 2ℓ in this case. Since the cylinder has shrunk to a line, it does not have this surface factor πa^2 anymore, so it does not have to be accounted for anymore in the value of C . This C value accommodates for the decrease of the volume (and thus the magnetisation) of the rod, thus preserving the magnetic moment of the object. The expression for C will prove useful again in equation (3.38).

3.2.3 Back to the potential

The potential calculated at (3.12) holds for every uniform magnetisation of the rod in $\mathbb{R}^3$, but to use it in this context it has to be calculated specifically for the three-dimensional rod, resulting in the following expression for the potential of the rod:

$$\psi(\mathbf{r}) = -\frac{1}{4\pi} \iiint_{\mathbb{R}^3} \frac{1}{\|\mathbf{r} - \mathbf{r}'\|} (\nabla' \cdot \mathbf{M}')(\mathbf{r}') d\mathbf{r}'. \quad (3.30)$$

Substituting the value of C in (3.24), yields the following for the magnetisation $\mathbf{M}'$ of the line:

$$\mathbf{M}' = \frac{m_x}{2\ell} \mathbf{u}_x \cdot [U(x' + \ell) - U(x' - \ell)] \delta(y') \delta(z'). \quad (3.31)$$

Calculating the divergence of this new magnetisation $\mathbf{M}'$, used to plug into (3.30), yields:

$$\nabla' \cdot \mathbf{M}' = \begin{bmatrix} \partial x' \\ \partial y' \\ \partial z' \end{bmatrix} \cdot \begin{bmatrix} M'_x \\ 0 \\ 0 \end{bmatrix} = \partial x' \cdot M'_x = \frac{m_x}{2\ell} [\delta(x' + \ell) - \delta(x' - \ell)] \delta(y') \delta(z'). \quad (3.32)$$

Note: the derivative of the Heaviside function is equal to the dirac delta function, hence $\delta(x' + \ell)$ and $\delta(x' - \ell)$. Plugging (3.32) into the potential (3.30) yields:

$$\psi(\mathbf{r}) = -\frac{1}{4\pi} \frac{m_x}{2\ell} \int_{x'=-\infty}^{\infty} \int_{y'=-\infty}^{\infty} \int_{z'=-\infty}^{\infty} \frac{1}{\|\mathbf{r} - \mathbf{r}'\|} [\delta(x' + \ell) - \delta(x' - \ell)] \delta(y') \delta(z') dx' dy' dz'. \quad (3.33)$$

Picking an arbitrary observation point in $\mathbb{R}^3$ for $\mathbf{r} = (x, y, z)$, and $\mathbf{r}' = (x', y', z')$, we are able to calculate the distance between the two explicitly:

$$\psi(\mathbf{r}) = -\frac{1}{4\pi} \frac{m_x}{2\ell} \int_{x'=-\infty}^{\infty} \int_{y'=-\infty}^{\infty} \int_{z'=-\infty}^{\infty} \frac{[\delta(x' + \ell) - \delta(x' - \ell)] \cdot \delta(y') \delta(z)}{\sqrt{(x - x')^2 + (y - y')^2 + (z - z')^2}} dx' dy' dz'. \quad (3.34)$$

Calculating this expression explicitly yields the final expression for the potential of the line:

$$\psi(\mathbf{r}) = -\frac{1}{4\pi} \frac{m_x}{2\ell} \left[\frac{1}{\sqrt{(x + \ell)^2 + y^2 + z^2}} - \frac{1}{\sqrt{(x - \ell)^2 + y^2 + z^2}} \right]. \quad (3.35)$$

In the next section, ℓ will be shrunk to zero, this will cause some problems with this definition of ψ (since m_x is being divided by 2ℓ). Luckily there is a solution to that.

Shrinking the line to a point

The process of shrinking the line to a point is a little different than from a rod to a line. The following definition of the derivative of a function f is used:

$$f'(x) = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x-h)}{2h}. \quad (3.36)$$

Above equation will prove useful if we rewrite (3.35) in the following way:

$$\psi(r) = -\frac{1}{4\pi} \frac{m_x}{2\ell} \left[\frac{1}{\sqrt{(x+\ell)^2 + y^2 + z^2}} - \frac{1}{\sqrt{(x-\ell)^2 + y^2 + z^2}} \right] \quad (3.37)$$

$$= -\frac{1}{4\pi} \frac{m_x}{2\ell} 2\ell \left[\frac{\frac{1}{\sqrt{(x+\ell)^2 + y^2 + z^2}} - \frac{1}{\sqrt{(x-\ell)^2 + y^2 + z^2}}}{2\ell} \right] \quad (3.38)$$

$$= -\frac{1}{4\pi} m_x \cdot \left[\frac{\frac{1}{\sqrt{(x+\ell)^2 + y^2 + z^2}} - \frac{1}{\sqrt{(x-\ell)^2 + y^2 + z^2}}}{2\ell} \right]. \quad (3.39)$$

If we now use (3.36) on expression (3.39), define $f(x)$ as:

$$f(x) = \frac{1}{\sqrt{x^2 + y^2 + z^2}} \quad (3.40)$$

$$f(x - \ell) = \frac{1}{\sqrt{(x - \ell)^2 + y^2 + z^2}} \quad (3.41)$$

$$f(x + \ell) = \frac{1}{\sqrt{(x + \ell)^2 + y^2 + z^2}} \quad (3.42)$$

Plugging the above expressions into (3.39) results in:

$$\psi(\mathbf{r}) = -\frac{1}{4\pi} m_x \frac{f(x + \ell) - f(x - \ell)}{2\ell} \quad (3.43)$$

Now, using the definition of the derivative (3.36), let $\ell \rightarrow 0$:

$$\psi(\mathbf{r}) = -\frac{1}{4\pi} m_x \lim_{\ell \rightarrow 0} \frac{f(x + \ell) - f(x - \ell)}{2\ell} \quad (3.44)$$

$$= -\frac{1}{4\pi} m_x f'(x) \quad (3.45)$$

$$= -\frac{1}{4\pi} m_x \cdot -\frac{1}{2} (x^2 + y^2 + z^2)^{-\frac{3}{2}} 2x \quad (3.46)$$

$$= -\frac{1}{4\pi} m_x \frac{x}{\|\mathbf{r}\|^3} \quad (3.47)$$

Now (3.47) is the expression of the potential for a magnetised line dipole shrunk to length 0 (i.e. a point dipole).

3.2.4 Calculating the $\mathbf{H}$ -field

With the potential (3.47), the magnetic field strength $\mathbf{H}$ follows from equation (3.3):

$$\mathbf{H} = -\nabla\psi. \quad (3.48)$$

Applying (3.48) leaves us with:

$$-\nabla\psi = -\nabla\left(-\frac{1}{4\pi}\mathbf{m}\frac{x}{\|\mathbf{r}\|}\right) \quad (3.49)$$

$$= -\frac{1}{4\pi}\mathbf{m}\nabla\left(\frac{x}{\|\mathbf{r}\|}\right). \quad (3.50)$$

Applying the partial derivatives in the nabla-operator, the magnetic field strength $\mathbf{H}$ is:

$$-\nabla\psi = \mathbf{H} = -\frac{1}{4\pi}\mathbf{m}\left[\left(\frac{1}{\|\mathbf{r}\|^3} - 3x^2\frac{1}{\|\mathbf{r}\|^5}\right)\mathbf{u}_x - 3xy\frac{1}{\|\mathbf{r}\|^5}\mathbf{u}_y - 3xz\frac{1}{\|\mathbf{r}\|^5}\mathbf{u}_z\right]. \quad (3.51)$$

Now, (3.51) defines the magnetic field strength for a magnetic point dipole with magnetic moment only defined in the x -direction. Let us now generalise this result to find the magnetic field strength $\mathbf{H}$ for a dipole with arbitrary magnetic moment $\mathbf{m}$:

$$\mathbf{m} = m_x\mathbf{u}_x + m_y\mathbf{u}_y + m_z\mathbf{u}_z \quad (3.52)$$

In addition suppose:

$$f(\mathbf{r}) = -\frac{1}{4\pi}\frac{1}{\|\mathbf{r}\|}. \quad (3.53)$$

f is Green's function for the Laplace operator, so the method proposed in equation 3.11 can be applied again to obtain:

$$\psi(\mathbf{r}) = -\frac{1}{4\pi}\left(m_x\frac{\partial f}{\partial x} + m_y\frac{\partial f}{\partial y} + m_z\frac{\partial f}{\partial z}\right) \quad (3.54)$$

$$= -\frac{1}{4\pi}\left(m_x \cdot -\frac{x}{\|\mathbf{r}\|^3} + m_y \cdot -\frac{y}{\|\mathbf{r}\|^3} + m_z \cdot -\frac{z}{\|\mathbf{r}\|^3}\right) \quad (3.55)$$

$$= \frac{1}{4\pi}\frac{m_x \cdot x + m_y \cdot y + m_z \cdot z}{\|\mathbf{r}\|^3} \quad (3.56)$$

$$= \frac{1}{4\pi}\frac{\mathbf{m} \cdot \mathbf{r}}{\|\mathbf{r}\|^3}. \quad (3.57)$$

Using (3.48) one obtains the $\mathbf{H}$ -field:

$$\mathbf{H} = -\nabla\left(\mathbf{m} \cdot \frac{\mathbf{r}}{4\pi\|\mathbf{r}\|^3}\right). \quad (3.58)$$

The dot product rule of vector calculus states:

$$\nabla(\mathbf{A} \cdot \mathbf{B}) = (\mathbf{A} \cdot \nabla)\mathbf{B} + (\mathbf{B} \cdot \nabla)\mathbf{A} + \mathbf{A} \times (\nabla \times \mathbf{B}) + \mathbf{B} \times (\nabla \times \mathbf{A}). \quad (3.59)$$

As mentioned earlier, $\mathbf{m}$ is independent from $\mathbf{r}$, so $(\mathbf{B} \cdot \nabla)\mathbf{A}$, $\mathbf{A} \times (\nabla \times \mathbf{B})$ and $\mathbf{B} \times (\nabla \times \mathbf{A})$ are zero. So:

$$\mathbf{H} = -[(\mathbf{A} \cdot \nabla) \cdot \mathbf{B}] \quad (3.60)$$

$$= - \left[(\mathbf{m} \cdot \nabla) \cdot \frac{\mathbf{r}}{4\pi \|\mathbf{r}\|^3} \right] \quad (3.61)$$

$$= \frac{1}{4\pi} \left[\frac{3(\mathbf{m} \cdot \mathbf{r})\mathbf{r}}{\|\mathbf{r}\|^5} - \frac{\mathbf{r}}{\|\mathbf{r}\|^3} \right] \quad (3.62)$$

The derivation of (3.61) to (3.62) can be found in the appendix. From this $\mathbf{H}$ -field the magnetic induction field $\mathbf{B}$ follows immediately by equation (3.1):

$$\mathbf{B}(\mathbf{r}) = \frac{\mu_0}{4\pi} \left[\frac{3(\mathbf{m} \cdot \mathbf{r})\mathbf{r}}{\|\mathbf{r}\|^5} - \frac{\mathbf{m}}{\|\mathbf{r}\|^3} \right] \quad (3.63)$$

Generalising (3.63) to accommodate a point dipole anywhere in $\mathbb{R}^3$ the final forward model is:

$$\mathbf{B}(\hat{\mathbf{r}}) = \frac{\mu_0}{4\pi} \left[\frac{3(\mathbf{m} \cdot \hat{\mathbf{r}})\hat{\mathbf{r}}}{\|\hat{\mathbf{r}}\|^5} - \frac{\mathbf{m}}{\|\hat{\mathbf{r}}\|^3} \right] \quad (3.64)$$

where $\hat{\mathbf{r}} = \mathbf{r} - \mathbf{d}$.

3.3 Symmetry

Something to bear in mind using this model are the symmetries that manifest in the magnetic induction field of a dipole. The four graphs below demonstrate this symmetry:

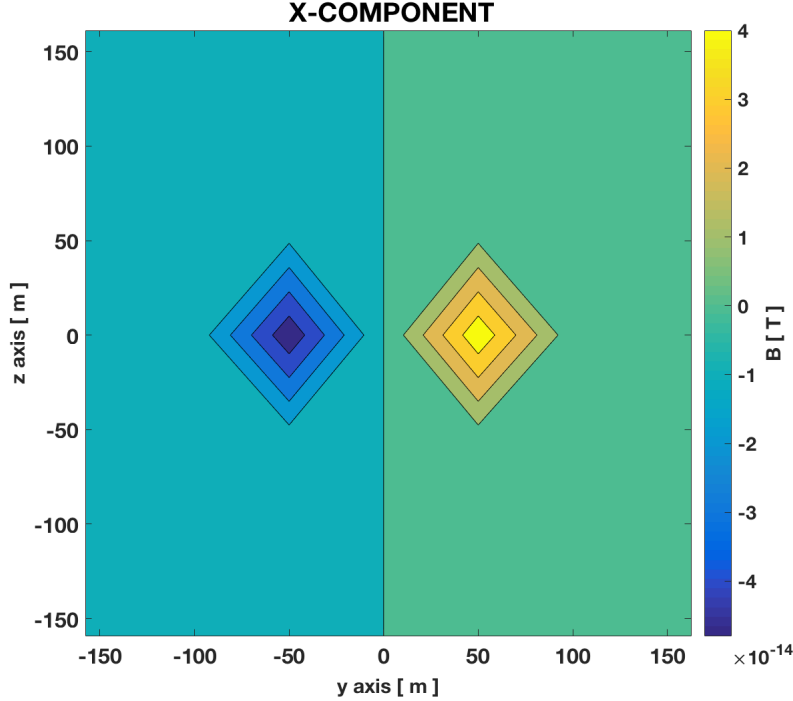


Figure 3.2: Contour plot of the x -component of a dipole in $[0, 1, 0]$ m.

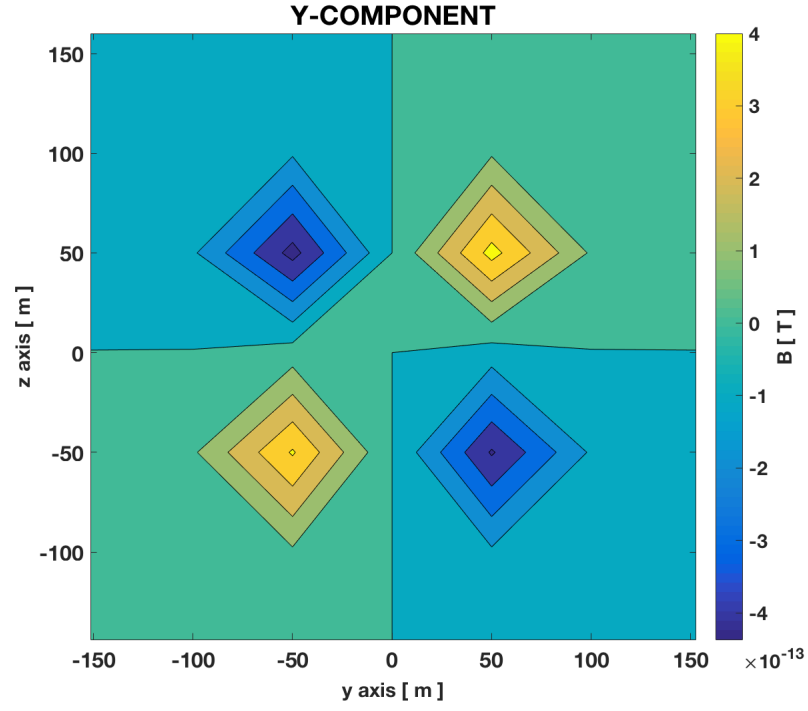


Figure 3.3: Contour plot of the y -component of a dipole in $[0, 1, 0]$ m.

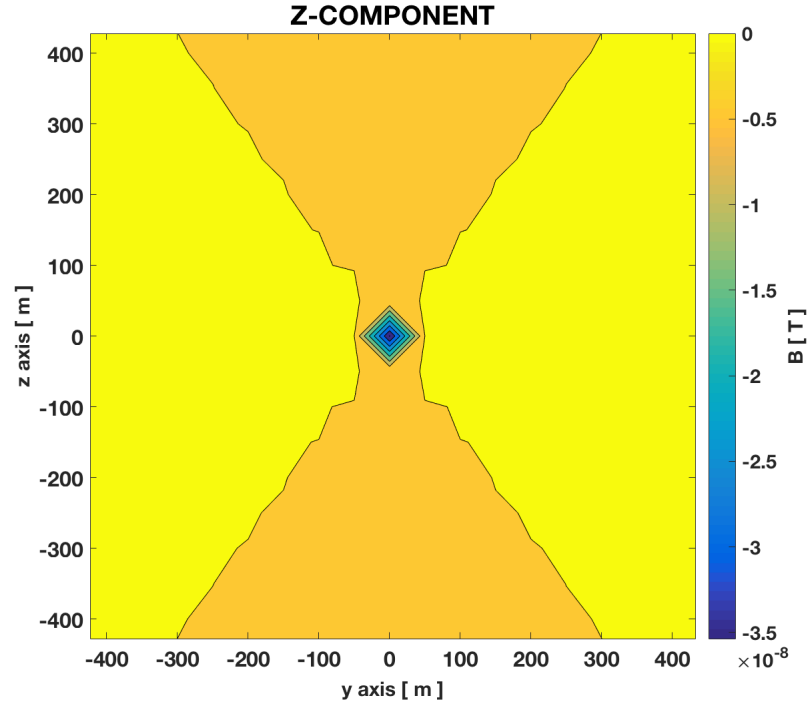


Figure 3.4: Contour plot of the z -component of a dipole in $[0, 1, 0]$ m.

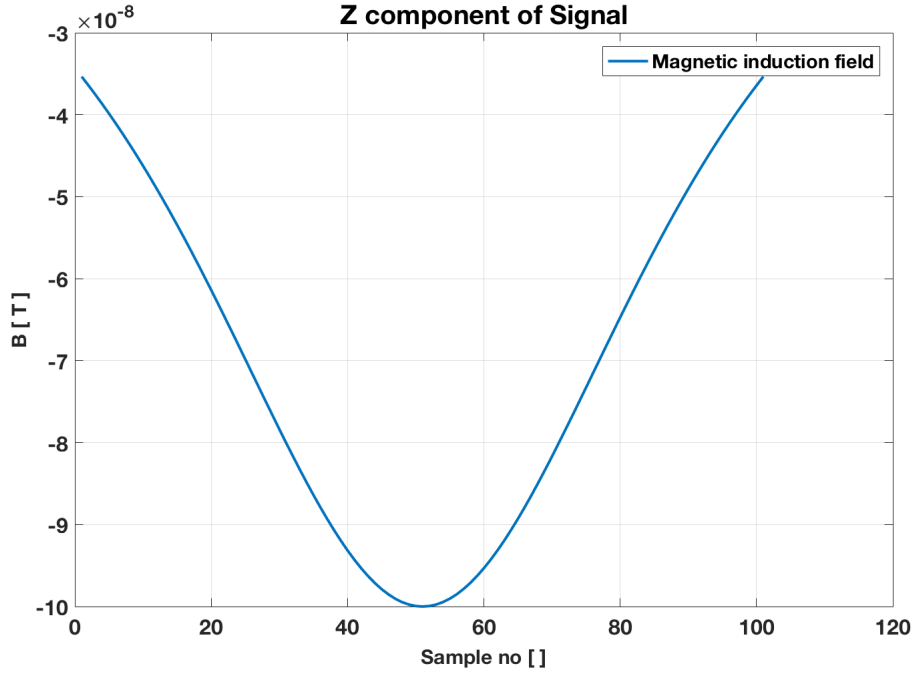


Figure 3.5: z component of magnetic induction field along the x axis.

The first three graphs are contour plots of a dipole at $[0, 1, 0]$ m, with a magnetic moment of $[0, 0, 1]$ Am². The observation plane is the $y z$ plane at $x = 1$. The fourth figure shows the z component of the magnetic induction field along a path which is the x axis from -1 to 1 . The other two components of the signal do not matter since they are equal to zero. The important remark to make is that if the dipole were to be located in $[0, -1, 0]$ m, the graphs would look exactly the same. This is of course no real world example, but such situations could definitely occur. It is a phenomenon worth taking into account in certain situations. In this thesis, it hasn't been taken into account since the drones would all have to fly exactly perpendicular to the magnetic moment of the shipping container which they are looking for. This is not the case, but such situations could definitely happen in real life.

The way to omit this problem is to make sure the drones fly in different directions, to accomplish a unique solution (position and magnetic moment). Another way to solve the non-uniqueness of the solution is to take as much samples as possible, to increase the resolution of the image sketched of the magnetic induction field. This would only benefit however, if the drones would fly in perpendicular paths to each other.

4. Inverse model

4.1 Inverse Problem

For the inverse problem one tries to connect the best forward model to a given set of real-world observations. This way certain important parameters defining the system can be deduced from the forward model. In this thesis the inverse problem uses a signal of a magnetometer describing the magnetic induction field $\mathbf{B}$, to deduce a forward model from it. This signal is heavily compromised by noise of the Earth or other ferromagnetic objects close by, which complicates matters. From this best-fitting forward model certain conclusions can be drawn such as the magnetic moment or position of the ferromagnetic object.

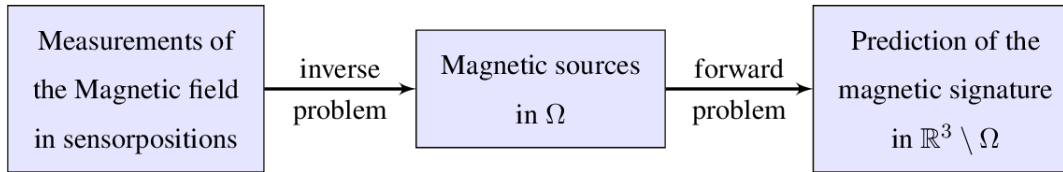


Figure 4.1: Flow chart of forward/inverse modelling. [5]

This section will describe the process of linking a forward model found in the previous chapter to a given (real-world) signal. First, the problem will be described as a set of non-linear equations. Next, the solution is presented by means of a search algorithm called Simulated Annealing (SA).

4.2 Problem formulation

This part is based upon the work of Arie Sheinker and Boaz Lerner, as mentioned in the bibliography[6]. Although there was no test set-up available for this thesis as in Sheinker's work, the starting point to formulate the problem remained the same. To envision this, picture a magnetometer running down a predetermined track on a cart, measuring the magnetic field in equidistant points. Next to the track a ferromagnetic object is located, pictured on the next page.

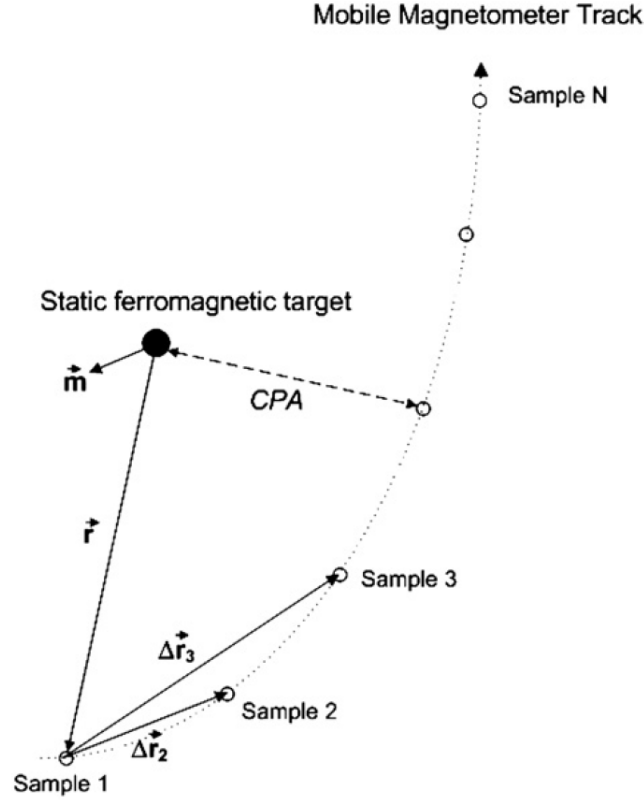


Figure 4.2: The test set-up in Sheinker's paper. [6]

This setup yields a set of equations to describe the magnetic B-field in each sample point on the track. A MATLAB simulation was used, using the forward model described in the previous section, to which a little random noise was added to produce the source signal. The setup in Figure 4.2 produces a set of equations describing the field in each of the N samples:

$$\begin{cases} \mathbf{B}(\mathbf{m}, \mathbf{r}) = \frac{\mu_0}{4\pi} \left[\frac{3(\mathbf{m} \cdot \mathbf{r})\mathbf{r}}{||\mathbf{r}||^5} - \frac{\mathbf{m}}{||\mathbf{r}||^3} \right] + \mathbf{B}_E \\ \mathbf{B}(\mathbf{m}, \mathbf{r}_2) = \frac{\mu_0}{4\pi} \left[\frac{3(\mathbf{m} \cdot \mathbf{r}_2)\mathbf{r}_2}{||\mathbf{r}_2||^5} - \frac{\mathbf{m}}{||\mathbf{r}_2||^3} \right] + \mathbf{B}_E \\ \vdots \\ \mathbf{B}(\mathbf{m}, \mathbf{r}_N) = \frac{\mu_0}{4\pi} \left[\frac{3(\mathbf{m} \cdot \mathbf{r}_N)\mathbf{r}_N}{||\mathbf{r}_N||^5} - \frac{\mathbf{m}}{||\mathbf{r}_N||^3} \right] + \mathbf{B}_E \end{cases} \quad (4.1)$$

With $\mathbf{B}_E$ being the ambient Earth's magnetic field. An important thing to note now is that this set of equations has $N + 1$ unknown vectors ($\mathbf{r}, \mathbf{r}_2, \dots, \mathbf{r}_N$ and $\mathbf{m}$), which means there are $3 * (N + 1)$ unknown variables (since the vector are three-dimensional and every dimension has its own scalar equation). Thus it is not guaranteed that a unique solution will be found, since it is a set of nonlinear equations. However, if $\mathbf{r}$ is known, the rest of the place vectors $\mathbf{r}_2, \dots, \mathbf{r}_N$ are know as well, since the track of the magnetometer is predetermined. To state this mathematically, the following substitution can be made:

$$\mathbf{r}_i = \mathbf{r} + \Delta \mathbf{r}_i \quad (4.2)$$

Where $\Delta \mathbf{r}_i$ is the distance from the first sample position to the i -th sample position, a value which is known for $i = 1, \dots, N$. So now the set of equations contains N equations, and it has only two unknown variables, $\mathbf{r}$ and $\mathbf{m}$. Although the problem stated in equation (4.1) is still nonlinear this reduction of unknown variables helps in searching for a solution.

4.3 Simulated Annealing

Simulated annealing (SA) is an algorithm based on the physical process of annealing. Annealing is a method often used by blacksmiths to lower the total stress in a piece of metal, by repeatedly heating the metal and letting it cool down slowly. This method toughens the piece of metal in multiple iterations, and lowers the overall energy in the piece of metal. SA works analogous to this blacksmith technique, and is used to find global maxima or minima. It is especially useful for these kinds of applications, since it has the favourable property that it is able to overcome local minima or maxima. This fundamental property of the algorithms comes from the fact that the key decision variable of the algorithm is stochastically defined. The decision variable is at the core of the algorithm, and it is the reason that SA is so useful. Simulated annealing will be used in this situation to find the best fitting dipole model for a given signal. For it to do this, it will need a method to compare different models with each other, something which will be addressed later on. SA compares so-called *states* with each other. A state is a snapshot of a situation, in this case the position and magnetic moment of a dipole. To calculate the forward model two values are needed: the magnetic moment $\mathbf{m}$ and position $\mathbf{r}$ of the dipole. These two vectors combined form the six-dimensional state:

$$S = (m_x, m_y, m_z, r_x, r_y, r_z) \quad (4.3)$$

To limit the number of neighbouring states the algorithm has to examine, boundaries and a resolution are set for the position and magnetic moment of the ferromagnetic object. These conditions are based on real-world demands to usefully identify and estimate the position and magnetic moment of the ferromagnetic object. Neighbouring states are states which differ in value no more than the resolution of the grid on which the algorithm is searching, and which lie between the specified boundaries.

The algorithm is designed to minimise the energy in a system. This energy must be thought of not as energy in the traditional sense of the word, but as a function which assigns a certain value to a state. This energy function allows the algorithm to compare states to each other, and to decide which state is better (i.e. has a lower energy). In this problem a solution must be sought which approximates the measured signal best, in other words has a minimal error relative to the measured signal. The mean squared error between the calculated model and the measured signal is the function which needs to be minimised, so this will be the energy function:

$$E(S_j) = \frac{1}{N} \sum_{i=1}^N \|\mathbf{B}_m^i - \mathbf{B}_c^i(S_j)\|^2, \quad (4.4)$$

where $\mathbf{B}_m^i$ is the measured signal at the i -th sample position, and $\mathbf{B}_c^i(S_j)$ is the calculated forward model of the dipole. The forward model is calculated using the position and magnetic moment information in the state S_j . In an ideal scenario SA would find the exact source signal of the ferromagnetic object by varying the position and magnetic moment over a given grid of possible options. In the case that the exact signal is found the following would happen with the

energy of that particular state:

$$E_{\text{optimal}} = \sum_{i=1}^N \|\mathbf{B}_m^i - \mathbf{B}_c^i(S_{\text{source}})\|^2 \quad (4.5)$$

$$= \sum_{i=1}^N \|\mathbf{B}_{\text{source}}^i + \mathbf{B}_{\text{noise}}^i - \mathbf{B}_c^i(S_{\text{source}})\|^2 \quad (4.6)$$

$$= \sum_{i=1}^N \|\mathbf{B}_{\text{noise}}^i\|^2 \quad (4.7)$$

In the optimal scenario, the measured source field and the calculated source field would cancel out, causing the value of the energy function to be equal to the noise level. To determine whether or not to transition into a neighbouring state an acceptance value or decision variable is used. The special feature of this algorithm, as mentioned before, is that the decision variable is stochastically based. It compares two neighbouring states S_j and S_k (with energies E_j and E_k) with each other:

$$a_{kj} = \begin{cases} 1, & E_j < E_k \\ \exp\left(\frac{E_j - E_k}{T}\right) & E_j > E_k \end{cases} \quad (4.8)$$

To recite the blacksmith analogy, the decision variable is also dependent on the temperature T of the algorithm. The algorithm starts off with a high temperature, and gradually brings it down using a cooling parameter α . This causes the effect that at the beginning of the process a state which, paradoxically, has a higher energy (and thus is a worse approximation of the location and magnetic moment of the object), still has a relatively large chance to be favoured by the acceptance value. The corollary of this tactic is that in the beginning the entire area is coarsely examined as to overcome local minima, and that as the temperature drops the chosen states converge towards the global minimum. As the temperature $T \rightarrow 0$, the term $\exp\frac{E_k - E_n}{T} \rightarrow 0$, $E_k - E_n < 0$. This means the probability that S_n is favoured over S_k —although the energy of S_n is higher than the energy of S_k —will converge to 0 as well (line 16).

Algorithm 1: Simulated annealing.

input : A crude signal from a distant ferromagnetic object.

output: The optimal state (magnetic moment and position) fitting the crude signal.

```

1 Set parameters:
2  $T$  = starting temperature;
3  $\alpha$  = cooling rate  $0 < \alpha < 1$ ;
4  $\eta$  = total number of neighbouring states to be examined;
5  $\zeta$  = total number of running cycles;

6 Choose initial state  $S_k$ 
7 for  $i = 1$  to  $\zeta$  do
8    $E_k = \text{energy}(S_k)$ ;
9   for  $n = 1$  to  $\eta$  do
10     $S_n = \text{random neighbor of } S_k$ ;
11     $E_n = \text{energy}(S_n)$ ;
12    if  $\text{energy}(S_n) < \text{energy}(S_k)$  then
13       $S_n = S_k$ 
14    else
15       $R = \text{Random}(0,1)$ ;
16      if  $R < \exp \frac{E_k - E_n}{T}$  then
17         $S_n = S_k$ 
18      end
19    end
20  end
21   $T = \alpha T$ ;
22 end

23 def  $\text{energy}(S_1, S_2)$ :
24   return  $\frac{1}{N} \sum (|S_1 - S_2|^2)$ ; ; //  $N$  is the number of elements in each array
25 end

```

5. Numerical results

5.1 Introduction

The implementation in MATLAB produced some very similar results to the finding in the work of Sheinker et al. This is favourable since it means that the algorithm is able to reliably predict the location and magnetic moment of a ferromagnetic object, based on its (disturbed) magnetic field signal. This property is vital in light of the swarm application later on. To demonstrate how the algorithm is working, consider the following graph:

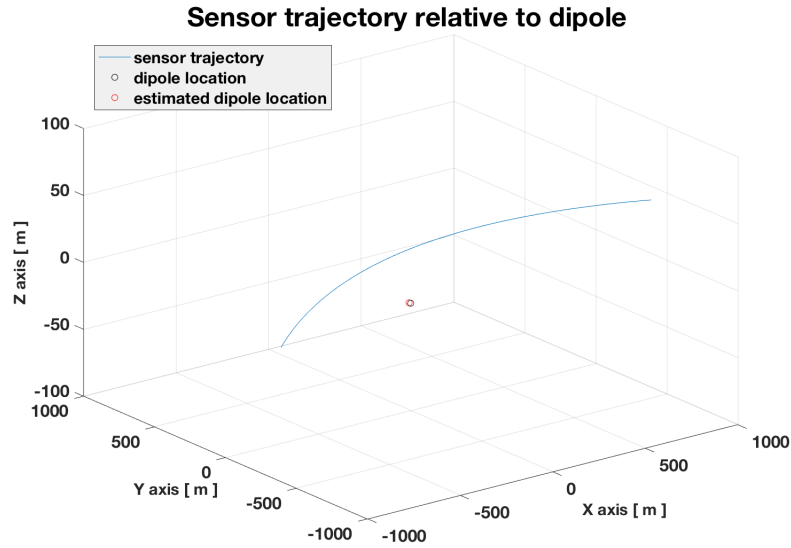


Figure 5.1: Sensor trajectory with estimation and actual dipole location.

This graph is a simulation –with real world values– of the test setup sketched in figure 4.2. The search space of the drone to find the dipole is defined in metres and measures 1000m by 1000m by 100m. The search space was chosen such that the sea-level is located in the $z = 0$ plane. The dipole is located at $[0, 0, -20]$ (20 metres below sea-level), and it has a magnetic moment of $[-1000, 1300, -800] \text{ Am}^2$. These values are loosely based upon real world values. Earlier work estimated the magnetic moment of a steel shipping container somewhere in the order of magnitude of 10^3 Am^2 . The path of the drone starts at sea level and curves up to a height of around 40 metres, which is quite an easy task for most drones. Along its path the drone took a measurement every 10 metres approximately. In each of these sample points a magnetic field value is found with three components: x , y and z . These components are plotted

against the samples in the graphs below. As in the test setup, a sensor is dragged along a parabolic path. It curves around the actual location of the dipole, as to gain more information about its magnetic field. This is an advantage since with more information the algorithm is able to determine the position and magnetic moment of the dipole with more accuracy. The curved path also omits the possibility of flying perpendicular in one direction to the magnetic moment of the dipole, a problem addressed in section 3.3. Along the path (the blue line), the sensor took 201 samples of the magnetic field, these samples are depicted in Figures 5.2 to 5.4 as the actual signal, subject to noise. Since no signal from an actual ferromagnetic object was available at the time of implementing, an 'actual' fictitious signal was created. This has been done by setting a position and magnetic moment, then calculating the magnetic induction field, but adding some random noise to it. A good notion to explain concerning noise is the *signal-to-noise ratio* (SNR). This ratio indicates the amount of noise a signal contains. The higher the ratio, the clearer the signal. Sometimes it is denoted in decibel (dB), but for the sake of simplicity this will not be done in this thesis. The level of noise was set in advance at a signal-to-noise ratio of 10. This means the field of the dipole was 10 times stronger than the noise added. Below the three components of the magnetic field signal and model are plotted along the path travelled by the sensor:

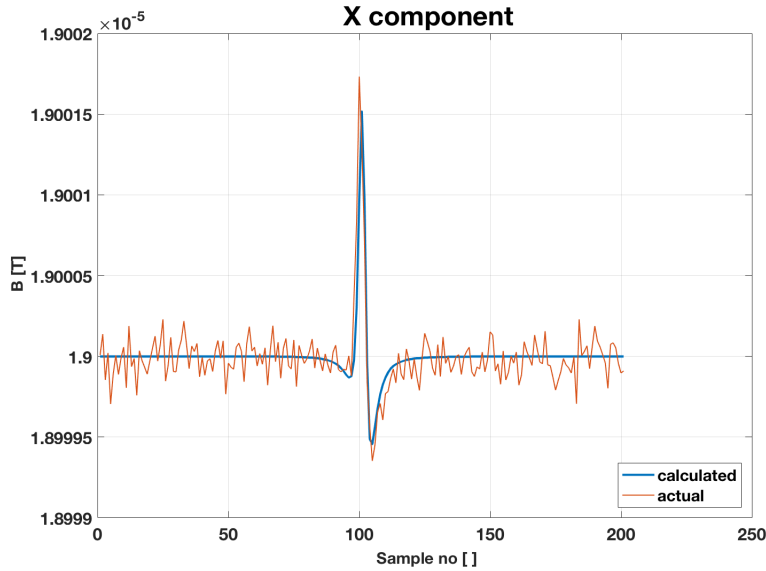
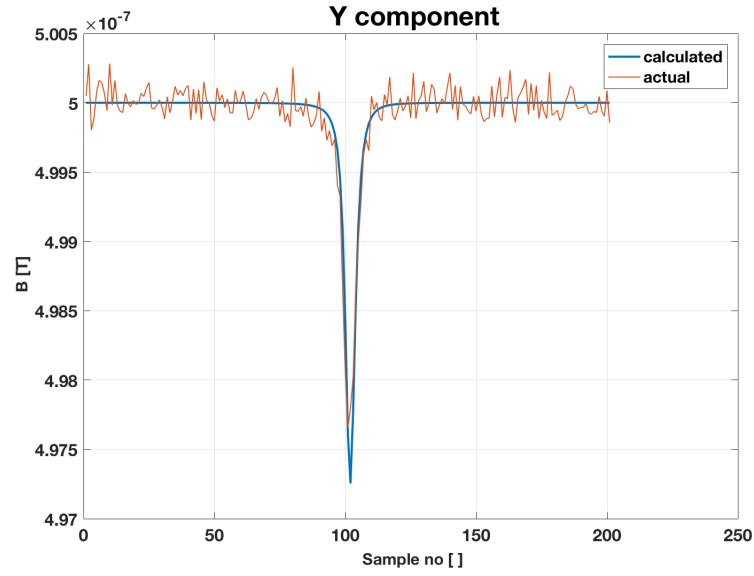
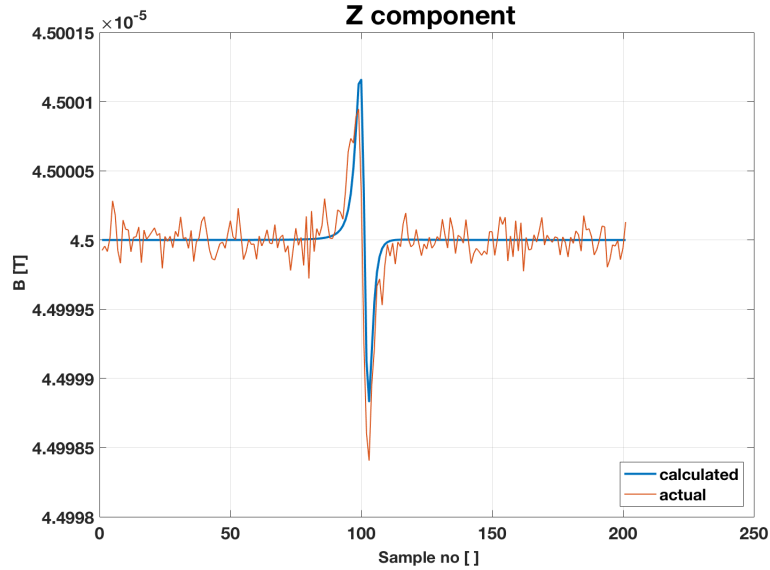


Figure 5.2: x -component of the magnetic induction field.

Figure 5.3: *y*-component of the magnetic induction field.Figure 5.4: *z*-component of the magnetic induction field.

Sample 0 indicates the start of the path, and sample number 201 the end. Each of the three components shows a clear anomaly around sample number 100 approximately, which represents the middle of the path travelled by the sensor. This matches Figure 5.1, since the dipole is roughly located at the middle of the path. For the most part, however, the signal seems constant, apart from the noise. These constant parts of the graph represent the ambient Earth's magnetic field. This ubiquitous field is known at every position at the Earth's surface, and can be calculated using an online Earth magnetic field calculator.¹ In this particular case the

¹The Earth's magnetic field used in this thesis was calculated using the Geomagnetic calculator provided by the National Centers for Environmental Information (NCEI), calculated at 2020-06-06 in Delft (4.3571 E, 52.0116 N). Web: <https://www.ngdc.noaa.gov/geomag/calculators/magcalc.shtml#igrfwmm>

Earth's magnetic field was calculated at $[19, 0.5, 45] \mu\text{T}$, which explains the scale on the y axis of the graphs in Figures 5.2, 5.3 and 5.4. The blue curve is a result of the state vector which the algorithm varies, which contains the information about the magnetic moment and location of the dipole. In this instance, the algorithm terminated with this as the optimal solution which it could find considering the time it was given. The dipole which produces the blue curve is located at $[0, -5, -20] \text{ m}$, and has magnetic moment $[-1300, 1400, -800] \text{ Am}^2$. This means that the algorithm determined the location of the dipole up to 5 metres, and the magnetic moment up to 400 Am^2 . The reason that these differences are multiples of 5 and 50 has to do with the parameter settings of the algorithm, which are worth discussing.

5.2 Implementation

The algorithm, as implemented in MATLAB had the following structure:

```
SA(params,poslim,estp,estm,B,path)
params = [T,alpha,cycles,mlim,mlim,pres,mres,nb]
```

The algorithm takes in an array of parameters defined in the line below it, which consists of the following values:

Parameter	Description
T	start temperature of algorithm
alpha	cooling rate
cycles	number of cycles the algorithm goes through
mlim	lower and upper bound for the magnetic moment estimation
pres	distance between two neighbouring states concerning position
mres	distance between two neighbouring states concerning magnetic moment
nb	number of neighbours examined in each cycle

Table 5.1: Declaration of parameters.

Most of these parameters speak for themselves, but there are a couple worth discussing more elaborately, since they have a bigger impact on the functioning of the algorithm. This will be done later. Besides the `params` array, the SA algorithm takes in the variables:

Variable	Description
poslim	a 3x2 matrix containing the upper and lower bounds in all positional directions.
estm	starting approximation of magnetic moment
estp	starting approximation of position
B	received signal by the sensor
path	path flown by the sensor

Table 5.2: Declaration of variables.

`estm` and `estp` together form the initial state of the algorithm from which it starts 'walking' towards the optimal solution with stepsize `pres` and `mres`. These two represent the initial state chosen in line 6 of algorithm 1. The reason `poslim` is defined separately from `mlim` is that the positional limits differ from each other (1000 m in x and y , and 100 m in z), while the limits

of the magnetic moment may be the same in every direction. Figures 5.1 to 5.4 were produced with the following params settings:

```
params = [50,0.1,200,-1500,1500,5,50,20]
```

5.3 Sensitivity analysis of alpha

One of the parameters worth discussing more in depth is alpha, the cooling rate of the algorithm. As mentioned earlier, this parameter determines the amount of random states the algorithm takes into account. The lower the cooling rate, the more direct the algorithm will 'walk' towards its perceived optimal solution. This is not always desirable, especially when many local minima exist. The following graphs illustrates alpha's influence on the functioning of the algorithm very well:

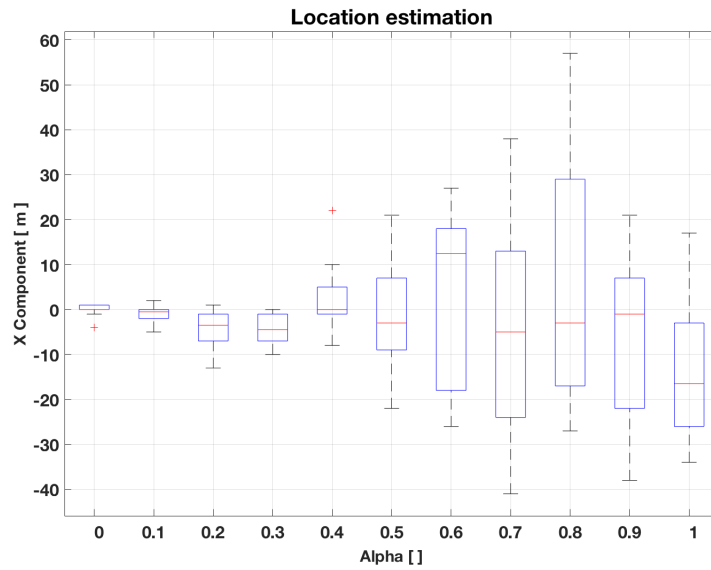


Figure 5.5: x -component of location estimate for different values of alpha

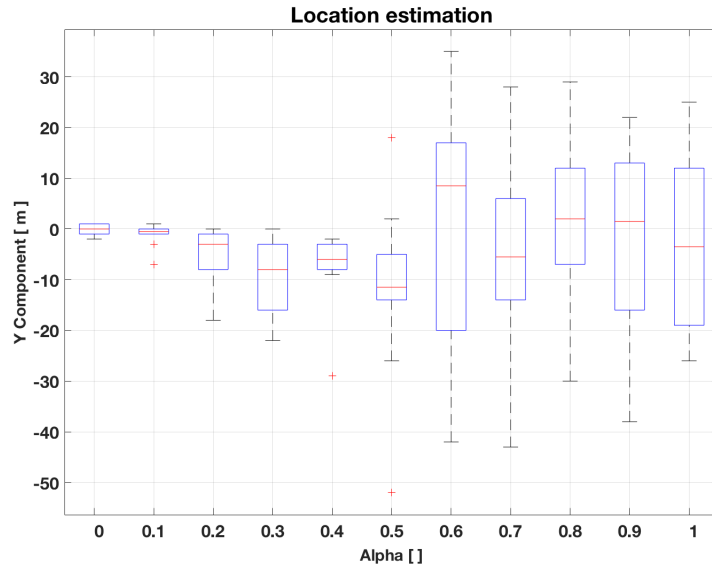


Figure 5.6: y -component of location estimate for different values of alpha

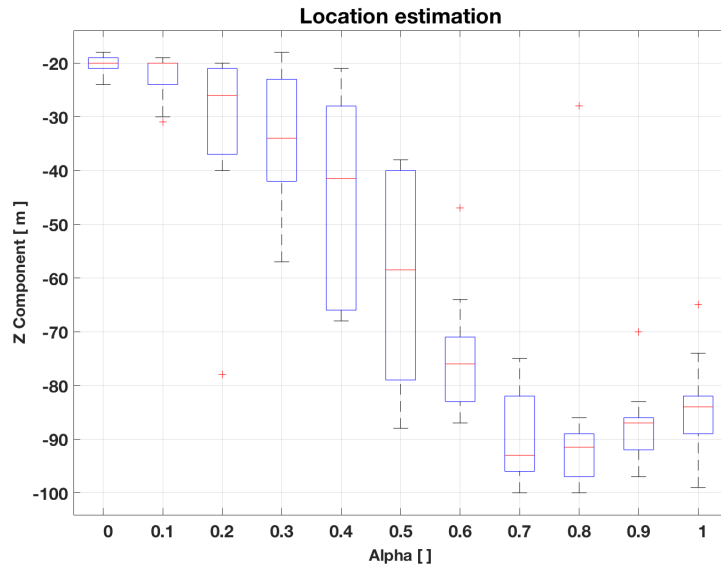


Figure 5.7: z -component of location estimate for different values of alpha

These figures show that the variance of the estimation made by SA increases in all directions as alpha increases. The same observation can be made for the magnetic moment estimation of the dipole shown on the next page.

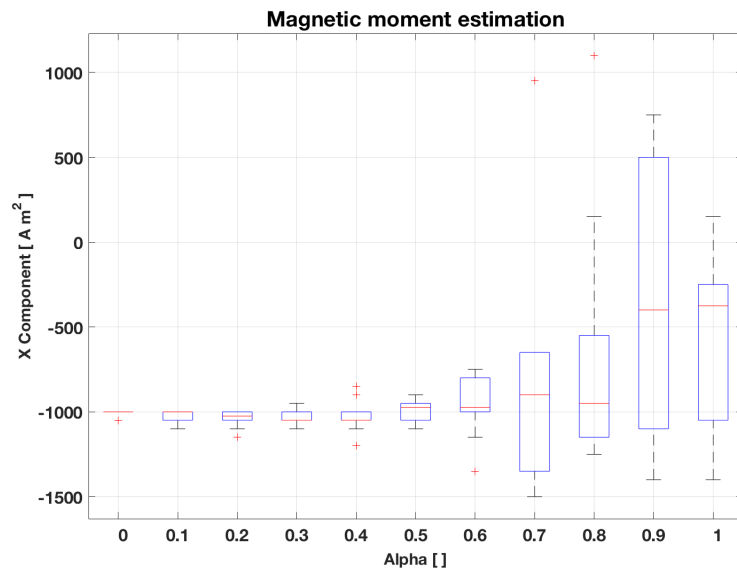


Figure 5.8: x -component of magnetic moment estimate for different values of alpha

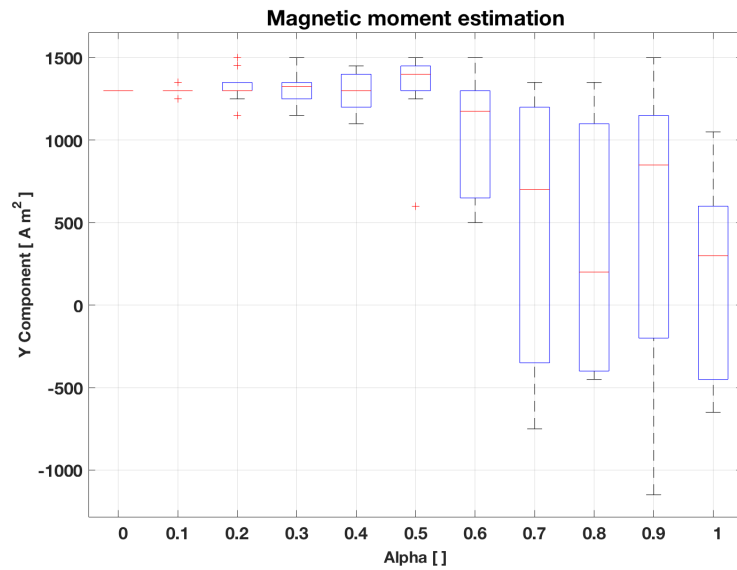


Figure 5.9: y -component of magnetic moment estimate for different values of alpha

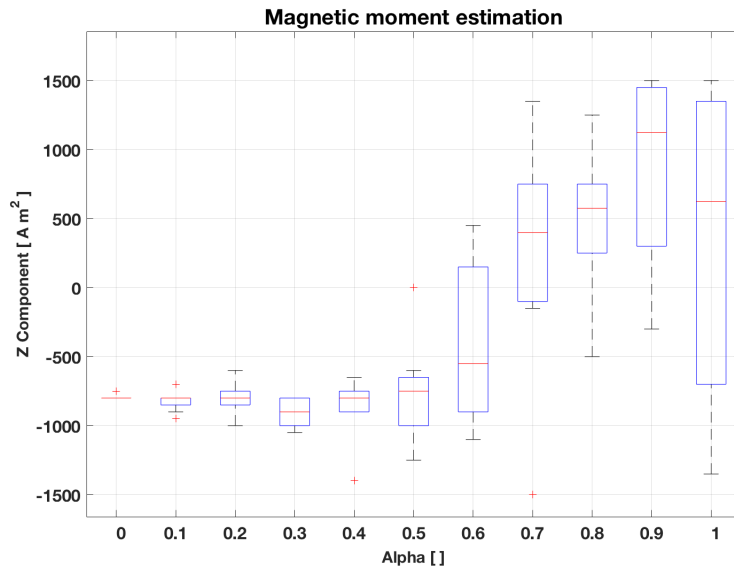


Figure 5.10: z -component of magnetic moment estimate for different values of alpha

The reason the variance increases as alpha increases is the fact that it could end up in a totally wrong area of the solution space at the beginning of the process. Since the temperature would decrease while the algorithm is 'walking' in the wrong corner, it could get 'stuck' there and produce a wrong outcome.

5.4 Statistical analysis based on Monte Carlo simulations

To discuss the individual role of certain important parameters a very clarifying graph is the one below, which visualises the reliability of the algorithm, and repeatability of the results.

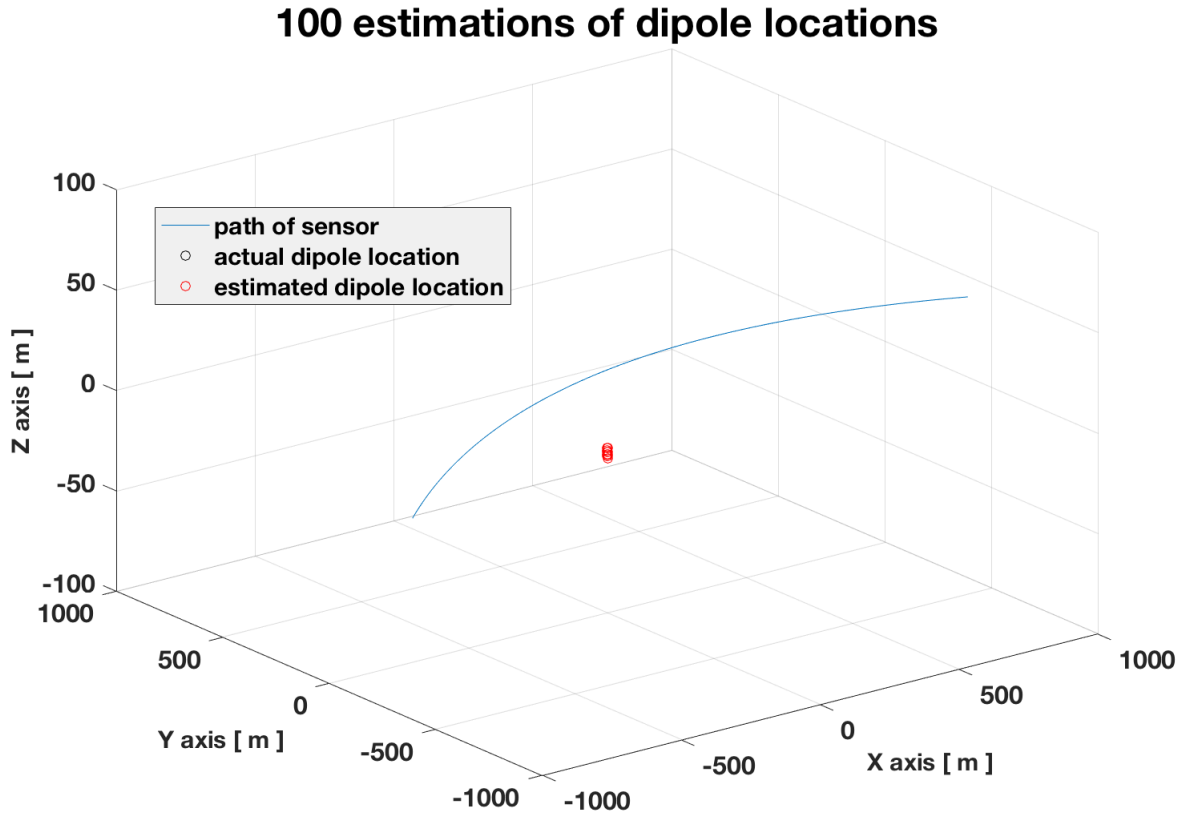


Figure 5.11: 100 estimations of the dipole location by SA.

The figure is a result of 100 runs of SA, each with their own approximation of the position of the dipole. The graph is produced with the following `params` settings:

```
params = [50,0.1,200,-1500,1500,1,50,20]
```

Now, figure 5.11 showcases the accuracy of the algorithm, but it doesn't show the variance of the location estimates very well. A zoomed in version is shown on the next page to visualise the spread in location estimation due to the stochastic base of the algorithm.

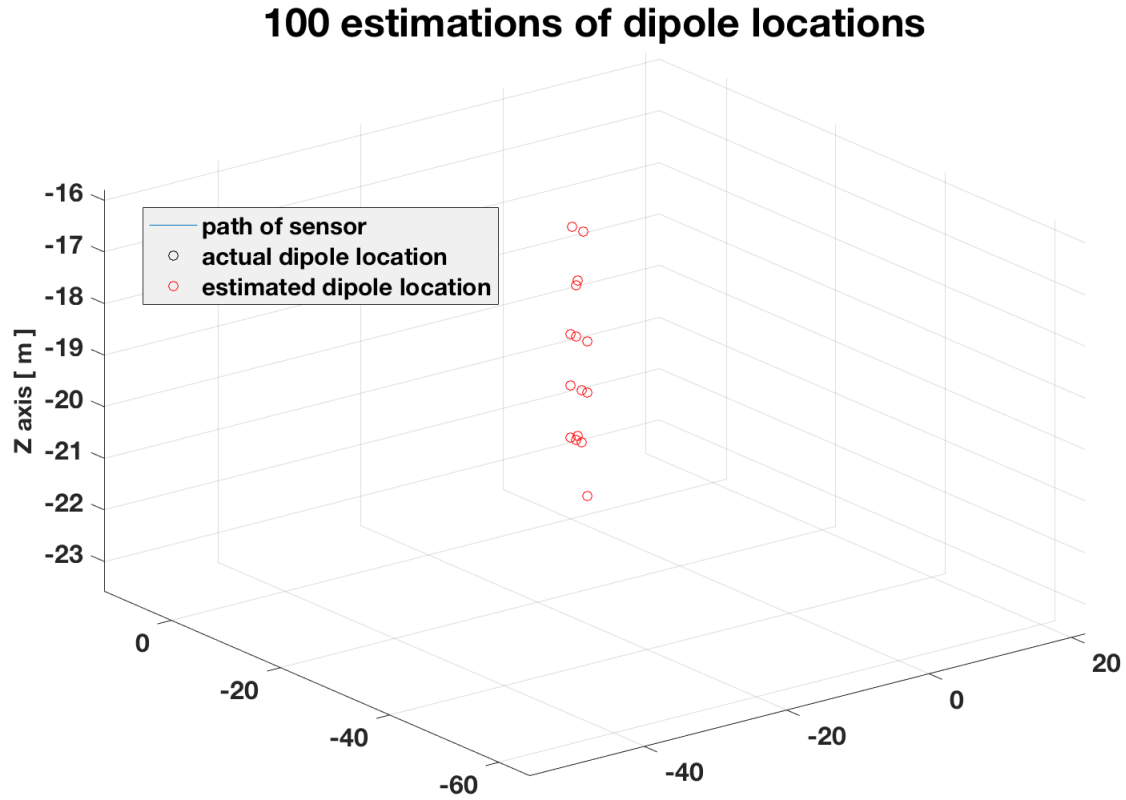


Figure 5.12: Zoomed in estimation.

The reason the estimations are grouped together in what appears to be different z planes, is that poslimis equal to 1, which means that different approximations all differ by 1 metre. The reason that the approximations appear to be closer to each other in the xy plane is that the scale of the xy axes is 10 times larger than that of the z axis. Another thing to note is that the zoomed in graph does not seem to show 100 approximations, but far less. This is due to the fact that the algorithm landed on the approximation a number of times, which is a good thing since it showcases the repeatability of the results, in spite of the stochastic base of the algorithm.

6. Swarm Implementation

The next objective is to use this Simulated Annealing algorithm with multiple sensors (the drones) at the same time, and to exploit the benefits of multiple sensor locations at the same time as much as possible.

6.1 Introduction

In the next sections, different methods are discussed to optimise the advantage of multiple sensor locations at the same time. Prior to discussing these different methods, it makes sense to sketch the setup and the algorithm to give the next section more context. Starting with the initial setup of this first test. The magnetic moment of the dipole remains the same as those in previous sections: $[-1000, 1300, -800]$ Am², but the position will be changed, to make the individual differences between the methods more clear: $[60, -20, -5]$ m. The search space for the drones has been altered as well, a factor 10 smaller in every direction: 100 m by 100 m by 10 m. This had to do with the real world as well, if the sample space had been 1000 m by a 1000 m by 100 m, the drones would have to find a signal in the order of 10^{-13} T at their starting positions, which no sensors are able to do at present. In the current set-up the signal of the dipole at the starting positions is somewhere in the order of 10^{-10} T, which would still go unexpected with the sensors currently available. To omit this problem, the drones are programmed in this example to fly the first step towards their average location, before they start estimating. Of course this could be altered to any arbitrary step. The noise setup was altered as well, instead of using a random noise based on a SNR, the noise was defined using a standard normal distribution. This as well to mimic reality better. The noise was defined in such a way that the 3σ value was in the order of 5-10 nT ($1 \text{ nT} = 10^{-9} \text{ T}$):

```
noise = [normrnd(0,2e-09) normrnd(0,2e-09) normrnd(0,2e-09)]
```

It was added to the pure signal as a three dimensional vector, to make the noise distribution the same in every direction. The parameter settings for Simulated Annealing are the same as in the previous example:

```
params = [50,0.1,75,-1000,1000,1,10,20]
```

As the reader might observe, some values have been changed relative to the parameters which produced the results shown in figure 5.11. This has been done to exaggerate the differences between the methods. The number of cycles was decreased from 200 to 75, this has been done to make the methods faster as well. The iterative process in which the drones find their way

through space is:

Algorithm 2: Drone control process

input : Space, dipole info, path lengths and starting positions of drones
output: Drone trajectories

- 1 **Set parameters:**
- 2 **step_no** = number of times the drones alter their course;
- 3 **drone_no** = number of drones;
- 4 **init_estp** = estimated location of dipole;
- 5 **init_estm** = estimated magnetic moment of dipole;
- 6 **drone_path** = array which will hold positions of all drones at each step at the end;
- 7 **rand_obs** = position of drone over which can be looped ;
- 8 **Choose initial path**
- 9 **for** $i = 1$ **to** $step_no$. **do**
- 10 **for** $j = 1$ **to** $drone_no$. **do**
- 11 add position **rand_obs** to **drone_path** of drone j at step i ;
- 12 **rand_obs** = last entry of path of drone;
- 13 **course** = $(init_estp - rand_obs) \div norm((init_estp - rand_obs))$;
- 14 **endofpath** = **rand_obs** + **course** · **pathlength** ;
- 15 **path_it** = path from **rand_obs** to **endofpath** ;
- 16 **source_field** = field along **path_it** ;
- 17 [**field_it**, **init_estp**, **est_m**] = SA(**params**, **poslim**, **init_estp**, **init_estm**, **source_field**,
 path_it);
- 18 **end**
- 19 **end**

The values chosen for implementation are mentioned in the Table 6.1:

Variable	Value
step_no	20
drone_no	5
init_estp	[20, 0, 0]
init_estm	[0, 0, 0]

Table 6.1: Variable choice

The starting positions of the drones, together forming the **rand_obs** , array were: [100, 100, 10] m; [-100, -100, -10] m; [-100, 100, 10] m; [100, -100, -10] m and [100, 0, 0] m. In Figure 6.1 the **drone_path** array is plotted. The fact that drones are starting from 10 metres below the water surface has been done for demonstrative purposes. Now from the graph it certainly looks like the drones are roughly moving towards the same spot, but it is hard to see exactly.

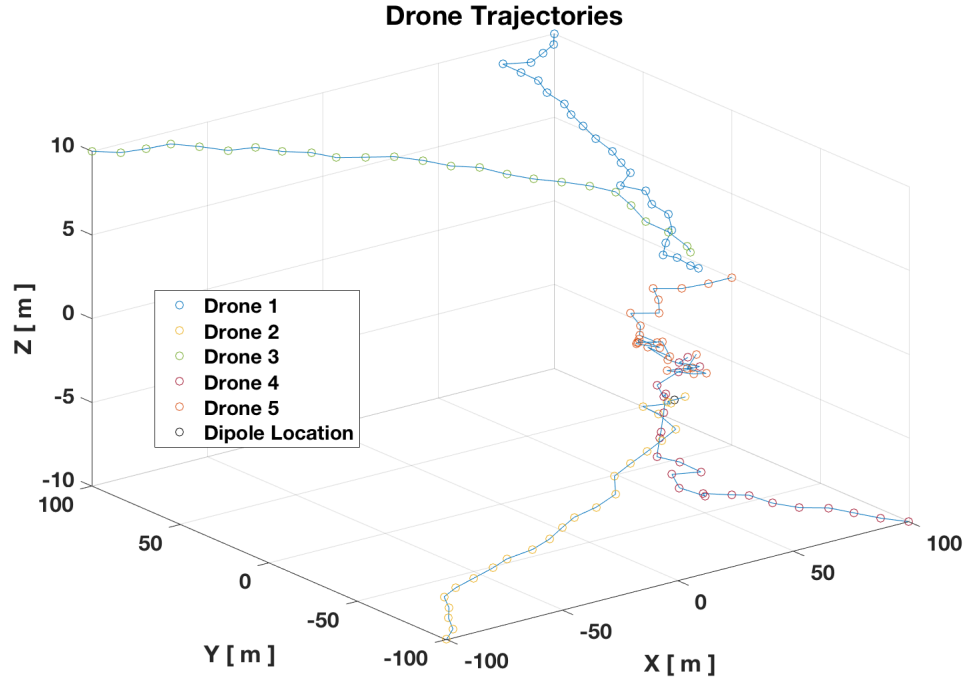
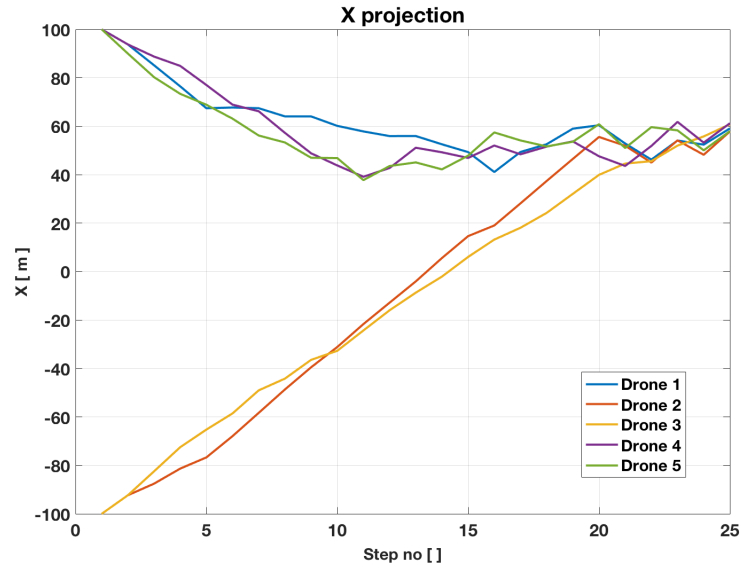
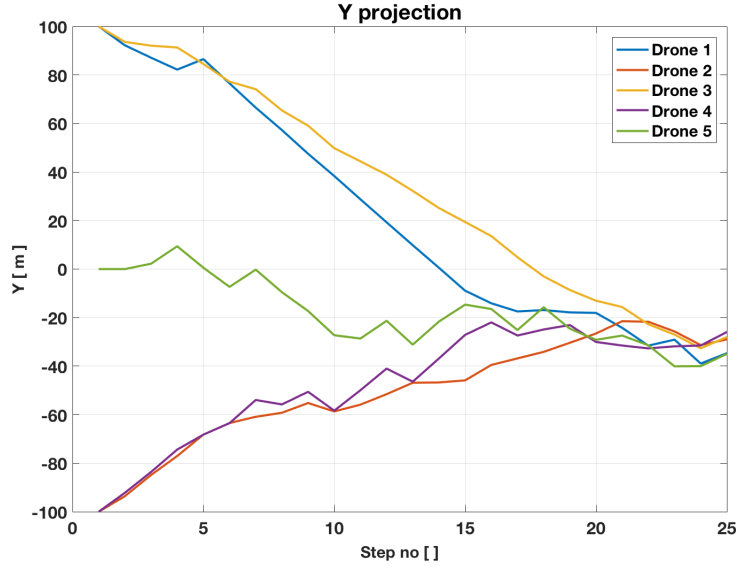
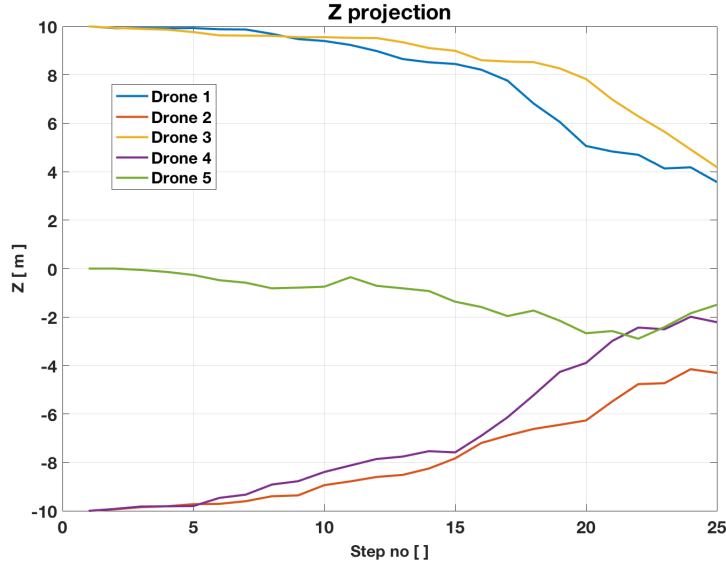


Figure 6.1: Drone trajectories

To see the drone convergence towards the dipole more clearly, each of the components of every drone has been plotted as well in figure 6.2 to 6.4:

Figure 6.2: x -component of path travelled by drones

Figure 6.3: y -component of path travelled by dronesFigure 6.4: z -component of path travelled by drones

The component plots show that the drones are definitely moving in the right direction, although they do not necessarily end in the right spot. Especially in the z direction the motion towards the dipole is not particularly convincing. This is why some methods have been applied to exploit the benefits of multiple sensors in different planes.

6.2 Method 1: best guess method

One of these methods compares the guesses of each of the drones, thus finding increasingly better approximations of the location of the dipole. The energy function is of use again here, since it provides a method to compare different approximations with each other, just as in Simulated

Annealing itself. During the implementation of this method however, one problem arose. Since the drones still start at considerable distances from the dipole, unlike in the example in figure 5.1, the magnetic field values differ a lot, as mentioned earlier as well. In other words, the magnetic dipole model is very space dependent. This messes with the perceived energy levels of the different states. There were cases that while the approximation was actually better, the energy was a lot higher. This has to do with the fact that the energy is relative to the number of sample points, but not to the order of magnitude of the signal. So while the absolute error might be bigger, the approximation might well be a lot better, since the drone would be closer to the dipole, and would sense much higher magnetic field values. These higher magnetic field values cause a higher absolute error. There are two ways to solve this complication, only one will be put into action here. The one left unexplored will be explained more elaborately in the recommendations section. The method used is to multiply the magnetic field model by a value which takes out some of that space dependency (the space dependency is caused by the factors $\frac{1}{\|r\|^5}$ and $\frac{1}{\|r\|^3}$ in equation 3.64). Multiplying the dipole model with a factor $\frac{1}{\|r\|^3}$ solved this problem to a very good extent. To apply the method of saving the best approximation, the following code had to be added before the first 'end' statement in algorithm 2:

```
current = energy(field_it,source_field);
if current < best_energy
    best_energy = current;
    bestp = init_estp;
    bestm = init_estm;
end
```

The energy function used is the one defined in algorithm 1. The course defining line in algorithm 2 would have to be changed to:

```
course = (bestp-init_estp)/norm(bestp-init_estp)
```

Where the `norm` function calculates the Euclidean norm of the difference vector between the approximation and the current location. This produces a path much more stable than the paths in the initial run of the setup. The drones flying in figure 6.1 altered their course at every step, since they made the approximation again every step. In this new case, all the drones are being updated with the best approximation all the time, as a result they have to alter their course a lot less. The table below showcases the process of coming to the best approximation:

no.	best guess	best energy
1	[26, 12, 3]	9.0494e-10
2	[48, 8, -7]	3.8907e-12
3	[60, -50, -7]	3.1901e-12
4	[59, -18, -5]	2.5222e-12

Table 6.2: Best guesses and their energies (mean squared errors)

The algorithm had to adjust its approximation 4 times to come to a reasonably close approximation. This means that all the drones had to change their course 4 times as well, whereas the drones in the example without a method all had to change their course 20 times, at every step.

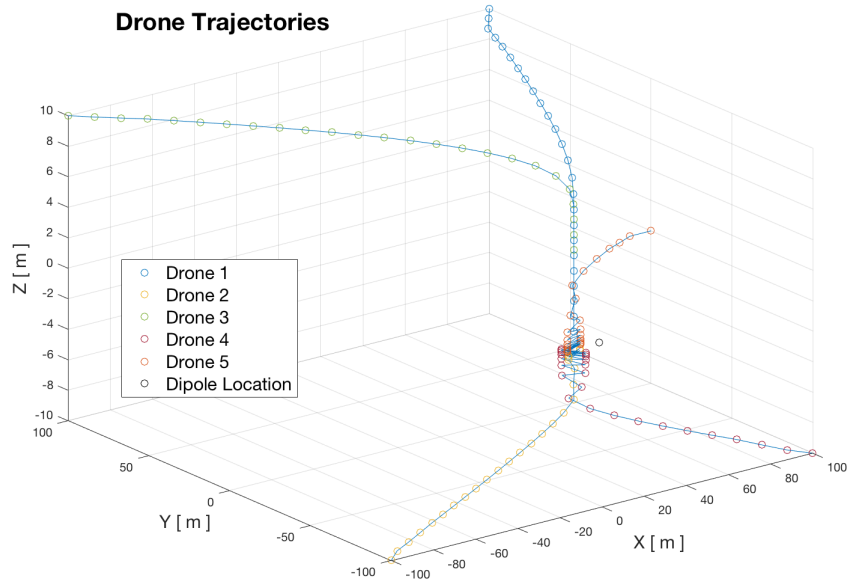
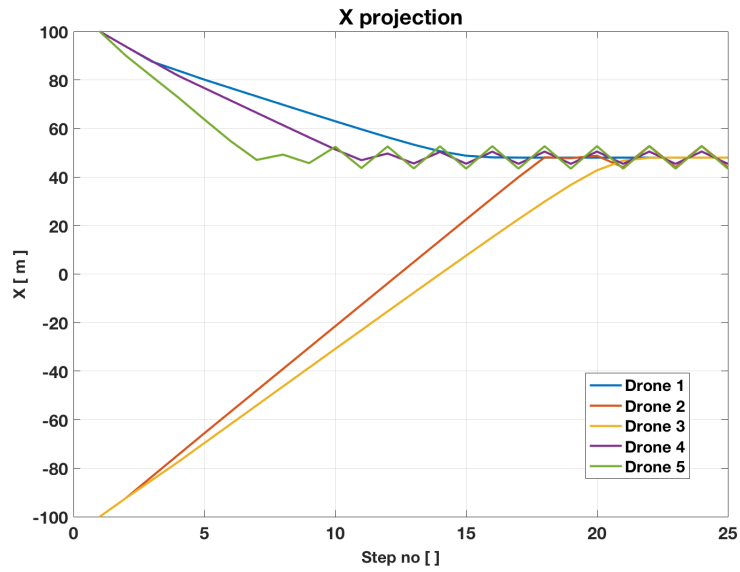
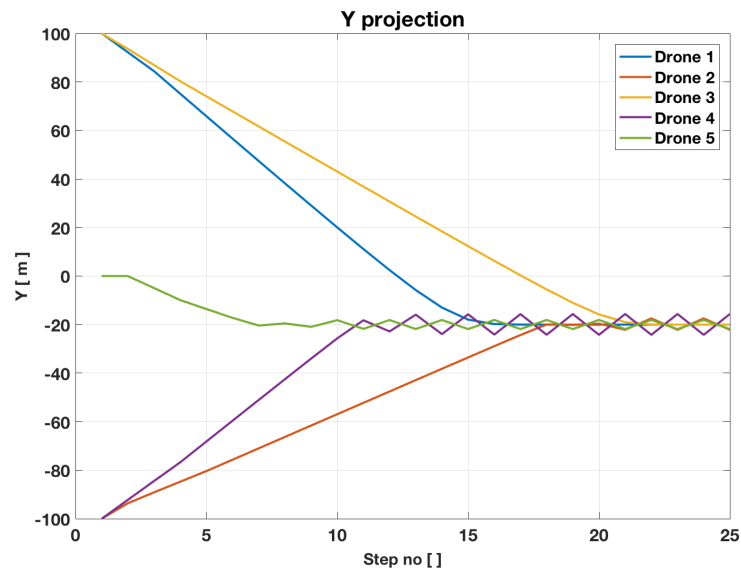
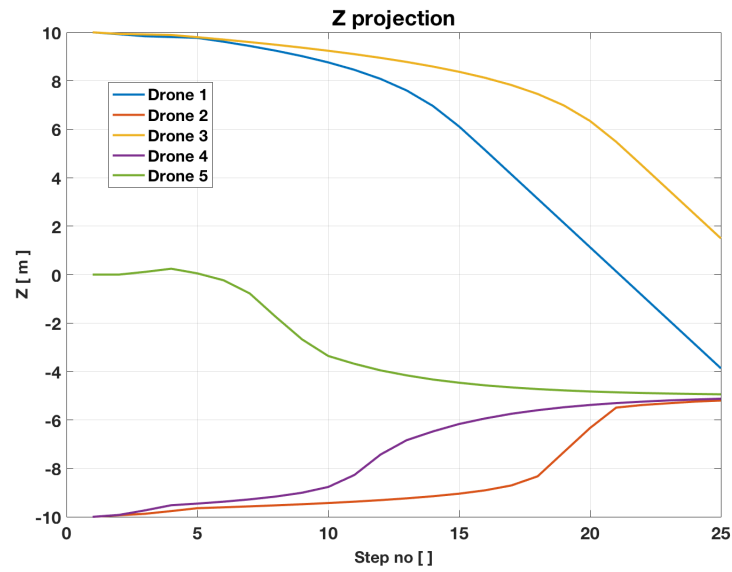


Figure 6.5: Drone trajectories

Figure 6.6: x -component of path travelled by drones

Figure 6.7: y -component of path travelled by dronesFigure 6.8: z -component of path travelled by drones

6.3 Method 2: average guess method

Another way to take better advantage of multiple sensors is to look at their average estimation, to rule out some of the stochastic fluctuations of the algorithm. The stochastic fluctuations are random by nature, so they should cancel out if the average is taken, which would make the algorithm perform more consistently. For the average to be implemented, all estimations of each drone must be saved at every course alteration. The following code snippets were added:

```
allp(:,j) = init_estp ;
allm(:,j) = init_estm ;
```

These two lines had to be added in the innermost loop to save all the estimations side by side. Now after the first end, in the outer loop, the average will be calculated, using this code:

```
avgp = mean(allp);
avgm = mean(allm);
```

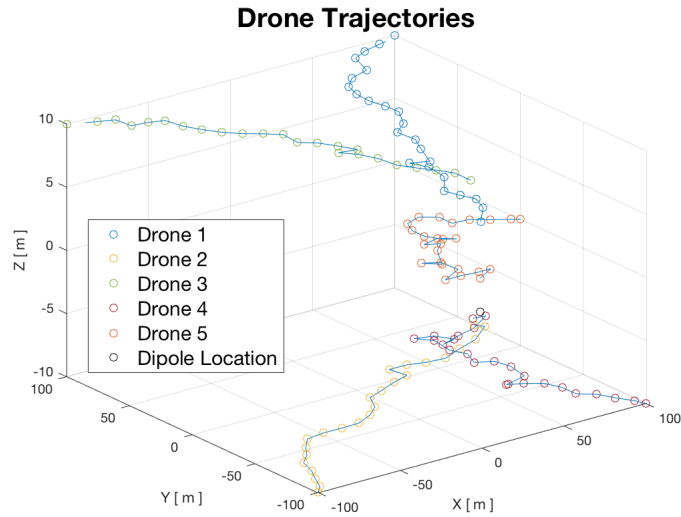
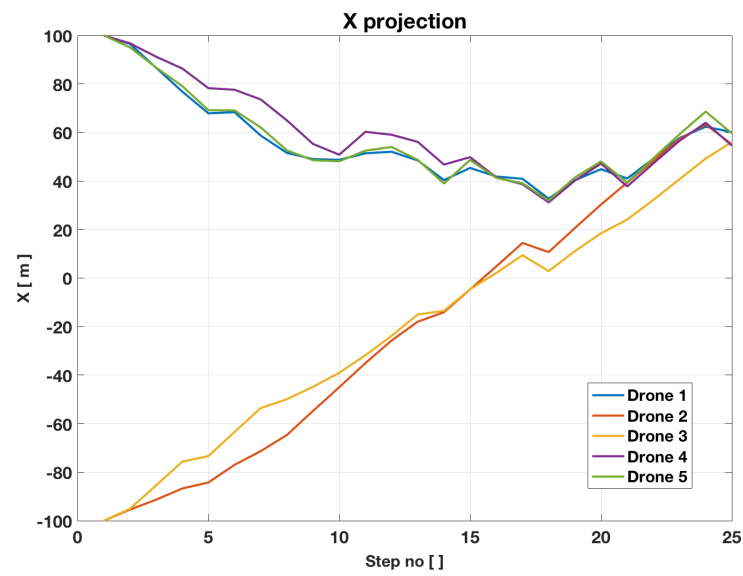
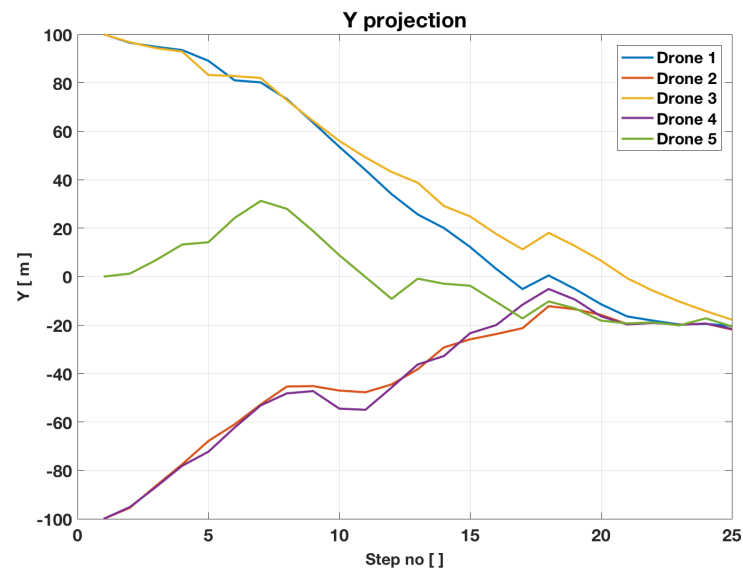


Figure 6.9: Drone trajectories

Figure 6.10: x -component of path travelled by dronesFigure 6.11: y -component of path travelled by drones

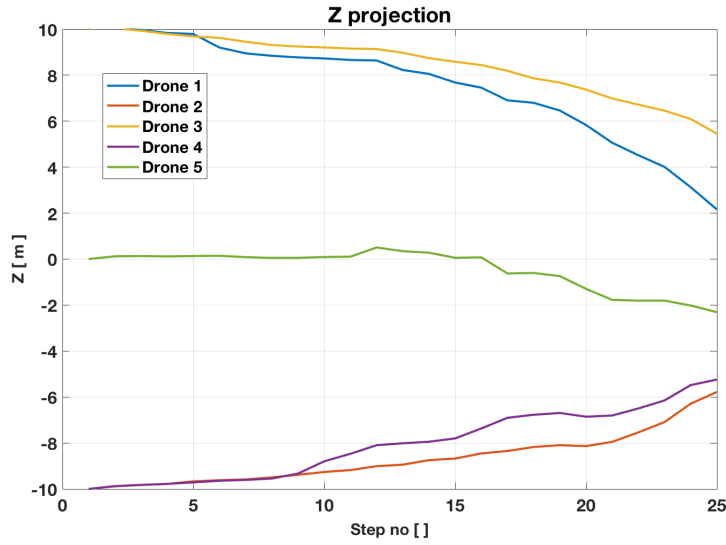


Figure 6.12: z -component of path travelled by drones

Using the average of the estimations of the 5 sensors means that the drones have to alter their course 20 times as well, as in the first example. However, the average is a much better prediction for the location of the dipole than all estimations individually. This has to do with the fact that the algorithm is stochastically based, which means that the expectation of the error of the estimations is 0. This difference is most clear in the x and y direction compared to the x and y graphs in Figures 6.2 and 6.3.

6.4 Method 3: total signal method

A third and last method would be to collect all of the sensor data of the drones and combine these signals into one longer signal, and to perform SA on this longer signal. Since there are more segments in the longer signal to find a fit to, in theory more information could be extracted from the signal. Two cases have been further developed, one where the best approximation wasn't saved, and another where it was saved analogous to section 6.2. Instead of plotting the 3D plot, the final signals are plotted, to give an idea of how the segments link up to make a longer signal.

6.4.1 Method 3.1: without best guess

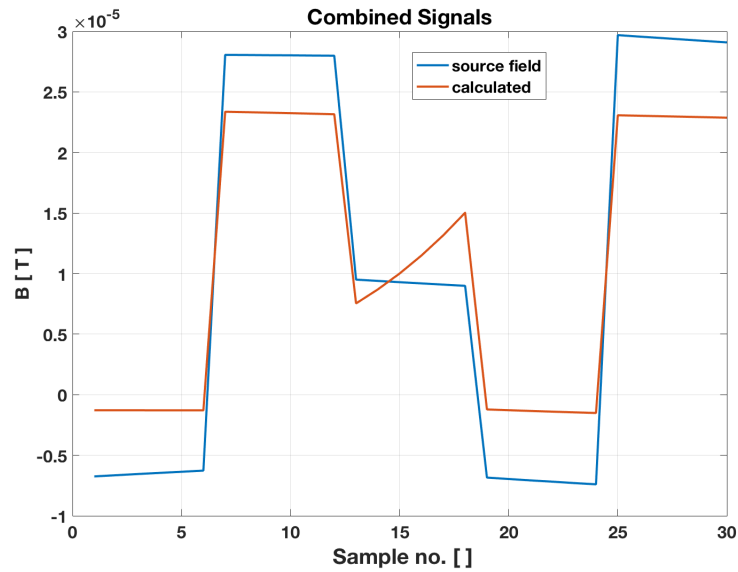
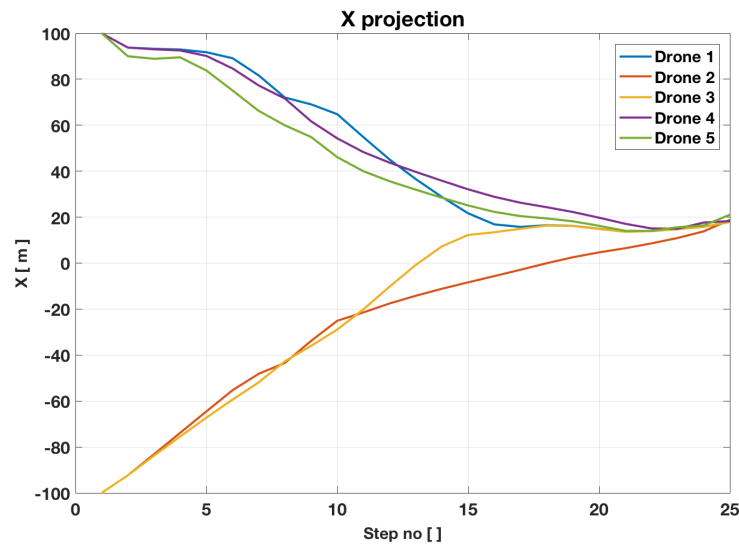
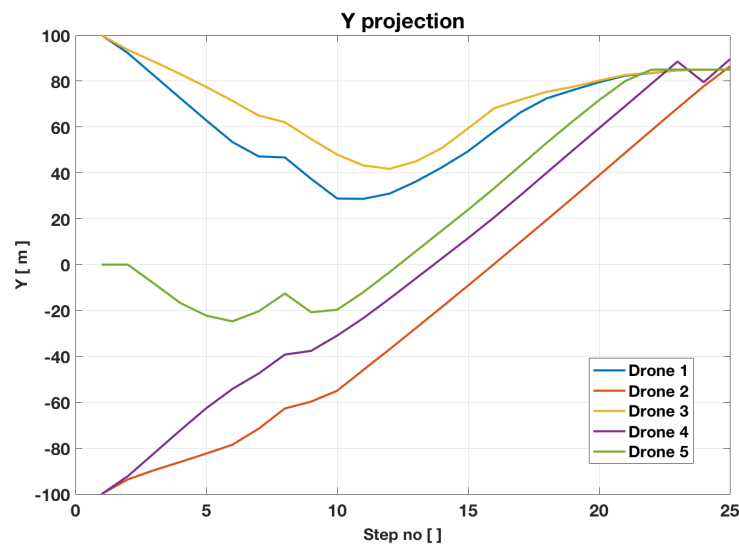
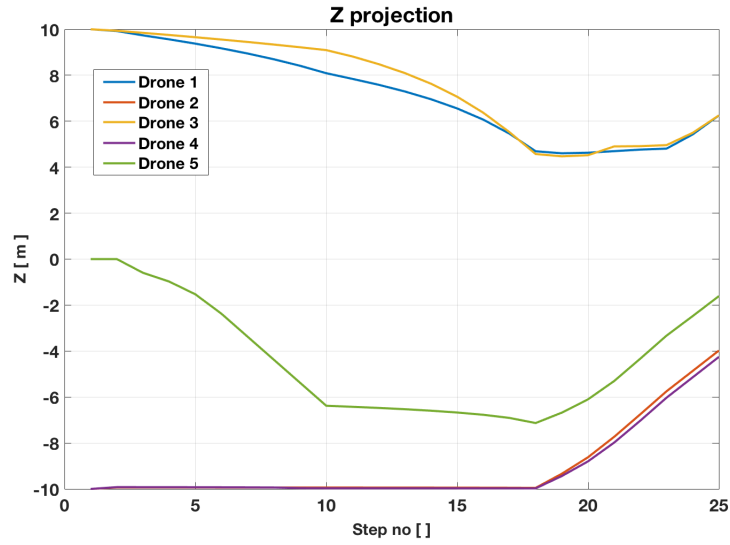


Figure 6.13: Total field signal along the last leg of the path measured and estimated by the 5 drones

Figure 6.14: x -component of path travelled by dronesFigure 6.15: y -component of path travelled by drones

Figure 6.16: z -component of path travelled by drones

So far this method hasn't produced as convincing results as the other two, this may have to do with the fact that all the different signals from the drones vary in order of magnitude as well, but this is just a conjecture. This definitely needs some further research, since it may well be the most promising method of the three.

6.4.2 Method 3.2: with best guess

One might wonder why the signals in Figures 6.13 and 6.17 differ a factor 10. The fact that combined signal in Figures 6.13 and 6.17 differ a factor 10 has to do with the factor added to minimise the space dependency to better assess the energies of different states, this causes the above signal to be 10 times stronger than the signal in figure 6.13.

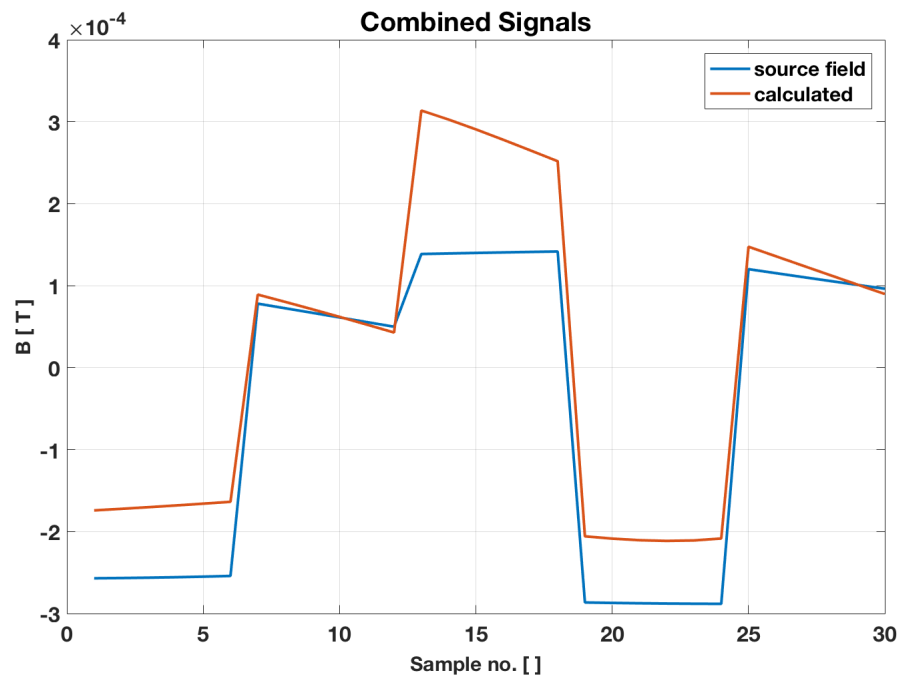


Figure 6.17: Total field signal along the last leg of the path measured and estimated by the 5 drones

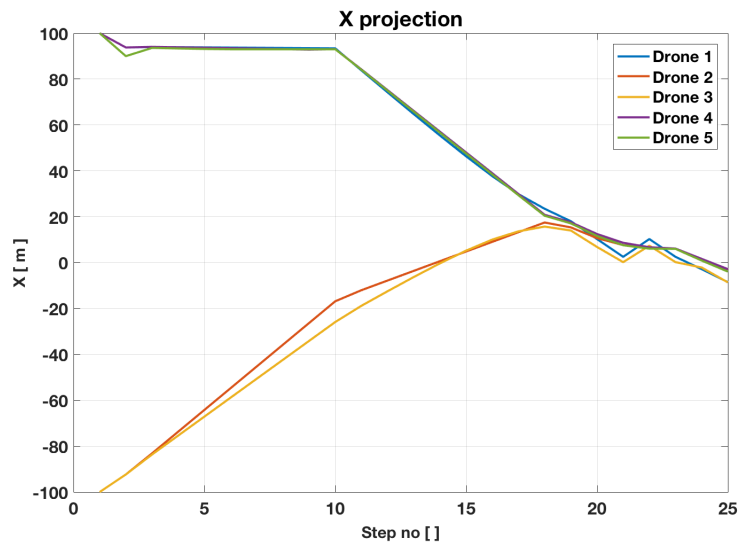
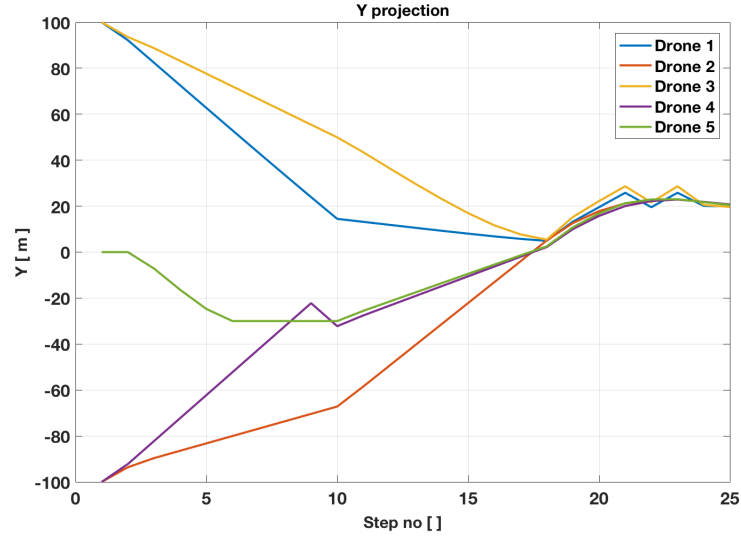
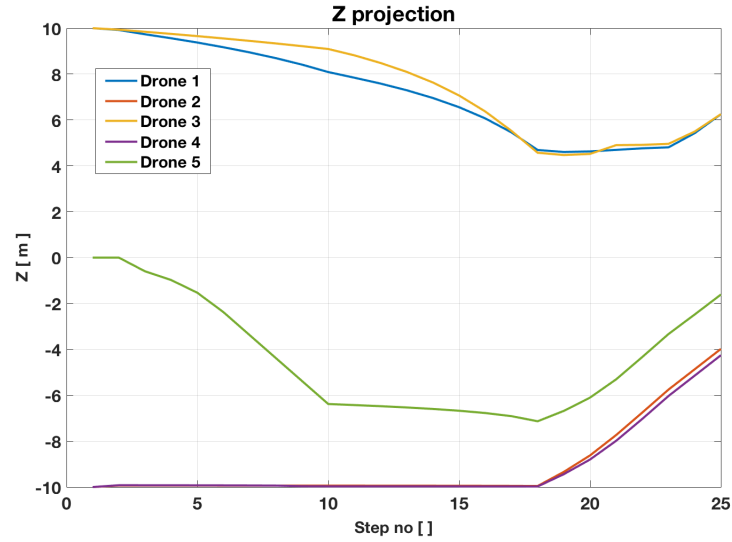


Figure 6.18: x -component of path travelled by drones

Figure 6.19: y -component of path travelled by dronesFigure 6.20: z -component of path travelled by drones

As shown in the above plot, combining the total signal method with the best guess method, produces slightly better results, but still not as good as the best guess or average method on their own.

6.5 Twin Experiment

The concept of adding noise into the equation has been mentioned before as well, but not in light of a twin experiment. In a twin experiment, one obtains two sets of data, one with noise, and another without noise. The same methods are applied to both data sets, and afterwards the results are compared. In this setup however it is quite hard to add the noise in a realistic manner. Since the set-up deals with large distances from the dipole location, the magnetic field

values vary a factor 10^3 from the starting position of the drones to the point where they are very close to the dipole. This makes it hard to test the setup rigorously since noise is not space dependent, while the magnetic field signal is. This is a reflection of reality, which in itself is a good thing, since it must be as realistic as possible, however the dimensions of space chosen in figure 5.1, will not work in this application. This is why the space has been adjusted for this implementation to 100m x 100m x 10m to reflect the real world better.

6.6 Comparison of the localisation methods

Multiple methods have been discussed, and although there might not be one of them which is absolutely best, it does make sense to compare them to each other. The table below summarises their differences:

Method	Number of course alterations	B -field used	Positional error	Magnetic moment error
Initial	drone_no*step_no	normal	0.972e03	1.380e06
Best guess	depends	amplified	0.264e03	1.622e06
Average	step_no	normal	0.544e03	9.455e05
Total signal	step_no	normal	2.023e03	1.401e06
Total signal with best guess	depends	amplified	3.371e03	2.524e06

Table 6.3: Comparison of the methods.

The number of course alterations for the best guess method depends on how often the best estimate is altered. Because SA is stochastically based, this will differ a bit. Comparing the energy of each method does not make much sense, since the methods using a best guess will have amplified errors because of the amplified **B**-field. A better comparison is to take the mean squared error of the estimations of position and magnetic moment and the actual values. In real life this is not be possible, since the position and magnetic moment of the dipole would be unknown, but for comparison in this thesis it is very useful.

As shown in Table 6.3 the method with the best positional estimation is the best guess method. As mentioned earlier, the total signal method still had some trouble with estimating the properties of the dipole. This probably has to do with the fact that some drones were closer than others. Since they are closer their relative error is much less, but their absolute error may be multiple times higher than the error of drones further away. To apply the total signal method optimally the errors should be scaled to the signal amplitude, something talked about earlier. The method of amplifying the signal and making it less space dependent talked about earlier, did not work in this setting.

7. Assessment

7.1 Conclusion

In conclusion, the Simulated Annealing algorithm works very reliable under certain conditions. It proved to perform reasonably well in the case of the drone control process. If the drones could bring down the search space for a shipping container from a couple hundred metres in every direction to a couple of meters in every direction this is a useful result. By comparing the estimations of the methods in section 6.6 it became clear that at the moment the best guess method was best in estimating position. The method which averages the estimations turned out to be the best in estimating the magnetic moment of the dipole.

Another conclusion which may be drawn is that certain parameters have a considerable influence on the functioning of the algorithm. This is important to consider since it will influence the outcome of the drone search a lot, especially since the SA algorithm is called a lot during every cycle. All in all, two of the three methods proposed produced results which were better than the drone search presented in section 6.1, without using any information sharing. In real-life the less the drones have to alter their course the more efficient their search for a shipping container is. This is exactly what the average and the best guess methods achieved.

7.2 Discussion

Everything discussed in this thesis is based upon a simulation, and not real-world data. This is one of the main setbacks for checking if everything that was thought of here is actually possible for today's sensors and drones. Most of the parameters chosen, such as the number of data-points taken over a given path, were deduced from other articles where they seemed to produced favourable results, but it is not sure that in the real world all these parameters would produce a desirable outcome as is. The SA algorithm for instance could turn out to be too slow to be of any meaning in altering the course of a drone.

The total signal method is not working properly the way it has been implemented at the moment, but with more time the problem of wrongly scaled errors could definitely be solved. The method has been mentioned because it is a very promising method.

Another thing that could have improved the thesis is the control of the SA algorithm. Since there are many parameters influencing its working, it was a bit of struggle at times to find the perfect settings. More research into this topic maybe could have helped achieve even better results. Apart from the alpha analysis, efficiency analyses could have been performed on the parameters `mres` and `posres`. These parameters determine the amount of distance the algorithm can maximally cover in one cycle. It varies a lot if these parameters are altered a factor 10.

7.3 Recommendation

In section 6.2 a second method to tackle the space dependency of the drone process was briefly mentioned. It is worth explaining it more elaborately here, in light of potential future research into this matter. Another way to address the space dependency would be to look at the ratio of the maximum of a signal versus its mean value. As shown in the graphs in Figure 5.2 to 5.4, if the sensor location is close to the location of the dipole (in the order of 10-50 metres) the graph shows a sharp spike at the location of the dipole. This property could be exploited in a way that the drones would fly towards their average position, until the ratio between the maximum or minimum value of the signal reaches a certain level. The premise of this approach is that all drones are roughly equally distributed around the dipole location, this however is not unrealistic to presume. Another premise would be that the drones should fly a path roughly perpendicular to the dipole location, to generate a clear spike, this is something which is harder to implement. It is harder, since the symmetries discussed in section 3.3 could play a role. A number of tests pointed out that as the sensor location gets closer and closer to the dipole, the ratio of the absolute value relative to the mean of the signal did increase rapidly. In light of the time given however this method was left untapped.

Instead of the average being calculated at each step in the path for all drones, a running average could be taken as well, in which case the location would be updated more frequently, thus accounting for possible changes in the position of the dipole. Aside from the running average, the average or running average method could also be combined with the total signal method, maybe that this would work better than the best approximation.

There are a number of other premises which could be made, but for which there was no time to implement them into the drone control process. If it is known at forehand that the drones are flying above sea level, and the shipping container which they are looking for is located underwater, a number of simplifications could be made of course. If the drones are looking for something on the seabed, the position could even be fixed to the known depth of the sea floor in that area, which would reduce the computation time of the process.

SA has so many parameters influencing its working –think of the analysis of alpha– that in the span of this project not all optimal values for the parameters in every situation could have been found. It would be very interesting to continue with the analysis of certain parameters with a noteworthy influence on the working of SA. This to make sure the algorithm is best equipped with the most suitable parameters for every implementation of it.

As mentioned in the discussion, another recommendation would be to test everything discussed in this thesis to real-world values. This is the only way to find out if everything proposed is even feasible. In this case, it would be wise to let the drones hover in the same z plane above the location of the shipping container under the sea level, since otherwise they would probably crash into the sea. Another idea was to use a combination of drones underwater and drones flying above sea level. Although the underwater drones probably won't cover as much ground as the flying drones, they would offer an extra sensory dimension. Simulated Annealing could be adjusted in such a way that the flying drones would do a quick crude estimation by covering much ground in a short period of time, whilst the underwater drones would go for the accuracy of pinpointing the shipping container to its exact location. One thing is for sure, this project really appeals to ones imagination and there are many directions to move on with this project, likely with some very exciting outcomes.

Bibliography

- [1] Unknown. Alleen globale inhoud containers msc zoe in openbaarheid. *Omrop Fryslân*, 2019.
- [2] X.; Direkogly C.; Hurley D.J. Nixon, M.S.; Liu. On using physical analogies for feature and shape extraction in computer vision. *The Computer Journal*, 54(1):11–25, 2011.
- [3] StackExchange. Curl vector: What exactly is rotating? <https://math.stackexchange.com/questions/1054132/curl-vector-what-exactly-is-rotating>.
- [4] T. Whyntie. A magnetic dipole and a magnetic monopole (figure). <http://doi.org/10.5281/zenodo.215520>, December 2016.
- [5] A.R.P.J. Vijn. Inverse modeling for magnetic signature monitoring of naval ships. 2016.
- [6] Arie Sheinker, Boaz Lerner, Nizan Salomonski, Boris Ginzburg, Lev Frumkis, and Ben-Zion Kaplan. Localization and magnetic moment estimation of a ferromagnetic target by simulated annealing. *Measurement Science and Technology*, 18:3451, Sep 2007.
- [7] A.C. Lumadjeng. Determination of the magnetic background field on naval ships. August 2017.
- [8] H.A. Haus and J.R. Melcher. *Electromagnetic Fields and Energy*. Massachusetts Institute of Technology, Cambridge, MA, USA, 1998.
- [9] StackExchange. What does forward modelling mean? <https://astronomy.stackexchange.com/questions/19687/what-does-forward-modeling-mean>.
- [10] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Proceedings of ICNN'95 - International Conference on Neural Networks*, volume 4, pages 1942–1948 vol.4, 1995.

A. Derivation of equation (3.61) to (3.62)

First note that $\nabla \cdot \mathbf{r}$ (a scalar value) and $\mathbf{r} \cdot \nabla$ (linear combination of derivative operators) are two different expressions. Important for this derivation is the definition of ∇ :

$$\nabla = \begin{bmatrix} \frac{\partial}{\partial r_x} \\ \frac{\partial}{\partial r_y} \\ \frac{\partial}{\partial r_z} \end{bmatrix} \quad (\text{A.1})$$

Let's define $\mathbf{m}$ and $\mathbf{r}$ as:

$$\mathbf{m} = \begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix}, \quad \mathbf{r} = \begin{bmatrix} r_x \\ r_y \\ r_z \end{bmatrix} \quad (\text{A.2})$$

The definition of $\mathbf{m} \cdot \nabla$ thus becomes:

$$\mathbf{r} \cdot \nabla = \left(m_x \cdot \frac{\partial}{\partial r_x} + m_y \cdot \frac{\partial}{\partial r_y} + m_z \cdot \frac{\partial}{\partial r_z} \right) \quad (\text{A.3})$$

Then expression (3.61) becomes:

$$\mathbf{H} = - \left[(\mathbf{m} \cdot \nabla) \cdot \frac{\mathbf{r}}{4\pi ||\mathbf{r}||^3} \right] \quad (\text{A.4})$$

$$= -\frac{1}{4\pi} \left[\left(\begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix} \cdot \nabla \right) \cdot \frac{\begin{bmatrix} r_x \\ r_y \\ r_z \end{bmatrix}}{(r_x^2 + r_y^2 + r_z^2)^{\frac{3}{2}}} \right] \quad (\text{A.5})$$

$$= -\frac{1}{4\pi} \left[\left(m_x \cdot \frac{\partial}{\partial r_x} + m_y \cdot \frac{\partial}{\partial r_y} + m_z \cdot \frac{\partial}{\partial r_z} \right) \cdot \begin{bmatrix} \frac{r_x}{(r_x^2 + r_y^2 + r_z^2)^{\frac{3}{2}}} \\ \frac{r_y}{(r_x^2 + r_y^2 + r_z^2)^{\frac{3}{2}}} \\ \frac{r_z}{(r_x^2 + r_y^2 + r_z^2)^{\frac{3}{2}}} \end{bmatrix} \right] \quad (\text{A.6})$$

For clarifying, first the first row of this vector will be calculated, the other rows are calculated analogous to this first row. Calculating the first row of the vector, using the product rule for

differentiation obtains:

$$= \left(m_x \cdot \frac{\partial}{\partial r_x} + m_y \cdot \frac{\partial}{\partial r_y} + m_z \cdot \frac{\partial}{\partial r_z} \right) \cdot \frac{r_x}{(r_x^2 + r_y^2 + r_z^2)^{\frac{3}{2}}} \quad (\text{A.7})$$

$$= m_x \cdot \frac{\partial}{\partial r_x} \left(\frac{r_x}{(r_x^2 + r_y^2 + r_z^2)^{\frac{3}{2}}} \right) + m_y \cdot \frac{\partial}{\partial r_x} \left(\frac{r_x}{(r_x^2 + r_y^2 + r_z^2)^{\frac{3}{2}}} \right) + m_z \cdot \frac{\partial}{\partial r_x} \left(\frac{r_x}{(r_x^2 + r_y^2 + r_z^2)^{\frac{3}{2}}} \right) \quad (\text{A.8})$$

$$= m_x \left(\frac{1}{(r_x^2 + r_y^2 + r_z^2)^{\frac{3}{2}}} - \frac{3r_x^2}{(r_x^2 + r_y^2 + r_z^2)^{\frac{5}{2}}} \right) - m_y \cdot \frac{3r_x r_y}{(r_x^2 + r_y^2 + r_z^2)^{\frac{5}{2}}} - m_z \frac{3r_x r_z}{(r_x^2 + r_y^2 + r_z^2)^{\frac{5}{2}}} \quad (\text{A.9})$$

Calculating the other two rows is analogue to the first, and so (A.6) becomes one large vector (with the substitution $\sqrt{r_x^2 + r_y^2 + r_z^2} = ||\mathbf{r}||$):

$$\mathbf{H} = -\frac{1}{4\pi} \begin{bmatrix} m_x \left(\frac{1}{||\mathbf{r}||^3} - \frac{3r_x^2}{||\mathbf{r}||^5} \right) - m_y \cdot \frac{3r_x r_y}{||\mathbf{r}||^5} - m_z \frac{3r_x r_z}{||\mathbf{r}||^5} \\ -m_x \frac{3r_y r_x}{||\mathbf{r}||^5} + m_y \left(\frac{1}{||\mathbf{r}||^3} - \frac{3r_y^2}{||\mathbf{r}||^5} \right) - m_z \frac{3r_y r_z}{||\mathbf{r}||^5} \\ -m_x \frac{3r_z r_x}{||\mathbf{r}||^5} - m_y \frac{3r_z r_y}{||\mathbf{r}||^5} + m_z \left(\frac{1}{||\mathbf{r}||^3} - \frac{3r_z^2}{||\mathbf{r}||^5} \right) \end{bmatrix}$$

If we rearrange this it becomes:

$$\mathbf{H} = -\frac{1}{4\pi} \begin{bmatrix} \frac{m_x}{||\mathbf{r}||^3} - m_x \frac{3r_x^2}{||\mathbf{r}||^5} - m_y \frac{3r_x r_y}{||\mathbf{r}||^5} - m_z \frac{3r_x r_z}{||\mathbf{r}||^5} \\ \frac{m_y}{||\mathbf{r}||^3} - m_x \frac{3r_y r_x}{||\mathbf{r}||^5} - m_y \frac{3r_y^2}{||\mathbf{r}||^5} - m_z \frac{3r_y r_z}{||\mathbf{r}||^5} \\ \frac{m_z}{||\mathbf{r}||^3} - m_x \frac{3r_z r_x}{||\mathbf{r}||^5} - m_y \frac{3r_z r_y}{||\mathbf{r}||^5} - m_z \frac{3r_z^2}{||\mathbf{r}||^5} \end{bmatrix}$$

Now, the first terms of each row together form the vector $\frac{\mathbf{m}}{||\mathbf{r}||^3}$:

$$\mathbf{H} = -\frac{1}{4\pi} \left(\frac{\mathbf{m}}{||\mathbf{r}||^3} + \frac{3}{||\mathbf{r}||^5} \begin{bmatrix} -m_x r_x^2 - m_y r_x r_y - m_z r_x r_z \\ -m_x r_y r_x - m_y r_y^2 - m_z r_y r_z \\ -m_x r_z r_x - m_y r_z r_y - m_z r_z^2 \end{bmatrix} \right)$$

Consider that:

$$(\mathbf{m} \cdot \mathbf{r}) = m_x r_x + m_y r_y + m_z r_z \quad (\text{A.10})$$

Rewrite $\mathbf{H}$:

$$\mathbf{H} = -\frac{1}{4\pi} \left(\frac{\mathbf{m}}{||\mathbf{r}||^3} + \frac{3}{||\mathbf{r}||^5} \begin{bmatrix} -r_x(m_x r_x + m_y r_y + m_z r_z) \\ -r_y(m_x r_x + m_y r_y + m_z r_z) \\ -r_z(m_x r_x + m_y r_y + m_z r_z) \end{bmatrix} \right) = \frac{1}{4\pi} \left[\frac{3(\mathbf{m} \cdot \mathbf{r})\mathbf{r}}{||\mathbf{r}||^5} - \frac{\mathbf{m}}{||\mathbf{r}||^3} \right]. \quad (\text{A.11})$$

List of Figures

2.1	The different divergence scenarios for a vector field. [2]	13
2.2	The different curl scenarios in a vector field. [3]	13
2.3	Schematic representation of a classic permanent magnet. [4]	15
3.1	Spatial clarification of the vectors.	18
3.2	Contour plot of the x -component of a dipole in $[0, 1, 0]$ m.	24
3.3	Contour plot of the y -component of a dipole in $[0, 1, 0]$ m.	25
3.4	Contour plot of the z -component of a dipole in $[0, 1, 0]$ m.	25
3.5	z component of magnetic induction field along the x axis.	26
4.1	Flow chart of forward/inverse modelling. [5]	27
4.2	The test set-up in Sheinker's paper. [6]	28
5.1	Sensor trajectory with estimation and actual dipole location.	33
5.2	x -component of the magnetic induction field.	34
5.3	y -component of the magnetic induction field.	35
5.4	z -component of the magnetic induction field.	35
5.5	x -component of location estimate for different values of alpha	37
5.6	y -component of location estimate for different values of alpha	38
5.7	z -component of location estimate for different values of alpha	38
5.8	x -component of magnetic moment estimate for different values of alpha	39
5.9	y -component of magnetic moment estimate for different values of alpha	39
5.10	z -component of magnetic moment estimate for different values of alpha	40
5.11	100 estimations of the dipole location by SA.	41
5.12	Zoomed in estimation.	42
6.1	Drone trajectories	45
6.2	x -component of path travelled by drones	45
6.3	y -component of path travelled by drones	46
6.4	z -component of path travelled by drones	46
6.5	Drone trajectories	48
6.6	x -component of path travelled by drones	48
6.7	y -component of path travelled by drones	49
6.8	z -component of path travelled by drones	49
6.9	Drone trajectories	50
6.10	x -component of path travelled by drones	51
6.11	y -component of path travelled by drones	51

6.12	z -component of path travelled by drones	52
6.13	Total field signal along the last leg of the path measured and estimated by the 5 drones	53
6.14	x -component of path travelled by drones	54
6.15	y -component of path travelled by drones	54
6.16	z -component of path travelled by drones	55
6.17	Total field signal along the last leg of the path measured and estimated by the 5 drones	56
6.18	x -component of path travelled by drones	56
6.19	y -component of path travelled by drones	57
6.20	z -component of path travelled by drones	57

List of Tables

2.1	Declaration of variables and their units.	11
3.1	Variables defining the dipole model.	17
5.1	Declaration of parameters.	36
5.2	Declaration of variables.	36
6.1	Variable choice	44
6.2	Best guesses and their energies (mean squared errors)	47
6.3	Comparison of the methods.	58

List of Algorithms

1	Simulated annealing.	31
2	Drone control process	44