

PERMANENT MAGNET DESIGN AND IMAGE RECONSTRUCTION
TECHNIQUE FOR MAGNETIC RESONANCE IMAGING IN
INHOMOGENEOUS MAGNETIC FIELDS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

HÜSEYİN YİĞİTLER

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

SEPTEMBER 2006

Approval of the Graduate School of Natural and Applied Sciences.

Prof. Dr. Canan ÖZGEN
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Prof. Dr. İsmet ERKMEN
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Prof. Dr. B. Murat EYÜBOĞLU
Supervisor

Examining Committee Members

Prof. Dr. M. Kemal Leblebicioğlu (METU,EEE) _____

Prof. Dr. B. Murat EYÜBOĞLU (METU,EEE) _____

Prof. Dr. Adnan KÖKSAL (Hacettepe Un., EEE) _____

Prof. Dr. Nevzat G. Gencer (METU,EEE) _____

Assoc. Prof. Dr. Sencer Koc (METU,EEE) _____

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name : Hüseyin YİĞİTLER

Signature :

ABSTRACT

PERMANENT MAGNET DESIGN AND IMAGE RECONSTRUCTION TECHNIQUE FOR MAGNETIC RESONANCE IMAGING IN INHOMOGENEOUS MAGNETIC FIELDS

YİĞİTLER, Hüseyin

M. Sc., Department of Electrical and Electronics Engineering

Supervisor: Prof. Dr. B. Murat Eyüboğlu

September 2006, 170 pages

Recently, the use of permanent magnets as magnetic field sources in biomedical applications has become widespread. However, usage of permanent magnets in magnetic resonance imaging (MRI) is limited due to their inhomogeneous magnetic field distributions. In this thesis, shape and geometry optimization of a magnet is performed. Moreover, placement of more than one magnet is optimized to obtain desired magnetic field distribution in specific region of space. However, obtained magnetic field distribution can not be used in the conventional MRI image reconstruction techniques. Consequently, an image reconstruction technique for MRI in inhomogeneous magnetic fields is developed. Apart from these, since any reconstruction technique requires signal data, an MRI simulator in inhomogeneous magnetic fields is constructed as a part of this thesis. Obtained results show that the theory developed in this thesis is valid. Consequently, new MRI devices that have permanent magnets as magnetic field sources can be constructed in the future.

Keywords: Magnetic Resonance Imaging, MRI in Inhomogeneous Magnetic Fields, Permanent Magnet Shape Optimization, MRI Simulator, Image Reconstruction

ÖZ

HETEROJEN MANYETİK ALANLARDA MANYETİK REZONANS GÖRÜNTÜLEME İÇİN KALICI MIKNATIS TASARIMI VE GÖRÜNTÜ OLUŞTURMA TEKNİĞİ GELİŞTİRİLMESİ

YİĞİTLER, Hüseyin

Yüksek Lisans, Elektrik ve Elektronik Mühendisliği

Tez Yöneticisi: Prof. Dr. B. Murat Eyüboğlu

Eylül 2006, 170 sayfa

Son zamanlarda, biyomedikal uygulamalarında manyetik alan kaynağı olarak kalıcı mıknatıs kullanımı artmıştır. Ancak, manyetik rezonans görüntüleme (MRG) kalıcı mıknatıs kullanımı, mıknatısların heterojen manyetik alanlarından dolayı sınırlıdır. Bu tezde, kalıcı mıknatısların şekil ve geometri eniyilemesi yapılmıştır. Elde edilen şekil ve geometrideki mıknatısların birden fazlasının uzaydaki yerleşimleri de eniyilenerek uzayın belli bir bölgesinde istenilen manyetik alan dağılımı elde edilmiştir. Ancak, elde edilen manyetik alan dağılımı klinik MRG sistemlerinde kullanılan görüntü oluşturma teknikleri için uygun değildir. Bu nedenle, heterojen manyetik alanlarda MRG için görüntü oluşturma tekniği geliştirilmiştir. Bununla beraber, herhangi bir görüntü oluşturma tekniği sinyal bilgisini gerektirdiğinden, heterojen manyetik alanlarda MRG benzetimi de yine bu tezde yapılmıştır. Elde edilen sonuçlar, bu tezde geliştirilen kuramların doğru olduğunu göstermektedir. Dolayısıyla, ileride, geliştirilen kalıcı mıknatısları manyetik alan kaynağı olarak kullanan MRG sistemleri geliştirilebilecektir.

Anahtar Kelimeler: Manyetik Rezonans Görüntüleme, Heterojen Manyetik Alanlarda MRG, Kalıcı Mıknatıs Şekil Eniyilemesi, MRG benzetimi, Görüntü Oluşturma

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my supervisor Prof. Dr. Murat Eyüboğlu for his guidance and support throughout this study.

I would like also to express my sincere gratitude to my professor and mentor Prof. Dr. Kemal Leblebicioğlu for his guidance and support throughout this study

I would like to express my thanks to Assoc. Prof. Dr. Sencer Koç for his guidance and advices in hopeless times.

My friends Volkan Emre Arpınar and Sedat Doğru deserve to be second and third writers of this thesis. Thanks for their help and support. I would like to apologize for being aggressive during the thesis work from them.

I would like to express my gratitude to my colleagues Ayhan Ozan Yılmaz, Evren Değirmenci and Dario Martin Lorca for their patients.

I would express my thanks to Buket Tunç, who owns my heart, for her support and patients during the most stressful times of my life.

Finally, I would like to thank my family for their great support. I am sure I would not be able to finish this work without their preys. I would express my gratitude to my father and my mother for their motivation throughout my life.

TABLE OF CONTENTS

ABSTRACT	iv
Öz.....	v
ACKNOWLEDGMENTS	vi
TABLE OF CONTENTS	vii
CHAPTER 1 INTRODUCTION	1
1.1. Magnetic Resonance Imaging	1
1.2. Objective of the study	2
1.3. Outline of the study	4
1.4. References	5
CHAPTER 2 PERMANENT MAGNET DESIGN	6
2.1. Introduction	6
2.2. Theory	9
2.3. Method	16
2.3.1. Genetic Algorithms	17
2.3.2. Genetic Algorithm Operations	20
2.3.2.1. The Crossover Operation of a Real Coded Genetic Algorithm	20
2.3.2.2. The Mutation Operation of a Real Coded Genetic Algorithm	21
2.3.2.3. The Crossover Operation of a Binary Coded Genetic Algorithm	25
2.3.2.4. Mutation Operation of a Binary Coded Genetic Algorithm -1-	25
2.3.2.5. Mutation Operation of a Binary Coded Genetic Algorithm -2-	27
2.3.2.6. Mutation Operation of a Binary Coded Genetic Algorithm -3-	28
2.3.2.7. Selection Operation	31
2.3.3. Near Global Optimum Algorithm	32
2.4. Implementation	34
2.4.1. Problem Definition	34
2.4.2. Determination of Unit Magnets	36
2.4.3. Placement of Unit Magnets in Space	40

2.4.3.1. Variable Definitions	41
2.4.3.2. Region Definitions	45
2.4.3.3. Fitness Function Derivation	47
2.4.3.4. Identification Systems	50
2.4.3.5. Fitness Function Evaluation	51
2.4.3.6. Used Genetic Algorithms	53
2.4.4. Solution Steps	54
2.4.4.1. Determination of Unit Magnets' Solution Steps	54
2.4.4.2. Solution Steps of Placement of Unit Magnets'	55
2.5. Results	58
2.5.1. Results of Unit Magnet Determination Part	58
2.5.2. Results of Placement of Unit Magnets Part	68
2.6. Conclusion	81
2.7. References	83
CHAPTER 3 MAGNETIC RESONANCE IMAGING SIMULATOR AND IMAGE RECONSTRUCTION ALGORITHM IN INHOMOGENEOUS MAGNETIC FIELDS	85
3.1. Introduction	85
3.2. Theory	91
3.2.1. Conventional MRI Formulation	95
3.2.2. Extension to Inhomogeneous Magnetic Fields	101
3.2.3. MR Formulation in Inhomogeneous Magnetic Fields	103
3.2.4. Image Reconstruction Technique Formulation	111
3.3. Implementation	116
3.3.1. Implemented Pulse Sequence	116
3.3.2. Generation Of Magnetic Fields	122
3.3.3. MRI Simulator In Inhomogeneous Magnetic Fields	123
3.3.4. MRI In Inhomogeneous Magnetic Fields Implementation	128
3.3.5. Image Reconstruction Technique Implementation	132
3.4. Results	133
3.4.1. Results Of Homogeneous Magnetic Fields Case	134
3.4.2. Results of Inhomogeneous Magnetic Fields Case	147
3.5. Conclusion	163

3.6. References	165
CHAPTER 4 CONCLUSION.....	168
PUBLICATIONS.....	171
Publications Prior to M.Sc. Study	171
Publications During M.Sc. Study.....	171

CHAPTER 1

INTRODUCTION

1.1. Magnetic Resonance Imaging

Magnetic resonance imaging is a tomographic imaging technique that utilizes the response of the spinning atomic particle under influence of static and radio frequency magnetic fields in order to form images that are extremely rich in information content.

Nuclear magnetic resonance (NMR) phenomenon was first discovered independently by Bloch *et al* [1, 2] and Purcell *et al* [3] in 1946. This is followed by the discovery of the chemical shift which enables the identification of the nuclei in different chemical environments. Therefore nuclear magnetic resonance has become a powerful tool for chemistry to investigate the structure of the solids and liquids.

The discovery of NMR phenomenon and the discoveries of spectroscopic techniques are followed by the discoveries related with imaging the spin distribution in tissues. The principle of using gradient fields to spatially encode spin distribution in measured spin response was proposed by Lauterbur [4] and by Mansfield and Grannell [5] in 1973. The first whole body image is published by Damadian *et al* [6] in 1977. Their pioneering research triggers the development of many other discoveries such as the fast imaging methods, T_1 - T_2 contrast imaging etc.

1.2. Objective of the study

The proposed imaging techniques for conventional NMR imaging (MRI) systems require homogeneous distribution of strong, external, static magnetic field. This obligatory requirement arises from the fact that identical spinning frequency throughout the object being imaged can be perturbed in a controlled manner easier. However, to create homogeneous distribution of strong and static magnetic field requires large and expensive coils which restrict the size of object being imaged and require high running cost. On the other hand, recently, the usage of high gradient magnetic fields for separation and localization has become widespread due to the fact that inhomogeneous magnetic field systems have many advantages over the commercial resistive or superconductive magnet systems [7].

The presence of magnetic field gradients allows determination of transverse and longitudinal relaxation and translational diffusion which are vital parameters used in NMR and MRI systems [7]. The main advantages of high gradient magnetic fields over the homogenous magnetic fields are size and cost due to the fact that high gradient magnetic fields can be obtained using permanent magnets. Contrary to the resistive and superconductive magnets used in commercial systems, permanent magnets do not require any running cost, and the size of the magnet is small so that the magnetic field source itself can be moved on the object, allowing unlimited object size. On the other hand, the usage of permanent magnets is restricted to be used in near surface applications due to their high gradient magnetic fields.

Magnetic field distribution of permanent magnets can be altered by changing their surfaces. Therefore, the first object of this study is to develop a method to design permanent magnets structures.

Since the magnetic field in a region of space satisfies superposition principle for linear media, more than one magnet can be used to create desired magnetic field. Magnetic field distribution due to more than one magnet can be altered by changing positions of magnets. Therefore, the second aim of this study is to develop a method

to determine locations of more than one magnet to obtain desired magnetic field distribution.

Permanent magnet surface optimization and placement of more than one magnet have been studied by researchers. The attempts to obtain desired magnetic field distribution are given in introduction section of Chapter 2.

The use of inhomogeneous magnetic field sources in MRI requires development of novel image reconstruction techniques; because conventional techniques require homogeneous magnetic field distribution. Since the physical principle behind MRI allows usage of high gradient magnetic fields, a different mathematical model of MRI system is needed to form images from acquired signals. Since the usage of inhomogeneous magnetic field distribution in MRI requires much more complicated models, image reconstruction requires high computation cost.

To be able to observe the signal shape and characteristics when the atomic particles are under the influence of inhomogeneous magnetic fields in the ideal case, a computer simulator should be constructed. However, this requires a relation between measured signal and the object being imaged. This relation can be developed by extending the conventional MRI formulation using new coordinate system. Therefore, the third objective in this study is to develop and construct an MRI simulator in inhomogeneous magnetic fields.

Several MRI simulators for educational and research purposes have been constructed, so far. These simulators are shortly explained in introduction section of Chapter 3.

In this study to be able to construct images from the MRI simulator developed in inhomogeneous magnetic fields requires development of new image reconstruction techniques. Since any image reconstruction technique requires a mathematical model of the system, the fourth aim of this study is to develop a mathematical model of MRI in inhomogeneous magnetic fields system. Consequently, the final aim is to develop an image reconstruction technique which makes use of the model of the system.

In literature, it is not possible to find many works which aim to develop a novel reconstruction technique for MRI in inhomogeneous magnetic fields. However, there are some attempts to correct image artifacts due to magnetic field inhomogeneities. The aim to modify the signals obtained in a conventional image reconstruction technique. On the other hand, there are some NMR devices constructed using permanent magnets as magnetic field sources. Moreover, there have been attempts to obtain images using these devices by the conventional image reconstruction techniques. Therefore, the main constraint on construction of these devices is to obtain magnetic fields as homogeneous as possible. These works are also summarized in the introduction section of Chapter 3.

1.3. Outline of the study

As described above, there are five objectives in this study. These five aims can be grouped into two; first one is the permanent magnet design (objectives one and two) and the second one is the construction of an MRI simulator in inhomogeneous magnetic fields (objective three) and the development of an image reconstruction technique (objectives four and five).

In Chapter 2, permanent magnet design procedure is described. In its subsections, firstly, the motivation behind such a study, objectives and an outline of previous works are given. Secondly, the theory of magnetic field calculation is given. Thirdly, used methods are introduced. Fourthly, implementation details are given. Lastly, obtained results are demonstrated

In Chapter 3, the construction of an MRI simulator in inhomogeneous magnetic fields and the development of an image reconstruction technique are described. In its subsections, firstly, the motivation behind such a study, objectives and an outline of previous works are given. Secondly, theory of the simulator and image reconstruction technique are given. Thirdly, implementation details of the simulator

and image reconstruction technique are given. Fourthly, obtained results are given for both homogeneous and inhomogeneous magnetic field cases are given.

Finally, Chapter 4 is the conclusion chapter.

Since the thesis is divided into two main parts, the references of each part are given at the end of each chapter.

1.4. References

- [1]. F. Bloch, “Nuclear induction”, Physical Review, Vol. 70, pp. 460-474, 1946.
- [2]. F. Bloch, W. W. Hansen, and M. Packard, “The nuclear induction experiment”, Physical Review, Vol. 70, pp. 474-485, 1946.
- [3]. E. M. Purcell, H. C. Torrey, and R. V. Pound, “Resonance absorption by nuclear magnetic moments in a solid”, Physical Review, Vol. 69, pp. 37-38, 1946.
- [4]. P. C. Lauterbur, “Image formation by induced local interactions - Examples employing nuclear magnetic-resonance”, Nature, Vol. 242, pp. 190-191, 1973.
- [5]. P. Mansfield and P. K. Grannell, “NMR 'diffraction' in solids”, Journal of Physics C: Solid State Physics, Vol. 6, pp. 422-426, 1973.
- [6]. R. Damadian, M. Goldsmith, and L. Minkoff, “NMR in cancer: XVI. FONAR image of the live human body”, Physiological Chemistry and Physics, Vol. 9, pp. 97-100, 1977.
- [7]. Gareth P. Hatch, “Magnet design considerations for devices and particles for biological high-gradient magnetic separation (HGMS) system”, Journal of Magnetism and Magnetic Materials, Vol. 225, pp. 262-276, 2001.

CHAPTER 2

PERMANENT MAGNET DESIGN

2.1. Introduction

Magnetic resonance imaging is based on physical phenomenon of nuclear magnetic resonance, which is related to magnetic properties of atoms' nucleus. For paramagnetic materials, magnetization of microscopic particles becomes observable in macroscopic level only when sufficient number of nuclei is aligned to strong external magnetic field [1].

In commercial nuclear magnetic resonance (NMR) and magnetic resonance imaging (MRI) systems, large and expensive magnets are used to obtain a highly homogeneous magnetic field in order to observe resonance phenomena throughout the object being imaged. On the other hand, recently, the use of high gradient magnetic fields, for separation and localization, has become widespread due to the fact that inhomogeneous magnetic field systems have many advantages over the commercial resistive or superconductive magnet systems [7].

In NMR and MRI systems, magnetic field gradients are an asset for spatial resolution [7]. Their presence allows determination of transverse and longitudinal relaxation and translational diffusion which are vital parameters used in NMR and MRI systems. The main advantages of high gradient magnetic fields over the homogenous magnetic fields are size and cost due to the fact that high gradient magnetic fields can be obtained using permanent magnets. Contrary to the resistive and

superconductive magnets used in commercial systems, permanent magnets do not require any running cost, and their sizes are so small that the magnetic field source itself can be moved on the object, allowing unlimited object size. Moreover, some permanent magnets can produce up to 1.2 T of polarization fields allowing to use them in applications which require high magnetic fields. On the other hand, the use of permanent magnets is restricted to be used in near surface applications due to their highly inhomogeneous magnetic fields.

The magnetic field distribution of permanent magnets is a function of construction material, geometry, and initial magnetization of a magnet [2, 3]. Hence, the magnetic field distribution can be changed by fine tuning both micro and macro level properties of magnet. Since magnetization and magnetic materials are related to production of bulk magnet, it is assumed that these parameters are fixed and leave them out of scope of this work. Another important reason to leave these parameters out of scope of this work is due to the fact that it is not possible to reach a unique magnetization distribution which generates a specific magnetic field distribution [2]. Consequently, in the first part of this work it is aimed to obtain permanent magnet geometry which yields desired magnetic field properties.

Geometry optimization of permanent magnets is studied by researchers previously. In [7] and [8], it is assumed that the surface of the magnet is flat, and the geometry is optimized for the height and the side lengths of the magnet. However, this kind of optimization is not adequate to design permanent magnets for NMR and MRI applications. Since the magnetic field distribution generated by a permanent magnet is not unique, manufacturability becomes the most important constraint. In previous works, manufacturability is defined as regularity [9, 10, 11, 12]. In [9], a novel method is presented which makes use of finite element method (FEM) to calculate magnetic field distribution. Changes in geometry are performed by moving the mesh elements under the application of artificial forces. This idea guarantees the continuity of surface, and manufacturability is indirectly assured. In [10], the surfaces are parameterized using spline functions adaptively, and the number and locations of control points are generated automatically to avoid jagged surfaces. In [11], an analytical solution of magnetic field of permanent magnets is developed. The surface

is described by equipotential lines of scalar potential function, scalar magnetic potential. The use of equipotential lines of potential field assures continuity of the surface since any equipotential line of a potential function should be continuous. In [12], an inverse methodology is followed. From the desired magnetic field distribution, they try to obtain a magnet surface which generates the desired magnetic field. In that work, analytical solution derived in [11] is used. Scalar magnetic potential is calculated from desired magnetic field to obtain the surface.

In the first part of this work it is aimed to design a permanent geometry which can be jagged or not to generate a magnetic field with desired properties in some specific region of space. Contrary to previous works, manufacturability is not equated to continuity of the surface, since some discontinuous surfaces can be manufactured easier than continuous ones.

In the second part of this work, placement of more than one permanent magnet with surfaces obtained in the first part is tried to be optimized. The placement of more than one permanent magnet in space has been studied previously. It is accepted that Halbach magnet arrangement is the one that produces homogeneous magnetic field distribution [13]. However, these arrangements are developed for 2D problems and they produce homogeneous magnetic fields in the interior of closed structures. In [14], a Halbach array is designed to produce homogeneous magnetic field inside a tube like 3D structure.

In the second part of the study, it is aimed to design an optimum arrangement of permanent magnets which does not form a closed surface so that the structure is open, produces a magnetic field with desired properties.

Geometry optimization of permanent magnets and placement of more than one magnet in space to obtain desired magnetic field distribution problem is very hard to solve by standard nonlinear programming algorithms. These algorithms require near optimum initial solution, gradient or Hessian matrix of optimization function, a line search algorithm, and a nearly convex optimization function. The most important restriction of these algorithms comes from the initial solution requirement. Since it is

not possible to predict a near global optimum solution for such problems directly, an algorithm which can carry the solution towards the global optimum is needed. In fact, evolutionary algorithms like “Genetic Algorithms” which belong to “Weak Methods” in Artificial Intelligence community are the best tools to solve these kinds of problems [15].

Although genetic algorithms provide a means to get closer to global optimum solution, they should not be used to reach global optimum points in a finite number of iterations (generations). When it is thought that the solution is close enough to global optimum one, a gradient based optimization algorithm should be utilized to reach it. In this work, these algorithms are called “Near Global Optimum Algorithms”.

In this chapter, a new method to design permanent magnet structures for generation of magnetic fields with desired properties is described. Firstly, the theory of magnetic field calculations is given. Secondly, details of genetic algorithms and their operations and near global optimum algorithms are described. Thirdly, implementation details are given. Fourthly, obtained results are demonstrated. Finally, this chapter is concluded.

2.2. Theory

In this part, calculation of magnetic field distribution of permanent magnet using scalar magnetic potentials is described in detail.

Magnetic field distribution in space can be calculated using two different potentials: vector magnetic potentials, and scalar magnetic potentials. Vector magnetic potential arises from the fact that magnetic field is solenoidal, i.e., the divergence of magnetic induction is 0. Using the identities of vector calculus:

$$\nabla \cdot \vec{B} = 0 \quad (2.1)$$

$$\nabla \cdot (\nabla \times \vec{A}) = 0 \quad (2.2)$$

$$\vec{B} = \nabla \times \vec{A} \quad (2.3)$$

for any vector field, \vec{A} is defined as vector magnetic potential [6].

The definition implies that, magnetic vector potential exists whenever a magnetic induction exists.

On the other hand, for current free regions,

$$\nabla \times \vec{B} = \mu_0 \vec{J} \Big|_{\vec{J}=0} = 0 \quad (2.4)$$

Hence, using the null identity of vector calculus \vec{B} can be defined as:

$$\nabla \times (\nabla V_m) = 0 \quad (2.5)$$

$$\vec{B} = -\mu_0 \nabla V_m \quad (2.6)$$

for any scalar field V_m . V_m is defined as the scalar magnetic potential [6].

Since permanent magnets create magnetic field without a current source, magnetic scalar potential can be used to calculate magnetic field distribution. General formulation of scalar magnetic potential is given in [4]. Here a slightly modified version is given. Before giving the formula of the scalar magnetic potential, definition of magnetic moment should be made.

Amperé's theorem states that any current carrying loop can generate a magnetic field. Hence, any magnetic material can be considered as a combination of infinitesimally small current carrying loops, making these loops being the most elementary elements of magnetism [2].

By defining $Id\vec{s}$ as $d\vec{m}$, $d\vec{m}$ can be used as the elementary unit of magnetism instead of infinitesimally small current loop:

$$d\vec{m} = Id\vec{s} \quad (2.7)$$

This term is defined as the magnetic dipole moment [2].

Magnetic induction at any point in the space due to a current carrying loop can be found using Biot-Savart law;

$$d\vec{B} = \frac{\mu_0}{4\pi} (Id\vec{l} \times \frac{\vec{R}}{\|\vec{R}\|^3}) \quad (2.8)$$

where \vec{R} is defined as distance vector;

$$\vec{R} = \vec{r} - \vec{r}' \quad (2.9)$$

where \vec{r}' is the position vector of source points, and \vec{r} is the position vector of field points.

Hence, the change in scalar vector potential, V_m , described by (2.6), when the field point is moved by an amount of $\delta\vec{d}$, the change in scalar magnetic potential is given by;

$$\Delta V_m = -\frac{1}{\mu_0} \vec{B} \cdot \delta\vec{d} = -I \oint \frac{\delta\vec{d} \cdot (d\vec{l} \times \vec{R})}{4\pi \|\vec{R}\|^3} \quad (2.10)$$

The same change in potential should be observed when the field point is fixed and the source point is displaced by an amount of $-\delta\vec{d}$. The area swept by such a displacement is:

$$d\vec{s} = -(\delta\vec{d} \times d\vec{l}) \quad (2.11)$$

Subtended solid angle by area $d\vec{s}$ is given by:

$$\Omega = -\int_s \nabla \left(\frac{1}{\|\vec{R}\|} \right) \cdot d\vec{s} \quad (2.12)$$

$$\nabla \left(\frac{1}{\|\vec{R}\|} \right) = -\frac{\vec{R}}{\|\vec{R}\|^3} \quad (2.13)$$

$$d\Omega = -\frac{\vec{R} \cdot (\delta\vec{d} \times d\vec{l})}{\|\vec{R}\|^3} = -\frac{\delta\vec{d} \cdot (d\vec{l} \times \vec{R})}{\|\vec{R}\|^3} \quad (2.14)$$

Substituting (2.14) in (2.10) yields:

$$\Delta V_m = I \frac{d\vec{s} \cdot \vec{R}}{4\pi \|\vec{R}\|^3} = \frac{I}{4\pi} d\Omega \quad (2.15)$$

Using (2.15) and (2.7):

$$V_m = \int_s \frac{d\vec{m} \cdot \vec{R}}{4\pi \|\vec{R}\|^3} \quad (2.16)$$

The detailed version of derivations can be found in [4].

In macroscopic level, magnetic dipole moments can not be observed but magnetic intensity can be observed. Magnetic intensity is defined as:

$$\vec{M} = \frac{\vec{m}}{V} \quad (2.17)$$

where \vec{M} is magnetic intensity, \vec{m} is magnetic dipole moment, and V is the volume where magnetic dipole moments are distributed. Using this definition \vec{m} can be written as:

$$d\vec{m} = \vec{M}dV \quad (2.18)$$

Substituting (2.18) into (2.16) and using (2.13) yields:

$$V_m = \int_v \frac{\vec{M} \cdot \vec{R}}{4\pi \|\vec{R}\|^3} dV = -\frac{1}{4\pi} \int_v \vec{M} \cdot \nabla \left(\frac{1}{\|\vec{R}\|} \right) dV \quad (2.19)$$

Using the vector identity:

$$\nabla \left(\frac{\vec{M}}{\|\vec{R}\|} \right) = \vec{M} \cdot \nabla \left(\frac{1}{\|\vec{R}\|} \right) + \frac{1}{\|\vec{R}\|} \nabla \cdot \vec{M} \quad (2.20)$$

and the divergence theorem, V_m turns out to be:

$$V_m = -\frac{1}{4\pi} \left[\int_v \nabla \cdot \left(\frac{\vec{M}}{\|\vec{R}\|} \right) dV - \int_v \frac{\nabla \cdot \vec{M}}{\|\vec{R}\|} dV \right] \quad (2.21)$$

$$V_m = -\frac{1}{4\pi} \left[\int_s \frac{\vec{M}}{\|\vec{R}\|} \cdot d\vec{s} - \int_v \frac{\nabla \cdot \vec{M}}{\|\vec{R}\|} dV \right] \quad (2.22)$$

The magnetic field distribution can be calculated using (2.22) and (2.6).

To be able to say that any function is a scalar magnetic potential, uniqueness of scalar magnetic potential in terms of its formulation, (i.e., equation (2.22)), and boundary conditions should be shown.

Boundary conditions of scalar magnetic potential are derived from ordinary magnetostatic boundary conditions. Therefore, a relationship between magnetic induction, \vec{B} , and magnetic field, \vec{H} , in whole space (inside and outside of the magnet) should be defined. Interior of ferromagnetic body contains some aligned dipole moments, which contributes to the exterior magnet field, \vec{H} . This fact is represented by relation:

$$\vec{B} = \mu(\vec{H} + \vec{M}) \quad (2.23)$$

Outside the magnetic medium \vec{M} is 0 and (2.23) becomes:

$$\vec{B} = \mu_0 \vec{H} \quad (2.24)$$

Then, generally scalar magnetic potential is written as:

$$\vec{H} = -\nabla V_m \quad (2.25)$$

to make the definition valid inside the ferromagnetic body. Then, using divergence free nature of \vec{B} :

$$\nabla \cdot \vec{B} = \mu \nabla \cdot (\vec{M} + \vec{H}) = 0 \quad (2.26)$$

Inside the magnetic body the differential equation of scalar potential becomes:

$$-\nabla \cdot \vec{H} = \nabla^2 V_m = \nabla \cdot \vec{M} \quad (2.27)$$

Outside the magnetic body:

$$\nabla^2 V_m = 0 \quad (2.28)$$

Using the continuity of tangential component of \vec{H} and normal component of \vec{B} , the boundary conditions for the scalar magnetic potential are obtained as:

$$V_{m_{in}} = V_{m_{out}} \quad (2.29)$$

$$\frac{\partial V_{m_{in}}}{\partial n} - \frac{\partial V_{m_{out}}}{\partial n} = \vec{M} \cdot \hat{n} \quad (2.30)$$

Any function satisfying the equations from (2.27) to (2.30) is a solution of these equations and it is unique. Proof of this statement is given in [3].

To be able to calculate magnetic field from scalar magnetic potential using (2.22), some assumptions about magnetic intensity distribution of magnet should be made. That follows from the fact that magnetic field is given by:

$$\vec{H} = \frac{1}{4\pi} \nabla \left(\int_s \frac{\vec{M}}{\|\vec{R}\|} \cdot d\vec{s} - \int_v \frac{\nabla \cdot \vec{M}}{\|\vec{R}\|} dV \right) \quad (2.31)$$

If magnetic field intensity is assumed to be uniform inside the volume of permanent magnet, volume term in (2.31) can be dropped. Let us assume that magnetic field intensity of a permanent magnet is defined as:

$$\vec{M} = M_0 \hat{e}_z \quad (2.32)$$

where \hat{e}_z is the direction parallel to the height of magnet and M_0 is the magnitude of magnetic field intensity. Therefore, using the fact that gradient operator in (2.31)

operates on field points, \vec{r} , whereas the integral operates on source points, \vec{r}' , (2.31) can be written as:

$$\vec{H} = \frac{1}{4\pi} \nabla \left(\int_s \frac{\vec{M}}{\|\vec{R}\|} \cdot d\vec{s} \right) = \frac{1}{4\pi} \int_s \nabla \left(\frac{1}{\|\vec{R}\|} \right) \vec{M} \cdot d\vec{s} \quad (2.33)$$

Using (2.13), (2.33) can be rewritten as:

$$\vec{H} = -\frac{1}{4\pi} \int_s \frac{\vec{R}}{\|\vec{R}\|^3} \vec{M} \cdot d\vec{s} \quad (2.34)$$

where \vec{R} in terms of coordinates of source point and observation point is defined as:

$$\vec{R} = (x - x')\hat{e}_x + (y - y')\hat{e}_y + (z - z')\hat{e}_z \quad (2.35)$$

Surface integral in (2.34) operates on the whole surface of the permanent magnet. The surface of the permanent magnet is represented as a function $z' = z'(x', y')$. Since, z' may or may not be a continuous function, the function space of surfaces can not be restricted. Consequently, a methodology which covers all possible surfaces, like spline function parameterization, or finite element meshes should be followed. In this work, finite element meshes are generated. To calculate magnetic field superposition principle is used.

Triangular elements are chosen to partition the domain of surface function, z' . A sample of mesh is shown in Figure 2.1.

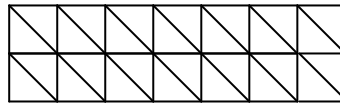


Figure 2.1: A sample that shows the elements' projection on domain of surface

Since the mesh segments of domain is represented by 3 points which are not on the same line the equation of the plane passing through these 3 points can be found. This

equation can be used to calculate the value of $z' = z'(x', y')$ when x' and y' are known and fixed. This methodology will increase the magnetic field calculation accuracy. On the other hand, linear interpolation parameters for triangular mesh segments can also be found to calculate surface function value.

To find linear interpolation parameters, a slightly different method which makes use of the vector algebra is used. A clearer plot of a mesh element is given in Figure 2.2.

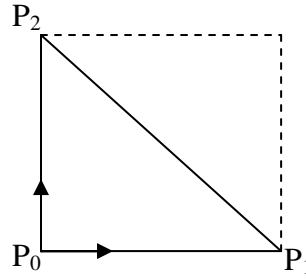


Figure 2.2: A triangular element with edge points specified

Vectors $\overrightarrow{P_0P_1}$ and $\overrightarrow{P_0P_2}$ are defined as shown in Figure 2.2, so that any point inside the triangle can be expressed as a linear combination of these vectors as shown in Figure 2.3.

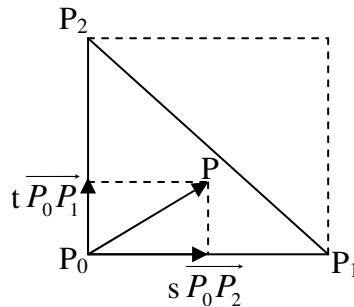


Figure 2.3: Representation of any point inside the element as linear combination of $\overrightarrow{P_0P_1}$ and $\overrightarrow{P_0P_2}$

Since P is restricted to be inside the triangle, s and t can not take every value. They are restricted to satisfy:

$$0 \leq s + t \leq 1 \quad (2.36)$$

Then $\overrightarrow{P_0P}$ can be written as:

$$\overrightarrow{P_0P} = t \cdot \overrightarrow{P_0P_1} + s \cdot \overrightarrow{P_0P_2} \quad (2.37)$$

For the problem at hand, x' and y' are assumed to be known for every point, P , inside the triangle. z' value of P is given by:

$$z' = z'_0 + t \cdot (z'_1 - z'_0) + s \cdot (z'_2 - z'_0) \quad (2.38)$$

s and t are found from equations of x' and y' , by solving 2 equations with 2 unknowns:

$$x' = x'_0 + t \cdot (x'_1 - x'_0) + s \cdot (x'_2 - x'_0) \quad (2.39)$$

$$y' = y'_0 + t \cdot (y'_1 - y'_0) + s \cdot (y'_2 - y'_0) \quad (2.40)$$

Area of a surface element in a mesh is given by:

$$ds = -\frac{1}{2} (\overrightarrow{P_0P_1} \times \overrightarrow{P_0P_2}) \quad (2.41)$$

The magnetic field due to a source point $P(x', y', z')$ at point (x, y, z) is given by:

$$\vec{H}_{P_i}(x, y, z) = \frac{M_0}{8\pi} \frac{(x-x')\hat{e}_x + (y-y')\hat{e}_y + (z-z')\hat{e}_z}{((x-x')^2 + (y-y')^2 + (z-z')^2)^{\frac{3}{2}}} \hat{e}_z \cdot (\overrightarrow{P_0P_1} \times \overrightarrow{P_0P_2}) \quad (2.42)$$

Using the superposition principle, total magnetic field at point (x, y, z) is found by the vector sum of individual magnetic fields due to source points P_i , where $i = 1..N$ and N is the number of source points in the whole surface mesh.

2.3. Method

In this section, used algorithms are explained in detail. Firstly, a general description of genetic algorithms is given. Secondly, operations of genetic algorithms used in

this work are described. Finally, the overall algorithm which is expected to converge to near global optimum points is explained.

2.3.1. Genetic Algorithms

Genetic algorithms are a class of optimization (or search) technique inspired from the evolutionary biology. These techniques are stochastic in nature, and they simulate main concepts of evolutionary biology such as mutation, selection and crossover. Consequently, they try to do what nature does to individuals to transform species to better ones in the sense of survival [16].

Mathematically genetic algorithms are simulations of a population of abstract representations of solutions which aim to evolve to a better solution. Typically, these algorithms require two things [17]:

1. An abstract representation of a solution,
2. A fitness function to evaluate them

These two requirements are an asset of genetic algorithms, but not enough to guarantee an evolution. Evolution of population in genetic algorithms is guaranteed by selection, mutation and crossover operations. The first operation is used to inherit the solution to the next generation. Mutation and crossover operations are used to introduce diversity of a population of solutions. These operations are performed for all generations.

A typical genetic algorithm steps are shown in Figure 2.4. The first step of genetic algorithms is to define an abstract representation of solutions. This representation of solutions is called a “Chromosome”. Chromosomes are nothing but a sequence of numbers, which have number of elements equal to the dimension of the solution space. Each element of a chromosome is called a “Gene”. Genes can be real numbers, floating numbers or binary numbers depending on the representation of solutions.

Fitness function gives a measure of fitness of chromosomes. In other words, it assigns a value to the abstract representation of a solution.

```
define an abstract representation of solutions
    (form chromosomes)
define a fitness function
generate initial population
evaluate fitness function (population)
i=1
while i≤maximum iteration number
    find and hold the elitist chromosome
        (best solution in the population)
    perform crossover operation
    perform mutation operation
    perform selection for next generation
    evaluate fitness function (population)
    add the elitist chromosome to next generation
    i++
end
```

Figure 2.4: Genetic algorithm steps

Transformation from solution to chromosome is called “Encoding” and transformation from chromosome to solution is called “Decoding”. When chromosomes are sequences of real numbers the genetic algorithm is called “Real Coded Genetic Algorithm”; when the chromosomes are sequences of binary numbers the genetic algorithm is called “Binary Coded Genetic Algorithm”. The type of encoding is determined according to the problem. In this work, both, binary and real coded genetic algorithms are used.

After defining chromosomes and fitness function, the next step is to generate the initial population. Initial population is usually constructed randomly. However, sometimes some chromosomes with higher fitness are added to it, to be able increase the convergence rate of the genetic algorithm. The size of the initial population is

defined according to the problem and it is preserved till the end of simulation. After creating initial population, the fitness of the population is calculated. To guarantee monotonic increase or decrease, the elitist chromosome of the population is stored to directly pass it to the next generation. After that step, genetic algorithm operations are repeated.

As in crossover during mitosis in biology two chromosomes come side-by-side, and exchange their genes. This structure defines the general nature of crossover operation. At the beginning of the simulation a probability of crossover occurrence is defined for a chromosome. The usual procedure, that any crossover follows, begins with the determination of chromosomes that will take part in crossover. The procedure continues with random selection of two chromosomes among selected ones that will exchange their genes. These two chromosomes are called “Parents”. After that, a random gene is selected as the binding or breaking point (also called “Crossover Point”) after which the parent chromosomes will exchange their genes. Finally, the parents exchange the genes and offspring chromosomes are created. Whether the parent chromosomes cease to exist or continue to exist depends on the problem.

The mutation operation is also an analogue of mutation in genetics. Since the mutation in genetics has a variety of forms, in genetic algorithms, this operation is not uniquely defined. However, generally, as in crossover operation at the beginning of simulation, a probability of mutation is defined. Genes or chromosomes that will have mutation operation constitute a set of genes or chromosomes called “Mutation Pool”. After that, selected genes or chromosomes are imposed to the indicated mutation operation. As in crossover, whether the parent chromosomes cease to exist or continue to exist depends on the problem.

Chromosomes for next generation are obtained by applying the selection generation. There is a variety of selection procedures, but it is customary to use roulette wheel that has slots sized according to fitness values of chromosomes which enter to selection mechanism. Then, a shot is made to select a chromosome. This procedure of selection is the most frequently used one because of Schema Theorem. This

theorem states that the chromosomes with above average fitness receive exponentially increasing trials in subsequent generations of genetic algorithms [16]. The details of selection procedure are given in next section.

2.3.2. Genetic Algorithm Operations

As previously stated, in this work, binary coded and real coded genetic algorithms are used. The genetic algorithm operations vary according to genetic algorithm type and according to problem. In this work, one type of operation has been used for real coded genetic algorithms and for binary coded genetic algorithms' operations one type of crossover and three kinds of mutation have been used. Only one type of selection operation is used.

2.3.2.1. The Crossover Operation of a Real Coded Genetic Algorithm

The chromosomes in real coded genetic algorithm are sequences of real numbers. Therefore, by just exchanging genes of chromosomes does not increase the population diversity. Crossover operation of real coded genetic algorithms is modified. This modified crossover is known as "Arithmetic Crossover". In figure 2.5 arithmetic crossover is shown. In Figure 2.5, X and Y are parents, and Z_1 and Z_2 are offspring. The crossover is performed at the i^{th} gene of the parent chromosomes.

In arithmetic crossover, when the parents are exchanging their genes, genes are multiplied with a random number between 0 and 1. This multiplication is performed according to rules given below:

$$\begin{array}{l} Z_{11} = a \cdot X_1 + (1-a) \cdot Y_1 \\ Z_{12} = (1-a) \cdot X_2 + a \cdot Y_2 \\ Z_{21} = (1-a) \cdot X_1 + a \cdot Y_1 \\ Z_{22} = a \cdot X_2 + (1-a) \cdot Y_2 \end{array}$$

where a is any random number in the interval $[0,1]$.

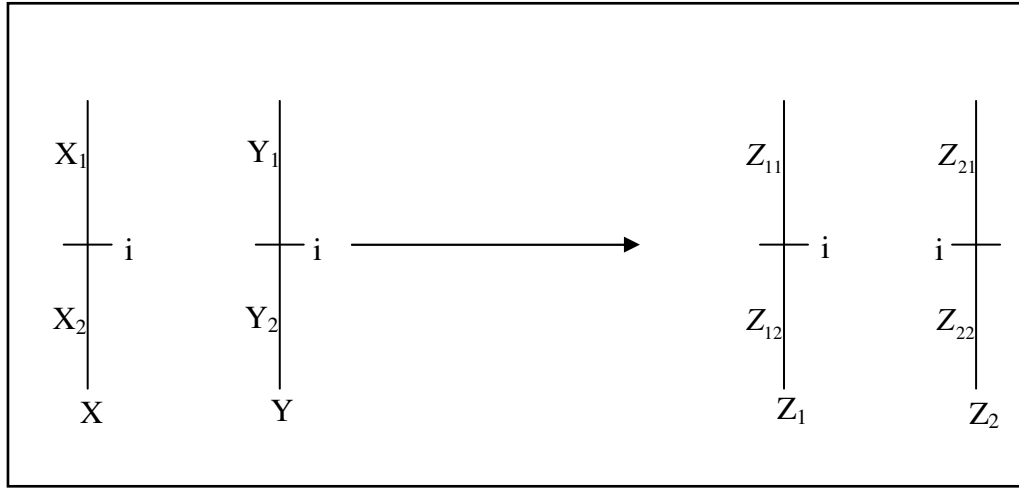


Figure 2.5: Real coded genetic algorithm

However, this multiplication with random number causes the chromosomes to converge to an average value of a population as the generations are performed. Convergence to average value is prevented by adding boundary chromosomes to population. This procedure is known as “Boundary Mutation”, and will be explained in real coded mutation section.

An example of real coded genetic algorithm crossover operation is given on the next page.

2.3.2.2. The Mutation Operation of a Real Coded Genetic Algorithm

As described previously, mutation operations can operate on genes or chromosomes. Since the chromosomes of real coded genetic algorithm are composed of real numbers, choosing mutation that operates on chromosomes is more convenient.

Chromosomes length = N

Select chromosomes for crossover

$$\text{selected chromosome set} = \{C_3, C_7, C_9, C_{18}\}$$

Select two of selected chromosomes randomly and remove them from the set

$$\text{parents} = \{C_3, C_{18}\}$$

$$C_3 = \begin{Bmatrix} 1.2 \\ 2.1 \\ \vdots \\ 2.9 \\ 1.22 \\ 1.45 \end{Bmatrix}, C_{18} = \begin{Bmatrix} 2.2 \\ 1.1 \\ \vdots \\ 1.9 \\ 1.11 \\ 2.54 \end{Bmatrix}$$

Select crossover point

$$\text{crossover point} = N - 2$$

Generate a random number in interval $[0,1]$

$$a = 0.1$$

Perform the multiplication and form offspring

$$O_1 = \begin{Bmatrix} 0.1 \cdot \begin{Bmatrix} 1.20 \\ 2.10 \\ \vdots \\ 2.90 \end{Bmatrix} + 0.9 \cdot \begin{Bmatrix} 2.20 \\ 1.10 \\ \vdots \\ 1.90 \end{Bmatrix} \\ 0.9 \cdot \begin{Bmatrix} 1.22 \\ 1.45 \end{Bmatrix} + 0.1 \cdot \begin{Bmatrix} 1.11 \\ 2.54 \end{Bmatrix} \end{Bmatrix}, O_2 = \begin{Bmatrix} 0.9 \cdot \begin{Bmatrix} 1.20 \\ 2.10 \\ \vdots \\ 2.90 \end{Bmatrix} + 0.1 \cdot \begin{Bmatrix} 2.20 \\ 1.10 \\ \vdots \\ 1.90 \end{Bmatrix} \\ 0.1 \cdot \begin{Bmatrix} 1.22 \\ 1.45 \end{Bmatrix} + 0.9 \cdot \begin{Bmatrix} 1.11 \\ 2.54 \end{Bmatrix} \end{Bmatrix}$$

$$O_1 = \begin{Bmatrix} 2.10 \\ 1.20 \\ \vdots \\ 2.00 \\ 1.121 \\ 2.431 \end{Bmatrix}, O_2 = \begin{Bmatrix} 1.30 \\ 2.00 \\ \vdots \\ 2.80 \\ 1.209 \\ 1.559 \end{Bmatrix}$$

Continue till the selection set is empty

An example of the crossover operation of a real coded genetic algorithm

A mutation operation that operates on chromosomes should start with the selection of chromosomes that are supposed to undergo a mutation operation. Preceding part of mutation procedures is problem dependent. For this problem, mutation has been implemented such that a new chromosome with random gene distribution is added to the mutating chromosome with some weight. The new chromosome consists of genes of real number whose values are between -1 and 1. To prevent extreme boundary exceeding of genes, new chromosomes' genes are multiplied with a number which is 5% to 10% of original chromosomes' gene values. In other words, the weight is chosen so that it is 5% to 10% of variable's (the variable which is represented by chromosome) interval length of definition. Finally, the genes of the output chromosome are checked for boundary exceeding. If a boundary exceeding occurs for a gene, that genes value is moved to near boundary. In Figure 2.6, these concepts are shown.

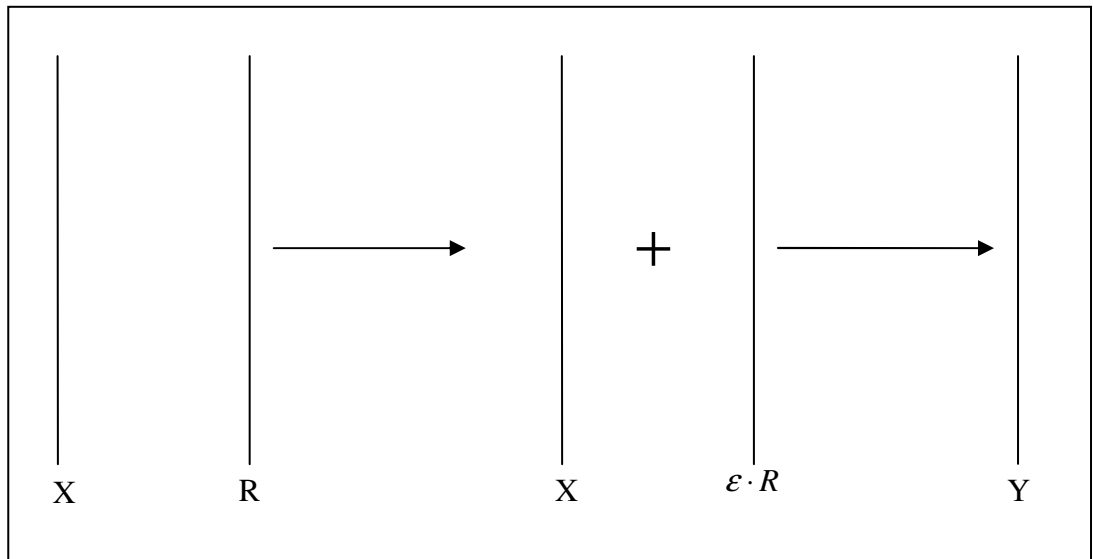


Figure 2.6: Real coded genetic algorithm mutation operation

In Figure 2.6, X shows a chromosome that is selected for mutation; R is the random chromosome whose elements are in the interval $[-1,1]$; Y, which is given by $X + \epsilon \cdot R$, is the output.

The determination of ε is critical for the mutation operation. Referring to Figure 2.6, if the genes of X are in the interval $[a, b]$, then ε is calculated by $\varepsilon = k \cdot (b - a)$, where k is in the range $[0.05, 0.10]$.

An example of the mutation operation of a real coded genetic algorithm is given below:

Chromosomes' genes interval $[1, 3]$

Define ε

$$\varepsilon = 0.05 \cdot (3 - 1) = 0.1$$

Select chromosomes for mutation

selected chromosome set = $\{C_1, C_9, C_{15}\}$

Select one of selected chromosomes randomly and remove it from the set

mutation chromosome = $\{C_9\}$

$$X = C_9 = \begin{Bmatrix} 1.2 \\ 2.1 \\ \vdots \\ 2.9 \\ 1.22 \end{Bmatrix}$$

Form a random chromosome

$$R = \begin{Bmatrix} 0.12 \\ -0.21 \\ \vdots \\ 0.02 \\ -0.22 \end{Bmatrix}$$

Generate the output chromosome

$$Y = \begin{Bmatrix} 1.20 \\ 2.10 \\ \vdots \\ 2.90 \\ 1.22 \end{Bmatrix} + 0.1 \cdot \begin{Bmatrix} 0.12 \\ -0.21 \\ \vdots \\ 0.02 \\ -0.22 \end{Bmatrix} = \begin{Bmatrix} 1.212 \\ 2.079 \\ \vdots \\ 2.902 \\ 1.198 \end{Bmatrix}$$

Check the output chromosome's genes for boundary exceedings

Continue till the selected chromosome set is empty

An example of the mutation operation of a real coded genetic algorithm

As stated earlier, another mutation which prevents convergence to average value of a population is boundary mutation. Boundary mutation takes its name from the fact that it forces certain components of relocated chromosomes to take values at the boundary of the feasible region. Therefore, new chromosomes with genes that have only 2 values, upper boundary and lower boundary of variable, are created. The number of genes that have upper boundary or lower boundary is random and determined randomly.

2.3.2.3. The Crossover Operation of a Binary Coded Genetic Algorithm

The first crossover operation for the binary coded genetic algorithm is the customary one. It is the direct analogue of crossover in mitosis. The operation associated with this crossover operation are as in for the crossover operation ones in real coded genetic algorithm, which is described above, except multiplication with a random number part. The operations associated with this crossover operation are illustrated in the example shown in the next page.

2.3.2.4. Mutation Operation of a Binary Coded Genetic Algorithm -1-

The first mutation operation is the customary one. This mutation operation is performed on genes rather than chromosomes.

Before starting to describe this operation it should be noted that the probability of mutation in this operation is defined for individual genes rather than chromosomes. This operation begins with the formation of the mutation pool. This pool is the set of chromosomes whose genes will be candidates of mutation. Then a sequence of random numbers between 0 and 1 is generated so that each gene in the mutation pool has a corresponding number. Then, genes whose random number is less than or equal to mutation probability toggles its value. An example of the steps of this mutation operation is given.

Select chromosomes for crossover

selected chromosome set = $\{C_3, C_7, C_9, C_{18}\}$

Select two of selected chromosomes randomly and remove them from the set

parents = $\{C_3, C_{18}\}$

$$C_3 = \begin{Bmatrix} 1 \\ 1 \\ 0 \\ \vdots \\ 1 \\ 0 \end{Bmatrix}, C_{18} = \begin{Bmatrix} 0 \\ 0 \\ 1 \\ \vdots \\ 0 \\ 1 \end{Bmatrix}$$

Select crossover point

crossover point = 2

Form offspring

$$O_1 = \begin{Bmatrix} \begin{Bmatrix} 1 \\ 1 \end{Bmatrix} \\ \begin{Bmatrix} 1 \\ \vdots \\ 0 \\ 1 \end{Bmatrix} \end{Bmatrix}, O_2 = \begin{Bmatrix} \begin{Bmatrix} 0 \\ 0 \end{Bmatrix} \\ \begin{Bmatrix} 0 \\ \vdots \\ 1 \\ 0 \end{Bmatrix} \end{Bmatrix}$$

Continue till the selection set is empty

An example of the crossover operation of the binary coded genetic algorithm

Probability of mutation = 0.05
Chromosomes length = N
Form mutation pool
Mutation pool = $\{C_1, C_7, C_8, C_{14}\}$
Take one chromosome from pool end remove it from pool
Taken chromosome = C_1
Create N uniformly distributed random numbers between 0 and 1
Random Numbers = $\{0.01, 0.21, 0.30 \dots, 0.05, 0.51\}$
Determine genes for mutation and toggle their values

$$C_1 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}, R = \begin{bmatrix} 0.01 \\ 0.21 \\ 0.99 \\ \vdots \\ 0.05 \\ 0.51 \end{bmatrix} \rightarrow \begin{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} & \begin{bmatrix} 0.01 \\ 0.21 \\ 0.99 \\ \vdots \\ 0.05 \\ 0.51 \end{bmatrix} \end{bmatrix} \rightarrow Y = \begin{bmatrix} 1 \\ 1 \\ 0 \\ \vdots \\ 1 \\ 1 \end{bmatrix}$$

Repeat till mutation pool is empty

An example of the mutation operation of the binary coded genetic algorithm -1-

2.3.2.5. Mutation Operation of a Binary Coded Genetic Algorithm -2-

The second mutation operation used operates on chromosomes rather than genes. In the previous mutation operation, the number of 1s and 0s alters during mutation. In some problems a mutation operation which preserves number of 1s and 0s is needed. The simplest idea is to perform usual mutation and check and correct the number of 1s and 0s. However, this kind of check and correction reduces population diversity which mutation operation assures. Therefore, since this mutation operation operates on a single chromosome and preservation of 1s and 0s is needed, an operation on the chromosome itself is used.

This operation starts with the selection of chromosomes for mutation. Then, two genes from selected chromosome are determined. The elements of the chromosome genes between the determined genes are extracted from the chromosome. The reduced chromosome is called “Intermediate Chromosome” and the extracted part of genes is called “Mutation Genes”. Then, again a gene from the intermediate chromosome is selected. The mutation genes are inserted to the intermediate chromosome of the selected. This forms the output chromosome. Note that for this operation mutation probability is defined for chromosomes. An example of the steps of this operation is given on the next page.

2.3.2.6. Mutation Operation of a Binary Coded Genetic Algorithm -3-

The third mutation operation used also operates on chromosomes. This operation also preserves number of 1s and 0s. The operation starts with selection of chromosomes for mutation. Then, two sets of indices are constructed. The first set contains indices of 0s on the mutating chromosome; whereas, the second set contains indices of 1s of the same chromosome. Then, one element from each set is determined. Then, the determined element of each set is interchanged. Number of elements that will be interchanged depends on the problem. Finally, from the index sets output chromosome is formed. This operation is illustrated in the following example.

Chromosomes length = N

Number of 1s in chromosomes = N_0

Number of 0s in chromosomes = N_1

Select chromosomes for mutations

selected chromosome set = $\{C_3, C_7, C_9, C_{18}\}$

Select one of selected chromosomes randomly and remove it from the set

selected chromosome = $\{C_3\}$

$$C_3 = \begin{Bmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \\ 1 \end{Bmatrix}$$

form the sets of indices

$S_0 = \{1, 3, \dots, N-1\}$ set of 0s

$S_1 = \{2, \dots, N\}$ set 1s

Select one element from each set

S_0 element no = 2

S_1 element no = N_1

Interchange the elements of sets

$S_0 = \{1, N, \dots, N-1\}$

$S_1 = \{2, \dots, 3\}$

Form the output from the sets

$$Y = \begin{Bmatrix} 0 \\ 1 \\ 1 \\ \vdots \\ 0 \\ 0 \end{Bmatrix}$$

Continue till selected chromosome set is empty

An example of mutation operation for the binary coded genetic algorithm -

Probability of mutation = 0.05

Chromosomes length = N

Select chromosomes for mutations

selected chromosome set = $\{C_3, C_7, C_9, C_{18}\}$

Select one of selected chromosomes randomly and remove it from the set

selected chromosome = $\{C_3\}$

$$C_3 = \begin{Bmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \\ 1 \end{Bmatrix}$$

Select two genes from selected chromosome randomly

selected genes = $\{2, 3\}$

Form intermediate chromosome and mutation genes

$$\text{Intermediate chromosome} = \begin{Bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{Bmatrix}$$

$$\text{Mutation genes} = \begin{Bmatrix} 1 \\ 0 \end{Bmatrix}$$

Determine a gene in intermediate chromosome randomly

selected gene = $(N - 2) - 1$

Paste mutation genes after determined gene and form the output chromosome

$$Y = \begin{Bmatrix} 0 \\ \vdots \\ 0 \\ \begin{Bmatrix} 1 \\ 0 \end{Bmatrix} \\ 1 \end{Bmatrix} \rightarrow \begin{Bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ 1 \end{Bmatrix}$$

Continue till the selection set is empty

An example of mutation operation for the binary coded genetic algorithm -2-

2.3.2.7. Selection Operation

The selection operation used can be thought as a simulation of roulette wheel as stated previously. This operation can also be thought as the game of dart. As in the roulette, the dart board has slots proportional to the fitness of chromosomes. To perform a selection, a dart spear is thrown without aiming any specific slot on board. Consequently, a probability proportional to the size of each slot on the dart should be defined to perform selection properly.

Let us define F_i as the fitness of chromosome i in the population composed of N chromosomes. Then, the selection probability of each chromosome is defined as follows:

$$F_{total} = \sum_{n=1}^N F_n \quad (2.43)$$

$$p_i = \frac{F_i}{F_{total}} \quad (2.44)$$

However, slots are not defined by just finding probabilities, since a slot is defined as an interval with a starting point q_i . The starting slot (i.e. interval is defined as) :

$$q_i = \sum_{j=1}^i p_j \quad (2.45)$$

Since uniformly distributed real random numbers can be generated in computers, to simulate a dart spear throw the uniformly distributed random number generators can be used. Therefore, a real random number, r , between 0 and 1 is generated to simulate dart spear throw. After that, by checking the interval it falls into using the standing points of the interval, $q_{i-1} < r \leq q_i$, which slot has been hit is found. When there is a hit in the i^{th} interval, the i^{th} chromosome is put to the next generation.

It should be noted that this operation allows us to select the number of chromosomes that will be carried to next generation, since the number of dart spear throw depends on the user. The number of selections depends on the problem, and should be determined by the programmer.

2.3.3. Near Global Optimum Algorithm

As stated previously, genetic algorithms do not guarantee a hit to a global optimum solution. When it is thought that the fitness of a solution is close enough to the global optimum value, it is convenient to use another algorithm to reach a global optimum solution quickly.

The selection of such an algorithm depends on the problem. However, for problems with very large solution spaces, and for optimization functions that are not continuous or that do not have 1st and 2nd derivatives, Steepest Descent or Newtonian like algorithms can not be utilized. Moreover, since these algorithms require line search techniques, and since function evaluation is costly for the problem at hand, these algorithms are not suitable for the present problem.

The suitable algorithms that can be used are the one that do not require line search and 1st or 2nd derivatives of the cost function. Although there are lots of algorithms in this class, “Cycling Coordinates” is one of the simplest and straightforward.

Cycling coordinates algorithm searches for increase (or decrease) in the direction of each orthonormal basis vector of the solution space. The direction of search is determined by one function evaluation, and step size of algorithm is determined using additional function evaluations. After the step size is determined, the solution is updated.

The update equation for this algorithm is:

$$x_{k+1} = x_k + \alpha_k e_j \quad (2.46)$$

where k is the iteration number, e_j is the j^{th} orthonormal basis vector in N dimensional solution space, α_k is a constant which determines the step size and the search direction. Note that j is an integer from 1.. N . It is determined according to:

$$j = \begin{cases} \text{mod}(k, N) & \text{if } k \neq nN \\ 1 & \text{otherwise} \end{cases} \quad \text{for an integer } n \quad (2.47)$$

The operations associated with the algorithm are shown in Figure 2.7.

```

Initialize;
k = 1;
j = 1;
while k ≤ iteration No
    j = mod(k, N);
    if j = 0
        j = 1;
    endif
     $\alpha_{k,1} = \alpha_0$ ;
     $x_{k+1} = x_k + \alpha_{k,1}e_j$ ;
    if FunctionValue( $x_{k+1}$ ) < FunctionValue( $x_k$ )
         $\alpha_{k,1} = -\alpha_0$ 
         $x_{k+1} = x_k + \alpha_{k,1}e_j$ ;
    endif
    i = 2;
    while FunctionValue( $x_{k+1}$ ) > FunctionValue( $x_k$ )
         $\alpha_{k,i} = 2\alpha_{k,i-1}$ ;
         $x_{k+1} = x_k + \alpha_{k,i}e_j$ ;
        i = i + 1;
    endwhile
     $x_{k+1} = x_k + \alpha_{k,i-1}e_j$ ;
    k = k + 1;
endwhile

```

Figure 2.7: Cycling Coordinate algorithm
evaluation steps

In the initialization section of the algorithm, initial solution and initial step size, α_0 , are entered. In the direction determination part of the algorithm, the direction of increase is checked. This is due to the fact that the given algorithm is for a maximization problem.

The convergence rate of this algorithm is not high, but excluding the time needed for function evaluations of other nonlinear programming algorithms, this algorithm will converge faster.

2.4. Implementation

In this section, implementation details of the problem are given. Firstly, the parts of the problem are clarified. Secondly, fitness function derivations and implementation details of each part are given. Finally, evolution of the solution is given.

2.4.1. Problem Definition

The problem of generating magnetic field with desired properties is the most important step for all types of devices whose operation depends upon magnetic field distribution. For a system that uses permanent magnets for the generation of magnetic fields, a methodology for optimization of permanent magnet structures to obtain desired magnetic field distribution in some region of space should be developed.

Magnetic field distribution and strength of magnetic field of a permanent magnet is a function of magnetization and geometry of the magnet as stated previously. This can also be rephrased as; induction between a source and a target point is a function of properties of the media and the source. Therefore, source distribution and source strength are the parameters related to the source that change the magnetic field's distribution and strength at a target point. Although a single source (single magnet for this case) can be used, multiple sources can also be used to obtain the desired strength and magnetic field distribution at target points.

In this work, many unit magnets will be used to generate remote magnetic field with desired properties. This can be thought as a four part problem:

1. Determination of the shape of unit magnets that will be used as a unit source of magnetic field.
2. Optimization of the shape parameters to determine exactly the shape of the unit magnet.
3. Optimization of the surface of the unit magnet which has a shape determined in previous steps.

4. Optimization of placement of these unit magnets in a specific region to generate a magnetic field with desired properties at some region of space.

The first three steps determine exactly the unit magnets that are used in the final step. Therefore, in these steps problem should be defined so that the objective to generate magnetic field with desired properties will be achieved in smaller number of iterations in the final step.

The first step is not studied in this work. The main geometry of unit magnets is assumed to have square prism shape. Second step has studied in [7] and [8]. However, in this work, the results of [8] are used. Correctness of the fitness function is checked by trying to obtain the same results. Third and fourth parts of the problem are the main objectives of this work.

As stated previously, permanent magnet structures have a drawback that does not allow them to be used in applications that require high field strength in deeper regions (points of space far away from the surface of magnet). Therefore, the natural aim of the third step of the problem is to generate magnetic field that does not scatter through outer regions (outer regions will be described later). The dual of this objective is to localize the magnetic energy in space due to a permanent magnet. Therefore, the main problem of the third step is to *maximize magnetic energy in some region of space by adjusting the surface of magnetic energy source*.

The final step of this process is the most vital one since it directly aims to generate the magnetic field in a region of interest. In this step, resultant magnet of third step is used as an input and the problem is extended to many magnets as magnetic energy source. Placement of a unit magnet in some region to obtain optimized results with respect to previously defined requirements of the problem implies optimization of three independent variables. These variables are the location of centre of the magnet, the azimuth and elevation angles of the magnet.

The elevation angle of unit magnets directly determines the effective pole of magnet on the region of interest. Therefore, this parameter can be defined as a two state

variable instead of defining this parameter as a continuous variable; one state for N pole facing the region of interest and one state for S pole facing the region of interest. In this case, the elevation angle parameter is named as the orientation parameter. Consequently, the fourth step's objective is to *obtain a desired magnetic field distribution by adjusting locations, azimuth angle and orientation of unit magnets*.

The implementation details and fitness function derivations of these problems are given in the next sections.

2.4.2. Determination of Unit Magnets

As stated previously, the shape of magnets is not determined in this work. The results of [8] is directly used.

In [8], the shape of magnet is taken as square prism and side length and height is tried to be optimized in the sense of maximum magnetic field strength transferred to deeper regions of space. In Figure 2.8, results of [8] are visualized.

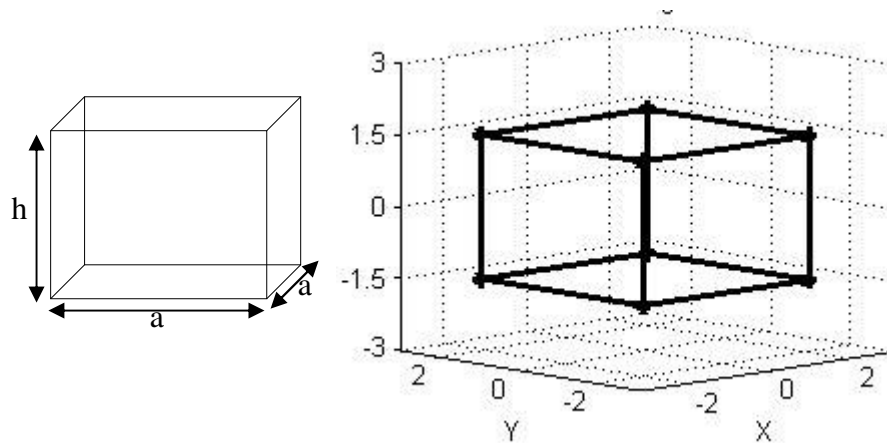


Figure 2.8: Unit magnet and its dimensions

As shown in Figure 2.8, the unit magnet shape is given by ratio $\frac{h}{a} = 0.75$. In Figure 2.8, the N pole is the top side of the prism with $+z$ values, and the S pole is the bottom side of the prism with $-z$ values.

Any surface can be described by $z = f(x, y)$. Therefore, z determines the surface of any geometrical object. Consequently, in this part of the problem the distribution of z is sought. To make the problem feasible and solvable, a range of z should be defined.

$h = 3, a = 4$ are taken as a test case. The magnet's centre is placed to the origin of laboratory frame of reference, as shown in Figure 2.8. Then, the program is allowed to adjust the surface of poles by changing $z = h$ variable $\pm 5\%$ of its original value, 1.5. In other words, z is defined in the range $[1.425, 1.575]$.

Magnetic energy in free space is given by:

$$E = \int_V \vec{B} \cdot \vec{H} dv \quad (2.48)$$

Using the definition of magnetic induction, a formula for energy in terms of the magnetic field strength is obtained.

$$\vec{B} = \mu_0 \vec{H} \quad (2.49)$$

$$E = \mu_0 \int_V \vec{H} \cdot \vec{H} dv = \mu_0 \int_V \|\vec{H}\|^2 dv \quad (2.50)$$

Since the main drawback of permanent magnets' magnetic fields is the rapid decrease in the magnetic field strength when the filed point is moved away from the surface of the magnet, the natural objective is to maximize the magnetic field strength in the region of interest. This corresponds to the maximization of magnetic energy due to the fact that maximization of magnetic field strength is equivalent to maximization of square of magnetic field strength since magnetic field strength is non-negative function. Therefore, by comparing the definition of magnetic energy, (2.50), with maximization function of square of magnetic field strength, this problem

can be defined as *maximize the magnetic energy in the specific region of space by changing z variable*.

This definition of problem implies minimization of magnetic energy in regions exterior to the desired region of space. Therefore, the problem should contain a minimization part. This part of the problem can be stated as *minimize the magnetic energy in exterior to the regions of region of interest*.

Magnetic energy of a permanent magnet, generally, is given by:

$$E = \int_{V_{magnet}} \vec{M} \cdot \vec{B} dv \quad (2.51)$$

For magnets placed in free space, (2.51) is modified:

$$E = \frac{1}{\mu} \int_{V_{magnet}} \|\vec{M}\|^2 dv \quad (2.52)$$

The magnetic energy in the space due to a permanent magnet can not exceed magnet's total energy which is given by (2.51) and (2.52). In (2.51) and (2.52) the integrals are evaluated in the volume of magnet. Therefore, the energy of magnet is a function of magnet's volume when the magnetization vector of the magnet is assumed known and constant. Consequently, the volume of the magnet should be constant during the optimization procedure.

The resultant definition of the problem becomes:

By adjusting the z variable maximize the magnetic energy in the region of interest while minimizing the magnetic energy in the regions exterior to the region of interest, keeping the volume of the magnet constant.

Before giving the mathematical representation of this problem, let us define the regions of space. From now on, the region of interest will be abbreviated by ROI. In Figure 2.9, average magnet and ROI (a), and magnet, ROI and exterior regions (b) are shown. In Figure 2.10, ROI and exterior regions are shown in 2D as the top view of Figure 2.9(b).

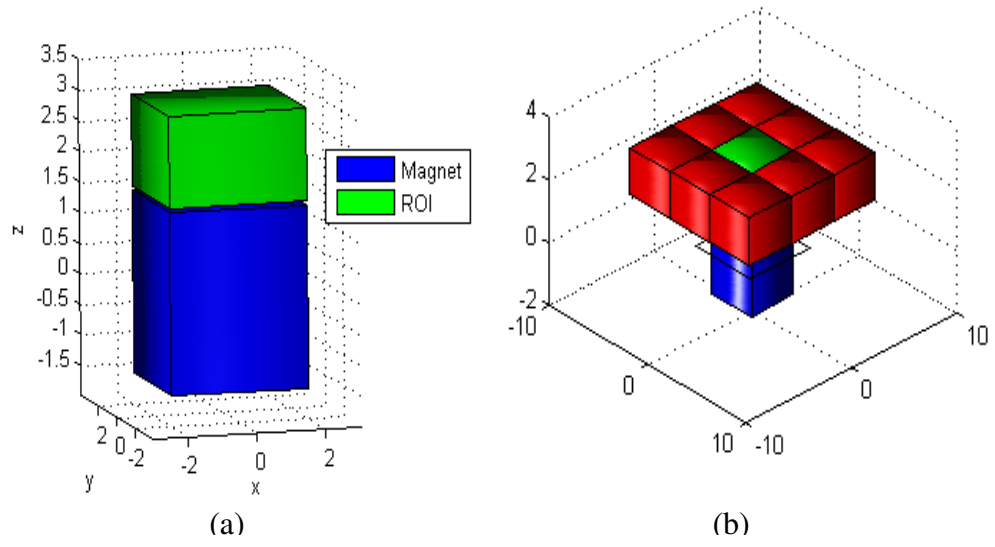


Figure 2.9: (a) Magnet (blue) and ROI (green);
(b) Magnet (blue), ROI (green) and exterior regions (red)

Exterior7	Exterior6	Exterior5
Exterior8	ROI	Exterior4
Exterior1	Exterior2	Exterior3

Figure 2.10: Region definitions in 2D

Any optimization problem including maximization and minimization objectives can be combined mathematically as:

$$F = \alpha_0 F_{\max} - \sum_{i=1}^N \alpha_i F_{\min_i} \quad (2.53)$$

where α_i s are some constants, satisfying $\sum_{j=0}^N \alpha_j = 1$, and N is the number of minimization or constraining functions.

For the problem at hand, F_{\max} is defined by:

$$F_{\max} = \int_{V_{ROI}} \|\vec{H}\|^2 dv \quad (2.54)$$

Similarly, the objectives of the problem to be minimized can be defined as:

$$F_{\min_1} = \sum_{i=1}^8 \int_{V_{outer_i}} \|\vec{H}\|^2 dv \quad (2.55)$$

Since the volume of the magnet is desired to be fixed, another function that guarantees this fix in the volume is required. This constraint is imposed on the problem as a penalty using the square norm function. Consequently, the second minimization function takes the form shown in (2.56).

$$F_{\min_2} = K (V_0 - V_{magnet})^2 \quad (2.56)$$

The resultant objective function of the problem is given below (2.57).

$$F = \alpha_0 \int_{V_{ROI}} \|\vec{H}\|^2 dv - \alpha_1 \sum_{i=1}^8 \int_{V_{outer_i}} \|\vec{H}\|^2 dv - \alpha_2 K (V_0 - V_{magnet})^2 \quad (2.57)$$

where K is any positive constant that reflects the importance of the constant volume constraint.

To maximize F , real coded genetic algorithm has been used. The chromosomes of this algorithm are z values of each node of surface grid shown in Figure 2.1. Therefore, the chromosome consists of the z coordinates of P_i 's in expression (2.42).

2.4.3. Placement of Unit Magnets in Space

Magnetic field properties generated at a region of space is a function of source strength, source location, distribution of sources and properties of medium.

Assuming linear isotropic free space as medium, optimization of magnetic field generated by permanent magnets requires optimization of locations and distributions of permanent magnets.

The problem of the placement of a unit magnet in a region of space contains optimization of three different variables. The first one is the point where the centre of the magnet will be placed. The second one is the angle at which the magnet will be rotated (azimuth angle). The third is the orientation of the magnet.

2.4.3.1. Variable Definitions

To determine the point at which the centre of the magnet will be placed is the problem of determination of a point in 3D space. However, since the magnets should not be placed on the top of one another, determination of the center point in the 2D plane and leaving the other spatial variable as another optimization variable is a better treatment.

In the plane magnet placement region can be defined as a grid. Each segment of the grid is centered at a point at which the centre of unit magnet can be placed. The grid is shown in Figure 2.11.

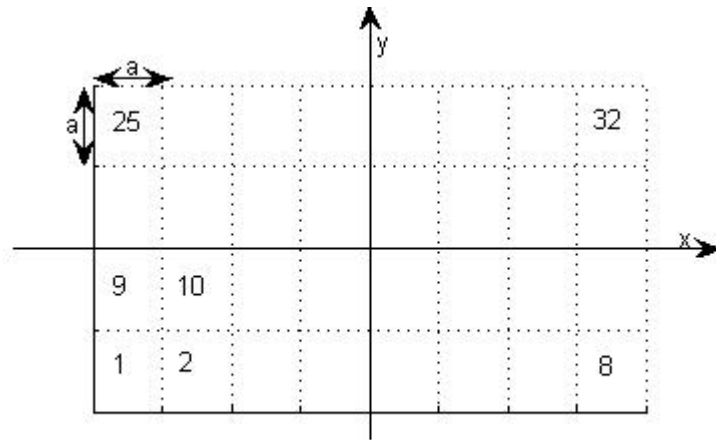


Figure 2.11: Magnet Placement Region (MPR) in 2D

From now on magnet placement region is abbreviated as MPR. As shown in Figure 2.11, each unit magnet can be placed in a segment which is a square with side length a . a should be chosen so that the magnet can be rotated inside the grid segment. Since the unit magnets are in square prism shape $a = \sqrt{2}\hat{a}$ should be satisfied to ensure the rotation of magnets inside a grid where \hat{a} is the side length of a unit magnet. The numbers shown in Figure 2.11 are the ID numbers of each location. Since this is a placement problem, 3rd coordinate variable (for this problem it is the z -coordinate) should be found. In Figure 2.12, MPR is shown in 3D.

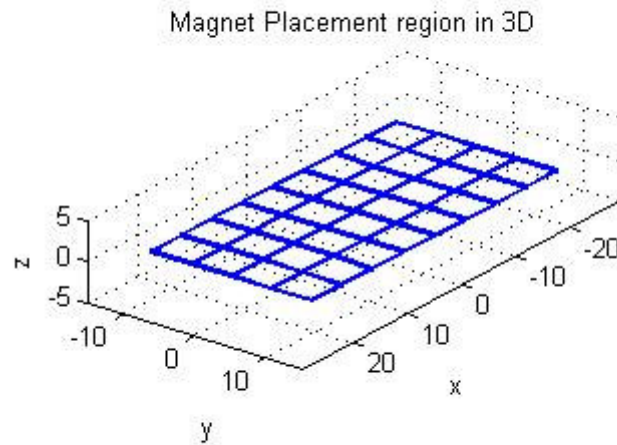


Figure 2.12: Magnet Placement Region (MPR) in 3D

Figure 2.12 shows only the grid of MPR, but not the 3rd variable. In figure 2.13, magnet placement region with two unit magnets placed having the same but opposite z values is shown.

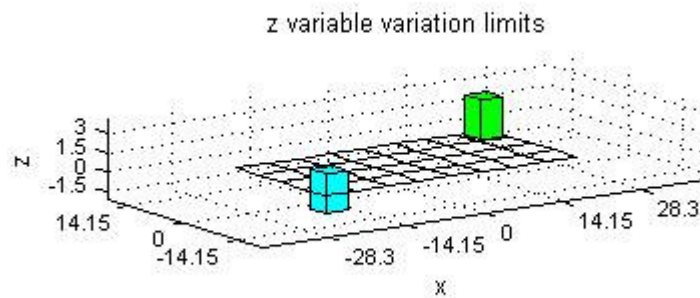


Figure 2.13: z limits

As it is shown in Figure 2.13, the variation of z variable is not limited to discrete levels; it is a continuous variable. Top view and side view of magnets shown in Figure 2.13 are given in Figure 2.14:

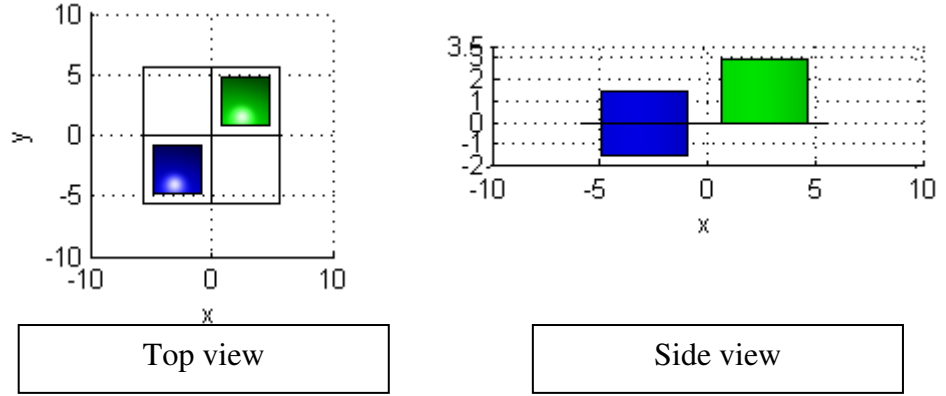


Figure 2.14: Top and side views of z limits

As shown in Figure 2.13 and in Figure 2.14, z is in the interval $[0, 1.5]$. In both figures, the unit magnets shown have side lengths and heights as defined in the previous section.

For the placement problem, a unit magnet can be rotated in a grid. Since the rotation of unit magnets around their principle axis changes the magnetic field distribution, angle of rotation is another variable of the problem. In Figure 2.15, angles of rotation for two unit magnets are shown. In Figure 2.15, blue magnet is rotated $\frac{\pi}{4}$ degrees and green one is rotated $\frac{\pi}{8}$ degrees.

Another variable which changes the magnetic field distribution in a region is the orientation of the unit magnets (by the orientation of a magnet, pole facing to the ROI is meant), due to the fact that the magnetic field in a region due to more than one magnet is the superposition of magnetic fields generated by each magnets.

Therefore, the magnetic field strength increases or decreases according to the relative alignment of magnets.

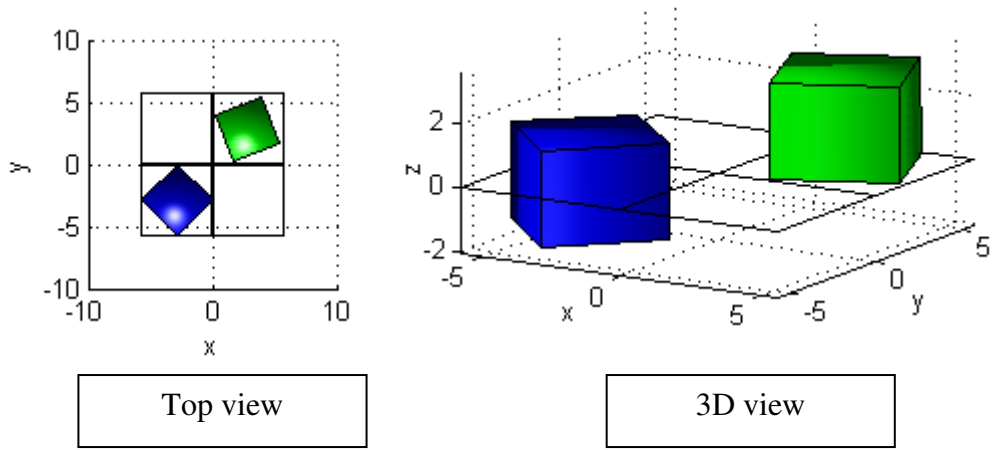


Figure 2.15: Different Angles of rotation

Consequently, two variables that vary in discrete steps, and two variables that vary continuously are at hand. If in a grid region there is a magnet then the value of the placement variable is 1, otherwise it is 0. The height, where the centre of the magnet is placed, can be changed continuously. The rotation angle is also allowed to change continuously which is a direct consequence of the effect of rotation angle of magnet on the magnetic field distribution. Lastly, the orientation variable is also discrete since it can only take two states representing which pole of the magnet points ROI.

Placement variable is associated with a grid region as shown in Figure 2.11. This variable is used to represent existence of a unit magnet in a grid segment. Therefore, a string of 0s and 1s can be formed to represent where the unit magnets are placed in MPR. In Figure 2.16, formation of strings is shown.

In Figure 2.16, No is the id number of a grid segment, Value shows whether a grid segment contains a unit magnet or not (1 if it contains, 0 if not), Position is the coordinate of grid segments' centre.

No	Value	Position
1	1	(x_1, y_1)
2	1	(x_2, y_2)
\vdots	\vdots	\vdots
31	0	(x_{31}, y_{31})
32	1	(x_{32}, y_{32})

Figure 2.16: Placement Strings

Center positions of the grid segments are determined when MPR is created. These values are stored to be used while fitness function is evaluated. It is the Value row that represents the variable of the placement problem. Therefore, the chromosomes of genetic algorithm are composed of the Value row of Figure 2.16.

Orientation variable represents which pole of magnet is facing ROI. Therefore, it is also a two state variable where 0 means S pole facing ROI, 1 means N pole facing ROI. As in placement variable, a string of 0s and 1s is constructed to represent the orientation of each unit magnet.

Height and rotation angle variables are real valued. Therefore, these variables are represented by real valued strings (sequences). The values in the strings should be in the feasible range of corresponding variables.

2.4.3.2. Region Definitions

As in the previous section, ROI with respect to MPR should be defined. In Figure 2.17, MPR and ROI are shown to illustrate how the regions are in 3D space. In Figure 2.18, ROI and MPR when MPR is filled with unit magnets is shown.

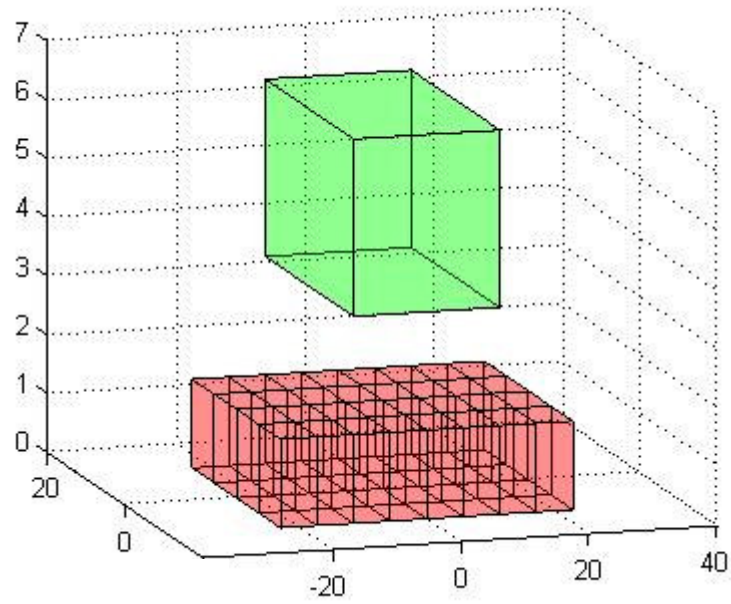


Figure 2.17: MPR (green object) and ROI (red object) in 3D

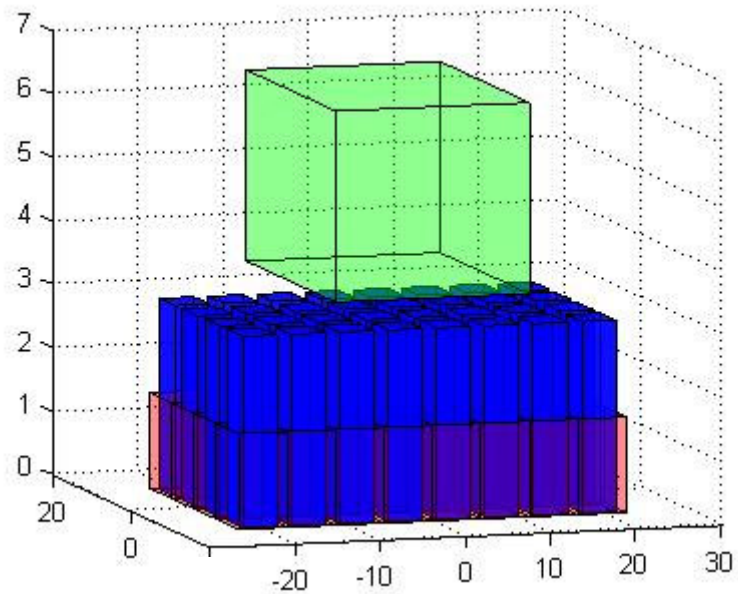


Figure 2.18: ROI (green object) and MPR (red object) with unit magnets (blue objects)

Since ROI affects the solution directly, the size of it should be determined to obtain good results. However, it is also possible to keep ROI sufficiently large so that the solution is not satisfactory but enables us to determine a portion of it so that magnetic field distribution in that portion has the desired properties.

2.4.3.3. Fitness Function Derivation

Since the aim is to generate a magnetic field distribution with desired properties in a specific region, magnetic field properties should be defined mathematically. As in the previous section, it is aimed to obtain maximum strength of magnetic field in the ROI. Remaining properties of the magnetic field in the ROI can vary according to the requirements of the problem. In general, the problem can be defined as the maximization of magnetic field energy in a specific region subject to some constraints on the solution space of the problem. Mathematically, these can be expressed as:

$$\text{maximize} \left\{ \int_{V_{ROI}} \|\vec{H}\|^2 dV - F_{\min} \right\}$$

where \vec{H} is the magnetic field intensity, and F_{\min} is a function that represents other requirements which varies according to the requirements of the problem.

For this problem, the objective function is given as:

$$F = \alpha_1 \int_{V_{ROI}} \|\vec{H}\|^2 dv - \sum_{i=2}^N \alpha_i F_{\min_{i-1}} \quad (2.58)$$

where $F_{\min_{i-1}}$ is defined according to problem requirements.

For F to represent a maximization function for magnetic field distribution that will be used in Magnetic Resonance Imaging (MRI), some constraints to be minimized related to either magnetic field strength or magnetic field components should be defined.

The dependence on magnetic field strength is due to the fact that, magnetic field distribution in MRI generates level surfaces of Larmor frequencies. This can be shown as:

$$\omega = \gamma \|\vec{B}\| \quad (2.59)$$

$$S_\omega = \{\vec{x} \mid \omega = \gamma \|\vec{B}(\vec{x})\|\} \quad (2.60)$$

On the other hand, dependence on magnetic field components can be explained by the fact that when a magnetic field generated has the same magnitude but different direction, for a sample containing nonzero susceptibility value will produce a counter magnetic field that changes the overall magnetic field distribution, and the shape of S_ω . This is one of the most vital problems in MRI due to the fact that for imaging modalities it is assumed that the frequencies at each point are known. However, one can assume existence of paramagnetic materials in the region of interest and applied magnetic field strength is not so high to take the materials to the saturation region. These assumptions will enable us to safely assume that, there is no change in orientation of magnetic fields due to some interactions of materials inside the ROI. Therefore, only the shape of S_ω surfaces (i.e., magnetic field strength distribution) is optimized.

Since the physical limits of change of direction of magnetic field that satisfy assumptions are not exactly known, it is assumed that the direction requirements can be satisfied easily, by adding some extra minimization functions to (2.58). In this work, magnetic field strength distribution in ROI is tried to be optimized.

A proper mathematical definition of the problem should be made. As stated earlier, the magnetic field strength in ROI should be maximized subject to some constraints on the magnetic field distribution. As a test case, linear variation of isosurfaces of magnetic field strength is chosen. This statement implies the derivatives of strength of magnetic field to be constant and second derivatives to be identically zero. In a more formal way:

$$\frac{\partial \|\vec{H}\|}{\partial x} = C_x, \frac{\partial \|\vec{H}\|}{\partial y} = C_y, \frac{\partial \|\vec{H}\|}{\partial z} = C_z \quad (2.61)$$

$$\frac{\partial^2 \|\vec{H}\|}{\partial x^2} = 0, \frac{\partial^2 \|\vec{H}\|}{\partial y^2} = 0, \frac{\partial^2 \|\vec{H}\|}{\partial z^2} = 0 \quad (2.62)$$

$$\frac{\partial^2 \|\vec{H}\|}{\partial x \partial y} = 0, \frac{\partial^2 \|\vec{H}\|}{\partial x \partial z} = 0, \frac{\partial^2 \|\vec{H}\|}{\partial z \partial y} = 0 \quad (2.63)$$

Therefore, for this part, the problem can be defined to *maximize magnetic energy while keeping the variation of strength of magnetic field constant in ROI*.

Expressions (2.61) to (2.62) should be defined as minimization functions. This can be achieved by introducing barrier and penalty functions. Penalty and barrier functions are defined using *max* and *min* functions when the limits of linear variation are defined. Before giving these functions, let us define limits of variation:

$$\begin{aligned} C_{xMIN} &\leq \frac{\partial \|\vec{H}\|}{\partial x} \leq C_{xMAX} \\ C_{yMIN} &\leq \frac{\partial \|\vec{H}\|}{\partial y} \leq C_{yMAX} \\ C_{zMIN} &\leq \frac{\partial \|\vec{H}\|}{\partial z} \leq C_{zMAX} \end{aligned} \quad (2.64)$$

Therefore, the barrier functions of these limits are:

$$\begin{aligned} P_x &= \left(\min \left(\left(C_{xMAX} - \left| \frac{\partial \|\vec{H}\|}{\partial x} \right| \right), 0 \right) + \min \left(\left(\left| \frac{\partial \|\vec{H}\|}{\partial x} \right| - C_{xMIN} \right), 0 \right) \right)^2 \\ P_y &= \left(\min \left(\left(C_{yMAX} - \left| \frac{\partial \|\vec{H}\|}{\partial y} \right| \right), 0 \right) + \min \left(\left(\left| \frac{\partial \|\vec{H}\|}{\partial y} \right| - C_{yMIN} \right), 0 \right) \right)^2 \\ P_z &= \left(\min \left(\left(C_{zMAX} - \left| \frac{\partial \|\vec{H}\|}{\partial z} \right| \right), 0 \right) + \min \left(\left(\left| \frac{\partial \|\vec{H}\|}{\partial z} \right| - C_{zMIN} \right), 0 \right) \right)^2 \end{aligned} \quad (2.65)$$

Since the second derivatives of magnitude of magnetic field components are desired to be identically 0, penalty functions of those constraints are defined as:

$$\begin{aligned}
P_{xx} &= \left(\max \left(\left| \frac{\partial^2 \|\vec{H}\|}{\partial x^2} \right|, 0 \right) \right)^2 & P_{xy} &= \left(\max \left(\left| \frac{\partial^2 \|\vec{H}\|}{\partial x \partial y} \right|, 0 \right) \right)^2 \\
P_{yy} &= \left(\max \left(\left| \frac{\partial^2 \|\vec{H}\|}{\partial y^2} \right|, 0 \right) \right)^2 & P_{xz} &= \left(\max \left(\left| \frac{\partial^2 \|\vec{H}\|}{\partial x \partial z} \right|, 0 \right) \right)^2 \\
P_{zz} &= \left(\max \left(\left| \frac{\partial^2 \|\vec{H}\|}{\partial z^2} \right|, 0 \right) \right)^2 & P_{yz} &= \left(\max \left(\left| \frac{\partial^2 \|\vec{H}\|}{\partial y \partial z} \right|, 0 \right) \right)^2
\end{aligned} \tag{2.66}$$

Therefore, minimizations functions in (2.58) can be defined as:

$$\begin{aligned}
F_{\min 1} &= P_x + P_y + P_z \\
F_{\min 2} &= P_{xx} + P_{yy} + P_{zz} + P_{xy} + P_{yz} + P_{xz}
\end{aligned} \tag{2.67}$$

(2.58) can be written as:

$$F = \int_{V_{ROI}} \left(\alpha_1 \|\vec{H}\|^2 - \alpha_2 F_{\min 1} - \alpha_3 F_{\min 2} \right) dv \tag{2.68}$$

2.4.3.4. Identification Systems

To keep the track of variables during function evaluation, two types of identification numbering systems have been used. First one is to give id numbers to grid segments; second one is to give id numbers to unit magnets. These numbering systems are visualized in Figure 2.19:

Segment No	Value	Position	Unit No
1	0	(x_1, y_1)	---
2	1	(x_2, y_2)	1
\vdots	\vdots	\vdots	\vdots
31	0	(x_{31}, y_{31})	---
32	1	(x_{32}, y_{32})	16

Figure 2.19: Segment No and Unit No

Figure 2.19 is an example of numbering systems for 32 grid segments and 16 unit magnets.

Grid segments' numbering is not allowed to change during the run of the program whereas unit numbering is allowed to change. ID number 1 in unit numbering system is always the first occurrence of 1 in the binary string of placement variable. Therefore, the units are numbered according to their positions in the placement strings. Grid numbering system is used to represent the placement variable, whereas unit numbering system is used in the representations of other variables.

2.4.3.5. *Fitness Function Evaluation*

Since the orientation, the height of centre of magnets and the angle of rotation are identified using unit magnet numbering system, the indices of 1s in placement strings are needed to be able to calculate the magnetic field in ROI. After finding the indices of 1s in placement string, the centre points of MPR segments are used from table that was previously stored. After that, for each 1 in the placement string, corresponding height, rotation angle and orientation value are found from corresponding strings. These values are used to calculate the magnetic field in ROI due to a single unit magnet. This procedure is repeated for the number of unit magnets and magnetic fields of all unit magnets are added to obtain the total magnetic field.

In, the magnetic field calculations it is assumed that the magnet is placed at the origin. Therefore, the centre points of unit magnets surface grid segments should be modified. Translation of origin from one point to another requires translation of every point that is defined referencing to the first origin. This translation is given by:

$$\begin{aligned} \bar{x} &= \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \bar{x}_0 = \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} \\ \hat{\bar{x}} &= \bar{x}_0 + \bar{x} \end{aligned} \quad (2.69)$$

where \bar{x}_0 is the coordinate of the new origin referenced to the previous one, \bar{x} is the original point, \hat{x} is the new point referencing to the previous origin. Therefore, only two vectors are added to find the centers of unit magnets' surface grid segments.

To find the effect of rotation of magnets in azimuth, before moving them to placement grid segments, (x, y) values of centre points of unit magnets' surface grid segments are multiplied by the rotation matrix of rotation R (2.70).

$$R = \begin{bmatrix} \cos \phi & -\sin \phi \\ \sin \phi & \cos \phi \end{bmatrix} \quad (2.70)$$

where ϕ is the angle of rotation.

In the theory part, it is assumed that a unit magnet's N pole is facing to the ROI. Therefore, when orientation is not 1, centre points of S pole of unit magnet surface grid segments should be interchanged with the ones of the N pole. Moreover, the direction of magnetization should be reversed if the S pole of the magnet is facing to the ROI.

After finding the centre points of unit magnet surface grid segments, the magnetic field is calculated using (2.41) for all nodes in the grid of unit magnet's surface. Afterwards, the same procedure is repeated for all unit magnets in a chromosome and the results are added to obtain the fitness value of that chromosome.

The derivatives of magnetic field magnitude in the ROI are found by using central differences between pixels. To find the neighboring pixels (voxel in 3D case), the neighboring pixels' indices are stored while constructing the ROI at beginning of the program. By using this convention, it is only needed to take differences of the values of magnitude of magnetic fields in the pixels and divide by the pixels size.

After calculating the magnetic field magnitude values and its first and second derivatives as explained, the only remaining thing is to add them up to obtain each part of (2.68) to find the fitness of a chromosome.

2.4.3.6. Used Genetic Algorithms

Four variables that should be optimized have been defined. Two of them (grid placement and orientation) are two state variables, whereas the other two (rotation angle and height of centre of magnet) are continuous variables which can take any value in the range specified. This definition of variables leads us to three different procedures of optimization; a procedure for optimization of real variables and two procedures for optimization of discrete variables. Although these variables are not independent when the solution of the problem using genetic algorithms is considered, it is quite safe to assume that these variables are independent.

A real coded genetic algorithm has been implemented to optimize the real valued variables. For optimization of binary coded chromosomes, two types of algorithms are needed due to the different nature of variables represented by these chromosomes.

Since the number of units is fixed, the genetic algorithm of the placement variable should not be able change number of 1s in a chromosome during crossover or mutation operations. In this work, for placement variable genetic algorithm, usual crossover (described in section 2.3.2.3) with number of 1s correction in the resultant chromosomes has been implemented. As a mutation operation, the permutation mutation operation (described in section 2.3.2.5) has been implemented.

Usual binary coded genetic algorithm operations have been used for the orientation variable optimization, since the number of magnets with specific orientation does not have to be fixed during the evolution of the algorithm. Therefore, the crossover is the one that was explained in the section 2.3.2.3. The mutation operation of the binary coded genetic algorithm is the one described in section 2.3.2.4.

2.4.4. Solution Steps

In this section the steps of the solution are given. First, surfaces of unit magnets are determined. After that, the steps of placement of unit magnets problem solution are given.

As stated previously, genetic algorithms do not guarantee convergence to a global optimum, but they provide a method to get closer to it. This fact leads us to use genetic algorithms till the solution gets closer to a global optimum solution. Afterwards, local optimum finding algorithms have been used. These two steps are followed in both parts of the solution. In the next sections, these steps are explained.

2.4.4.1. Determination of Unit Magnets' Solution Steps

Real coded genetic algorithm, which is explained previously, is summarized in Figure 2.20.

```
initialize;  
generate initial population  
evaluate fitness function (population)  
i=1  
while i≤maximum iteration number  
    find and hold the elitist chromosome  
    perform real coded genetic algorithm crossover operation  
    perform real coded genetic algorithm mutation operation  
    evaluate fitness function for new chromosomes  
    add B boundary chromosomes to chromosomes that are in selection set  
    perform selection for next generation  
    add the elitist chromosome to next generation  
    i=i+1  
end
```

Figure 2.20: Real coded genetic algorithm steps

The rate of change of the elitist chromosome's fitness value is used to determine the termination generation number of the genetic algorithm. If it is small enough to terminate the genetic algorithm, user passes the elitist chromosome of the last generation of the real coded genetic algorithm to the cycling coordinates optimization algorithm.

2.4.4.2. *Solution Steps of Placement of Unit Magnets'*

As stated previously, four genetic algorithms are performed at the same time for the optimization of placement of unit magnets problem. The steps of genetic algorithm are shown in Figure 2.21.

```

initialize;
create initial population
calculate fitness of initial population
i=1;
while i ≤ iterNo
  FOR HEIGHT, ANGLE, ORIENTATION AND PLACEMENT PARAMETERS
    perform crossover and create mutation pool
    perform mutation
  ENDFOR
  calculate fitness of offsprings of crossover operations
  calculate fitness of mutated chromosomes
  eliminate 2 chromosomes with worst fitness from real coded chromosomes
  create 2 boundary chromosomes for real coded parameters
  place generated chromosomes in the place of eliminated ones
  calculate fitness of voundary chromosomes
  perform selection for next generation
  add the previous population's elitist chromosome to next generation
endwhile

```

Figure 2.21: Evaluation of genetic algorithm

After the genetic algorithm, the output of genetic algorithm phase is transferred to near global optimum algorithm. For this part of the problem, two kinds of near global optimum algorithms are implemented. First algorithm is shown on Figure 2.22.

```
initialize;
while i ≤ iterNo
    find ellitest chromosome
    perform cycling coordinate on height parameter
    perform cycling coordinate on rotation angle parameter
    add obtained chromosome to population
    perform GA operations on placement and orientation parameters
    calculate fitness values of chromosomes
    perform selection for next generation
end
```

Figure 2.22: Evaluation of the first near optimum algorithm

Algorithm shown in Figure 2.22 does not converge fast enough due to the fact that four variables are not independent. This phenomenon can be understood better by defining dependent variables. It is obvious that height, rotation angle, and orientation variables are all independent of each other. However, these three variables are all dependent to the placement variable. In other words, orientation, height and rotation angle of unit magnets should be changed according to the places of unit magnets. Therefore, for example, if the placement order of any chromosome is changed and a better fitness for that chromosome is obtained, a better fitness is not assured while changing the height of the same chromosome. To obtain a better fitness value it is required to run another optimization algorithm. For specific height, orientation, and rotation angle strings of chromosomes, the genetic algorithm operations for the placement variable should preserve the other three variable values associated with

unit magnets. This idea leads us to define another near global optimum algorithm which takes into account the relation between variables.

The second near global optimum algorithm implemented takes into account the remarks made. It fixes three variables (height, rotation angle, and orientation variables) to unit magnets' ids. Therefore, when the place of a unit magnet is changed, the unit ids are also changed. If a change in the order of unit magnets occurred, the unit magnets' variables, associated with a specific unit magnet, are preserved. This algorithm preserves the number of unit magnets in MPR. Another important point that should be clarified is the independence of height, rotation angle and orientation variables. Since the algorithm is at a sub global optimum one, by just adjusting the places of these values the global optimum of whole problem can be reached. However, to guarantee the assumption of global optimum of these variables, cycling coordinate method is performed for the height and rotation angle variables.

The main difference of this algorithm from the previous one is in the operations of genetic algorithm used for the optimization of the placement variable. Permutation mutation has been used as crossover operation (described in section 2.3.2.5). Therefore, just one chromosome is needed for the crossover operation. Mutation operation described in section 2.3.2.6 is used. However, since the elitist chromosome has relatively high fitness value, only it can be used in the placement operations. On the other hand, it is quite possible for the elitist chromosome to be at a local optimum near to a global optimum. To prevent this kind of situations, more than one chromosome with high fitness values obtained from the genetic algorithm is used as an input for this algorithm. This algorithm is shown Figure 2.23.

As it is summarized in Figure 2.23, an improvement in the fitness value is expected after performing the placement operations. Due to this fact, an offspring chromosome which has a fitness value better than its parents and M additional chromosomes is expected. If there is no improvement for some specific steps, it is concluded that global optimum value is reached. Another important point that should be clarified is the fact that in this algorithm the orientation variable is not imposed to any operations. That is due to the fact that the orientation of unit magnets affects the

magnetic energy in ROI more than other variables. Therefore, orientation variable converges to its global optimum distribution faster than other variables. Consequently, it is assumed that this variable has already converged during the genetic algorithm phase.

```

initialize;
while i ≤ iterNo
    find M elitist chromosomes from population
    find maximum fitness valued M chromosome
    while offspring maximum fitness ≤ old maximum fitness
        perform placement parameter operations on M selected chromosomes
        calculate fitness of created offsprings
        find maximum fitness of offsprings
    end
    perform Cycling Coordinate algorithm on height and angle parameters
    update the population
end

```

Figure 2.23: Evaluation of the second near global optimum algorithm

2.5. Results

In this section, the results obtained for each part of the problem are given. First, the obtained results are shown for the unit magnet determination part. The results of the placement of unit magnets are shown.

2.5.1. Results of Unit Magnet Determination Part

First of all, the undefined parameter values should be defined. As stated previously, the magnets, shown in Figure 2.8, used in this work has dimensions of:

$$\begin{aligned} a &= 4 \\ h &= 3 \end{aligned}$$

units. Weighting coefficients in equation (2.57) are chosen to be:

$$\begin{aligned} \alpha_0 &= 0.25 \\ \alpha_1 &= 0.25 \\ \alpha_2 &= 0.5 \\ K &= 2000 \end{aligned}$$

The variable z is chosen to be in the interval $[1.425, 1.575]$. x and y values are discretized with step lengths 0.1 and 0.1, respectively. Therefore, the solution space is 1641 dimensional. The parameters of the real coded genetic algorithm are chosen as:

$$\begin{aligned} \text{mutation probability} &= 0.1 \\ \varepsilon \text{ of mutation operation} &= 0.015 \end{aligned}$$

In 1000 generations of the genetic algorithms, the fitness values of the elitist chromosomes are shown in Figure 2.24.

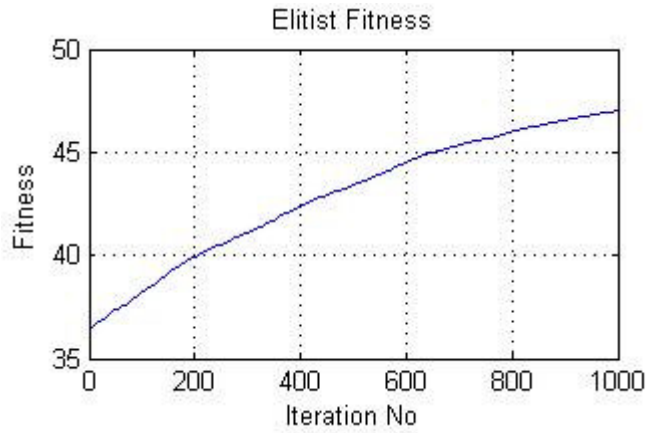


Figure 2.24: The evolution of elitist fitness values through generations

After 1000 generations of the genetic algorithm, the magnet's surface takes the form shown in Figure 2.25.

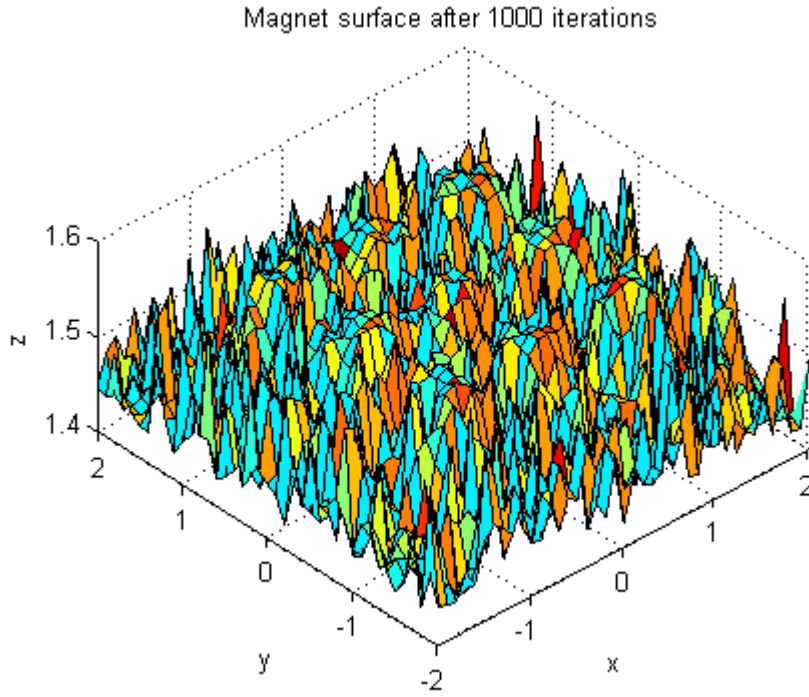


Figure 2.25: Obtained magnet surface after 1000 genetic algorithm generations

Surface shown in Figure 2.25 seems to be highly jagged. To eliminate high frequency changes on the surface, it has been passed through an ideal low pass filter that has cutoff frequency $\frac{\pi}{4}$. After this operation the surface becomes continuous as shown in Figure 2.26.

Surface shown in Figure 2.26 and the decrease in the rate of increase in elitist fitness values show that the genetic algorithm has brought the solution near to a global optimum. Therefore, the elitist chromosome of 1000th generation is transferred to the cycling coordinates method.

In the cycling coordinates method, the fitness function of the genetic algorithm is used. Initial step size, α_0 , is chosen as $\alpha_0 = 0.0015$. Since the solution space is 1641 dimensional, a full search cycle in the directions of all basis vector of the solution

space necessitates 1641 iterations. 1 cycle means 1641 iterations of the cycling coordinates method.

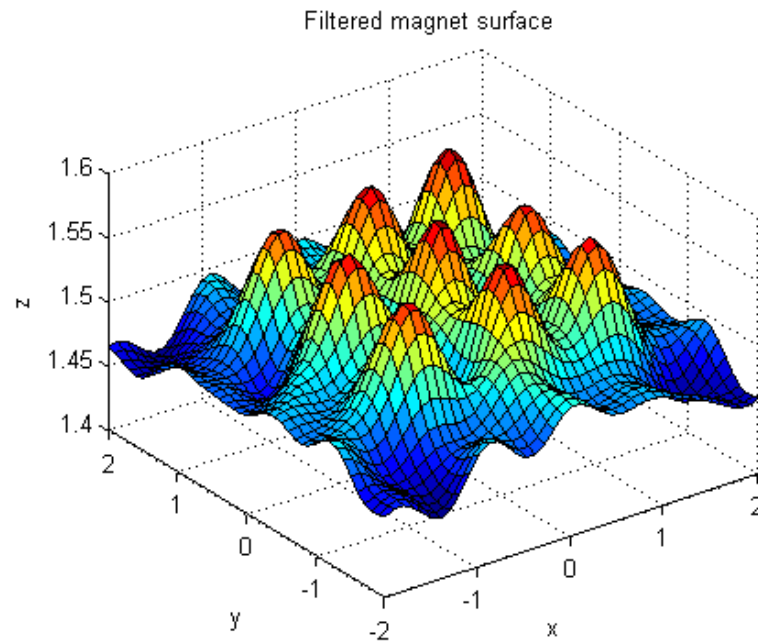


Figure 2.26: Filtered version of surface shown Figure 2.25

The fitness value variation in one cycle of iterations is shown in Figure 2.27.

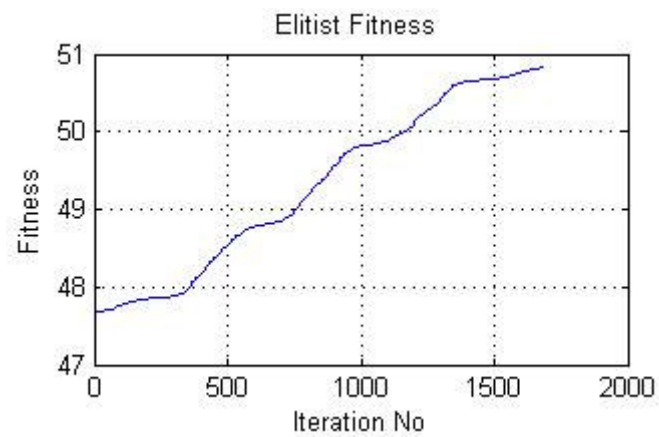


Figure 2.27: Fitness value variation in one cycle of iterations

Figure 2.27 shows that the algorithm has not converged, yet. Therefore, this algorithm should continue to run. The surface after one cycle of iterations of this algorithm is shown in Figure 2.28 in the following page.

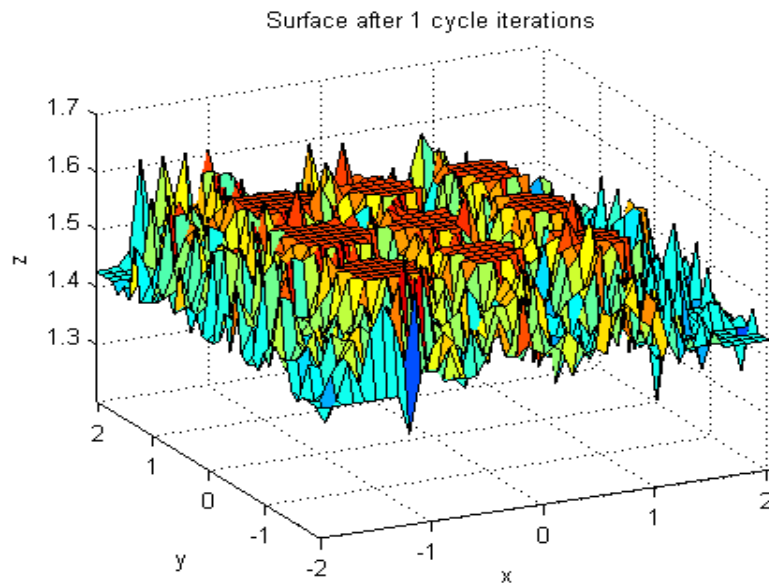


Figure 2.28: Surface after 1 cycle of iterations

After running this algorithm for 11 cycle iterations more, a sufficiently good surface is obtained. The variation of fitness values in 11 cycles of iterations is shown in Figure 2.29. The surface after 11 cycles of iterations is shown in Figure 2.30. As it is shown in Figure 2.29, the algorithm has converged.

The surface obtained after 1000 genetic algorithm generations and 12 cycles of the cycling coordinates method, shown in Figure 2.30, is manufacturable.

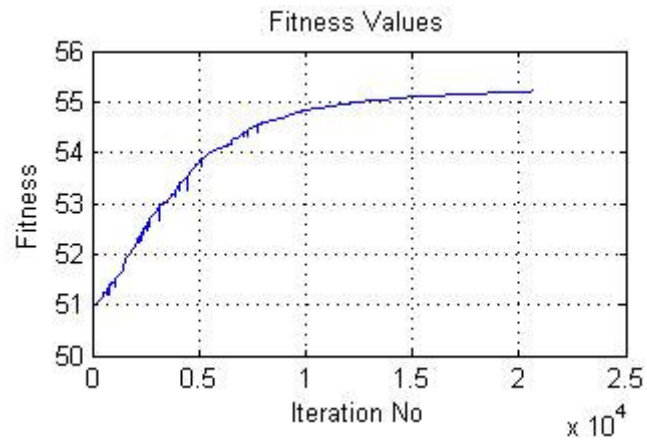


Figure 2.29: Fitness value variation in 11 cycles

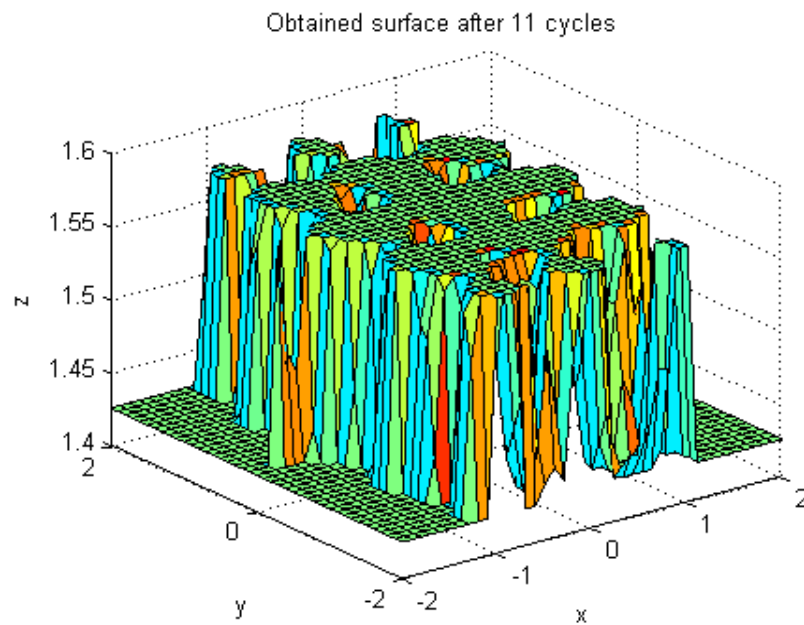


Figure 2.30: Obtained surface after 11 cycles of iteration

In the preceding figures, each run of algorithm accepts the initial solution as the last solution of the step described before it. For example, the initial solution of 11 cycle run is the surface shown in Figure 2.28.

The fitness function parts of magnet surfaces shown in Figures 2.25, 2.26, 2.28 and 2.30 are given in Table 2.1.

Table 2.1: Fitness function parts of obtained surfaces

$Fig.No$	F_{\max}	$F_{\min1}$	$F_{\min2}$	F
2.25	885.6343	697.6812	1.52E-06	46.98676
2.26	882.6575	698.4116	2.03E-05	46.04118
2.28	896.852	693.3342	4.74E-05	50.83208
2.30	906.0211	685.1394	7.58E-06	55.21284

In Table 2.1, $Fig.No$ means the figure number. F part is the fitness function value of the solution. As shown in Table 2.1, the fitness value is not affected drastically when the surface is filtered. Compared to the magnet surface after the genetic algorithm phase, the maximization part of the fitness is increased by 2.3% and minimization part is decreased by 1.8% and total fitness is increased by 17.51%.

In Figure 2.31, some surfaces are shown:

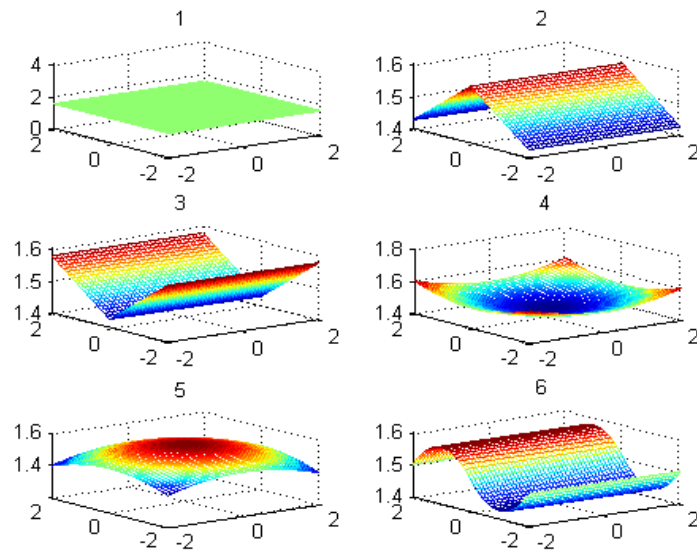


Figure 2.31: Some known surfaces

The fitness parts of these geometries are shown in Table 2.2.

Table 2.2: Fitness function parts of known surfaces

ID	F_{\max}	$F_{\min 1}$	$F_{\min 2}$	F
1	855.9116	713.8035	0	35.52703
2	873.3326	696.5576	0	44.19375
3	849.6079	738.2065	0	27.85035
4	847.2713	740.5302	0.000004	26.68126
5	870.1086	692.2395	0.000004	44.46329
6	862.3746	715.7831	0	36.64788

In Table 2.2, ID field represents numbers shown above the subfigures in Figure 2.31. Therefore, the average surface has an ID of 1. Comparing the best magnet surface with average magnet surface, the maximization part of the fitness has been increased by 5.85%, and minimization part has decreased by 4.02%. Fitness function value is increased by 55.41%. Therefore, the obtained result is satisfactory as long as increase in the fitness function is concerned.

Magnetic field distribution of magnet with surface shown in Figure 2.30 is shown in Figure 2.32.

In Figure 2.32, arrows show the magnetic field direction, and the colored parts show the magnetic field strength on the planes $x = 0$, $y = 0$ and $z = 2.35$.

To compare the change in the magnetic field distribution, side views of the magnetic field distributions of the average magnet and the resultant magnets are shown in Figure 2.33 and Figure 2.34, respectively.

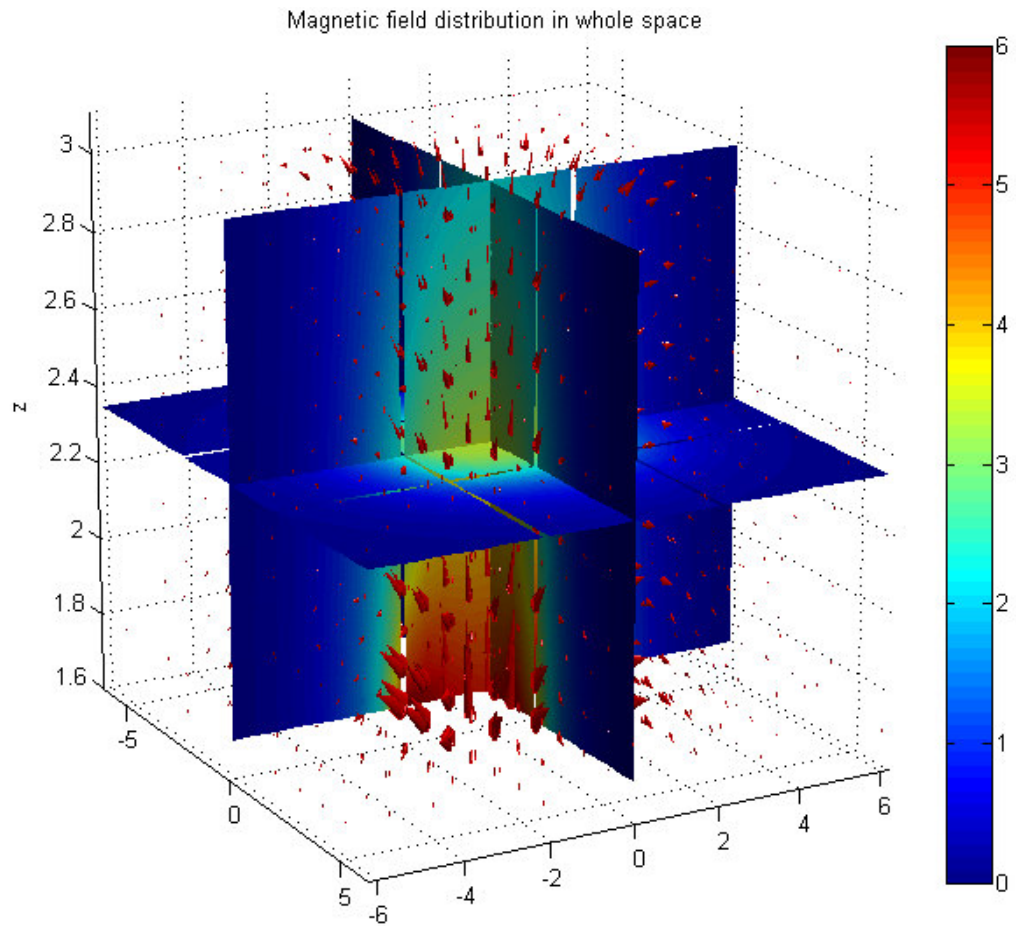


Figure 2.32: Magnetic field distribution of best solution in ROI and outer regions

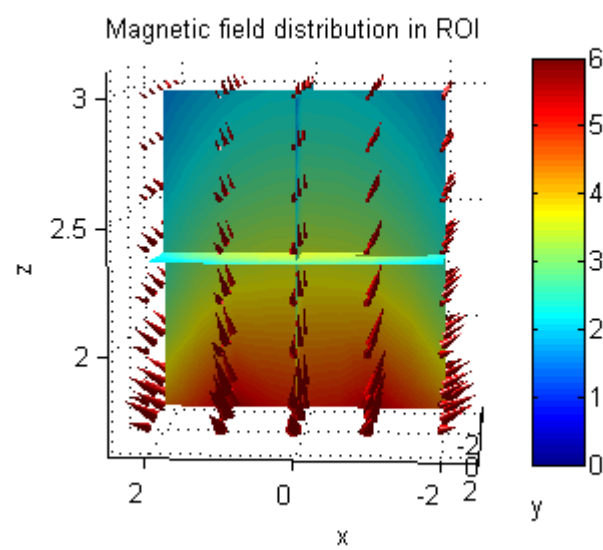


Figure 2.33: Magnetic field distribution of solution in ROI

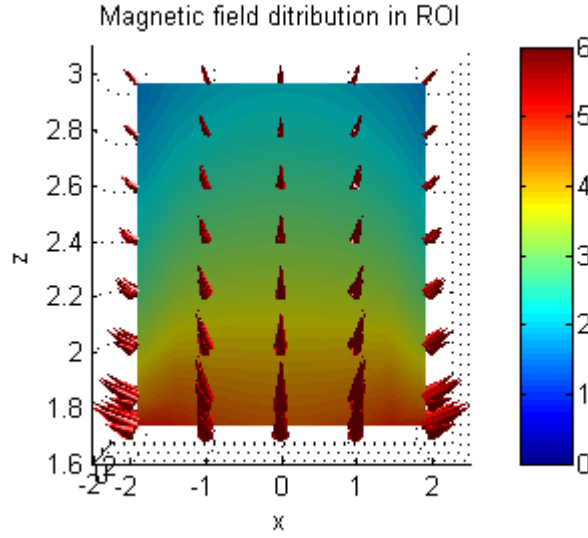


Figure 2.34: Magnetic field distribution of average magnet in ROI

When the preceding two figures are compared, the magnetic field strength shows slower variation with respect to z (magnetic field strength is higher in deeper regions) in the first figure. Therefore, the magnetic energy in ROI has increased.

As stated previously, to check the fitness function, shape parameters of the average magnet is optimized. There are two parameters that describe the shape of the magnet, a and h . When the volume of the magnet is fixed, these two parameters become dependent due to imposition of holonomic constraint on the principle coordinate system of the problem. Therefore the unknown of the problem can be defined as the ratio of h to a . Then, the problem is defined as “*maximizing magnetic energy in the ROI while minimizing magnetic energy in the outer regions by changing the ratio of h to a* ”.

This problem is solved by using the steepest descent algorithm. The volume of the magnet has been fixed to 1000 units and the initial side length has been chosen as 10 units. In other words, the optimization procedure is started with ratio 1.

For the steepest descent algorithm, the derivative step size is chosen as 10^{-4} , the accuracies of search, linear search and variable are 10^{-6} , 10^{-6} and 10^{-7} , respectively.

After 6 iterations, the program has converged to the value 0.7466, which is very near to the one obtained in [8].

2.5.2.Results of Placement of Unit Magnets Part

Before starting to present the unit magnet arrangements obtained, undefined parameters of the methods used should be defined.

As a test case, 16 unit magnets with surfaces shown in Figure 2.30 are aimed to be placed in 32 grid regions. Each unit magnet has side length, a , 4 units and height, h , 3 units. MPR is created so that there are 8 grids in one direction and 4 grids in the other, as shown in Figure 2.12.

The height of MPR segments, h , is defined to be in the range $[0,1.5]$. The rotation angle is defined in the interval $[-\frac{\pi}{4}, \frac{\pi}{4}]$.

Real coded genetic algorithms use 0.1 as the mutation probability. For real coded genetic algorithms, number of boundary mutations is 2. Mutation operation ε of height of MPR segments is taken as 0.075. Mutation operation ε of angle of rotation is taken as $\frac{\pi}{36}$. The mutation probabilities of placement and orientation parameters are taken as 0.25 and 0.05, respectively.

Weighting coefficients in equation (2.68) are chosen to be:

$$\begin{aligned}\alpha_1 &= 0.4 \\ \alpha_2 &= 0.3 \\ \alpha_3 &= 0.3\end{aligned}$$

The boundaries in (2.64) are chosen as:

$$C_{xMIN} = -0.5, C_{xMAX} = 0.5$$

$$C_{yMIN} = -0.5, C_{yMAX} = 0.5$$

$$C_{zMIN} = -2, C_{zMAX} = 0$$

Region of interest (ROI) for this part of the problem is defined by the intervals:

$$\begin{aligned} x &\in [-8\sqrt{2}, 8\sqrt{2}] \\ y &\in [-8\sqrt{2}, 8\sqrt{2}] \\ z &\in [3.5, 6.5] \end{aligned}$$

The differential changes in ROI are taken as:

$$\begin{aligned} dx &= 1 \\ dy &= 1 \\ dz &= 1 \end{aligned}$$

For these parameters, the variation of the elitist fitness in 350 generations of the genetic algorithm is shown in Figure 2.35.

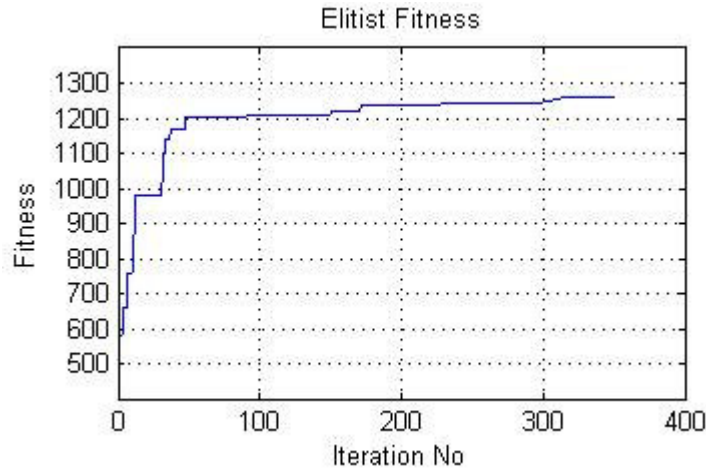


Figure 2.35: The variation of elitist fitness in 350 genetic algorithm iterations

The variation of the fitness of the elitist chromosome has a staircase shape in some regions. This is due to the fact that, more than one genetic algorithm is running. When a change in one variable occurs, the other variables are trying to reach and keep up that fitness value.

Since the fitness function is defined by the equation (2.68), the maximum height variable is expected. Since the ROI is defined in the middle of MPR, all magnets should be placed just below the ROI to maximize the magnetic energy. Since the maximum energy stored in the MPR occurs when the magnet arrangement is united in the sense of all unit magnets are touching each other, all the angles should be either $\frac{\pi}{4}$ or $-\frac{\pi}{4}$. Minimization parts of (2.68) will be effective on the orientation variable more than the other variables. Since linear variation in the magnitude of the magnetic field is desired, equal number of unit magnets with each possible orientations is expected.

The magnet arrangement after 350 genetic algorithm generations is shown in Figure 2.36 and 2.37. The chromosomes of each variable are shown in Table 2.3.

In Figures 2.36 and 2.37 blue objects represent unit magnets of which *S* pole is facing to the ROI, and green objects are represent unit magnets of which *N* pole is facing to the ROI. In Table 2.3, No represents the MPR segment number. Other four columns show the chromosomes of placement, angle, height and orientation variables. Some chromosomes do not have genes at grid nodes where magnet is not placed, due to the fact that the identification system of these chromosomes is different then the one used for placement chromosomes.

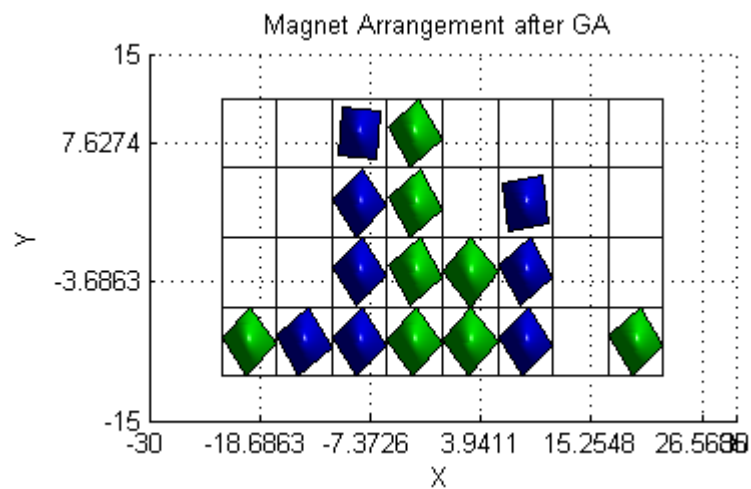


Figure 2.36: Top view of magnet placement after 350 genetic algorithm generations

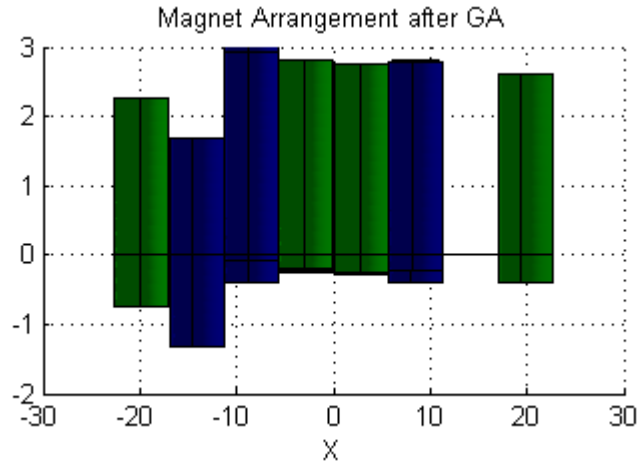


Figure 2.37: Side view of magnet placement after 350 genetic algorithm generations

Table 2.3: The chromosomes of obtained distribution after 350 genetic algorithm generations

No	Placement	Angle	Height	Orient
1	1	40.584	0.74985	1
2	1	34.809	0.18792	0
3	1	38.872	1.4229	0
4	1	40.981	1.2986	1
5	1	43.327	1.25	1
6	1	39	1.2765	0
7	0	--	--	--
8	1	36.429	1.1062	1
9	0	--	--	--
10	0	--	--	--
11	1	37.943	1.4919	0
12	1	35.23	1.2664	1
13	1	44.514	1.2247	1
14	1	35.254	1.11	0
15	0	--	--	--
16	0	--	--	--
17	0	--	--	--
18	0	--	--	--
19	1	40.464	1.1032	0
20	1	35.19	1.2565	1
21	0	--	--	--
22	1	9.1035	1.3233	0
23	0	--	--	--
24	0	--	--	--
25	0	--	--	--
26	0	--	--	--
27	1	-4.7881	1.4068	0
28	1	35.706	1.3165	1
29	0	--	--	--
30	0	--	--	--
31	0	--	--	--
32	0	--	--	--

As shown in Figure 2.35, the rate of increase in the fitness of the elitist chromosome is very small after 100th iteration. Therefore, after 350 generations of the genetic algorithms, 350th generation is transferred to the near global optimum algorithm.

In the near global optimum algorithms, since there are 16 unit magnets, the solution space of real valued variables is 16 dimensional. Therefore, one cycle of iterations means 16 iterations. In the cycling coordinates method part of this algorithm, the initial step size, α_0 , is chosen as 0.015 for the height variable and $\frac{\pi}{36}$ for the orientation angle variable. For algorithms summarized in Figure 2.20 and Figure 2.22, cycling coordinate part is performed for one cycle of iterations.

As stated previously, two types of the near global optimum algorithms are implemented for this part. The first algorithm does not converge fast enough. This fact is shown in the Figures 2.38, 2.39 and 2.40.

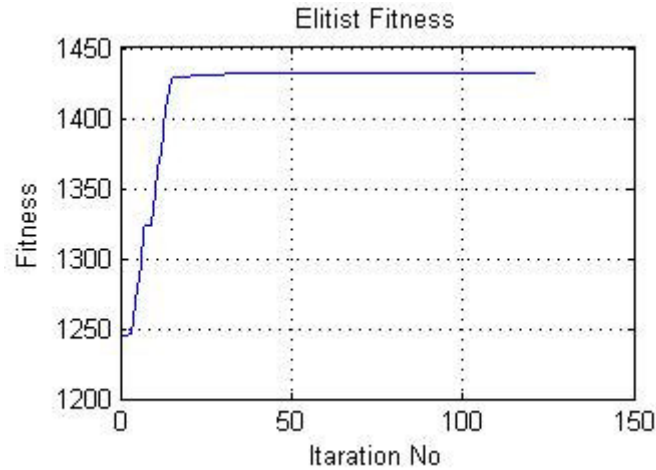


Figure 2.38: The variation of elitist fitness after 120 iterations of near global optimum algorithm 1

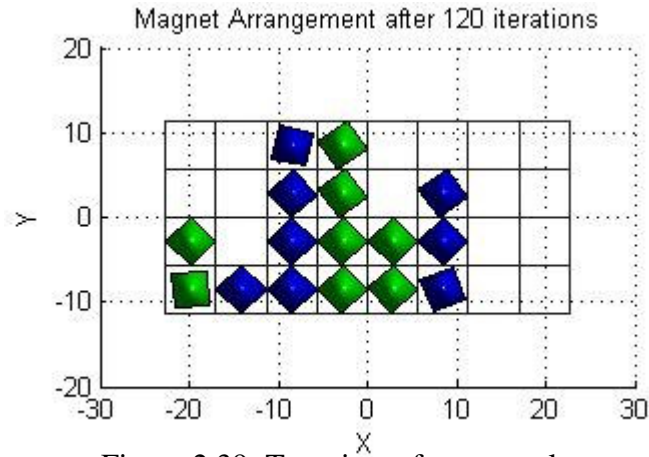


Figure 2.39: Top view of magnet placement after 120 iterations of near global optimum algorithm 1

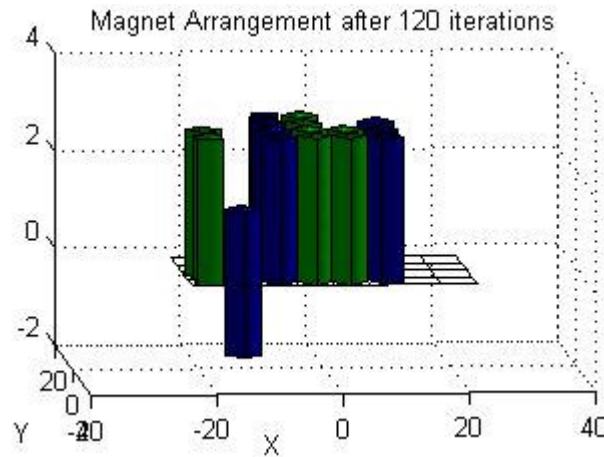


Figure 2.40: Side view of magnet placement after 120 iterations of near global optimum algorithm 1

Although the algorithm has been evaluated for 120 iterations, the placement has not converged as can be seen in Figure 2.39 and Figure 2.40. The increase in fitness function value shown in Figure 2.38 is due to a change in height and rotation angle parameters, which are optimized using cycling coordinates.

After concluding that this algorithm is not satisfactory for the problem defined, 350th generation of genetic algorithm is transferred to the 2nd near global optimum algorithm which is summarized in Figure 2.22.

In this algorithm, number of chromosomes that are imposed to placement algorithm operations, M , is chosen to be 5. Cycling coordinate parameters are the ones used in the previous near global optimum algorithm. If the fitness of the elitist chromosome increases or if 20 iterations of placement operations are performed, the program passes to the next steps.

In 48 iterations of this algorithm, the variation of fitness function value of the elitist chromosome is shown in Figure 2.41.

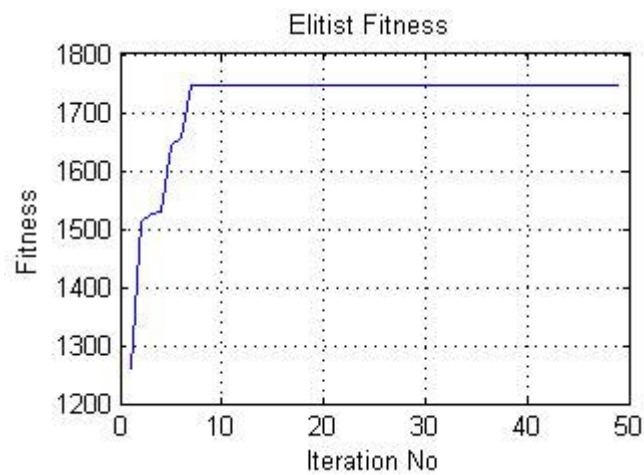


Figure 2.41: The variation of elitist fitness in 48 iterations of near global optimum algorithm 2

Obtained magnet arrangement is shown in Figure 2.42 and Figure 2.43, the values in the chromosomes are given in Table 2.4.

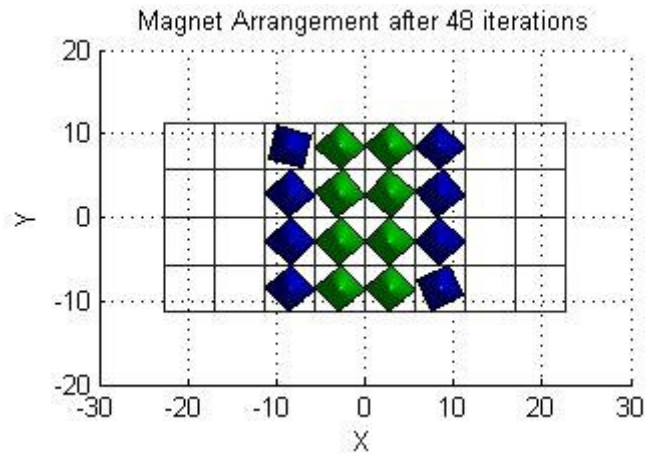


Figure 2.42: Top view of magnet placement in 48 iterations of near global optimum algorithm 2

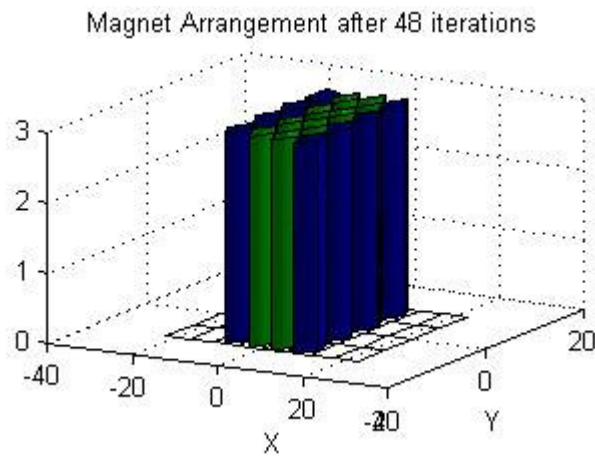


Figure 2.43: Magnet placement after 48 near global in 48 iterations of near global optimum algorithm 2

Table 2.4: The chromosomes of obtained arrangement

No	Place	Angle	Height	Orient
1	0	--	--	--
2	0	--	--	--
3	1	43.872	1.5	0
4	1	43.481	1.5	1
5	1	42	1.5	1
6	1	25.5	1.5	0
7	0	--	--	--
8	0	--	--	--
9	0	--	--	--
10	0	--	--	--
11	1	45	1.5	0
12	1	44	1.5	1
13	1	45	1.5	1
14	1	45	1.5	0
15	0	--	--	--
16	0	--	--	--
17	0	--	--	--
18	0	--	--	--
19	1	42.464	1.5	0
20	1	38.69	1.5	1
21	1	-41.5	1.5	1
22	1	39.103	1.5	0
23	0	--	--	--
24	0	--	--	--
25	0	--	--	--
26	0	--	--	--
27	1	-14.288	1.5	0
28	1	45	1.5	1
29	1	43	1.5	1
30	1	45	1.5	0
31	0	--	--	--
32	0	--	--	--

Figure 2.42 and Figure 2.43 show that the placement of more than one unit magnets has converged just after 8 iterations of 2nd near global optimum algorithm iterations.

Obtained magnet arrangement is the expected one. Although the rotation angle of all chromosomes is not the expected ones, other variables have the values as expected.

Table 2.5 is prepared to compare the fitness function of the obtained magnet arrangement with the ones predicted. The predicted arrangements are shown in Figures 2.44, 2.45 and 2.46.

Three arrangements in Figures 2.45, 2.46 and 2.47 have the same placement and height variable values as in the obtained arrangement.

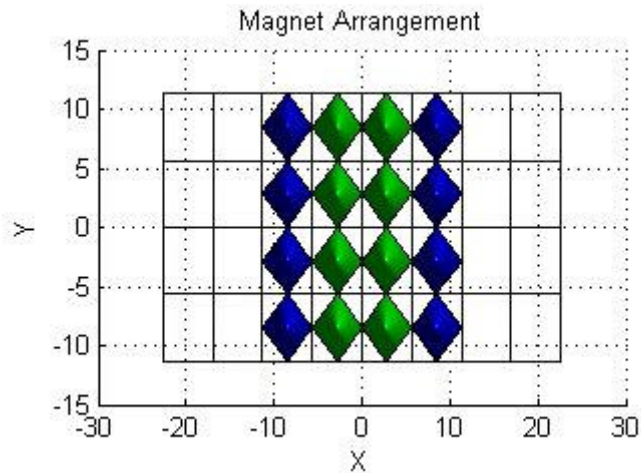


Figure 2.45: Predicted magnet arrangement 1

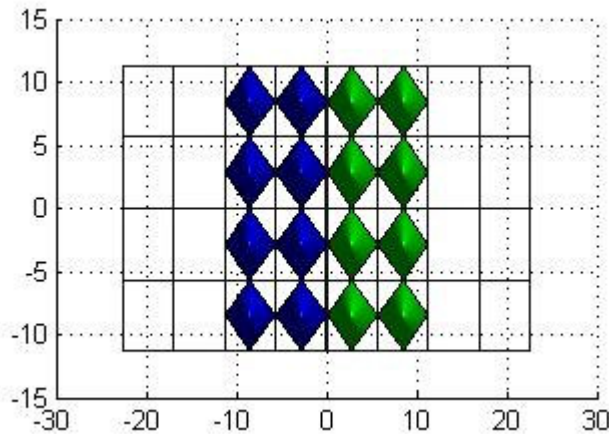


Figure 2.46: Predicted magnet arrangement 2

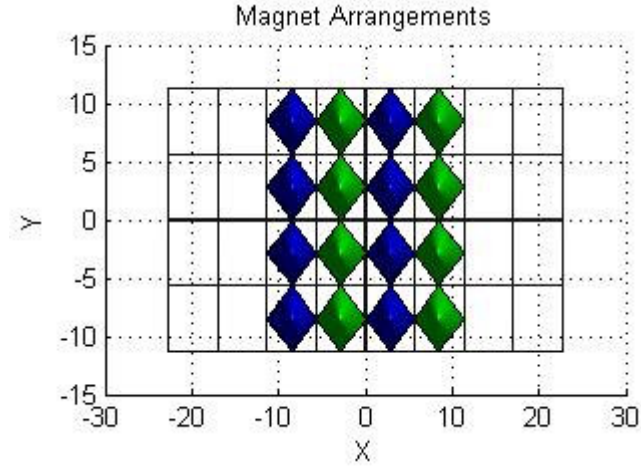


Figure 2.47: Predicted magnet arrangement 3

Figure 2.45 shows an arrangement very similar to the obtained one except that rotation angle is fixed to $\frac{\pi}{4}$ radians. Figure 2.46 shows an arrangement when unit magnets with different orientation are placed in two different regions of MPR to decrease the 1st and the 2nd derivatives of magnetic field magnitude. The arrangement shown in Figure 2.47 is different than others. In this arrangement, the orientation parameter is such that one row of the magnets with a fixed orientation are followed by the magnets with the other orientation. In Table 2.5 the values of fitness function of these magnet arrangements are shown.

Table 2.5: Fitness function parts of obtained and predicted magnet arrangements

<i>Fig.No</i>	F_{\max}	$F_{\min 1}$	$F_{\min 2}$	F
2.42	7835.60	1731.00	2890.70	1747.80
2.45	7837.50	1726.50	2903.10	1746.10
2.46	6838.90	1426.60	2793.30	1469.60
2.47	8180.70	2054.00	3115.90	1721.30

The *Fig.No* field in Table 2.5 shows the figure number.

The fitness function parts of the arrangements in Figures 2.42 and 2.45 have higher magnetic energy in ROI compared to arrangement in Figure 2.46 resulting in higher fitness values. These two arrangements have very similar results. Therefore, one of them can be chosen as the output of this study. Arrangement shown in Figure 2.46 has smaller minimization function parts. Therefore, the 1st and 2nd derivatives of the magnetic field magnitude in ROI are smaller for these arrangements. The arrangement shown in Figure 2.47 has the highest maximization part in its fitness. However, its minimization parts are higher than the others. The fitness is smaller than the fitness of the obtained arrangement. Consequently, these arrangements have been ignored.

The magnetic field distribution of magnet arrangement after 350 generations of the genetic algorithm is shown in Figures 2.48 and 2.50. In Figure 2.48 and in Figure 2.50, the magnetic field distribution is shown when looked from the Y axis and the X axis respectively. Similarly, the magnetic field distribution of magnet arrangement after 48 iterations of near global optimum algorithm is shown in Figures 2.49 and 2.51. In Figure 2.49 and in Figure 2.51, the magnetic field distribution is shown when looked from the Y axis and the X axis respectively.

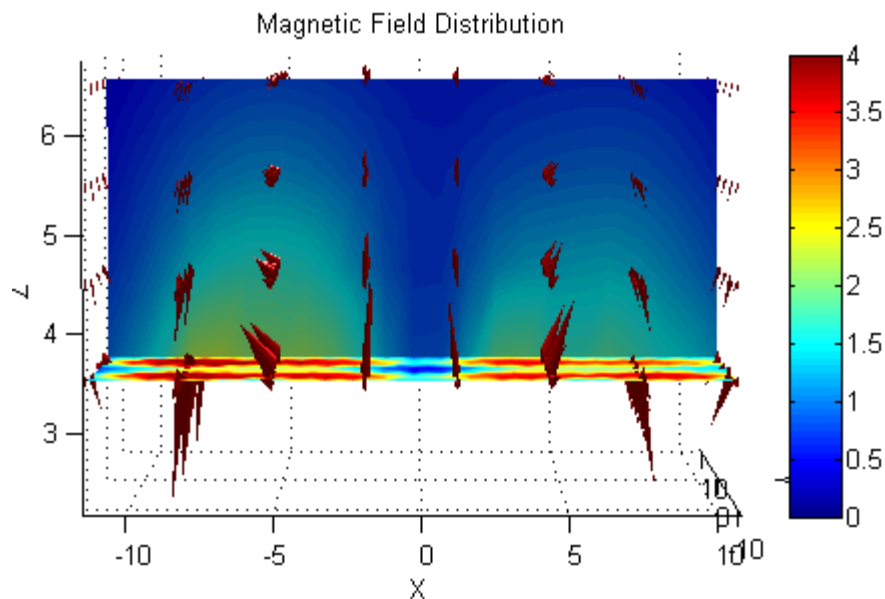


Figure 2.48: Magnetic Field distribution after 350 generations of genetic algorithm when viewed from Y axis

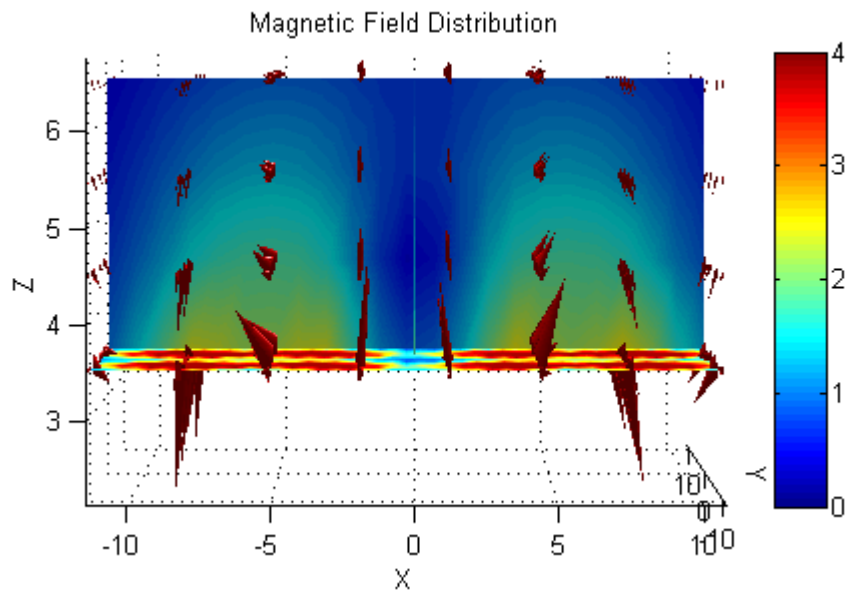


Figure 2.49: Magnetic Field distribution after 48 iterations of near global optimum algorithm when viewed from Y axis

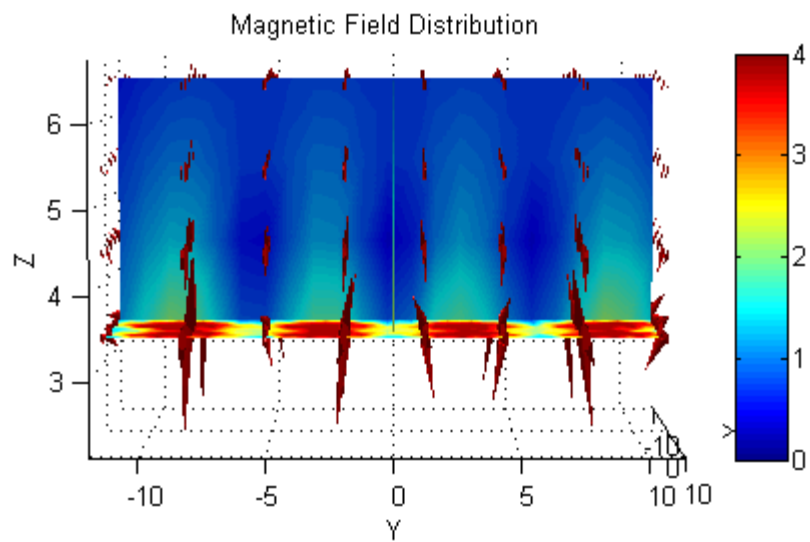


Figure 2.50: Magnetic Field distribution after 350 generations of genetic algorithm when viewed from X axis

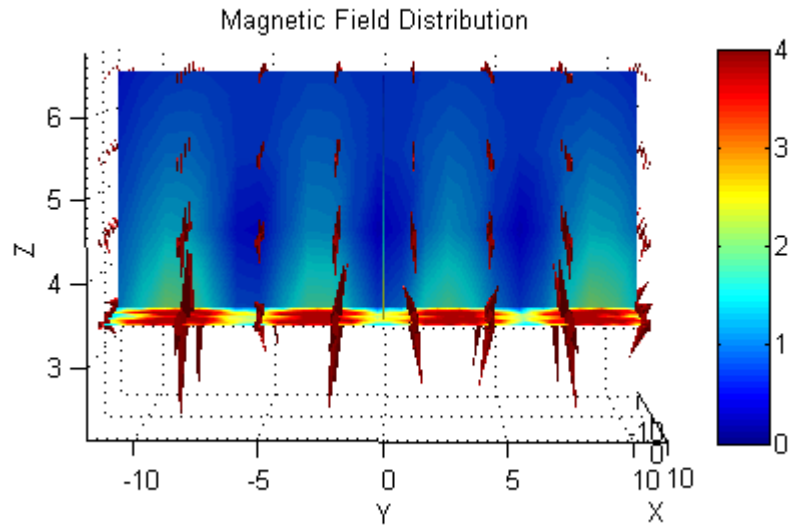


Figure 2.51: Magnetic Field distribution after 48 iterations of near global optimum algorithm when viewed from X axis

As it can be seen from the figures, magnetic field magnitude has increased in the deeper parts of the ROI while the variation of magnetic field with respect to x and y is kept nearly constant in larger areas of the ROI after 48 iterations of near global optimum algorithm.

These magnetic field distributions show that, the objective of the study has been reached.

2.6. Conclusion

In this part of the work, permanent magnet surface optimization and optimization of placement of more than one magnet to obtain a magnetic field distribution with desired properties have been performed. For this purpose, real and binary coded genetic algorithms, and an optimization routine which does not require gradient or Hessian of the fitness function have been utilized. These algorithms have been recombined to form a complete method for the design of a permanent magnet which

has a desired magnetic field distribution. The obtained results of each step have been presented.

Due to huge computing time requirements of magnetic field calculations in computer environment, scalar magnetic potential method has been used to calculate magnetic field distribution along with FEM like interpolation. For each part of the problem, variables of the fitness function that change the magnetic field distribution have been determined and optimized using the methods developed during this work. The methods used for the parts of this work are summarized in Figures 2.20 and 2.23. The unit magnets used in the placement of more than one magnet in space have the resultant surface of the surface optimization part. In this part of the thesis, the work done in [8] has also been verified. This verification has proved that the fitness functions used in this work have been chosen properly.

The surface of the unit permanent magnets has been shown in Figure 2.30. Although the resultant surface is discontinuous, it is quite suitable for manufacturing. This validates the assumption made about manufacturability of discontinuous surfaces. The fitness function of the obtained magnet surface has improved by 55.41% compared to a flat surface. In the second part of this work, 16 unit magnets with surfaces shown in Figure 2.30 have been placed in space as shown in Figures 2.42 and 2.43 for the fitness function and method parameters chosen as a test case. The magnetic field distribution of obtained magnet arrangement has been shown in Figures 2.49 and 2.51. The obtained arrangement has met the magnetic field property requirements of the test case fitness function. Since the obtained results are satisfactory, the developed methods are proven to be valid.

To sum up, the task of designing permanent magnet structures generating magnetic field with the desired properties in the region of interest has been accomplished. The resultant magnetic field distribution is used in the MRI simulator in inhomogeneous magnetic fields. In the next chapter, the development of MRI simulator in inhomogeneous magnetic fields is explained and the image reconstruction technique developed for MRI imaging in inhomogeneous magnetic fields is introduced.

2.7. References

- [1] Z. P. Liang and P. C. Lauterbur, “Principles of magnetic resonance imaging: A signal processing perspective”, SPIE Optical Engineering Press, 2000.
- [2] David Jiles, “Introduction to magnetism and magnetic materials”, 1st Edition Chapman & Hall, Iowa, USA, 1995
- [3] Amikam Aharoni, “Introduction to theory of ferromagnetism”, 1st Edition, Oxford University Press, New York, 1996
- [4] B. Bleaney and B. Bleaney, “Electricity and Magnetism”, 3rd Edition, Oxford University Press, London, 1976
- [5] R. Street, “Magnetism and Magnetic materials”, Lecture Notes, Dept. of Physics, The University of Western Australia, 2002
- [6] David Cheng, “Field and Wave Electromagnetics”, 2nd Edition, Addison Wesley Publishing Co, 1989.
- [7] Gareth P. Hatch, “Magnet design considerations for devices and particles for biological high-gradient magnetic separation (HGMS) system”, Journal of Magnetism and Magnetic Materials, Vol. 225, pp. 262-276, 2001.
- [8] H.Yigitler, A.O. Yilmaz and B. M. Eyuboglu: “An Approach to geometrical design of Permanent Magnets for Biomedical Applications”, Biomed2004, 11th International Biomedical Science and Technology Days, Ankara- Turkey, p.24, 2004.
- [9] Konrad Weeber, “Geometric Parameterization and Constrained Optimization Techniques in the Design of Salient Pole Synchronous Machines”, IEEE Transactions on Magnetics, Vol. 28 No 4, pp. 1948-1960, 1992.
- [10] Yingying Yao, Jae Seop Ryu and Chang Seop Koh, “Robust 3-D Shape Optimization of Electromagnetic Devices by Combining Sensitivity Analysis and Adaptive Geometric Parameterization”, IEEE Transactions on Magnetics, Vol. 40 No 2, pp. 1200-1203, 2004.
- [11] P. M. Glover, P. S. Aptaker, J. R. Bowler, E. Ciampi, and P. J. McDonald, “A Novel High-Gradient Permanent Magnet for the Profiling of Planar Films and Coatings”, Journal of Magnetic Resonance, Vol. 139, pp. 90-97, 1999.

- [12] Andrew E. Marble, Igor V. Mastikhin, Bruce G. Colpitts and Bruce J. Balcom, "An analytical methodology for magnetic field control in unilateral NMR", *Journal of Magnetic Resonance*, Vol. 174, pp. 78-87, 2005.
- [13] K. Halbach, "Design of permanent multipole magnets with oriented rare earth cobalt materials", *Nuclear Instruments and Methods*, Vol. 169, pp. 1–10, 1980.
- [14] H. Raich, P. Blumler, "Design and Construction of a Dipolar Halbach Array with a Homogeneous Field from Identical Bar Magnets: NMR Mandhalas", *Concepts in Magnetic Resonance Part B (Magnetic Resonance Engineering)*, Vol. 23B(1), pp. 16–25, 2004.
- [15] D. Whitley, "A Genetic Algorithm Tutorial"; Colorado State University, Computer Science Department, www.cs.uga.edu/~potter/CompIntell/ga_tutorial.pdf
- [16] Zbigniew Michalewicz, "Genetic Algorithms + Data Structures=Evaluation Programs", 3rd Edition, Springer, USA, 1996
- [17] Wikipedia, "Genetic algorithm", http://en.wikipedia.org/wiki/Genetic_algorithm, 05.08.2006

CHAPTER 3

MAGNETIC RESONANCE IMAGING SIMULATOR AND IMAGE RECONSTRUCTION ALGORITHM IN INHOMOGENEOUS MAGNETIC FIELDS

3.1. Introduction

Magnetic Resonance Imaging (MRI) is a tomographic imaging technique that produces images of microscopic magnetic characteristics of an object using measured signals. Physically, MRI is based on NMR phenomenon observed by Felix Bloch and Edward Purcell in 1946 [1]. This phenomenon describes the behavior of paramagnetic materials under the influence of strong and constant magnetic field, when the spinning frequencies of the microscopic particles are imposed to external RF energy. When the strong magnetic field is uniformly distributed over the object, the spinning frequencies of the same type of atoms are all the same. Paul Lauterbour developed an imaging method in 1973 [2], using the fact that when external magnetic field is not uniformly distributed the spinning frequencies of microscopic particles vary in accordance with non-uniformity of magnetic field. Since then, different techniques for image formation of magnetic properties of objects using spatial encoding in measured NMR signals have been developed.

The conventional imaging system is composed of three main parts, a main magnet, a gradient coil system, and a Radio Frequency (RF) system.

The main magnet's function is to generate strong static main magnetic field (\vec{B}_0). Although the strength of \vec{B}_0 directly effects the Signal to Noise Ratio (SNR) and the spatial resolution of the imaging system, it is limited with RF energy level to which the patient can be exposed. The order of \vec{B}_0 magnitude ranges between 0.3T and 3T for clinical application [3]. In these systems, the homogeneity of \vec{B}_0 over the imaging volume is very important, since the imaging modalities in use assume homogeneous magnetic field distribution. Therefore, to achieve almost uniform static magnetic field (inhomogeneity 10ppm to 50ppm [4]), these magnets are designed and constructed with special methods.

The gradient coil system, traditionally, consists of three orthogonal gradient coils. Each of these coils is designed to generate linearly varying spatial non-uniformity in the magnetic field that the object is imposed to. This linear variation is supposed to occur along a specific spatial coordinate axis selected during the system construction. Using these three coils, spatial information along each coordinate axis is encoded in measured NMR signals. This makes signal localization possible. The strength of the magnetic fields generated by these coils are smaller compared to static main magnetic field. The gradient coils' magnetic field strength is on the order of militesla per meter (mT/m) [4].

The RF system consists of a transmitter coil and a receiver coil or a single coil for transmission and reception of RF energy. The transmitter part of the system is capable of generating a time varying magnetic field in RF range used for perturbation of spins in object. Since this perturbation is possible only when the frequency of radiated RF energy and the spinning frequency of spins are in resonance, the transmitter part of the system is required to have a frequency bandwidth which contains this frequency. To obtain the desired perturbation, the magnetic field of the transmission part should be orthogonal to the main magnetic field. Therefore, since the main magnetic field is required to be homogeneous, the RF transmission system is required to generate homogeneous magnetic field. The receiver part of the RF system is used for NMR signal reception. In order not to

decrease the signal strength, it should be capable of uniform reception from the interior of the object being imaged.

When any of the magnetic fields (main magnetic field, gradient fields or RF field) has nonuniform distribution, the image quality lowers down. The correction methods for these cases have been developed [5, 6, 7, 8, 9]. In [5], a technique for the correction of magnetic field inhomogeneities is described. In this method, a model that describes the effect of inhomogeneity on the measurement is developed. This method requires two images of the same object with altered gradient. Using the model developed, the nonuniformity properties of magnetic fields are calculated and imaging accuracy is increased. In [6], an iterative method to reconstruct images is introduced. Since inhomogeneity in main and gradient magnetic field distribution changes the frequency content of the acquired signals, a linear mapping between image reconstruction and frequency domain trajectory of signals is developed. Since the frequency content of signal is not uniformly sampled for this case, the frequency domain trajectory of signals is not linear. Consequently, the inverse problem is solved iteratively. In [7], a band selective signal removal technique is presented. This technique is useful for applications requiring inhomogeneous RF transmitter, such as surface RF coil usage. In [8], a method for removing signal intensity inhomogeneity is developed. The signal intensity inhomogeneity is thought to be caused by surface RF receive coils. Obtained images are used to estimate the sensitivity map of surface RF coils. A wavelet transform is used for slowly varying sensitivity pattern estimation. In [9], a method for intensity nonuniformity due to RF field inhomogeneity in MRI data is presented. In this method, additive noise, normalize local intensity gradients and smooth local derivatives in the image are estimated to construct relative nonuniformity map. Then, true uniformity map is constructed.

The systems which do not require uniform distribution of magnetic fields have been investigated by a number of researchers. Although the analyses of the developed systems are done generally on NMR experiments, some efforts are made on image reconstruction.

The first method developed is known as stray-field imaging named as STRAFI, [10]. This method makes use of the large magnetic field gradient outside of central region of the main magnet in a conventional MRI system. Although the sensitive area of this method is restricted, it readily preserves dynamic contrast while offering very high spatial resolution.

The STRAFI method exploits the extremely large magnetic field gradient outside the central region of all high field superconducting NMR magnets. In this gradient, even a short radio frequency pulse only excites nuclei in a narrow slice of the sample orthogonal to the gradient direction, thus providing spatial localization of the NMR measurement. By stepping the sample through the gradient and repeating the measurement at each position, it is possible to build a profile of the sample in the gradient direction. By mechanical movement of the object and with the use of surface RF coils, 2D or 3D images have been constructed using this method [11].

When the system is not required to have uniform magnetic field distribution, obligations on main magnet are decreased. For example, permanent magnets can be used as magnetic field sources. Using this fact, a Mobile NMR Surface Scanner (NMR MOUSE) has been developed [12, 13]. This is a PC based NMR console with a permanent magnet probe. This design achieves portability and can investigate large objects. Also it eliminates the usage of super conducting magnet which is the most expensive part of the conventional MR system.

The basic setup of the MOUSE composed of mainly two parts: RF and electronic systems, and the probe [12]. RF pulse generation, amplification and signal detection are done by a PC controlled low-field (1-64MHz) NMR spectrometer, a duplexer and 1kW amplifiers. The PC is also used for pulse sequence generation, and data acquisition. The probe is composed of a permanent magnet and a solenoid RF coil. Due to low weight, the probe can be positioned arbitrarily. Sensitive depth range of the system is 0-2mm. The sensitive area is 3.5mm x 3.5mm for that depth. Since B1 and B0 fields generated by NMR-MOUSE are inhomogeneous, using the results of [14], Hahn echoes, CPMG type pulse sequences are used for signal generation.

Although the aim of NMR-MOUSE construction is to be used in separation and classification of materials, NMR-MOUSE had been used for explorative studies on in vitro biomedical samples [13, 15]. In [13], two dimensional image of a cross section through a pork leg is obtained. In order to obtain an image of the pork leg the NMR mouse is repositioned in a point-wise manner over the cross section of interest. Since the system is portable and has an open magnet, it is easy to make orientation dependent studies. Also the transverse relaxation time is measured by changing the orientation of the pork tendon [13]. In [15], anisotropy in tendon is investigated in vivo. In [16], a gradient coil is added to NMR-MOUSE to be able to construct 1D images. In [17], results of [16] are used as a basis and another gradient coil is added to the system. In [18], the inhomogeneity of main magnetic field is redirected so that the main magnetic field magnitude varies linearly in one direction. In [18], as in [17], two gradient coils are used for spatial encoding, and the gradient in main magnetic field is used as another gradient to be able to construct 3D images.

There are some theoretical studies on imaging in inhomogeneous main magnetic field and on inhomogeneity limits. In [19], very detailed theoretical study on magnetic resonance in inhomogeneous magnetic fields is presented. In that study, the theoretical limits of main magnetic field inhomogeneity is found using signal to noise ratio (SNR) and specific absorption rate (SAR) concepts.

The imaging experiments using the developed systems [13, 15, 16, 17, 18] and the theoretical study [19], all try to obtain a uniformly sampled frequency domain signals. Since MRI system which does not require homogeneous magnetic fields can not assume uniformly sampled frequency signals, these studies are trying to obtain limited imaging systems. On the other hand, in [20], some imaging techniques for MRI in inhomogeneous magnetic fields are developed. In that study, the imaging modality in MRI in inhomogeneous magnetic field are not restricted to the ones using frequency content transformation. For this purpose, a relation between measured signal and object properties is derived. The feasibility of the derived relation is investigated using the singular values of the matrix representation of this relation.

The works presented so far, do not aim to develop a generalized image reconstruction technique for MRI (except [20]). In these works, the main aim is to design a system which has a main magnet which generates magnetic field as homogeneous as possible to be able construct images using inverse Fourier approach. This is due to the fact that inverse Fourier techniques that are being used for conventional NMR and MRI systems yield high resolution images in short computation time. However, under inhomogeneous magnetic fields, frequency bandwidth of measured signals not only change due to the gradient field, but also change due to the magnetic field inhomogeneity. Therefore, MRI in inhomogeneous magnetic fields requires some image reconstruction modalities that do not require homogeneous magnetic fields. Moreover, the technique should be able to fully satisfy the obligatory constraints imposed by the physical basis of MRI. Moreover, the feasibility of the reconstruction technique should be examined in the sense that it should be able to make use of all acquired data to produce images.

Development of any image reconstruction technique requires generation of test data and construction of a forward problem solver. The first requirement is signal data obtained from any source. The source may be a physical system or a simulator. In this work, a MRI simulator in inhomogeneous magnetic fields to generate test data has been developed. Since the physical principle in microscopic level of magnetic resonance is the same for both inhomogeneous and homogeneous external magnetic fields, the construction steps are straightforward when the imaging domain is discretized.

Some MRI simulators have been constructed for educational and research purposes [21 to 32]. [21] and [27 to 29] are general simulators for MRI. These simulators require high computation time. Therefore, some simulators have been implemented for parallel computing [23 and 24]. Simulators for NMR have also been developed, [25 to 27]. These NMR simulators are developed for pulse sequence design and for educational purposes. In [22], object specific field mapping is used to simulate artifacts in the images. In [30], a simulator is developed to investigate static field and RF field imperfections. In [32], a simulator which can exhibit nearly all imaging artifacts is developed. This simulator combines all properties of other simulators.

However, none of above aims to construct a simulator for MRI or NMR in inhomogeneous magnetic fields. So far, [34] is the only attempt to construct an NMR simulator in inhomogeneous magnetic fields.

In this chapter, the construction details of a magnetic resonance imaging simulator in inhomogeneous magnetic fields are given and the development of image reconstruction technique for magnetic resonance imaging in inhomogeneous magnetic fields is introduced. Firstly, theory of the simulator and the image reconstruction technique are given. Secondly, implementation details of the simulator and the image reconstruction technique are given. Thirdly, obtained results are given for both homogeneous and inhomogeneous magnetic field cases. Finally, this chapter ends with conclusion.

3.2. Theory

Nuclei with odd atomic weight possess an angular momentum. This fundamental property of nuclei is called “Spin”. Since all moving charged particles create magnetic field around themselves, microscopic charged particles create magnetic field. This is the basic physical principle of nuclear magnetism. This microscopic magnetic field is called the “Magnetic Moment”.

The angular momentum is related to magnetic moment with proportionality constant γ as shown below.

$$\vec{\mu} = \gamma \vec{J} \quad (3.1)$$

This constant is known as the “Gyromagnetic Ratio”. Gyromagnetic ratio varies from microscopic particle to particle. In above equation $\vec{\mu}$ is magnetic moment, \vec{J} is angular momentum.

The magnetic property of atoms is represented by spin quantum number, S . In quantum mechanics, spin quantum number can only take discrete values. The magnitude of magnetic moment is certain under any condition and related to the spin quantum number. However, the direction of magnetic moment is uncertain due to the

uncertainty principle of quantum mechanics. On the other hand, a component of the magnetic moment can be determined by imposing the spin to an external static magnetic field. In this case, the energy of spin is fixed, and that component of the magnetic moment is aligned to the external magnetic field. However, the external magnetic field can not change the fact that the particle is spinning. Other two components of the magnetic moment are still rotating, and their exact magnitude can not be determined. Therefore, the spins under the influence of the external magnetic field creates a magnetic moment that precesses around the external magnetic field.

The frequency of precession is known as the “Larmor Frequency”. Larmor frequency of a spin is a function of the magnitude of the external magnetic field and is represented by

$$\omega_0 = \gamma \|\vec{B}_0\| \quad (3.2)$$

where \vec{B}_0 is the external magnetic field.

An ensemble of spins with the same spin quantum number is called a “Spin System”. Total magnetic moment of a spin system is known as the “Magnetization”, \vec{M} , and is defined as the vector sum of all individual magnetic moments of spins.

$$\vec{M} = \sum_{i=1}^N \vec{\mu}_i \quad (3.3)$$

where N is the number of spins in the spin system.

When all spins of a spin system is imposed to the same external static magnetic field, the magnitude and the component of the magnetic moment parallel to external magnetic field is certain. Moreover, total magnitude of the components of magnetic moment perpendicular to external magnetic field is known. However, the angle between these two components is a uniformly distributed random variable that can take any value between $[-\pi, \pi]$. When the number of spins in the spin system approaches to infinity, total number of perpendicular components of magnetic moments of individual spins start to become deterministic due to the central limit theorem. The angle between two components is defined as random variable. As the number of spins in the spin system approaches infinity, in the average observable

perpendicular to the external magnetic field components of magnetization tend to zero. Therefore, macroscopic magnetization of the spin system in thermal equilibrium is defined as shown below.

$$\vec{M}^0 = \frac{\gamma^2 \hbar^2 N_s S(S+1)}{3KT_s} \vec{B}_0 \quad (3.4)$$

where K is Boltzmann constant, T_s is absolute temperature of the spin system, N_s is the number of spins in spin system, \hbar is normalized Planck's constant ($\hbar = \frac{h}{2\pi}$), \vec{B}_0 is the external magnetic field and \vec{M}^0 is the magnetization of spin system in thermal equilibrium. Detailed derivation can be found in [4].

For $S = \frac{1}{2}$ spin system, (3.4) becomes:

$$\vec{M}^0 = \frac{\gamma^2 \hbar^2 N_s}{4KT_s} \vec{B}_0 \quad (3.5)$$

Since a hydrogen atom's nucleus consists of only one proton, its spin quantum number is $\frac{1}{2}$. Since in biological applications distribution of hydrogen atoms is imaged, from now on $S = \frac{1}{2}$ systems will be investigated.

As stated previously, angular momentum and magnetic moment are related to each other with gyromagnetic ratio. Torque created by magnetic moment and angular momentum should sum to zero. In classical mechanics, this statement is expressed as:

$$\frac{d\vec{M}}{dt} = \gamma \vec{M} \times \vec{B} \quad (3.6)$$

where \vec{B} is the magnetic field at any time instant that the spin system is imposed to. \vec{B} can be expressed as total magnetic field as shown below.

$$\vec{B} = \vec{B}_0 + \vec{B}_{OTHER} \quad (3.7)$$

where \vec{B}_{OTHER} represents other magnetic fields that effects the spin system. More detailed derivation can be found in [4].

Equation (3.6) implies that whenever \vec{M} and \vec{B} are not parallel to each other, a torque is created to make them parallel. This torque can not change \vec{M} instantaneously. This relaxation process is known as “Longitudinal Relaxation” and the time constant that represents this relaxation process is generally called T_1 . Therefore, when \vec{B}_{OTHER} becomes 0, the component of \vec{M} (longitudinal component of \vec{M} , \vec{M}_{\parallel}) parallel to \vec{B}_0 should relax to the thermal equilibrium magnetization, \vec{M}^0 .

Another relaxation process occurs for the components of \vec{M} (transverse components of \vec{M} , \vec{M}_{\perp}) perpendicular to \vec{B}_0 . This relaxation is known as “Transverse Relaxation” and represents the loss of phase coherence among spins in the spin system. Loss of phase coherence occurs due to \vec{B}_0 inhomogeneity along the volume of the spin system and due to spin-spin interactions. This relaxation process is represented by another time constant, T_2 . The transverse components of \vec{M} should decay to 0, when phase coherence among spins in the spin system is lost.

When the relaxation terms are added to (3.6), Bloch equation is obtained.

$$\frac{d\vec{M}}{dt} = \gamma \vec{M} \times \vec{B} - \frac{\vec{M}_{\perp}}{T_2} - \frac{(\vec{M}_{\parallel} - \vec{M}^0)}{T_1} \quad (3.8)$$

The state of the spin system ($\vec{M}(t)$) is described by (3.8). However, a measurement equation is needed to fully describe the spin system.

For homogeneous magnetic fields case, the relation between state of spin system and measurements is formulated in the next subsection. Also, in the next subsection, utilization of measured data to obtain images is briefly described. After that subsection, a method to extend formulation of homogeneous magnetic fields case to inhomogeneous magnetic fields case is introduced. After that, derivation of a linear

relation between initial magnetization and measurements is described. Finally, the theory of developed image reconstruction technique is presented.

3.2.1. Conventional MRI Formulation

In Bloch equation, $\gamma\vec{M} \times \vec{B}$ part expresses the torque that tries to change the magnetization state of a spin system. An external magnetic field with non-zero perpendicular component to \vec{M} should be applied to create that torque. As stated previously, the external RF field should be in resonance with the spins to be effective on them.

Applied RF field has the form shown below.

$$\vec{B}_1 = 2B_1^e(t) \cos(\omega_{rf}t + \varphi) [B_{1,x}\hat{x} + B_{1,y}\hat{y} + B_{1,z}\hat{z}] \quad (3.9)$$

where \hat{x} , \hat{y} and \hat{z} are the unit vectors of left handed orthogonal reference frame. For homogeneous \vec{B}_0 field distribution, \vec{B}_1 is homogeneous and orthogonal to \vec{B}_0 . When \vec{B}_0 is defined as in (3.10), \vec{B}_1 takes the form:

$$\vec{B}_0 = B_0\hat{z} \quad (3.10)$$

$$\vec{B}_1 = 2B_1^e(t) \cos(\omega_{rf}t + \varphi)\hat{x} \quad (3.11)$$

Equation (3.11) can be decomposed into two rotating fields; one rotating in clockwise and the other rotating in counterclockwise direction as:

$$\begin{aligned} \vec{B}_1 = B_1^e(t) \left[\begin{aligned} &\cos(\omega_{rf}t + \varphi)\hat{x} + \sin(\omega_{rf}t + \varphi)\hat{y} + \\ &\cos(\omega_{rf}t + \varphi)\hat{x} - \sin(\omega_{rf}t + \varphi)\hat{y} \end{aligned} \right] \begin{aligned} &\} \rightarrow \text{counterclockwise} \\ &\} \rightarrow \text{clockwise} \end{aligned} \end{aligned} \quad (3.12)$$

Since counterclockwise rotating part of the RF field is in the opposite direction of spin rotation direction (the spin rotation is in clockwise direction), it produces negligible effect. Therefore, the effective RF field, \vec{B}_1 , is in the form:

$$\vec{B}_1 = B_1^e(t) [\cos(\omega_{rf}t + \varphi)\hat{x} - \sin(\omega_{rf}t + \varphi)\hat{y}] \quad (3.13)$$

B_1^e is the envelope of the magnetic field. It can be of any shape, but it is usually chosen to be in sinc shape, square shape or triangular shape. When \vec{B}_0 field is in the form shown in (3.10), RF field in (3.13) produces non-zero torque due to $\gamma\vec{M} \times \vec{B}$ part of the Bloch equation. Therefore, the magnetization vectors of spins, due to imposed RF field in duration T_{RF} , are rotated by an angle of α .

On the other hand, since \hat{x} and \hat{y} components of magnetization vector of spins in thermal equilibrium are rotating with the Larmor frequency, ω_0 , and the effective rotating component of the RF field (equation (3.13)), is rotating with frequency ω_{rf} , it is customary to define a rotating frame of reference.

The unit vectors of rotating frame of reference are functions of time and expressed as:

$$\begin{aligned}\hat{x}' &= \cos(\omega t)\hat{x} - \sin(\omega t)\hat{y} \\ \hat{y}' &= \sin(\omega t)\hat{x} + \cos(\omega t)\hat{y} \\ \hat{z}' &= \hat{z}\end{aligned}\tag{3.14}$$

where ω is the frequency of rotating frame. Therefore, the unitary transformation matrix, T_{rot} , from laboratory frame to rotating frame is defined as:

$$T_{rot} = \begin{bmatrix} \cos(\omega t) & -\sin(\omega t) & 0 \\ \sin(\omega t) & \cos(\omega t) & 0 \\ 0 & 0 & 1 \end{bmatrix}\tag{3.15}$$

The time derivative of any vector in the rotating frame, $\vec{M}_{rot} = T_{rot}\vec{M}$, can be expressed as:

$$\frac{d(T_{rot}\vec{M})}{dt} = -\omega \begin{bmatrix} \sin(\omega t) & \cos(\omega t) & 0 \\ -\cos(\omega t) & \sin(\omega t) & 0 \\ 0 & 0 & 0 \end{bmatrix} \vec{M} + T_{rot} \frac{d\vec{M}}{dt}\tag{3.16}$$

in laboratory frame. The first term in (3.16) can be expressed as a skew symmetric matrix product as shown below.

$$\begin{bmatrix} \sin(\omega t) & \cos(\omega t) & 0 \\ -\cos(\omega t) & \sin(\omega t) & 0 \\ 0 & 0 & 0 \end{bmatrix} = - \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} T_{rot}\tag{3.17}$$

The skew symmetric matrix in (3.17) is nothing but cross product matrix of the unit vector \hat{z}' . Consequently, the first term in (3.16) can be expressed as:

$$-\omega \begin{bmatrix} \sin(\omega t) & \cos(\omega t) & 0 \\ -\cos(\omega t) & \sin(\omega t) & 0 \\ 0 & 0 & 0 \end{bmatrix} \vec{M} = \omega \left[(-\hat{z}') \times (T_{rot} \vec{M}) \right] \quad (3.18)$$

The derivative of \vec{M} can be expressed as:

$$T_{rot} \frac{d\vec{M}}{dt} = \frac{d(T_{rot} \vec{M})}{dt} - \omega \left[(-\hat{z}') \times (T_{rot} \vec{M}) \right] \quad (3.19)$$

The Bloch equation without relaxation terms is:

$$\frac{d(T_{rot} \vec{M})}{dt} = \gamma (T_{rot} \vec{M}) \times (T_{rot} \vec{B}) \quad (3.20)$$

Then, the Bloch equation in rotating frame can be expressed as:

$$T_{rot} \frac{d\vec{M}}{dt} = \gamma (T_{rot} \vec{M}) \times (T_{rot} \vec{B}) + \omega \left[(T_{rot} \vec{M}) \times (-\hat{z}') \right] \quad (3.21)$$

Combining the magnetization terms in (3.21) yields the Bloch equation without relaxation terms in the rotating frame.

$$T_{rot} \frac{d\vec{M}}{dt} = \gamma (T_{rot} \vec{M}) \times \left((T_{rot} \vec{B}) - \left(\frac{\omega}{\gamma} \hat{z}' \right) \right) \quad (3.22)$$

The effective magnetic field in the rotating frame is defined as:

$$\vec{B}_{eff} = \vec{B}_{rot} - \frac{\omega}{\gamma} \hat{z}' \quad (3.23)$$

The Bloch equation including relaxation terms turns out to be as:

$$\frac{\partial \vec{M}_{rot}}{\partial t} = \gamma \vec{M}_{rot} \times \vec{B}_{eff} - \frac{M_{x'} \hat{x}' + M_{y'} \hat{y}'}{T_2} - \frac{(M_{z'} - M^0) \hat{z}'}{T_1} \quad (3.24)$$

For static magnetic field as in (3.10) and for the case where the frequency of the rotating frame equals to Larmor frequency of spins:

$$(T_{rot} \vec{B}_0) = B_0 \hat{z}' \quad (3.25)$$

$$\omega = \omega_0 = \gamma B_0 \quad (3.26)$$

$$T_{rot} \frac{d\vec{M}}{dt} = \gamma (T_{rot} \vec{M}) \times \left(\left(B_0 - \frac{\omega}{\gamma} \right) \hat{z}' \right) = 0 \quad (3.27)$$

In the rotating frame, \vec{M} is stationary when, only, static magnetic field is applied.

In the rotating frame, the RF field in the form (3.13) turns out to be as shown in equation (3.28).

$$T_{rot} \vec{B}_1 = B_1^e(t) \hat{x}' \quad (3.28)$$

When an RF pulse is applied to the spin system rotating around \vec{B}_0 , in rotating frame, only the torque due to the envelope of RF field is effective. The frequency of rotation caused by this torque is related to the instantaneous magnitude of the RF pulse by:

$$\omega_1(t) = \gamma \vec{B}_1^e(t) \quad (3.29)$$

Consequently, after the RF pulse the magnetization vector is rotated around \vec{B}_1 . The angle of rotation is given by:

$$\alpha(t) = \gamma \int_0^t B_1^e(\tau) d\tau \quad (3.30)$$

After an α RF pulse, magnetization vector can be found using rotation matrices. When the RF field is in \hat{x}' direction, the rotation matrix that represents the rotation of the magnetization vector is given by:

$$R_{x'}(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & \sin \alpha \\ 0 & -\sin \alpha & \cos \alpha \end{bmatrix} \quad (3.31)$$

In order to spatially encode the spins in the spin system gradient fields are used. These fields are in the direction of \vec{B}_0 , but have magnitudes that vary almost linearly with position as shown below:

$$\vec{B}_{gradient} = (Gx) \hat{z} \quad (3.32)$$

where G is the derivative of magnitude of gradient field with respect to x . When such inhomogeneity is introduced, the Larmor frequencies of spins vary linearly in x direction. Consequently, the phase angle due to the gradient field can be calculated using:

$$\phi(x, t) = \gamma \int_0^t G(\tau) x d\tau \quad (3.33)$$

After applying the gradient field for finite duration $\tau_{gradient}$, the magnetization vector can be calculated using the rotation matrix around z axis.

$$R_{gradient}(\phi(x)) = \begin{bmatrix} \cos \phi(x) & -\sin \phi(x) & 0 \\ \sin \phi(x) & \cos \phi(x) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.34)$$

When no magnetic field other than \vec{B}_0 is applied, the Bloch equation in rotating frame has a solution given by:

$$\begin{bmatrix} M_{x'}(t) \\ M_{y'}(t) \\ M_{z'}(t) \end{bmatrix} = \begin{bmatrix} M_{x'}(t_0)e^{-(t-t_0)/T_2} \\ M_{y'}(t_0)e^{-(t-t_0)/T_2} \\ M_{z'}(t_0)e^{-(t-t_0)/T_1} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ (1 - e^{-(t-t_0)/T_1})M^0 \end{bmatrix} \quad (3.35)$$

The matrix form of the relaxation process is given by:

$$\vec{M}_{rot}(t) = \begin{bmatrix} e^{-(t-t_0)/T_2} & 0 & 0 \\ 0 & e^{-(t-t_0)/T_2} & 0 \\ 0 & 0 & e^{-(t-t_0)/T_1} \end{bmatrix} \vec{M}_{rot}(t_0) + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & (1 - e^{-(t-t_0)/T_1}) \end{bmatrix} \vec{M}^0 \quad (3.36)$$

Signal detection in MR is based on the Faraday law of electromagnetic induction and the reciprocity principle. The Faraday law of induction states that time varying magnetic flux through the receiver coil induces a voltage that is equal to the rate of change in magnetic flux through the coil. The detection sensitivity of a receiver coil is determined through the principle of reciprocity. Assume that \vec{B}_2 is the magnetic field produced by a unit direct current flowing through the coil. Then the magnetic flux through the coil due to \vec{M} is given by:

$$\Phi(t) = \int_{V_{obj}} \vec{B}_2 \cdot \vec{M} dv \quad (3.37)$$

Using the Faraday law, voltage induced is obtained as:

$$V(t) = -\frac{\partial \Phi(t)}{\partial t} = -\frac{\partial}{\partial t} \int_{V_{obj}} \vec{B}_2 \cdot \vec{M} dv \quad (3.38)$$

Since \vec{B}_2 and integral are not functions of time, then the derivative term can be taken inside the integral and allowed to operate on \vec{M} .

$$V(t) = - \int_{V_{obj}} \vec{B}_2 \cdot \frac{\partial \vec{M}}{\partial t} dv \quad (3.39)$$

To be able to obtain any information from measured voltage, magnetization should be time dependent, and its derivative should be large enough. The time dependent components of the magnetization vector are the transverse components. Since the measured voltage is around the Larmor frequency, the voltage should be demodulated. However, since the magnetization vector has two transverse components, the demodulator should be designed so that both components are demodulated. Since the phase difference between two components of magnetization is $\frac{\pi}{2}$, the designed demodulator has to have two multiplier parts with sinusoids that are $\frac{\pi}{2}$ degrees out of phase with respect to each other. Figure 3.1 shows the demodulator.

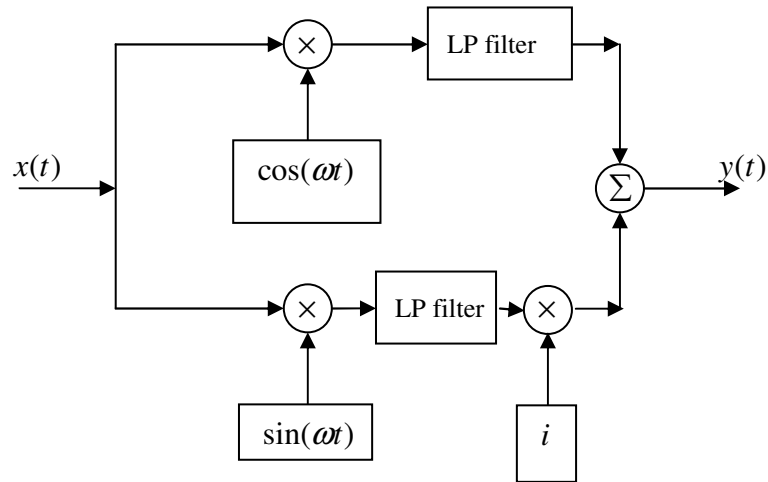


Figure 3.1: Demodulation system

In Figure 3.1, the input signal is multiplied by two sinusoids which are $\frac{\pi}{2}$ degrees out of phase with respect to each other. Then, each multiplication outputs are low pass filtered. One of the outputs of low pass filters is multiplied by i to obtain imaginary part of the signal. Then, two demodulated signals are added to obtain complete complex signal.

Since the output of the demodulator system is a band limited signal, it can be sampled according to the Nyquist criteria. The details will be given in the implementation part.

3.2.2. Extension to Inhomogeneous Magnetic Fields

Although the formulation given above is only valid for magnetic fields that are homogeneous throughout the object being imaged, this formulation can be used in inhomogeneous magnetic field case when the problem is discretized and a suitable transformation is defined.

When the volume of the object being imaged is discretized into small volume segments, the magnetic field variation in these volume segments can be ignored. In other words, magnetic fields in these volume segments (voxels) are considered as homogeneous magnetic fields. Therefore, the Bloch equation, (3.8), is valid in each voxel of the volume. However, since the directions of magnetic fields vary from voxel to voxel; a different reference frame for each pixel should be defined.

As in the homogeneous case, the local reference frame is defined referencing to \vec{B}_0 . In homogeneous case, the laboratory frame is defined according to direction of \vec{B}_0 and it is taken as z direction of laboratory frame. In inhomogeneous case, laboratory frame is not defined, but a global reference frame is used to represent \vec{B}_0 . This global reference frame can be considered as the reference frame used by magnetic field measurement device. The local reference frames are defined using a spherical coordinate system.

Any point in a Cartesian coordinate system (x, y, z) is represented by (θ, ϕ, r) in spherical coordinates. (θ, ϕ, r) are defined according to following expressions:

$$\begin{aligned}\theta &= \tan^{-1} \frac{\sqrt{x^2 + y^2}}{z} \\ \phi &= \tan^{-1} \frac{y}{x} \\ r &= \sqrt{x^2 + y^2 + z^2}\end{aligned}\tag{3.40}$$

While defining local reference frame, the same methodology is followed. \vec{B}_0 in each voxel is used for calculation of θ and ϕ angles, and its direction is taken to be in \hat{r} direction. The angles are calculated using:

$$\begin{aligned}\theta &= \tan^{-1} \frac{\sqrt{B_x^2 + B_y^2}}{B_z} \\ \phi &= \tan^{-1} \frac{B_y}{B_x}\end{aligned}\tag{3.41}$$

Then, a transformation matrix from global reference frame to local reference frame is defined as:

$$T_{spheric} = \begin{bmatrix} \cos \theta \cos \phi & \cos \theta \sin \phi & -\sin \theta \\ -\sin \phi & \cos \phi & 0 \\ \sin \theta \cos \phi & \sin \theta \sin \phi & \cos \theta \end{bmatrix}\tag{3.42}$$

Since $T_{spheric}$ is a unitary transformation its inverse is equal to its transpose.

\hat{r} component of local coordinate frame is not defined explicitly, but it is the unit vector in the direction of \vec{B}_0 .

$$\hat{r} = \frac{\vec{B}_0}{\|\vec{B}_0\|}\tag{3.43}$$

Within each voxel, \vec{B}_0 is in \hat{r} direction. Other vectors in the local reference are expressed with respect to the direction of \vec{B}_0 . In Figure 3.2, global and local reference frames are shown.

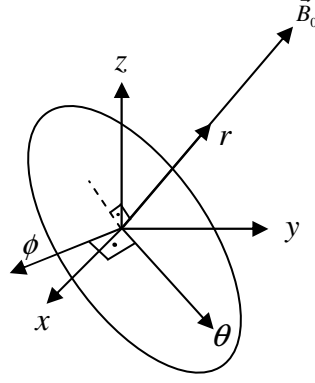


Figure 3.2: Global reference frame, (x, y, z) , and local reference frame, (θ, ϕ, r) .

Using the local reference frame, the Bloch equation (equation (3.8)) can be used in each voxel by the formulation derived for the homogeneous magnetic fields case.

3.2.3. MR Formulation in Inhomogeneous Magnetic Fields

The Bloch equation given in the form (3.8) can be expressed in terms of $\vec{M}, \vec{M}^0, \vec{B}_0, \vec{B}, T_1$ and T_2 . This representation is based upon the fact that transverse and longitudinal components of \vec{M} are defined referencing to \vec{B}_0 . The longitudinal component of \vec{M} is found by projection of \vec{M} onto \vec{B}_0 .

$$\vec{M}_{\parallel} = \left(\frac{\vec{M} \cdot \vec{B}_0}{\|\vec{B}_0\|} \right) \frac{\vec{B}_0}{\|\vec{B}_0\|} \quad (3.44)$$

The remaining component of \vec{M} is the transverse component, \vec{M}_{\perp} , defined below.

$$\vec{M}_{\perp} = \vec{M} - \left(\frac{\vec{M} \cdot \vec{B}_0}{\|\vec{B}_0\|} \right) \frac{\vec{B}_0}{\|\vec{B}_0\|} \quad (3.45)$$

Since \vec{M} and \vec{B}_0 are real valued vectors, the dot product is defined in \mathbb{R}^3 .

Consequently, (3.44) and (3.45) can be expressed as a matrix vector product.

$$\vec{M}_{\parallel} = \frac{\vec{B}_0 \vec{B}_0^T}{\|\vec{B}_0\|^2} \vec{M} \quad (3.46)$$

$$\vec{M}_\perp = \left(I - \frac{\vec{B}_0 \vec{B}_0^T}{\|\vec{B}_0\|^2} \right) \vec{M} \quad (3.47)$$

\vec{B} in (3.8) can be decomposed as in (3.7). Using the additivity property of cross product, cross product term in (3.8) can be expressed as:

$$\gamma \vec{M} \times \vec{B} = \gamma \vec{M} \times (\vec{B}_0 + \vec{B}_{OTHER}) = \gamma \vec{M} \times \vec{B}_0 + \gamma \vec{M} \times \vec{B}_{OTHER} \quad (3.48)$$

Using the following property of cross product:

$$\gamma \vec{M} \times \vec{B}_0 = -\gamma \vec{B}_0 \times \vec{M} \quad (3.49)$$

The cross product can be represented by a skew symmetric matrix operator.

$$[-\gamma \vec{B}_0 \times] \vec{M} = -\gamma \begin{bmatrix} 0 & -B_{0,z} & B_{0,y} \\ B_{0,z} & 0 & -B_{0,x} \\ -B_{0,y} & B_{0,x} & 0 \end{bmatrix} \vec{M} \quad (3.50)$$

As stated previously, \vec{B}_{OTHER} term represents the magnetic fields, other than \vec{B}_0 , the spin system is exposed to. The cross product matrix, $[-\gamma \vec{B}_{OTHER} \times]$, can be considered as the input matrix to the system represented by (3.8). Consequently, it is suitable to leave this term out of terms related to \vec{B}_0 , which represents the system.

\vec{B}_0 related terms in (3.8) can be collected to define system matrix A_0 .

$$A_0 = -\gamma \begin{bmatrix} 0 & -B_{0,z} & B_{0,y} \\ B_{0,z} & 0 & -B_{0,x} \\ -B_{0,y} & B_{0,x} & 0 \end{bmatrix} - \frac{1}{T_2} \left(I - \frac{\vec{B}_0 \vec{B}_0^T}{\|\vec{B}_0\|^2} \right) - \frac{1}{T_1} \frac{\vec{B}_0 \vec{B}_0^T}{\|\vec{B}_0\|^2} \quad (3.51)$$

(3.51) can be shown in more compact form as in (3.52).

$$A_0 = [-\gamma \vec{B}_0 \times] + \left(\frac{1}{T_2} - \frac{1}{T_1} \right) \frac{\vec{B}_0 \vec{B}_0^T}{\|\vec{B}_0\|^2} - \frac{I}{T_2} \quad (3.52)$$

The input matrix, A_u , is defined as follows:

$$A_u = [-\gamma \vec{B}_{OTHER} \times] \quad (3.53)$$

Using (3.52) and (3.53), the Bloch equation can be rewritten as:

$$\frac{d\vec{M}}{dt} = (A_0 + A_u)\vec{M} + \frac{\vec{M}^0}{T_1} \quad (3.54)$$

The summation of system matrix and input matrix can be combined to obtain the matrix, A . The constant term in (3.54) is called b .

$$A = A_0 + A_u \quad (3.55)$$

$$b = \frac{\vec{M}^0}{T_1} \quad (3.56)$$

Since \vec{M} and magnetic fields are functions of position, \bar{x} ($\bar{x} \triangleq (x, y, z)$), and time, t , (3.54) can be rewritten as:

$$\frac{d\vec{M}(\bar{x}, t)}{dt} = A(\bar{x}, t)\vec{M}(\bar{x}, t) + b(\bar{x}) \quad (3.57)$$

From now on, when \bar{x} is not specified explicitly, the derivations are given for one voxel (position is fixed to \bar{x}_0).

When A is time invariant, (3.57) has an analytical solution:

$$\vec{M}(t) = e^{A(t-t_0)}\vec{M}(t_0) + (e^{A(t-t_0)} - I)A^{-1}b \quad (3.58)$$

When A is time varying, one of the numerical methods can be utilized to solve the inhomogeneous differential equation shown in (3.57).

When $t \gg t_0$ in equation (3.58), for $A_u = 0$ case, exponential terms in (3.58) will approach to 0 due to the fact that all eigenvalues of A_0 have negative real parts. Consequently, (3.58) can be approximated to:

$$\vec{M}(t) \simeq -A_0^{-1}b, \quad t \gg t_0 \quad (3.59)$$

On the other hand, when $t \approx t_0$ in equation (3.58), exponential terms approach to I .

In this case, (3.58) can be approximated to:

$$\vec{M}(t) \simeq e^{A_0(t-t_0)}\vec{M}(t_0), \quad t \approx t_0 \quad (3.60)$$

Therefore, for long durations the inverse of A_0 is needed, whereas for short durations the inverse of A_0 is not needed. For the case $t \gg t_0$, it is required to be shown that A_0 has an inverse. To be able to prove this, eigenvalues and eigenvectors of A_0 are needed.

Proposition:

Eigenvalues of A_0 are:

$$\begin{aligned}\lambda_1 &= -\frac{1}{T_1} \\ \lambda_2 &= i\gamma\|\vec{B}_0\| - \frac{1}{T_2} \\ \lambda_3 &= -i\gamma\|\vec{B}_0\| - \frac{1}{T_2}\end{aligned}\tag{3.61}$$

Proof:

Eigenvalues of $[-\gamma\vec{B}_0 \times]$ are:

$$\begin{aligned}\lambda_{1,1} &= 0 \\ \lambda_{1,2} &= i\gamma\|\vec{B}_0\| \\ \lambda_{1,3} &= -i\gamma\|\vec{B}_0\|\end{aligned}\tag{3.62}$$

The eigenvectors associated with each eigenvalue of $[-\gamma\vec{B}_0 \times]$ are:

$$e_1 = \begin{bmatrix} B_{0,x} \\ B_{0,y} \\ B_{0,z} \end{bmatrix}; e_2 = \begin{bmatrix} \frac{B_{0,x}B_{0,y} + iB_{0,z}\|\vec{B}_0\|}{B_{0,y}B_{0,z} - iB_{0,x}\|\vec{B}_0\|} \\ \frac{B_{0,x}^2 + B_{0,z}^2}{B_{0,y}B_{0,z} - iB_{0,x}\|\vec{B}_0\|} \\ 1 \end{bmatrix}; e_3 = \begin{bmatrix} \frac{B_{0,x}B_{0,y} - iB_{0,z}\|\vec{B}_0\|}{B_{0,y}B_{0,z} + iB_{0,x}\|\vec{B}_0\|} \\ \frac{B_{0,x}^2 + B_{0,z}^2}{B_{0,y}B_{0,z} + iB_{0,x}\|\vec{B}_0\|} \\ 1 \end{bmatrix}\tag{3.63}$$

Since $[-\gamma\vec{B}_0 \times]$ is a cross product matrix (i.e., it is a skew symmetric matrix), it is obvious that \vec{B}_0 is an eigenvector associated with 0 eigenvalue of this matrix. Since the other two eigenvalues are nonzero and cross product produces nonzero results when it operates on vectors that have components orthogonal to \vec{B}_0 , the other two eigenvectors should be orthogonal to \vec{B}_0 .

$\left(\frac{1}{T_2} - \frac{1}{T_1}\right) \frac{\vec{B}_0 \vec{B}_0^T}{\|\vec{B}_0\|^2}$ is a rank 1 matrix. Consequently, it has only one nonzero

eigenvalue. Eigenvalues of this matrix are:

$$\begin{aligned}\lambda_{2,1} &= \frac{1}{T_2} - \frac{1}{T_1} \\ \lambda_{2,2} &= 0 \\ \lambda_{2,3} &= 0\end{aligned}\tag{3.64}$$

The eigenvector associated with the nonzero eigenvalue this matrix is \vec{B}_0 . This part of A_0 is, infect, a projection operator onto the subspace spanned by \vec{B}_0 . Therefore, \vec{B}_0 forms a basis for the range space of this matrix. Since this part is a rank 1 matrix, other eigenvectors form a basis for the null space of this matrix. In other words, these two eigenvectors are orthogonal to \vec{B}_0 . Therefore, e_2 and e_3 in (3.62) are suitable choices. Consequently, the eigenvectors in (3.62) are also eigenvectors of this part.

$-\frac{I}{T_2}$ part of A_0 has three nonzero eigenvalues all equal to $-\frac{1}{T_2}$.

$$\begin{aligned}\lambda_{3,1} &= -\frac{1}{T_2} \\ \lambda_{3,2} &= -\frac{1}{T_2} \\ \lambda_{3,3} &= -\frac{1}{T_2}\end{aligned}\tag{3.65}$$

Since this part is the identity matrix, any vector in \mathbb{C}^3 is an eigenvector of it.

Since all three parts of A_0 have the same set of eigenvectors, the eigenvalues of A_0 are sum of the eigenvalues of each part. This proves the proposition.

Since λ_2 and λ_3 have magnitudes much larger than λ_1 (due to the fact that γ is on the order of 10^6), A_0 is ill conditioned. Therefore, the inverse A_0 is nearly a rank 1 matrix. In this case, the pseudo inverse can be computed using singular value decomposition concept.

The singular value decomposition is based on the fact that any matrix of which number of columns is greater than or equal to number of rows can be written as product of three matrices. If the dimensions of a matrix A , are defined as $M \times N$, A can be written as a product of $M \times N$ column orthogonal matrix U , an $N \times N$ diagonal matrix S , transpose of a $N \times N$ orthogonal matrix V [33]. Therefore, A can be written as:

$$A = USV^T \quad (3.66)$$

For square matrices the inverse of A is given by:

$$A^{-1} = VS^{-1}U^T \quad (3.67)$$

Since S is a diagonal matrix its inverse is given by:

$$S^{-1} = \begin{bmatrix} \frac{1}{s_{1,1}} & 0 & \dots & 0 & 0 \\ 0 & \frac{1}{s_{2,2}} & 0 & \vdots & 0 \\ \vdots & 0 & \ddots & 0 & \vdots \\ 0 & \vdots & 0 & \frac{1}{s_{N-1,N-1}} & 0 \\ 0 & 0 & \dots & 0 & \frac{1}{s_{N,N}} \end{bmatrix} \quad (3.68)$$

For a square matrix A , the columns of U forms an orthonormal set of basis vectors for the range space of A . The columns of V forms an orthonormal set of basis for the null space of A . S is a diagonal matrix whose elements are the eigenvalues of A .

Generally, U is formed from the eigenvectors of AA^T and V is formed from the eigenvectors of $A^T A$. All parts of the A_0 matrix, defined in (3.52), have the same set of eigenvectors defined in (3.62). The transpose of each part also has the same set of eigenvectors. Therefore, U and V are formed using the vectors in (3.62).

In equation (3.59), the inverse of A_0 matrix should be calculated. An approximate inverse of A_0 can be found using singular value decomposition by setting singular values with very small multiplicative inverse to zero. When this operation is

performed on the eigenvalues given by (3.61), the multiplicative inverses of λ_2 and λ_3 should be set to zero. Therefore, after inversion in (3.67) is performed, the nonzero column of inverse will be e_1 , in (3.62). Consequently, the range space of A_0^{-1} is the span of \vec{B}_0 . When the range space of A_0^{-1} is composed with null space of it, a basis set for \mathbb{R}^3 will be obtained. Therefore, A_0^{-1} is obtained only when two vectors which are orthogonal to \vec{B}_0 and to each other, are replaced with zero columns of A_0^{-1} .

The eigenvalues of A_0 implies that one component of \vec{M} relaxes through \vec{B}_0 with time constant T_1 , and other two components are rotating in a plane perpendicular to \vec{B}_0 and losing their magnitude with time constant T_2 . This mathematical conclusion is in accordance with the physical fact, described previously. This shows us that the model exhibits the physical basics of the system.

When an input magnetic field is applied to the system ($A_u \neq 0$), the eigenvalues of the matrix A in (3.55) will be different than those in (3.65). In MRI pulse sequences, applied inputs have a finite duration that is neither small, nor large. Since none of the assumptions about durations hold for $A_u \neq 0$ case, it is more suitable to perform matrix inversion using (3.67) directly.

Any image reconstruction technique needs a relation between the property being imaged and signals measured. Therefore, an operator that transfers the magnetization from the initial state to the final state is needed. These transformations are called state transition functions, and for the finite dimensional case they are called state transition matrices, $\Phi(t, t_0)$. A state transition matrix satisfies:

$$\begin{aligned}\vec{M}(t) &= \Phi(t, t_0)\vec{M}(t_0) \\ \Phi^{-1}(t, t_0) &= \Phi(t_0, t) \\ \Phi(t_0, t_0) &= I \\ \Phi(t_2, t_0) &= \Phi(t_2, t_1)\Phi(t_1, t_0)\end{aligned}\tag{3.69}$$

Using state transition matrix concept and substituting (3.56), (3.58) can be rewritten as:

$$\vec{M}(t) = \left(e^{A(t-t_0)} \Phi(t_0, 0) + \frac{(e^{A(t-t_0)} - I) A^{-1}}{T_1} \right) \vec{M}^0 \quad (3.70)$$

Consequently, the state transition matrix from $t = 0$ to t is defined as:

$$\Phi(t, 0) = e^{A(t-t_0)} \Phi(t_0, 0) + \frac{(e^{A(t-t_0)} - I) A^{-1}}{T_1} \quad (3.71)$$

Therefore, for known input magnetic fields, \vec{B}_0 , T_1 and T_2 , $\Phi(t, 0)$ can be found.

(3.71) is only valid when A is time invariant. When A is time varying, state transition matrix cannot be found directly. In the time varying A case, \vec{M}^0 should be known to find the state transition matrix using numerical methods. For the reconstruction problem \vec{M}^0 is not known. To be able to find a relation between $\vec{M}(t)$ and \vec{M}^0 , each orthogonal basis vector of \mathbb{R}^3 is assigned as \vec{M}^0 and (3.57) is solved for each \vec{M}^0 . The calculated $\vec{M}(t)$ s are the columns of $\Phi(t, t_0)$. Therefore, to find $\Phi(t, t_0)$, differential equation (3.57) is solved three times for each unit basis vector of \mathbb{R}^3 .

Since the state transition matrix is known at time the measurement is made, (3.57) can be rewritten as:

$$\frac{d\vec{M}(t)}{dt} = \left(A(t) \Phi(t, 0) + \frac{1}{T_1} I \right) \vec{M}^0 \quad (3.72)$$

Measured voltage also satisfies (3.39). Substituting (3.72) to (3.39) yields:

$$V(t) = - \int_{V_{obj}} \vec{B}_2(\vec{x}) \cdot \left[\left(A(\vec{x}, t) \Phi(\vec{x}; t, 0) + \frac{1}{T_1(\vec{x})} I \right) \vec{M}^0(\vec{x}) \right] d\vec{x} \quad (3.73)$$

After measuring the voltage, it is demodulated to obtain the low frequency content. The demodulator is the same as the one shown in Figure 3.1. However, the cutoff frequency of the low pass filters should be chosen according to the bandwidth of data; because the magnitude variation of \vec{B}_0 field may be large causing large variations in the Larmor frequencies of spins.

3.2.4. Image Reconstruction Technique Formulation

If $\vec{M}^0(\bar{x})$ is known for all $\bar{x} \in \Omega$ (Ω is the volume of the object being imaged), the problem is to find $V(t)$. On the other hand, if $V(t)$ is known, the problem is to find $\vec{M}^0(\bar{x})$, which is a function of N_s . The first problem is called as forward problem, and the latter one is the inverse problem.

Forward problem can be solved by (3.73). However, a relation between $\vec{M}^0(\bar{x})$ and $V(t)$ is needed to solve the inverse problem. The relation given by (3.73) can not be used directly to find $\vec{M}^0(\bar{x})$. The easiest way is to discretize the relation (3.73) with respect to time and position, so that a linear transformation from $\vec{M}^0(\bar{x})$ to $V(t)$ is obtained.

As in the previous discretization, the volume being imaged, Ω , is divided into voxels, in each of which magnetic fields are assumed to be uniform. Let P denote number of voxels in Ω . Then, (3.73) can be rewritten as:

$$V(t) = - \sum_{p=1}^P \vec{B}_2(\bar{x}_p) \cdot \left[\left(A(\bar{x}_p, t) \Phi(\bar{x}_p; t, 0) + \frac{1}{T_1(\bar{x}_p)} I \right) \vec{M}^0(\bar{x}_p) \right] \Delta v \quad (3.74)$$

where Δv is the volume of one voxel and \bar{x}_p is the position of the centre of p -th voxel. Since \vec{B}_2 and the bracketed term in (3.74) are both real valued, dot product is the standard inner product in \mathbb{R}^3 . Consequently, (3.74) can be rewritten as:

$$V(t) = \sum_{p=1}^P \left[-\vec{B}_2^T(\bar{x}_p) \left(A(\bar{x}_p, t) \Phi(\bar{x}_p; t, 0) + \frac{1}{T_1(\bar{x}_p)} I \right) \Delta v \right] \vec{M}^0(\bar{x}_p) \quad (3.75)$$

The bracketed term in (3.75) can be defined as:

$$\vec{F}(\bar{x}_p, t) = \left[-\vec{B}_2^T(\bar{x}_p) \left(A(\bar{x}_p, t) \Phi(\bar{x}_p; t, 0) + \frac{1}{T_1(\bar{x}_p)} I \right) \Delta v \right]^T \quad (3.76)$$

Since $\vec{F}(\bar{x}_p, t)$ is a vector for each \bar{x}_p , a vector in \mathbb{R}^{3P} can be defined by concatenating each $\vec{F}(\bar{x}_i, t)$, $i = 1, 2, \dots, P$ one under another as follows:

$$\tilde{F}(t) = \begin{bmatrix} F_x(\bar{x}_1, t) \\ F_y(\bar{x}_1, t) \\ F_z(\bar{x}_1, t) \\ F_x(\bar{x}_2, t) \\ \vdots \\ F_y(\bar{x}_P, t) \\ F_z(\bar{x}_P, t) \end{bmatrix} \quad (3.77)$$

Similarly, another vector for the components of $\vec{M}^0(\bar{x}_p)$ can be defined as:

$$\tilde{M}^0 = \begin{bmatrix} \tilde{M}_x^0(\bar{x}_1) \\ \tilde{M}_y^0(\bar{x}_1) \\ \tilde{M}_z^0(\bar{x}_1) \\ \tilde{M}_x^0(\bar{x}_2) \\ \vdots \\ \tilde{M}_y^0(\bar{x}_P) \\ \tilde{M}_z^0(\bar{x}_P) \end{bmatrix} \quad (3.78)$$

Using (3.77) and (3.78), (3.75) can be written as a dot product of these two vectors.

$$V(t) = \tilde{F}^T(t) \tilde{M}^0 \quad (3.79)$$

When time is discretized using time step (sampling duration) that does not destroy the frequency content of data, (3.79) can be written as:

$$V[n] = \tilde{F}^T[n] \tilde{M}^0 = V(t_n), \quad n\Delta t < t \leq (n+1)\Delta t \quad (3.80)$$

where n is defined using $n\Delta t < t \leq (n+1)\Delta t$ (Δt is sampling duration).

In MRI, voltage is measured for finite duration of time. The measurement duration is called “Acquisition Time”, T_{acq} . If N samples are taken in T_{acq} , (3.80) can be extended to contain all the acquired data in T_{acq} . Defining a vector \tilde{V} and a matrix S , a linear relation between \tilde{M}^0 and measured voltage is obtained.

$$\tilde{V} = \begin{bmatrix} V[1] \\ V[2] \\ \vdots \\ V[N-1] \\ V[N] \end{bmatrix} \quad (3.81)$$

$$S = \begin{bmatrix} \frac{\tilde{F}^T [1]}{\tilde{F}^T [2]} \\ \vdots \\ \frac{\tilde{F}^T [N-1]}{\tilde{F}^T [N]} \end{bmatrix} \quad (3.82)$$

$$\tilde{V} = S\tilde{M}^0 \quad (3.83)$$

The inverse of S should be found to determine \tilde{M}^0 . However, since both right and left hand sides of (3.83) contain modulated data and since demodulator is a linear system, both sides can be demodulated. Left hand side can be demodulated directly using the demodulator in Figure 3.1. However, right hand side can not be demodulated directly. However, since right hand side is a matrix vector product, right hand side of (3.83) can be rewritten as:

$$S\tilde{M}^0 = \sum_{i=1}^{3P} \tilde{M}_i^0 \bar{s}_i \quad (3.84)$$

where \tilde{M}_i^0 is the i^{th} component of \tilde{M}^0 and \bar{s}_i is the i^{th} column of S . Therefore, the right hand side is demodulated by demodulating each column of S .

Let $\underline{\tilde{V}}$ denote the demodulated signal and \underline{S} denote the S matrix with demodulated columns.

The direction of $\vec{M}^0(\bar{x}_p)$ is known from (3.4). The number of unknowns can be reduced using the direction of $\vec{M}^0(\bar{x}_p)$. Another matrix can be constructed to represent the direction information to eliminate the direction of $\vec{M}^0(\bar{x}_p)$ among the unknowns.

$\vec{M}^0(\bar{x}_p)$ is in the direction of $\vec{B}_0(\bar{x}_p)$. Consequently, a $3P \times P$ direction matrix can be constructed as follows:

$$D = \begin{bmatrix} \frac{B_{0,x}(\bar{x}_1)}{\|\vec{B}(\bar{x}_1)\|} & 0 & \dots & 0 \\ \frac{B_{0,y}(\bar{x}_1)}{\|\vec{B}(\bar{x}_1)\|} & 0 & & \\ \frac{B_{0,z}(\bar{x}_1)}{\|\vec{B}(\bar{x}_1)\|} & 0 & & \\ 0 & \frac{B_{0,x}(\bar{x}_2)}{\|\vec{B}(\bar{x}_2)\|} & & \vdots \\ & \frac{B_{0,y}(\bar{x}_2)}{\|\vec{B}(\bar{x}_2)\|} & & \\ & \frac{B_{0,z}(\bar{x}_2)}{\|\vec{B}(\bar{x}_2)\|} & & \\ \vdots & 0 & \ddots & 0 \\ & \vdots & & \frac{B_{0,x}(\bar{x}_P)}{\|\vec{B}(\bar{x}_P)\|} \\ & & & \frac{B_{0,y}(\bar{x}_P)}{\|\vec{B}(\bar{x}_P)\|} \\ 0 & \dots & 0 & \frac{B_{0,z}(\bar{x}_P)}{\|\vec{B}(\bar{x}_P)\|} \end{bmatrix} \quad (3.85)$$

The unknown vector is defined as:

$$X = \begin{bmatrix} \|\vec{M}^0(\bar{x}_1)\| \\ \|\vec{M}^0(\bar{x}_2)\| \\ \vdots \\ \|\vec{M}^0(\bar{x}_P)\| \end{bmatrix} \quad (3.86)$$

Using these definitions (3.83) can be rewritten as:

$$(\underline{SD}) X = \underline{\tilde{V}} \quad (3.87)$$

Since S does not have full rank for a single measurement, solving (3.87) iteratively is more suitable than trying to compute the pseudo inverse of (\underline{SD}) . In the iterative solution, initial solution of each measurement data will be taken as the solution of previous measurement's set of data.

Any reconstruction technique that utilizes finite set of data to reconstruct an image of an infinite dimensional physical parameter, aims to find a solution in the least squares sense. Similarly, iterative reconstruction techniques also try to find a solution in the least squares sense.

The problem of solving a linear set of equations, as in (3.87), can be generalized as a minimization problem:

$$\text{Minimize } \|(\underline{SD})X - \tilde{\underline{V}}\|^2.$$

Since \underline{S} and $\tilde{\underline{V}}$ are complex valued the norm used in the objective function above is induced from the following complex inner product:

$$f(X) = ((\underline{SD})X - \tilde{\underline{V}})^H ((\underline{SD})X - \tilde{\underline{V}}) \quad (3.88)$$

where H denotes complex conjugate transpose (tranjugate). When multiplication is performed, (3.88) can be rewritten as:

$$f(X) = X^T D^T \underline{S}^H \underline{SD} X - X^T D^T \underline{S}^H \tilde{\underline{V}} - \tilde{\underline{V}}^H \underline{SD} X + \tilde{\underline{V}}^H \tilde{\underline{V}} \quad (3.89)$$

where for real valued vector and matrix (D and X), tranjugate is used as transposition.

The gradient of $f(X)$ with respect to X is given by:

$$\nabla(f(X)) = 2(D^T \underline{S}^H \underline{SD})X - 2(\underline{SD})^H \tilde{\underline{V}} \quad (3.90)$$

As in any unconstrained optimization problem the gradient $\nabla(f(X))$ should be equal to 0 at a local optimum point. Then, the problem is reduced to the solution of a linear set of equations. However, for the problem at hand, it is not possible to obtain a minimum norm solution using one set of data $((D^T \underline{S}^H \underline{SD})$ and $\tilde{\underline{V}}^H \underline{SD})$. Therefore, an iterative method, like steepest descent or conjugate gradient, will be more suitable to extend the solution obtained using one set of data to the next set of data.

3.3. Implementation

In this section, implementation details are given.

3.3.1. Implemented Pulse Sequence

In any MRI system, input magnetic fields should be applied in an order to obtain desired responses from the spin system. The order of application and the duration of inputs is called a “Pulse Sequence”.

There is a variety of pulse sequences used in MRI. However, the basic pulse sequence is known as the “Spin Echo” pulse sequence. In this work, the spin echo pulse sequence is implemented as a test case to verify that the developed theory is valid. Figure 3.3 shows the order of inputs of the spin echo pulse sequence.

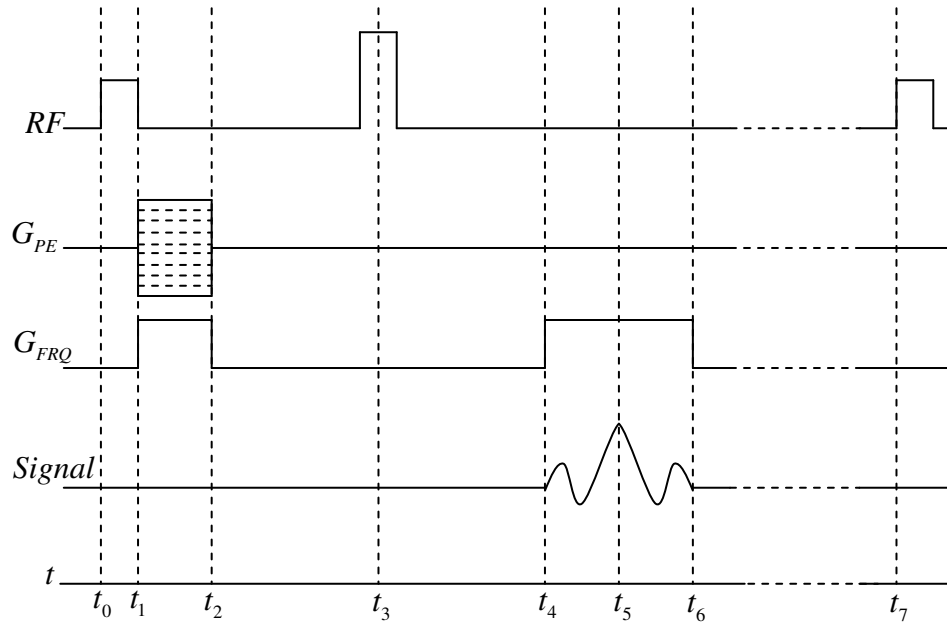


Figure 3.3: Implemented pulse sequence (order of inputs and durations)

In Figure 3.3, RF represents RF field, G_{PE} represents the magnitude of phase encoding gradient field derivative with respect to phase encoding direction, G_{FRQ} represents the magnitude of frequency encoding gradient field derivative with respect to the frequency encoding direction, $Signal$ represents the measured signal and t represents time.

The signal is obtained after an application of appropriate input magnetic fields. As shown in Figure 3.3, the signal is acquired only between specific time instants. As stated previously, the duration between these time instants is known as the signal acquisition time, T_{acq} .

In this pulse sequence two RF pulses are applied. First RF pulse is required to rotate \vec{M} through the transverse plane; whereas the second one is used for the reversal of the direction of \vec{M} . The first RF pulse is named as $\frac{\pi}{2}$, and second pulse is named as π pulse. In order to obtain a π pulse from the RF field applied to obtain a $\frac{\pi}{2}$ pulse, from equation (3.30), the magnitude of the RF field (described by the envelope function) can be doubled. In this case, the durations of RF pulses become equal to each other. The duration of $\frac{\pi}{2}$ pulse can be calculated using (3.30). However, in this study the envelope of the RF field is considered to be in rectangular shape. Consequently, (3.30) can be rewritten as:

$$\alpha(t) = \gamma \|\vec{B}_1\| t \quad (3.91)$$

and the duration of $\frac{\pi}{2}$ pulse can be calculated using:

$$T_{RF} = \frac{\pi}{2\gamma \|\vec{B}_1\|} \quad (3.92)$$

The RF pulse is designed so that it has a frequency bandwidth, $2\Delta\omega$. A typical RF envelope function in frequency domain is shown in Figure 3.4.

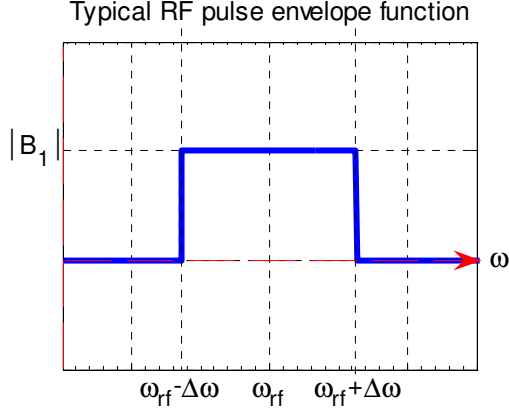


Figure 3.4: Typical RF pulse envelope function in the frequency domain

Since the resonance condition imposes the fact that all spins in the object should have a precession frequency that is covered by RF pulse's bandwidth, RF pulse is used to narrow the imaging domain in the object. The imaging region imposed by RF field's bandwidth is given by:

$$\Omega = \left\{ \bar{x} \mid \omega_0(\bar{x}) \in [\omega_{rf} - \Delta\omega, \omega_{rf} + \Delta\omega] \right\} \quad (3.93)$$

where $\omega_0(\bar{x})$ is the Larmor frequency in the point \bar{x} . This region is also called the “Field of View” (FOV). By changing the center frequency of the RF pulse, FOV is changed. Each FOV imposed by the RF pulse's bandwidth is named as “Slice”. If the number of slices is greater than one (3.93) can be rewritten as:

$$\Omega_i = \left\{ \bar{x} \mid \omega_0(\bar{x}) \in [\omega_{rf,i} - \Delta\omega, \omega_{rf,i} + \Delta\omega] \right\} \quad (3.94)$$

for the i^{th} slice. Hence, the whole imaging domain is the union of Ω_i s.

$$\Omega = \bigcup_{i=1}^{N_{slice}} \Omega_i \quad (3.95)$$

where N_{slice} is the number of slices. Consequently, although in conventional MRI this pulse sequence only produces 2D images, in inhomogeneous case it yields 3D images.

The phase encoding gradient field's magnitude is varied from measurement to measurement to obtain linearly independent measurements. The variation of the magnitude is chosen properly to prevent aliasing in the images. Although the

duration of the phase encoding gradient can be chosen arbitrarily, it is customary to chose it $\frac{T_{acq}}{2}$.

The frequency encoding is applied two times in this pulse sequence. These two frequency encoding gradient fields are applied to encode spatial information in the frequency encoding direction. The second one is applied for T_{acq} duration. The magnitude of the frequency encoding gradient can be chosen arbitrarily; but the magnitude of it directly effects the duration of signal detection, T_{acq} . Consequently, it should neither be too small nor too large.

In Figure 3.3, after acquiring the signal another RF pulse is shown to signify that the pulse sequence is repeated.

The time at which the maximum amplitude of signal occurs is known as “Echo Time”, T_E . The duration that the spin system is let to relax through the initial state is known as “Relaxation Time”, T_R .

Certain instants of the time axis are labeled in Figure 3.3. t_0 is the instant at which the sequence starts with the application of $\frac{\pi}{2}$ RF pulse. t_1 represents the end of $\frac{\pi}{2}$ RF pulse and the start of application of phase encoding and frequency encoding gradients. The duration between t_0 and t_1 is:

$$T_{RF} = t_1 - t_0 \quad (3.96)$$

At t_2 gradients fields are closed. Consequently, the duration between t_1 and t_2 is:

$$\frac{T_{acq}}{2} = t_2 - t_1 \quad (3.97)$$

The middle of π pulse is at t_3 . The π pulse is applied so that the middle of it is at $\frac{T_E}{2}$. Consequently, t_3 is equal to $\frac{T_E}{2}$. At t_4 the signal acquisition and frequency

encoding start at the same time. t_6 is the end of signal acquisition and frequency encoding. Consequently, the duration between t_4 and t_6 is :

$$T_{acq} = t_6 - t_4 \quad (3.98)$$

At t_5 the maximum amplitude of the signal is obtained. t_5 is equal to T_E . At t_7 the current step of the pulse sequence is finished and the next step starts. The duration between t_0 and t_7 is given by:

$$T_R = t_7 - t_0 \quad (3.99)$$

To be able to use this pulse sequence, T_E , T_R , G_{FRQ} , ΔG_{PE} , $\|\vec{B}_1\|$ and T_{acq} should be determined.

T_E and T_R can be chosen so that the image is weighted with T_1 or T_2 . However, since the aim of the reconstruction algorithm is to find \vec{M}^0 , it is more suitable to choose T_E around T_2 , and T_R larger than T_1 so that in the image neither T_1 nor T_2 is weighted.

G_{FRQ} should be determined before the pulse sequence is applied. In the homogeneous magnetic fields case, since this encoding changes the frequency content of the signal, the sampling duration of the signal should be chosen according to it. Sampling duration is determined by the data acquisition hardware of the system. For known sampling duration, G_{FRQ} can be calculated by:

$$G_{FRQ} \leq \frac{2\pi}{\gamma W_{frq} \Delta t} \quad (3.100)$$

where W_{frq} is the width of object in frequency encoding direction and Δt is the sampling duration. This condition also satisfies the Nyquist criteria, when it is given in the form:

$$\Delta t \leq \frac{2\pi}{\gamma W_{frq} G_{FRQ}} \quad (3.101)$$

for known G_{FRQ} .

Since the size of the image is determined prior to collecting data, for a fixed sampling duration, T_{acq} can be determined using:

$$T_{acq} = N_{frq} \Delta t \quad (3.102)$$

where N_{frq} is the number of samples (the size of image in frequency encoding direction).

However, in inhomogeneous case the frequency content of the signal is not only determined by the frequency encoding gradient, but also from the width of the inhomogeneity of \vec{B}_0 . Therefore, in this case, (3.98) should be written as:

$$\Delta t \leq \frac{2\pi}{2\Delta\omega_s} \quad (3.103)$$

where $\Delta\omega_s$ is the half of frequency bandwidth of the signal. The frequency bandwidth of the signal is:

$$2\Delta\omega_s = \gamma \left(\max \left(\left\| \vec{B}_0(\bar{x}) + \vec{B}_{FRQ}(\bar{x}) \right\| \right) - \min \left(\left\| \vec{B}_0(\bar{x}) + \vec{B}_{FRQ}(\bar{x}) \right\| \right) \right) \quad (3.104)$$

for the points previously determined by RF pulse bandwidth in (3.93), $\bar{x} \in \Omega$, where \vec{B}_{FRQ} is the magnetic field of frequency encoding gradient.

As stated previously, phase encoding gradient is used to obtain linearly independent solutions. Each step of the phase encoding gradient's magnitude should be chosen carefully to obtain linearly independent measurements. In conventional MRI, the linearly independent measurements are guaranteed by the Nyquist criteria. By the Nyquist criteria, the phase introduced to spins in the spin system is at most 2π in each measurement. Consequently, this condition can be written as:

$$\Delta G_{PE} \leq \frac{2\pi}{\gamma W_{pe} T_{PE}} \quad (3.105)$$

where W_{pe} is the width of object in the phase encoding direction and T_{PE} is the phase encoding duration. In MRI in inhomogeneous magnetic fields, the same condition is used.

3.3.2. Generation Of Magnetic Fields

There are five magnetic fields required by the simulator and the reconstruction algorithm, \vec{B}_0 , \vec{B}_1 , \vec{B}_{FRQ} , \vec{B}_{PE} and \vec{B}_2 .

The main magnetic field, \vec{B}_0 , is the magnetic field generated by the permanent magnet arrangement obtained in Chapter 2. The main magnetic field is used as a reference in the generation of other magnetic fields.

The RF field, \vec{B}_1 , is generated perpendicular to \vec{B}_0 at all points in the object being imaged. To obtain a perpendicular field, \vec{B}_0 is represented in spherical coordinates by calculating θ and ϕ angles using (3.41). These angles are used in the calculation of transformation given in (3.42). In this representation, \vec{B}_0 has only \hat{r} component. Since \vec{B}_1 should be in the form given in (3.13) and should be perpendicular to \vec{B}_0 , a vector in spherical coordinates is created as :

$$\vec{B}_{1,spheric}(t) = \begin{bmatrix} B_{1,\theta} \\ B_{1,\phi} \\ B_{1,r} \end{bmatrix} = \|\vec{B}_1\| \begin{bmatrix} \cos(\omega_{rf}t) \\ -\sin(\omega_{rf}t) \\ 0 \end{bmatrix} \quad (3.106)$$

Then, \vec{B}_1 in the global frame is found using:

$$\vec{B}_1 = T_{spheric}^T \vec{B}_{1,spheric} \quad (3.107)$$

The magnitude of \vec{B}_1 is determined prior to these operations.

Similarly, \vec{B}_2 is created using the same procedure.

$$\vec{B}_2 = T_{spheric}^T \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \\ 0 \end{bmatrix} \quad (3.108)$$

Gradient fields are assumed to have homogeneous direction distribution throughout the object being imaged. Therefore, they are created in the global frame as:

$$\vec{B}_{FRQ} = \begin{bmatrix} 0 \\ 0 \\ G_{FRQ}x \end{bmatrix} \quad (3.109)$$

$$\vec{B}_{PE} = \begin{bmatrix} 0 \\ 0 \\ G_{PE}y \end{bmatrix} \quad (3.110)$$

where the frequency encoding gradient's magnitude varies linearly in x direction and the phase encoding gradient's magnitude varies linearly in y direction. These gradient fields can be used in regions where the magnitude of the magnetic field is much larger than magnitudes of gradient fields. In the regions where this assumption is valid, gradient fields' effective component is in the direction of \vec{B}_0 . Therefore, gradient fields are modified using this fact.

$$\vec{B}_{FRQ} = \frac{\vec{B}_0 \vec{B}_0^T}{\|\vec{B}_0\|^2} \vec{B}_{FRQ} \quad (3.111)$$

$$\vec{B}_{PE} = \frac{\vec{B}_0 \vec{B}_0^T}{\|\vec{B}_0\|^2} \vec{B}_{PE} \quad (3.112)$$

3.3.3. MRI Simulator In Inhomogeneous Magnetic Fields

The constructed MRI simulator in inhomogeneous magnetic fields makes use of the magnetic fields generated and the pulse sequence described previously.

The objects being imaged are created in the simulator. The objects being created are composed of $N_x \times N_y \times N_z$ voxels. N_s , T_1 and T_2 distributions are created randomly so that each slice of the object forms previously defined shapes.

The magnetic fields are created in each voxel of the object. The main magnetic field, \vec{B}_0 , is created using the obtained magnet arrangement in Chapter 2. Each unit magnet in the arrangement has a side length of $4cm$ and a height of $3cm$. Then, the

magnetic field is calculated using (2.42). After creating main magnetic field, other magnetic fields are created as described in the previous section.

Since measured signals from any MRI device are continuous, an appropriate discrete representation of these signals should be used. Measured signal is a band limited RF signal modulated with a sinusoid with center frequency of bandwidth. To be able to represent this signal appropriately, at least two samples of the highest frequency component of the signal is needed. This condition can be expressed as:

$$\Delta t_a \leq \frac{1}{2\gamma \max(\|\vec{B}_0\|)} \quad (3.113)$$

where Δt_a is the sampling duration. Moreover, any continuous signal in the simulator is represented in discrete time by the chosen Δt_a .

In Figure 3.5, the simulator is given in pseudo code form.

```

Get user Inputs
Initialize
While k ≤ Number of Slices
    Find kth Slice
    Slice Initialize
    While j ≤ Number of phase encodings
        Apply  $\frac{\pi}{2}$  RF pulse
        Apply Phase encoding gradient
        Wait for Tw1 duration (Free relaxation)
        Apply  $\pi$  RF pulse
        Wait for Tw2 duration (Free relaxation)
        Acquire and Save the Signal
        Wait for Tw3 duration (Free relaxation)
        j ++
    end
    k ++
end

```

Figure 3.5: Simulator in pseudo code

In *Get user Input* part, the simulator expects the user to define $\|\vec{B}_1\|, \|\vec{B}_2\|, G_{FRQ}, T_E, T_R$, the number of phase encoding steps, the number of frequency encodings, the size of the object in terms of pixels and the magnetic field distribution in FOV.

In *Initialize* part, main magnetic field is loaded; the object being imaged is created; other magnetic fields are created referencing to main magnetic field and are represented in the local reference frame as described. After that, \vec{M}^0 is calculated using equation (3.4). Using \vec{B}_0 distribution and the number of slices entered by the user, RF field's bandwidth and center frequencies are calculated. After that, (3.113) is used to calculate the sampling duration for discrete time signal representation. After converting all of the vectors in local reference frame, they are considered as represented in the rotating frame of reference as in the conventional MRI.

In *Find k^{th} Slice* part, equation (3.94) is used to find the current slice voxels.

In *Slice Initialize* part, using (3.92) T_{RF} , using (3.102) T_{acq} and using (3.105) ΔG_{PE} are calculated. Afterwards, the steps of the phase encoding field's magnitude derivative with respect to the phase encoding direction for each phase encoding are generated.

$\frac{\pi}{2}$ RF pulse is applied using the matrix representation of this operation given in (3.31). Although equation (3.31) is only valid for a RF field in $\hat{\theta}$ direction of local reference frame, it is easy to extend it for RF fields in other directions. In this work, RF field is assumed to be in $\hat{\theta}$ direction, although the simulator can handle RF fields in other directions. The matrix representing RF field application is created for all voxels in the current slice, and magnetization vector of each voxel is multiplied with a corresponding matrix.

Phase encoding is applied for both phase encoding gradient and frequency encoding gradient for the same duration at the same time. As a matrix representation of the phase encoding operation, (3.34), is used. Since T_{PE} is defined to be $\frac{T_{acq}}{2}$, for constant gradient field the phase angle introduced by gradient fields are:

$$\phi(\bar{x}) = \gamma \left\| \vec{B}_{FRQ}(\bar{x}) + \vec{B}_{PE}(\bar{x}) \right\| \frac{T_{acq}}{2} \quad (3.114)$$

The phase angles calculated using (3.114) are used to create matrices of the phase encoding in (3.34) for each voxel in the current slice. Similar to RF pulse application, the matrix of each voxel is multiplied with the magnetization vector of the same voxel.

Between the phase encoding and π RF pulse application the spin system is left for relaxation. The matrix representation of the relaxation is given in (3.36). The matrix representations of the phase encoding and RF application operations do not contain any relaxation terms. Since the matrix representation of free relaxation is composed of two diagonal matrices, it commutes with any matrix. Consequently, the relaxations during phase encoding and RF field application can be concatenated with free relaxation period. Hence, T_{w1} is:

$$T_{w1} = \frac{T_E}{2} \quad (3.115)$$

For this period, the half of π RF pulse duration is covered. Similar to other operations, this operation is performed for all of the voxels in the slice.

The application of π RF pulse is the same with $\frac{\pi}{2}$ pulse. After that, the spin system is relaxed for T_{w2} duration. This time relaxation duration is:

$$T_{w2} = \frac{T_E}{2} - \frac{T_{acq}}{2} \quad (3.116)$$

Signal acquisition part of the simulator is summarized in Figure 3.6.

```

While i ≤ SampleNo
    Apply frequency encoding gradient for  $\Delta t_a$  duration
    Relax for  $\Delta t_a$  duration
    Convert the magnetizations from local frame to global rotating frame
    Convert the magnetizations from rotating frame to static frame
    Calculate derivative of magnetizations using Bloch equation
    Calculate the voltage induced
    Store the result of  $i^{\text{th}}$  sample
    i ++
end
Save the resultant signal

```

Figure 3.6: Signal acquisition phase of the simulator

Application of the frequency encoding gradient is not different than the application of the phase encoding gradient. Consequently, the frequency encoding is applied for Δt_a duration using the matrix representation of the phase encoding. The phase angle introduced by the frequency encoding is given by:

$$\phi(\bar{x}) = \gamma \|\vec{B}_{FRQ}(\bar{x})\| \Delta t_a \quad (3.117)$$

After that, the system is relaxed for Δt_a duration as explained above. To be able to calculate the voltage induced on the receiver coil, the magnetization vector in each voxel of slice should be represented in the global reference frame. To be able to obtain magnetization vector in the global frame, first, it should be converted to the global rotating frame, then to the inertial global frame. The conversion to the global rotating frame is done by multiplying the magnetization vectors of each voxel with $T_{spheric}^T$ matrix (in (3.42)) of the same voxel. Similarly, the conversion to the inertial frame is done by multiplying the obtained magnetization in the rotating frame by $T_{rot}^T(t)$ matrix at time instant given by:

$$t = T_E - \frac{T_{acq}}{2} + n\Delta t_a \quad (3.118)$$

After that, the derivative of the magnetization vector in the global inertial frame is calculated using the Bloch equation, (3.8). After calculating the derivative of magnetization, the discrete form of (3.37) (given in (3.119)) is used to calculate the voltage induced at the receiver coil.

$$V[i] = - \sum_{p=1}^P \vec{B}_2(\bar{x}_p) \cdot \left. \frac{d\vec{M}(\bar{x}_p, t)}{dt} \right|_{t=i\Delta t_a} \quad (3.119)$$

The procedure is repeated for all of the samples in T_{acq} . The number of samples can be calculated using:

$$SampleNo = round\left(\frac{T_{acq}}{\Delta t_a}\right) \quad (3.120)$$

After collecting all the samples, the obtained voltage signal is stored to be used in the reconstruction algorithm.

After acquiring the voltage induced, the simulator waits for the relaxation of spins to the initial state for T_{w3} duration. This duration is given by:

$$T_{w3} = T_R - T_E - \frac{T_{acq}}{2} \quad (3.121)$$

Then, the same pulse sequence operations are repeated for all of the phase encodings. The whole procedure is repeated for the total number of slices.

3.3.4. MRI In Inhomogeneous Magnetic Fields Implementation

The same pulse sequence is implemented for magnetic resonance in inhomogeneous magnetic fields. The same magnetic fields are used as in the simulator implementation. The procedure is summarized as a pseudo code in Figure 3.7.

In *Get user Input* part in Figure 3.7, the simulator expects the user to define $\|\vec{B}_1\|$, $\|\vec{B}_2\|$, G_{FRQ} , T_E , T_R , T_1 , T_2 , the number of phase encoding steps, the number of frequency encodings and the path of magnetic field distribution in FOV. The relaxation time constants, T_1 and T_2 , are just the estimated ones since the exact distribution is not known.

In *Initialize* part in Figure 3.7, main magnetic field is loaded and other magnetic fields are created referencing to the main magnetic field as in the simulator. Using \vec{B}_0 distribution and the number of slices entered by the user, RF field's bandwidth and center frequencies are calculated. After that, the sampling duration for the discrete time signal representation is used as in the simulator.

```

Get user Inputs
Initialize
While k ≤ Number of Slices
    Find kth Slice
    Slice Initialize
    While j ≤ Number of phase encodings
        Find transformation matrix for  $\frac{\pi}{2}$  RF pulse application
        Find transformation matrix for phase encoding
        Find transformation matrix for free relaxation for Tw1 duration
        Find transformation matrix for  $\pi$  RF pulse application
        Find transformation matrix for free relaxation for Tw2 duration
        Acquire and Save the transformation from  $\tilde{M}^0$ 
        Find transformation matrix for free relaxation for Tw3 duration
        j ++
    end
    k ++
end

```

Figure 3.7: Pseudo code of MR in inhomogeneous magnetic fields

In *Find kth Slice* part, equation (3.94) is used to find the current slice voxels.

In *Slice Initialize* part, T_{RF} , T_{acq} and ΔG_{PE} are calculated. Afterwards, the steps of the phase encoding field's magnitude derivative with respect to the phase encoding

direction for each phase encoding are generated. After that, the initial state transition matrices for each voxel are created. The initial state transition matrices are identity matrices. Then, free relaxation durations are calculated. In this case, pulse sequence operations contain relaxation terms. Consequently, relaxation times are given by:

$$\begin{aligned}
T_{w1} &= \frac{T_E}{2} - \frac{T_{acq}}{2} - \frac{3T_{RF}}{2} \\
T_{w2} &= \frac{T_E}{2} - \frac{T_{acq}}{2} - \frac{T_{RF}}{2} \\
T_{w3} &= T_R - T_E - \frac{T_{acq}}{2}
\end{aligned} \tag{3.122}$$

Since the RF pulses yields time varying A in (3.57), the transition matrices for RF pulses are calculated using Runge-Kutta (4, 5) algorithm. This algorithm is provided by MATLAB and is not implemented in this work. As described previously, to find the state transition matrix from \vec{M}^0 to $\vec{M}(t)$, equation (3.72) is solved three times for the assignment of each unit vector of \mathbb{R}^3 as \vec{M}^0 .

The state transition matrices, $\Phi(\vec{x}; T_{RF} + t_0, 0)$, of $\frac{\pi}{2}$ RF pulse are found as described above. This pulse is assumed to be applied at t_0 .

Gradient fields are assumed to be time invariant in this work. Therefore, after turning on these fields, A in (3.57) is time invariant. Consequently, equation (3.71) is used to find the state transition matrix after the application of gradient fields.

Between the phase encoding and π RF pulse application the spin system is left for relaxation. In this duration no input is applied. In other words, A is time invariant and equal to A_0 . Consequently, again (3.71) is used to find state the transition matrix after the relaxation duration, T_{w1} .

The state transition matrix after the application of π RF pulse is calculated with the same procedure used in $\frac{\pi}{2}$ RF pulse. While calculating the state transition matrices after π pulse, the magnitude of RF field is doubled to keep the duration constant.

After π pulse, the spin system is relaxed for T_{w2} duration. The state transition matrix after T_{w2} is calculated using the same procedure as in the previous relaxation.

Calculation of S matrix in equation (3.83) is summarized in Figure 3.8.

```

While i ≤ SampleNo
    Calculate the state transition matrix using (3.60)
    Calculate  $\tilde{F}[i]$  using (3.76)
    Form  $i^{\text{th}}$  row of  $S$  matrix from  $\tilde{F}^T[i]$ 
    i ++
end
Save the resultant matrix  $S$ 

```

Figure 3.8: S matrix calculation steps

The state transition matrix is calculated using small duration approximation. This assumption is valid since the S matrix is calculated for Δt_a duration. $\tilde{F}[i]$ is defined in (3.77), but its components are calculated using (3.76). Then, the S matrix is formed as in (3.82); i.e, i^{th} row of the S matrix is formed by $\tilde{F}^T[i]$. All rows of the S matrix is filled with \tilde{F}^T and resultant matrix is saved to be used in the reconstruction algorithm.

After creating the S matrix, the spin system is relaxed for T_{w3} duration. This duration is very large compared to other intervals. Consequently, large duration approximation holds and the state transition matrix is calculated using (3.59). The inverse of matrix A , which is equal to A_0 , is calculated using the facts proven in the theory part.

Then, the same pulse sequence operations are repeated for the total number of the phase encodings. The whole procedure is repeated for all of the slices.

3.3.5. Image Reconstruction Technique Implementation

The image reconstruction technique is a procedure to solve equation (3.87). The steepest descent algorithm is used to solve the optimization problem defined in the theory part. The steps of the steepest descent algorithm are summarized in Figure 3.9.

$$\begin{array}{l}
 \underline{A} = (D^T \underline{S}^H \underline{S} D) \\
 \underline{b} = (\underline{S} D)^H \underline{\tilde{V}} \\
 i = 0 \\
 \underline{r}_i = \underline{b} - \underline{A} \underline{x}_i \\
 \text{While } i \leq \text{iterNo} \\
 \quad \alpha_i = \frac{\underline{r}_i^T \underline{r}_i}{\underline{r}_i^T \underline{A} \underline{r}_i} \\
 \quad \underline{x}_{i+1} = \underline{x}_i + \alpha_i \underline{r}_i \\
 \quad \underline{r}_i = \underline{b} - \underline{A} \underline{x}_i \\
 \text{end}
 \end{array}$$

Figure 3.9: The steepest descent algorithm

The steps of the reconstruction algorithm are summarized in Figure 3.10. In the initialization part, the variables used in the algorithm are initialized. For each slice, Ω_i and $\vec{B}_0(\bar{x}_j), \bar{x}_j \in \Omega_i$ are loaded. $\vec{B}_0(\bar{x})$ is used to calculate the demodulator's frequency and cutoff frequency of the low pass filter. After that, j^{th} phase encoding of k^{th} slice data is loaded both from simulator and MR in inhomogeneous magnetic field parts. Then, both signals are demodulated using the procedure described in the theory part. After that, the steepest descent algorithm is performed using \underline{x}_0 as the

initial solution for previously defined steps. The obtained solution of the j^{th} phase encoding is used as the initial solution of the steepest descent algorithm at $(j+1)^{th}$ phase encoding. After covering all of the phase encodings, the obtained solution is inserted into the appropriate components of $\tilde{M}^0(\bar{x}_j)$. The procedure is repeated for all slices.

```

Initialize
While k ≤ Number of Slices
    Load kth slice data
     $\underline{x}_0 = \bar{0}$ 
    While j ≤ Number of phase encodings
        Load S matrix of kth slice's jth phase encoding
        Load signal data of kth slice's jth phase encoding,  $\tilde{V}$ 
        Demodulate S and  $\tilde{V}$  and obtain  $\underline{S}$  and  $\underline{\tilde{V}}$ 
        Perform the steepest descent algorithm for  $\underline{x}_0$  and obtain  $\underline{x}$ 
         $\underline{x}_0 = \underline{x}$ 
        j++
    end
    k++
end

```

Figure 3.10: Pseudo code of the image reconstruction technique

3.4. Results

In this section, obtained results are shown. First, the results of the simulator and the image reconstruction technique are shown for homogeneous magnetic fields case. These results are shown to validate the theory developed. Second, the results of both the simulator and the image reconstruction technique for inhomogeneous magnetic fields are shown.

3.4.1. Results Of Homogeneous Magnetic Fields Case

The constructed simulator and MR in inhomogeneous magnetic fields theory is tested using homogeneous magnetic fields. Homogeneous magnetic field distribution can be thought as a special case of the inhomogeneous distribution of magnetic fields. Moreover, the use of homogeneous magnetic fields as a test case provides a basis to compare the performance of the image reconstruction technique with the conventional inverse Fourier technique.

The input parameters are chosen as shown in Table 3.1.

Table 3.1: The chosen input values

Slice No = 1	$N_x = 11mm$	$T_E = 5ms$ $T_R = 5s$	$\ \vec{B}_0\ = 0.1T$
Number of Phase Encoding = 11	$N_y = 11mm$		$\ \vec{B}_1\ = 1mT$
Number of samples (N_{freq}) = 11	$N_z = 1mm$		$\ \vec{B}_2\ = 1T$
	$dV = 1mm^3$		

The object being imaged is created as a square object as shown in Figure 3.11.

For the object shown in Figure 3.11, relaxation time constants are chosen as:

$$T_1 = 1s$$

$$T_2 = 100ms$$

\vec{M}^0 is calculated using the equation (3.4). The magnitude distribution of initial the magnetization, $\|\vec{M}^0\|$, is shown in Figure 3.12

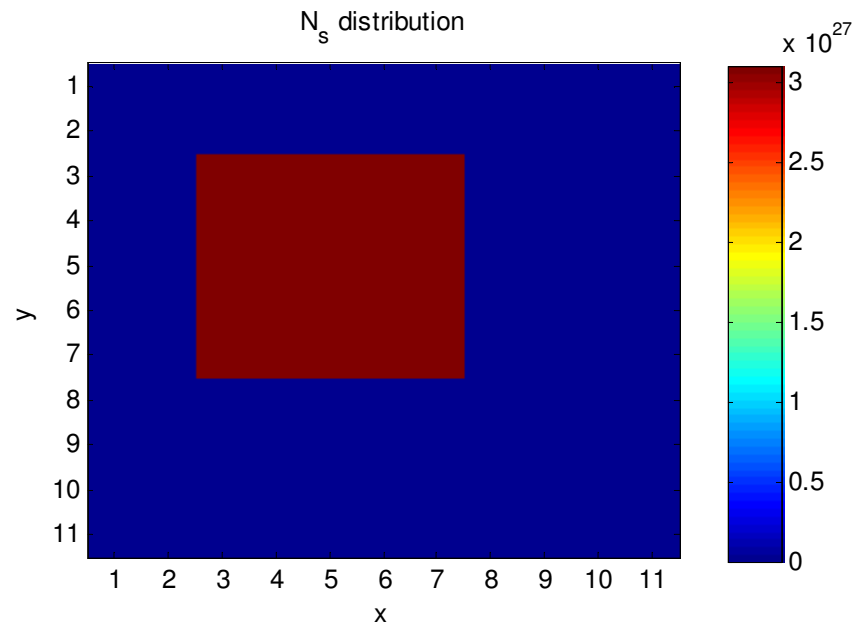


Figure 3.11: N_s distribution

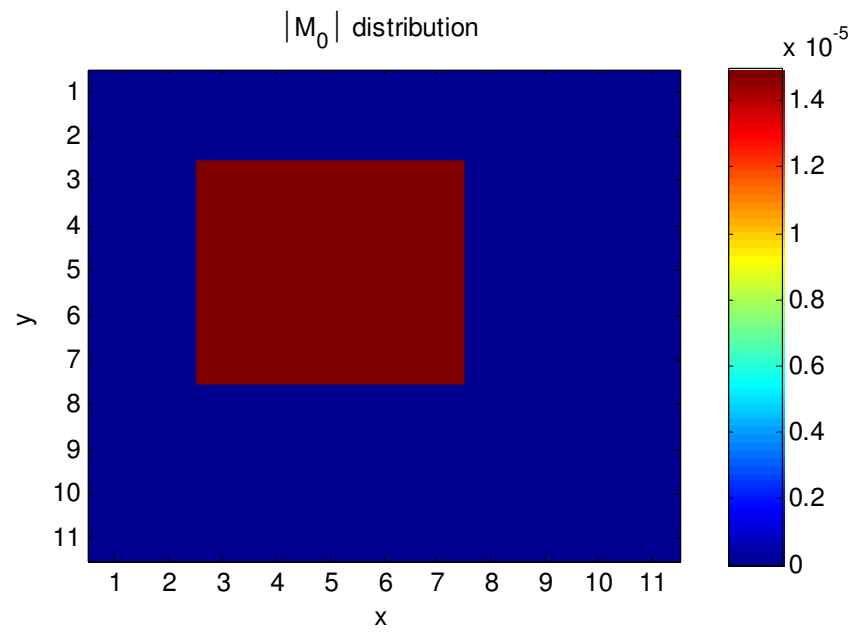


Figure 3.12: $\|\vec{M}^0\|$ distribution

The distributions of the gradient magnetic fields' magnitude are shown in Figures 3.13 and 3.14. Although the magnitude of the phase encoding gradient is varied during the evaluation of the pulse sequence, Figure 3.13 is given for unity phase encoding gradient's magnitude derivative.

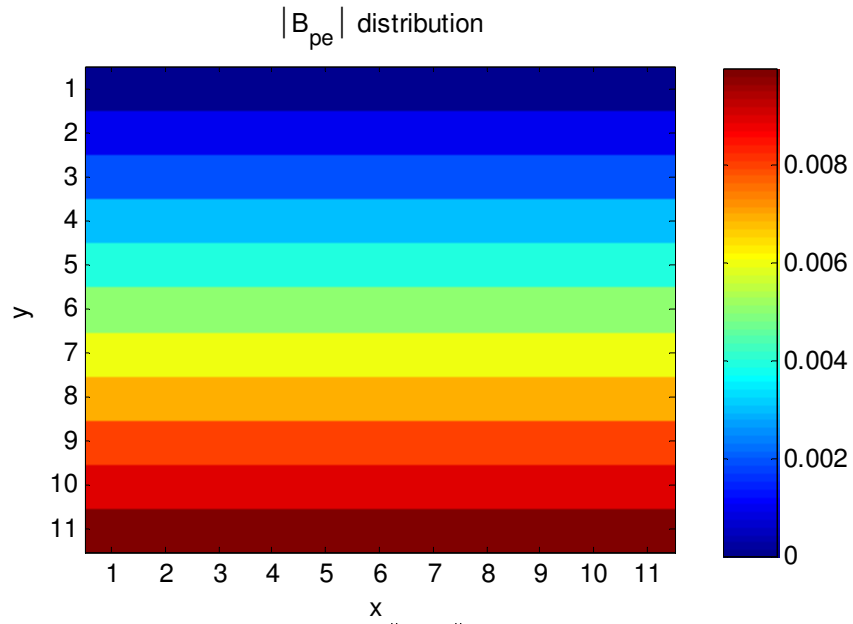


Figure 3.13: $\|\vec{B}_{PE}\|$ distribution

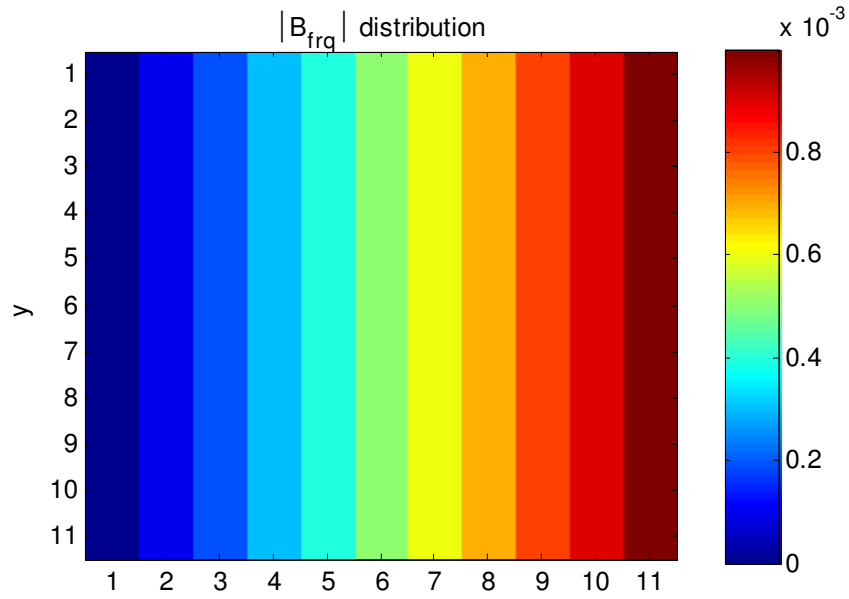


Figure 3.14: $\|\vec{B}_{FRQ}\|$ distribution

The reconstructed image using inverse Fourier technique is given in Figure 3.15.

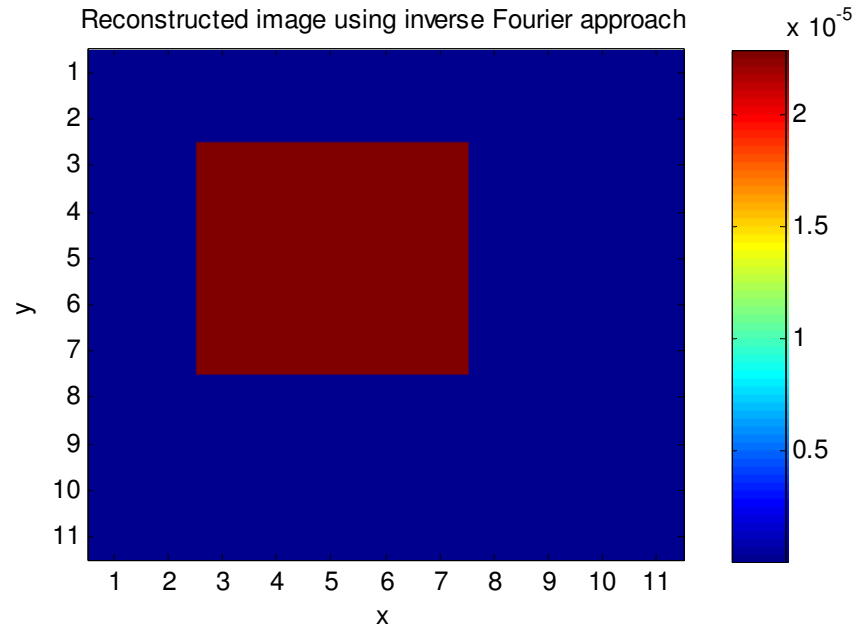


Figure 3.15: The reconstructed image using inverse Fourier technique

The reconstructed image using the developed technique is shown in Figure 3.16.

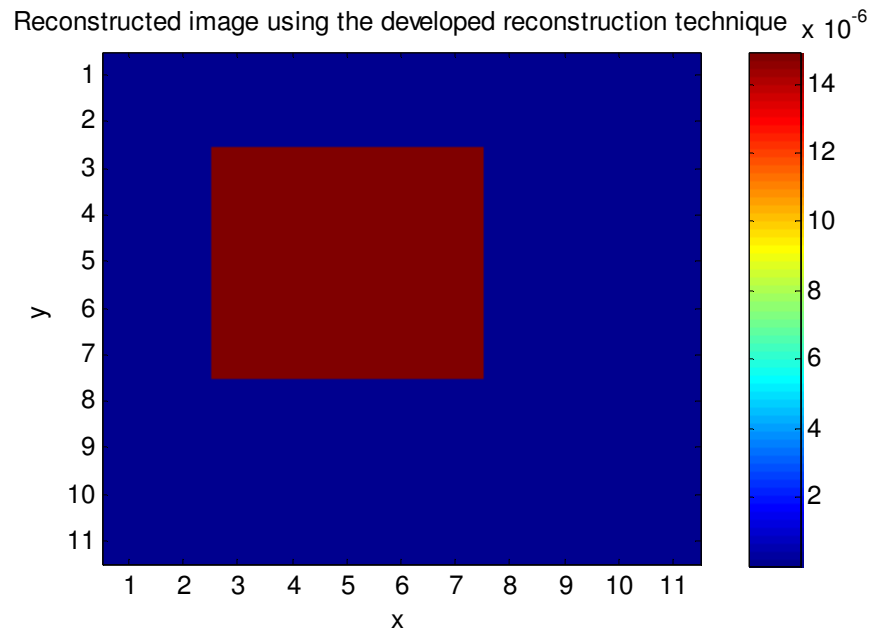


Figure 3.16: The reconstructed image using developed technique

The error of reconstructed image with respect to $\|\vec{M}^0\|$ distribution is calculated using:

$$err(\bar{x}_p) = \frac{\left(\left|\operatorname{Re}\{X_p\}\right| - \left\|\vec{M}^0(\bar{x}_p)\right\|\right)^2}{\left\|\vec{M}^0(\bar{x}_p)\right\|^2}, p = 1..P \quad (3.123)$$

where $X_p = \left\|\vec{M}^0(\bar{x}_p)\right\|_r$ is the calculated magnitude of \vec{M}^0 at the p^{th} pixel of image; $\left\|\vec{M}^0(\bar{x}_p)\right\|$ is the magnitude of \vec{M}^0 at p^{th} pixel of object being imaged. The calculated magnitude of \vec{M}^0 is in the modulus due to the fact that the reconstruction algorithm outputs complex numbers. The formula is only valid for voxels at which spins exist, i.e., $\left\|\vec{M}^0(\bar{x}_p)\right\|$ is nonzero.

Similarly, the conventional image reconstruction technique also aims to reconstruct the magnitude of \vec{M}^0 distribution. Consequently, the same error formula can be used to find the error distribution in the image which is reconstructed using inverse Fourier technique.

In Figures 3.17 and 3.18, the error distributions of reconstructed images are given for the developed technique and the inverse Fourier technique, respectively.

The error performance of the developed technique is better than the Fourier based image reconstruction technique. This is due to the fact that the developed technique is iterative. When an iterative technique converges to the solution, reconstruction error decreases. Due to this fact, the same set of data, with an initial solution obtained from previous sets of data, is used for more than one time in the procedure. Using this fact, the reconstruction technique has been run three times with the same set of data to obtain the image shown in Figure 3.16.

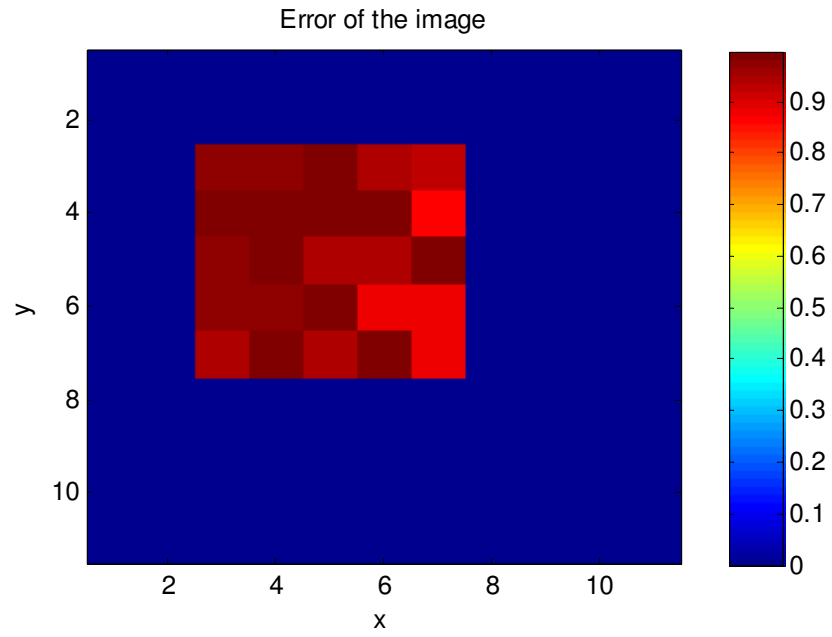


Figure 3.17: Error distribution of the image reconstructed with the developed technique

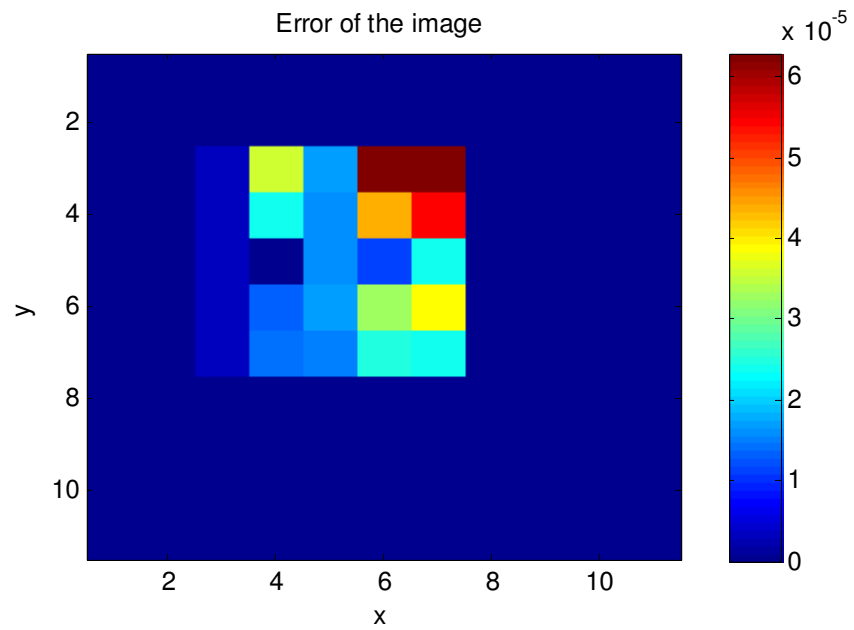


Figure 3.17: Error distribution of the image reconstructed with the developed technique

To investigate the noise performance of the technique, the developed technique is tested with noisy data. Zero mean Gaussian white noise is added to the measurement vector, \tilde{V} , to create a noisy data. In Figures 3.18 and 3.19, original measurement vector and noise added measurement vector for one phase encoding are shown, respectively. The noisy measurement shown in Figure 3.18 has a signal to noise ratio (SNR) of 10 dB.

White Gaussian noise has a constant power spectral density function. In other words, it contains all of the frequencies. Consequently, since the added noise is white Gaussian noise, low pass filter in the demodulator can not completely remove the noise from the measurement, but it increases the SNR. In Figures 3.20 and 3.21, the output of the demodulator for the original measurement and the noisy measurement are shown, respectively.

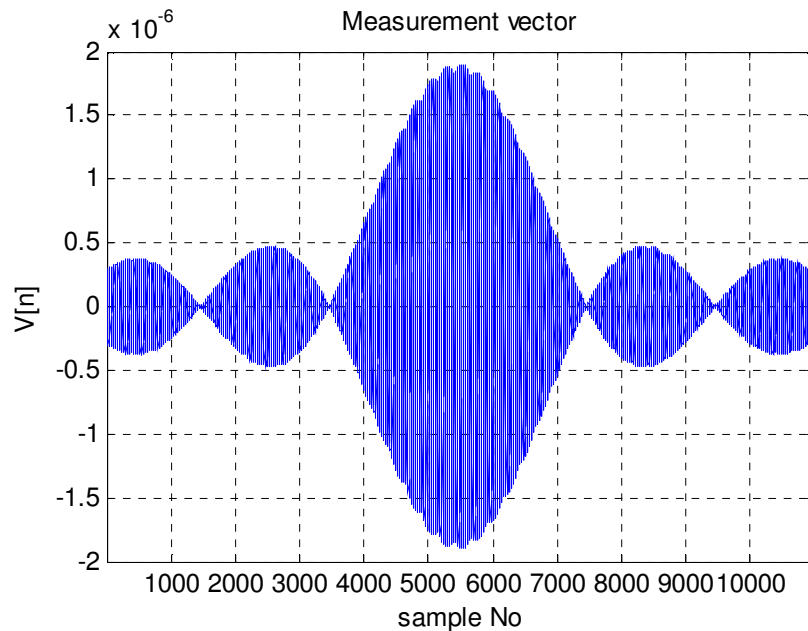


Figure 3.18: Noiseless measurement vector

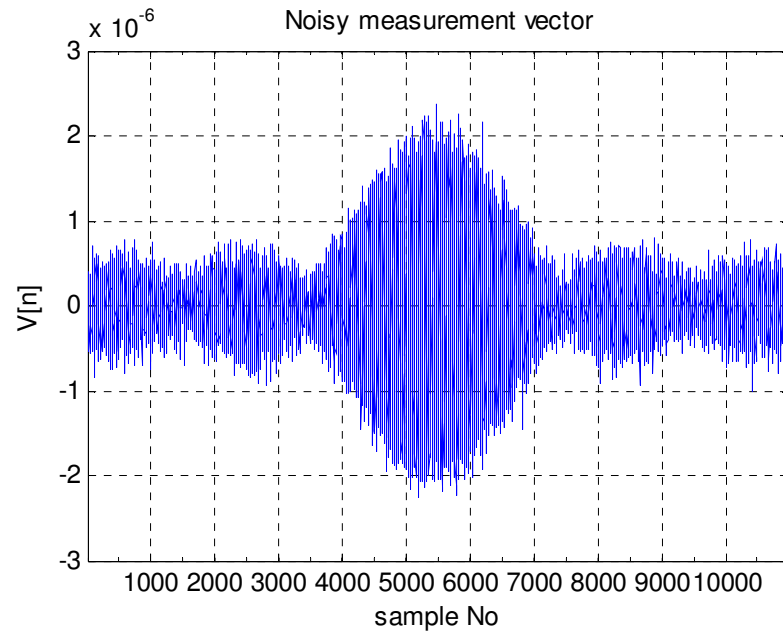


Figure 3.19: Noisy measurement vector
SNR = 10 dB

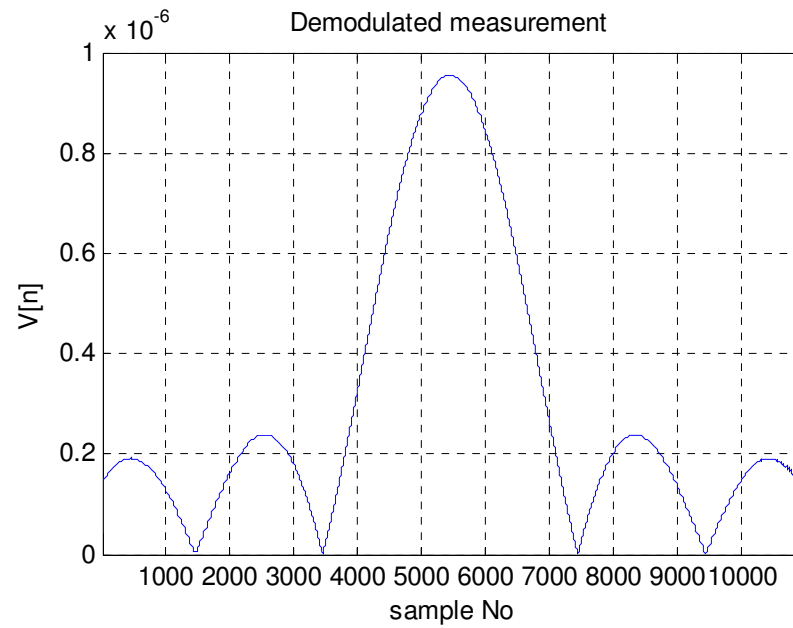


Figure 3.20: Demodulated noiseless measurement vector

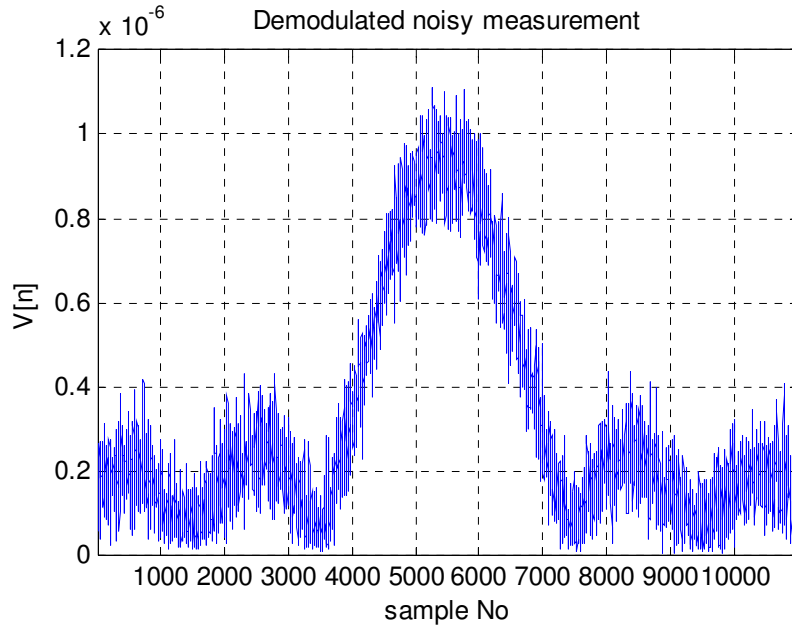


Figure 3.21: Demodulated noisy measurement vector

All of the measurement vectors for all phase encodings are exposed to noise and SNR value is kept constant at 10 dB. The reconstructed image using the developed technique and the inverse Fourier technique is given in Figures 3.22 and 3.23, respectively.

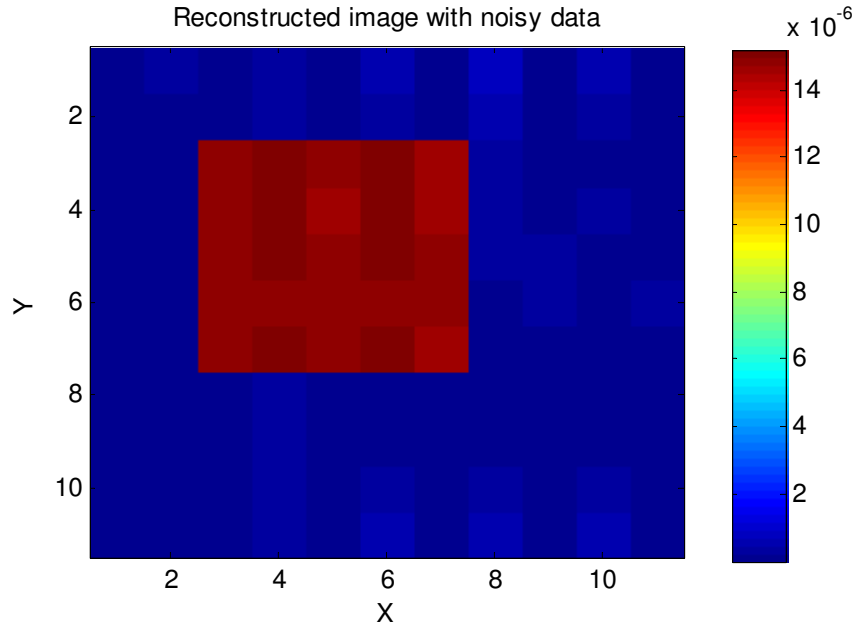


Figure 3.22: The image reconstructed with the developed technique using noisy data

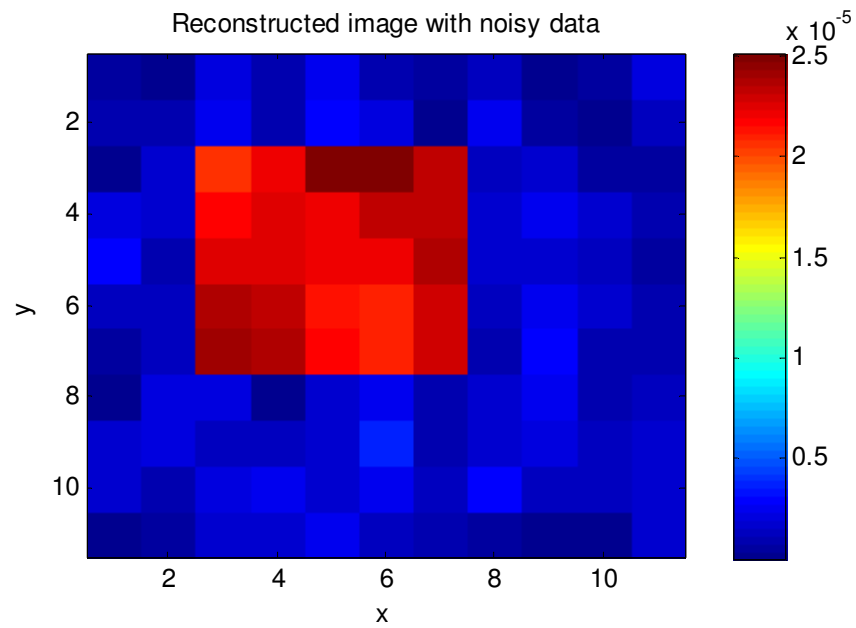


Figure 3.23: The image reconstructed with the inverse Fourier technique using noisy data

Figure 3.22 shows that the developed technique is less sensitive to noise than the conventional inverse Fourier technique.

Although obtained results are encouraging for the inhomogeneous case, before proceeding to inhomogeneous magnetic fields case, reconstruction of images of some other objects is required. The first object, shown in Figure 3.24, is known as the Shepp-Logan phantom.

For the chosen input values, given in Table 3.1, the image of the object is reconstructed with the developed technique. The image is shown in Figure 3.25.

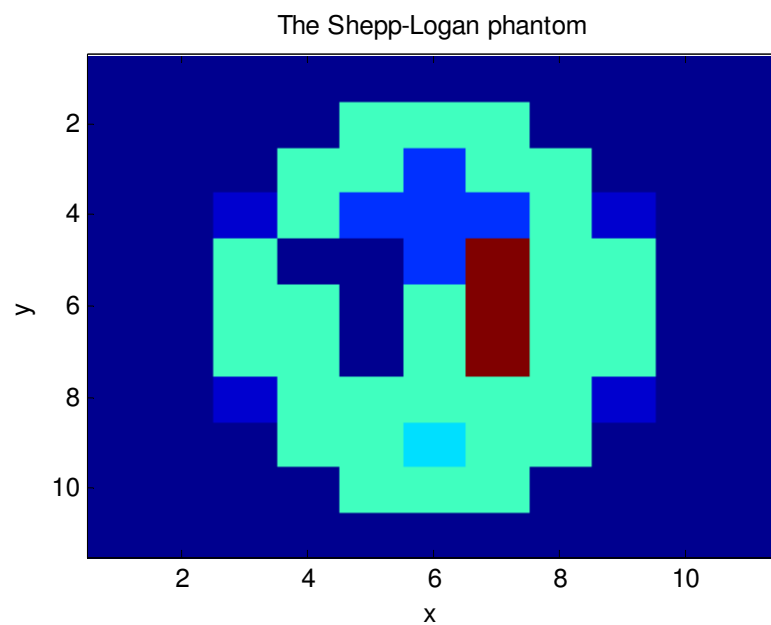


Figure 3.24: The Shepp-Logan phantom

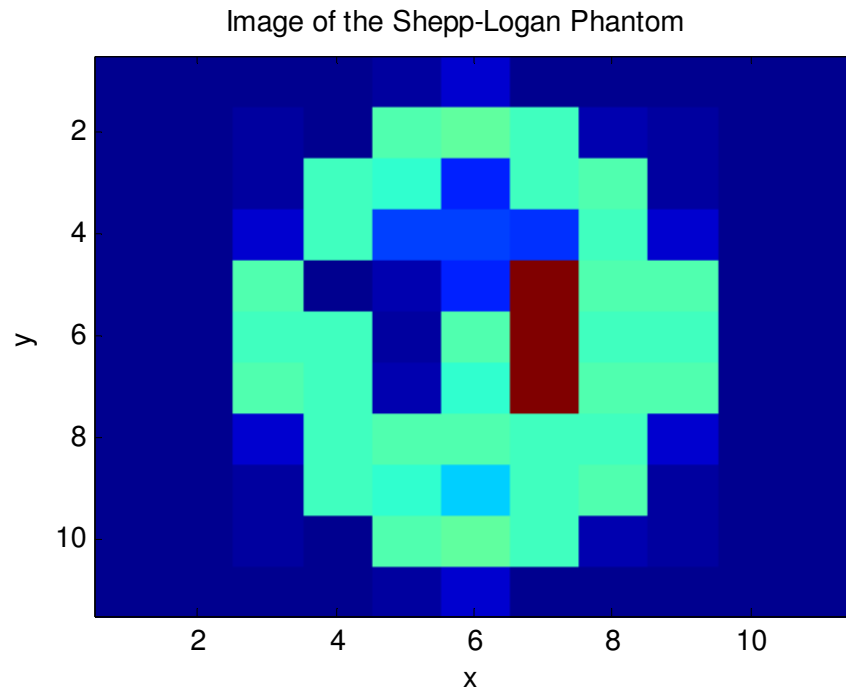


Figure 3.25: The reconstructed image of the Shepp-Logan phantom

Another object that is used for tests is shown Figure 3.26. The reconstructed image of the object shown above is shown in Figure 3.27.

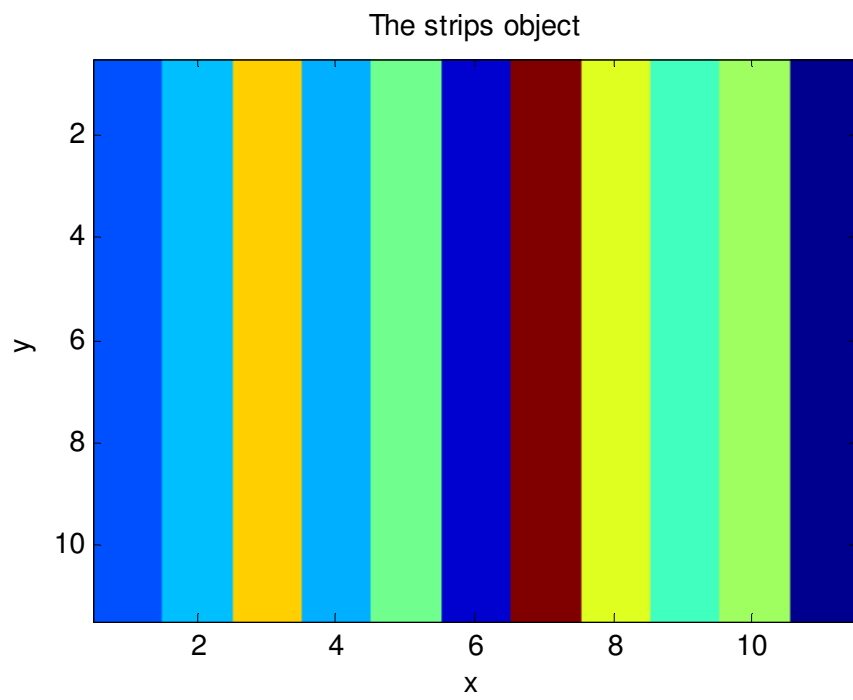


Figure 3.26: An object of strips

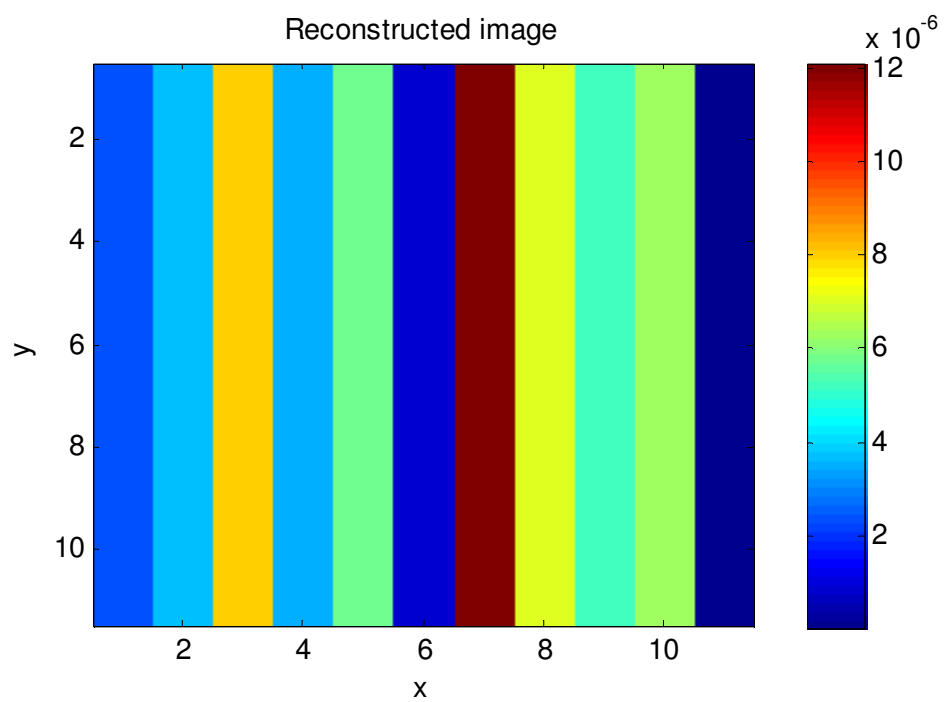


Figure 3.27: The reconstructed image of the object

The results obtained are satisfactory and shows that the developed technique can reconstruct images as good as the conventional technique does. Moreover, its performance for noisy data is better than the inverse Fourier technique. Consequently, the technique can be used in MRI in inhomogeneous magnetic fields.

3.4.2. Results of Inhomogeneous Magnetic Fields Case

In this part, the performance of the developed image reconstruction technique under inhomogeneous magnetic fields is demonstrated by presenting the obtained results.

The main magnetic field is calculated for the magnet structure obtained in Chapter 2. The dimensions of side length and height of unit magnets have not been defined. The magnetization strength, M_0 , of the unit magnets has been taken as 4π . To calculate a realistic magnetic field distribution, the side length, a , the height, h , and magnetization strength, M_0 , are defined as shown in Table 3.2.

The input parameters are chosen as shown in Table 3.2:

Table 3.2: The chosen input values

Slice No = 1 Number of Phase Encoding = 11 Number of samples (N_{freq}) = 11		$T_E = 5ms$ $T_R = 5s$
$N_x = 11mm$ $N_y = 11mm$ $N_z = 1mm$ $dV = 1mm^3$	$\ \vec{B}_1\ = 1mT$ $\ \vec{B}_2\ = 1T$	$a = 4cm$ $h = 3cm$ $M_0 = \frac{10^7}{4\pi} A/m$

The object is placed in the ROI of magnetic field calculations. Consequently, the magnetic field is calculated at each voxel of the object. The main magnetic field's

magnitude distribution is shown in Figure 3.28. The gradient fields are created as described in the implementation part. Their magnitude distributions are not the same with the ones shown in Figures 3.13 and 3.14 due to the fact that the effective component of gradient fields are assumed to be parallel to the main magnetic field. Consequently, according to equations (3.111) and (3.112), the magnitudes of generated gradient fields are as shown in Figures 3.29 and 3.30.

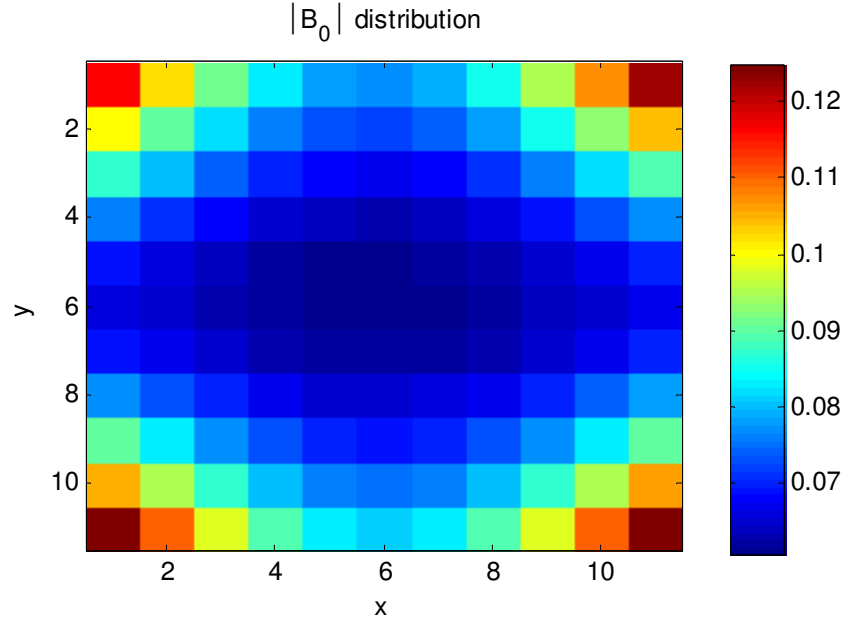


Figure 3.28: The main magnetic field's magnitude distribution

As a first test object, the one used in homogeneous magnetic fields case is used. The number of spin distribution is shown in Figure 3.11. Since the main magnetic field is not uniformly distributed through the object, $\|\vec{M}^0\|$ distribution is different than number of spins distribution. The initial magnetization distribution is shown in Figure 3.31.

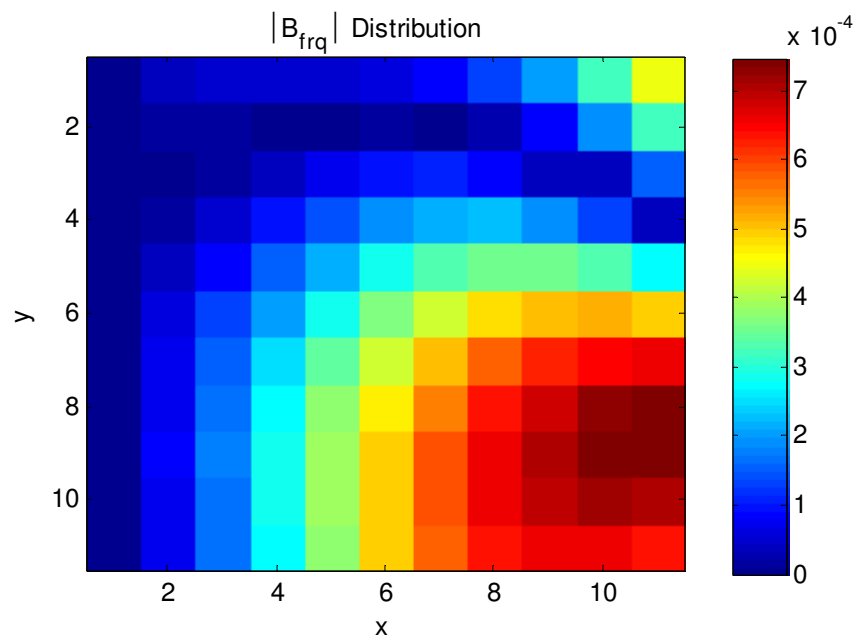


Figure 3.29: Frequency encoding gradient's magnitude distribution

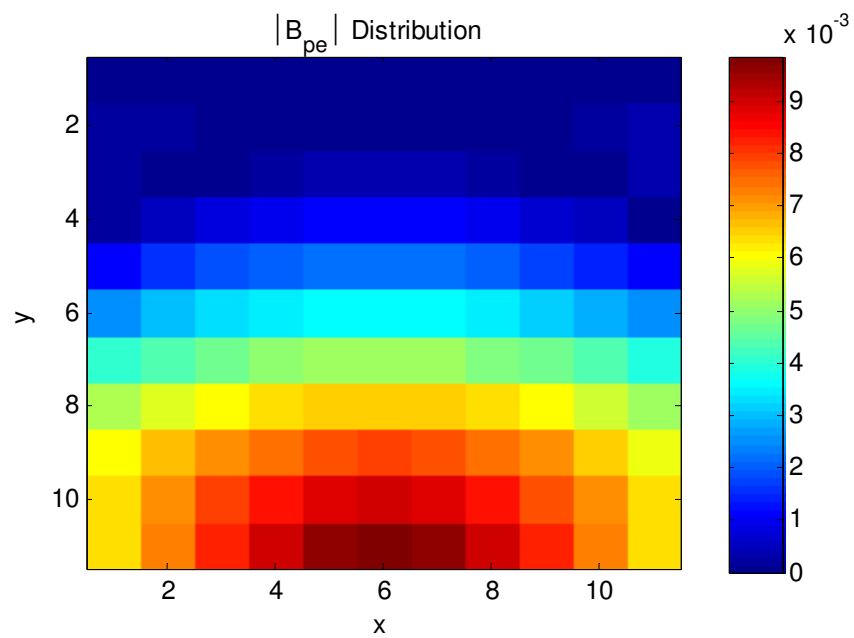


Figure 3.30: Phase encoding gradient's magnitude distribution

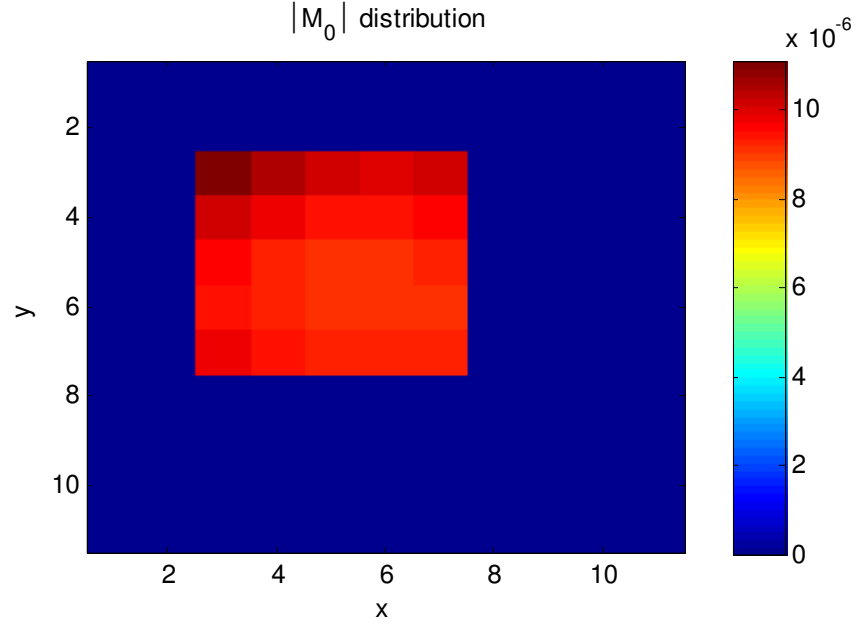


Figure 3.31: $\|\vec{M}^0\|$ distribution of the object being imaged

Since it is not possible to reconstruct images in inhomogeneous magnetic fields using inverse Fourier technique, only images reconstructed using the developed technique are given.

For the parameters given in Table 3.2 and the object shown in Figure 3.31, the reconstructed image after one run of the data sets in the developed technique is shown in Figure 3.32.

The error of reconstructed image is calculated using (3.123). The error distribution is shown in Figure 3.33.

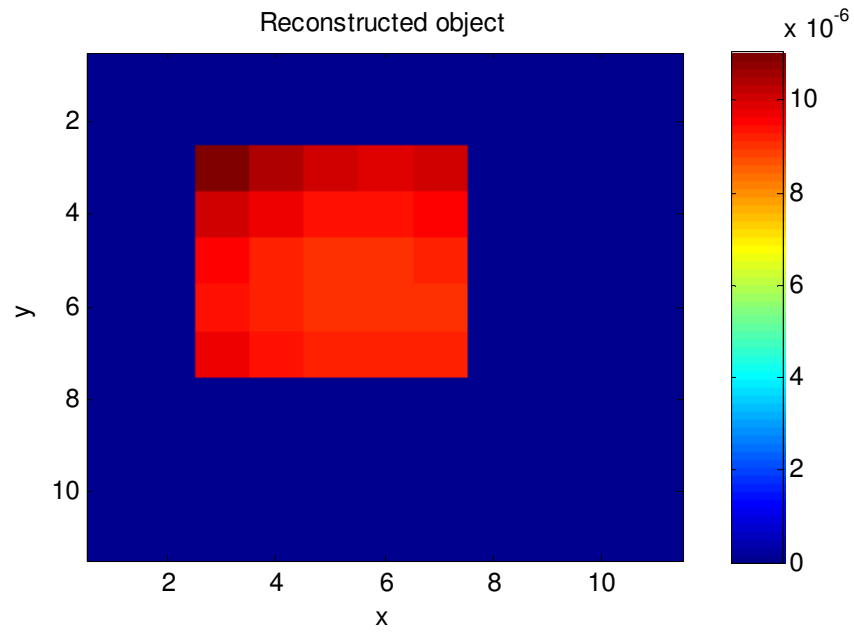


Figure 3.32: The reconstructed image one tour of data sets

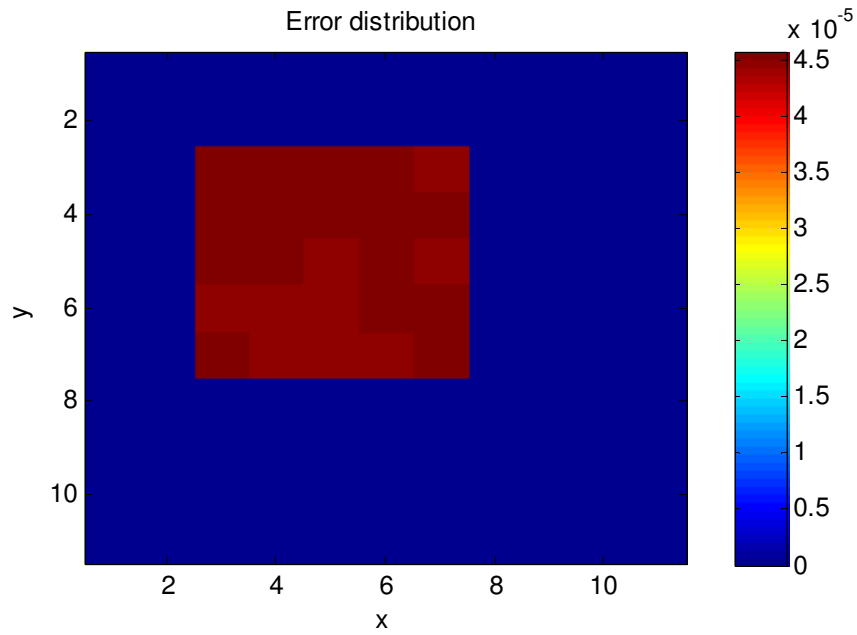


Figure 3.33: The distribution of error

To check the noise performance of the technique under noisy measurements, again the measurement vector is imposed to a white Gaussian noise. The power of the

noise is adjusted so that the SNR is 10 dB. In Figures 3.34 and 3.35, noiseless and noisy measurement vectors for the first phase encoding are shown, respectively.

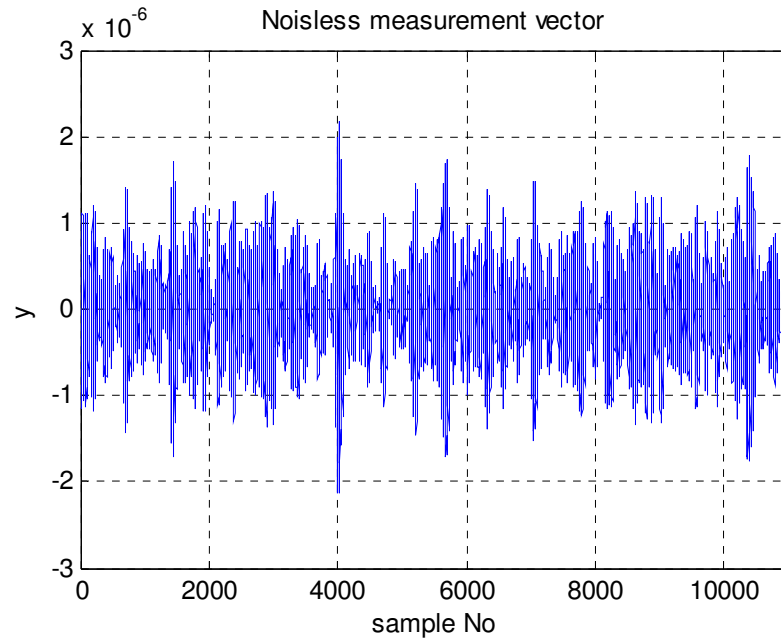


Figure 3.34: Noiseless measurement vector

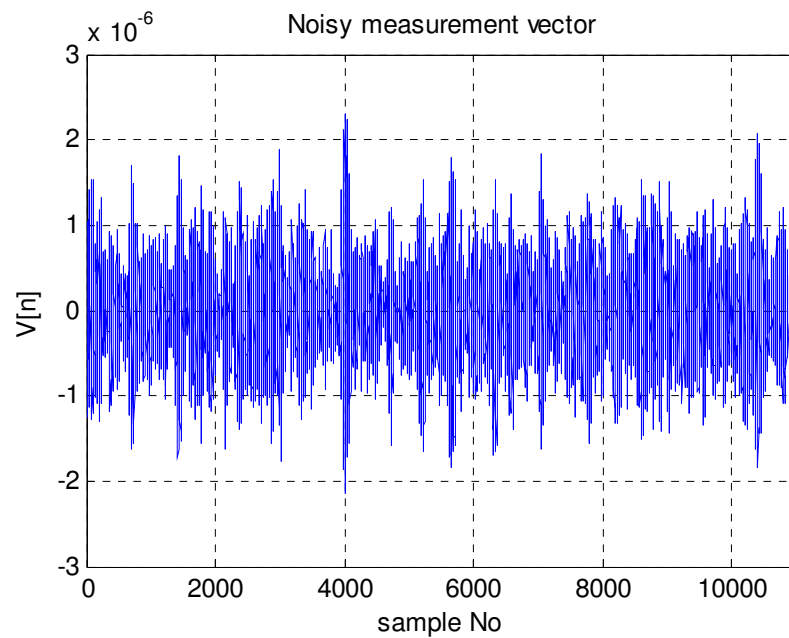


Figure 3.35: Noisy measurement vector

The low pass filter in the demodulator can not filter out the white noise added to measurements. The demodulator outputs for noiseless and noisy measurements are shown in Figures 3.36 and 3.37, respectively.

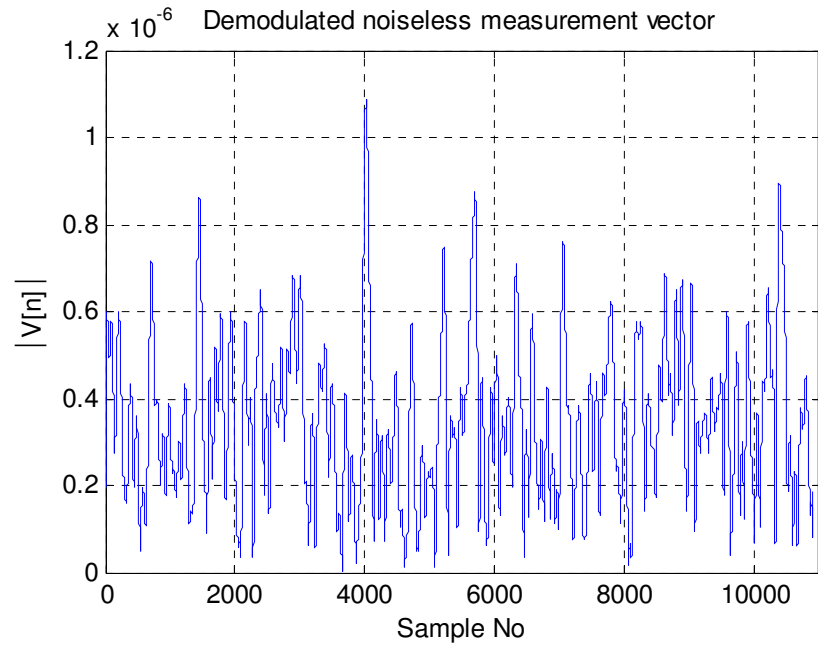


Figure 3.36: The demodulator output for noiseless measurements

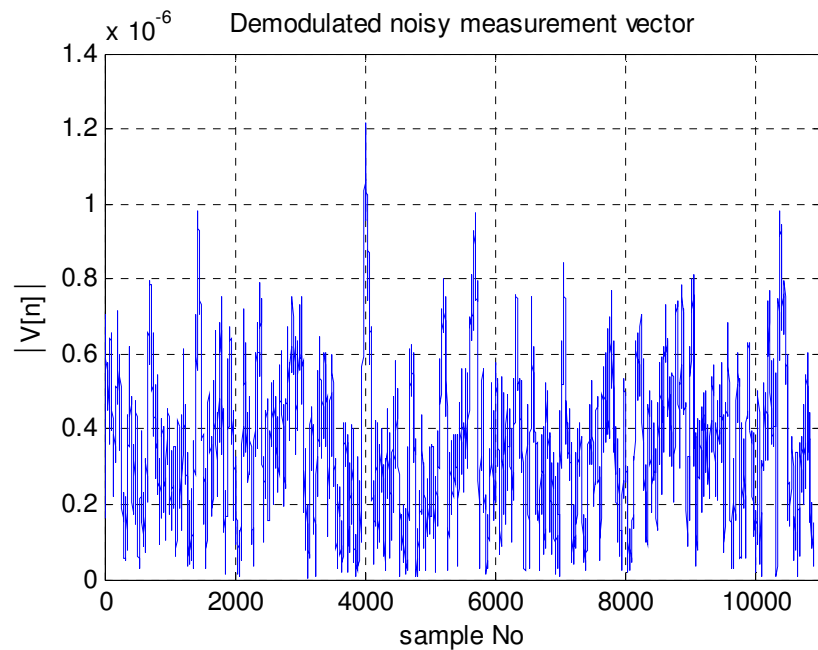


Figure 3.37: The demodulator output for noisy measurements

The reconstructed image using noisy data is shown in Figure 3.38.

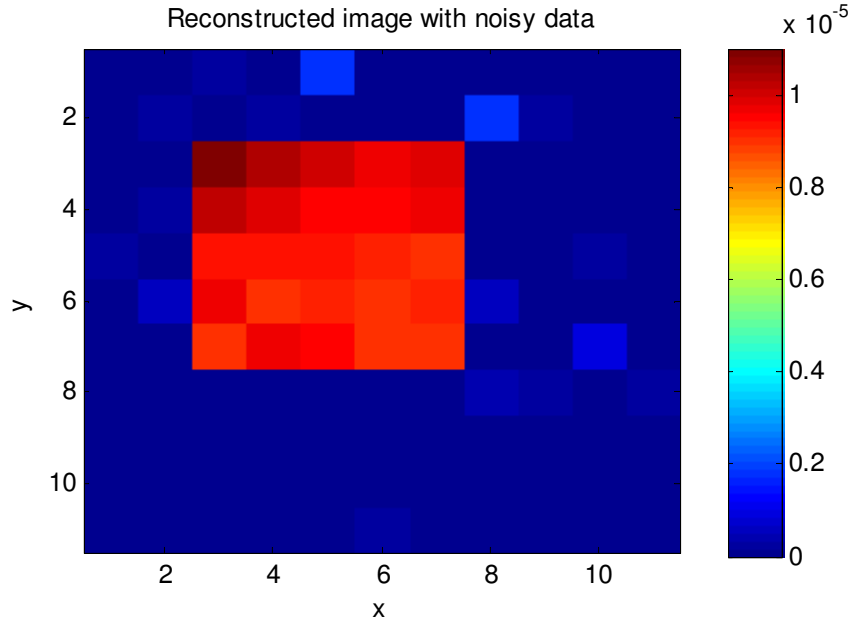


Figure 3.38: The reconstructed image using noisy data

The error of this image should be calculated within the whole object. However, (3.123) should be modified for this purpose.

$$err(\bar{x}_p) = \frac{\left(\left| \text{Re}\{X_p\} \right| - \left\| \vec{M}^0(\bar{x}_p) \right\| \right)^2}{\left\| \vec{M}^0(\bar{x}_p) \right\|^2 + \varepsilon}, p = 1..P \quad (3.124)$$

where ε is a very small number (i.e. $\varepsilon = 10^{-17}$) real value of the computer. The error distribution is shown in Figure 3.39.

The error distribution shown in Figure 3.39 is scaled so that the maximum error is unity. The error distribution of the image in the voxels with nonzero $\left\| \vec{M}^0 \right\|$ is shown in Figure 3.40.

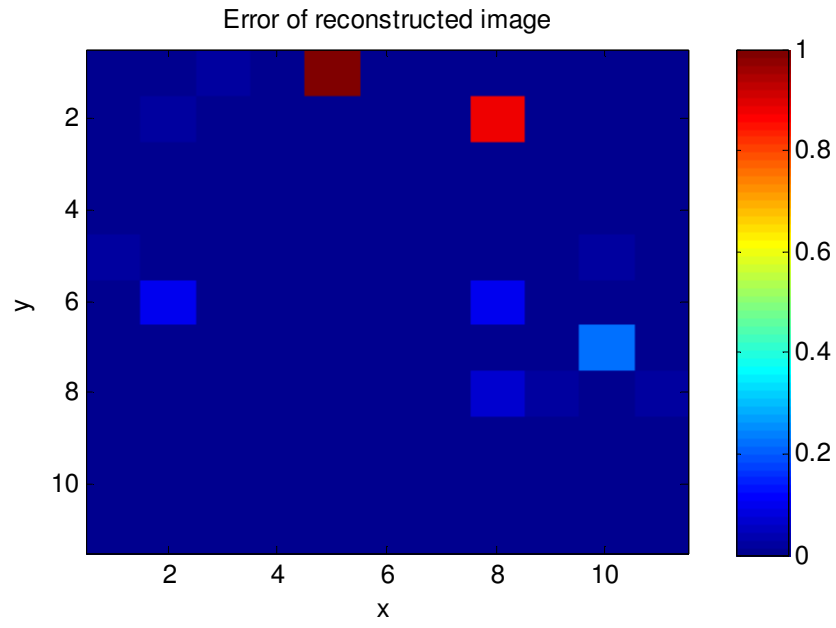


Figure 3.39: The error distribution of image reconstructed with noisy data

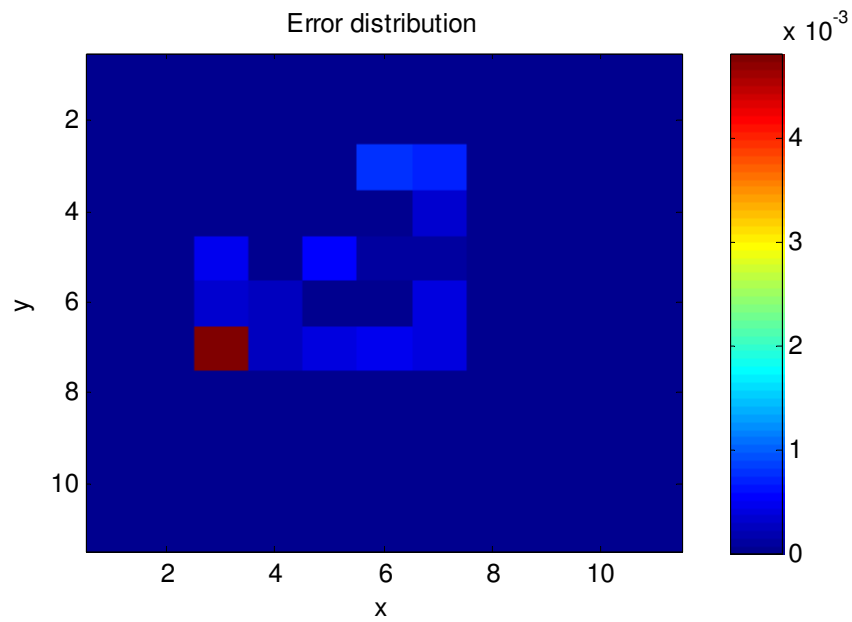


Figure 3.40: The error distribution of object in the voxels where $\|\vec{M}^0\|$ is nonzero

The Shepp-Logan object is used as an input object. The $\|\vec{M}^0\|$ distribution of the Shepp-Logan object is shown in Figure 3.41. The reconstructed image of this phantom is shown in Figure 3.42.

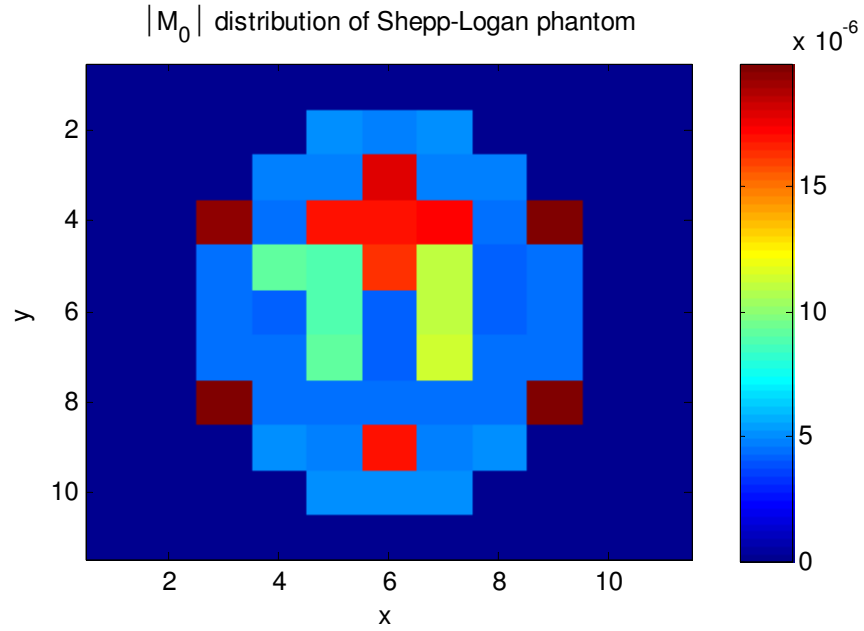


Figure 3.41: $\|\vec{M}^0\|$ distribution of the Shepp-Logan phantom

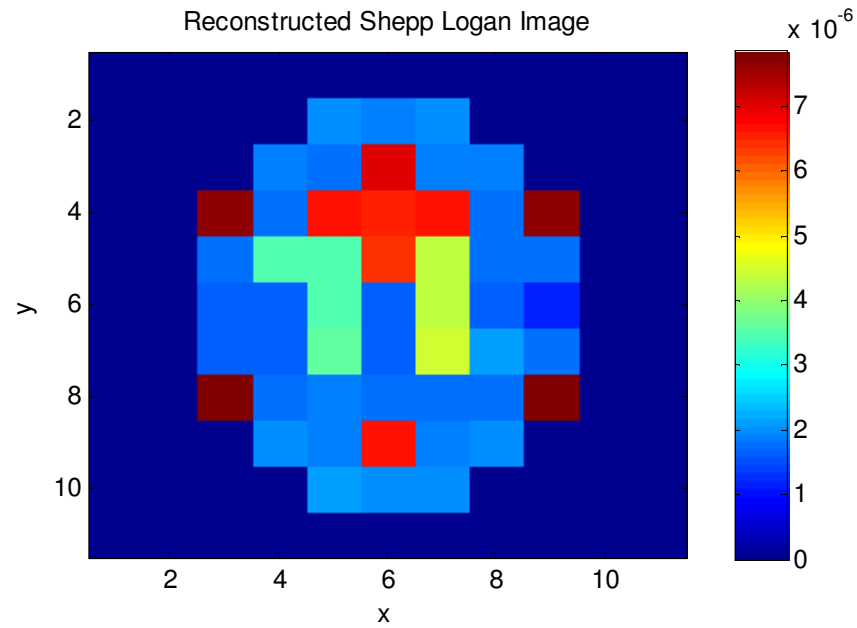


Figure 3.42: The reconstructed image of the Shepp-Logan phantom

The reconstructed Shepp-Logan object has a scale error. Although the image has the same distribution, the $\|\vec{M}^0\|$ varies from the original object. The error formula (3.123) is modified so that it can be used in scaled images.

$$err(\bar{x}_p) = \frac{\left(\alpha |\text{Re}\{X_p\}| - \|\vec{M}^0(\bar{x}_p)\|\right)^2}{\|\vec{M}^0(\bar{x}_p)\|^2}, p = 1..P \quad (3.125)$$

where α is the scale factor between original distribution and reconstructed image. The error distribution of the Shepp-Logan image is shown in Figure 3.42.

Although the reconstruction algorithm was used run for one cycle of the measurements, its output is satisfactory. On the other hand, it is possible to change the field of view (FOV) by narrowing RF field's bandwidth. To test this case, the object shown in Figure 3.11 is used. For this object 5 slices are taken and the object is reconstructed. The center frequencies of RF are shown in Figure 3.43. The reconstructed object is shown in Figure 3.44.

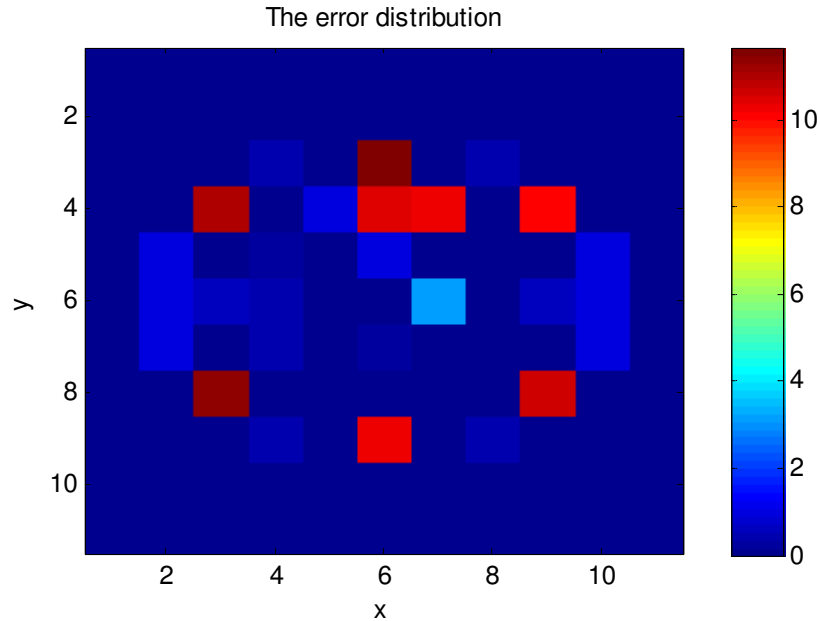


Figure 3.42: The error distribution of the Shepp-Logan image

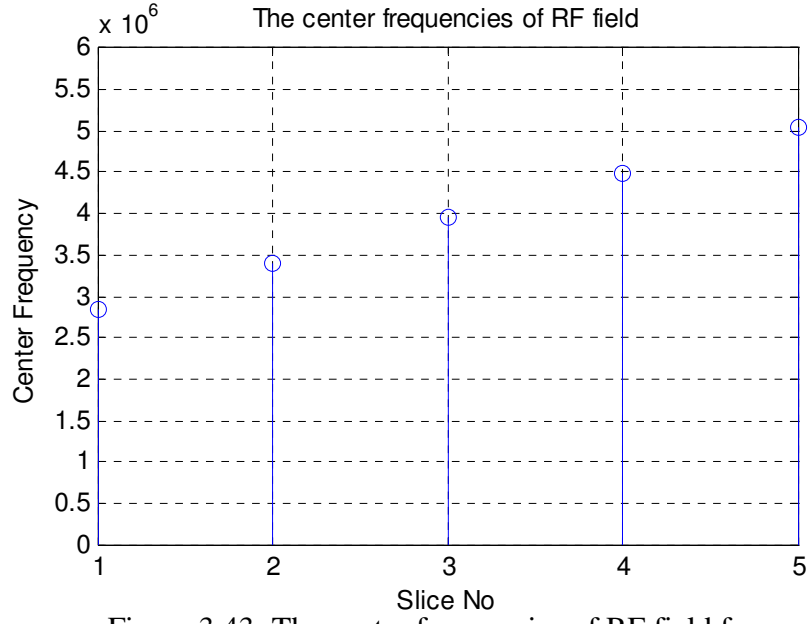


Figure 3.43: The center frequencies of RF field for a bandwidth of 548.16 kHz

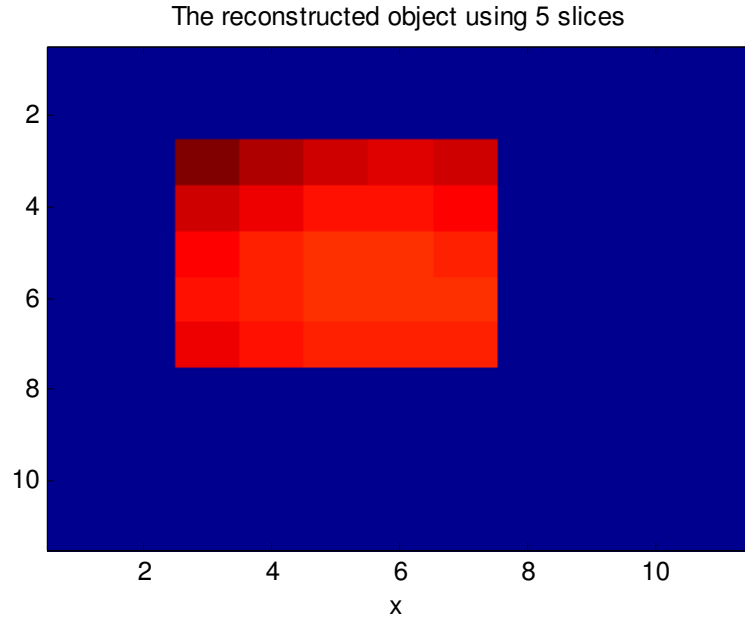


Figure 3.44: The object reconstructed by using 5 slices

When the image size is increased from 11 *mm* to 32 *mm*, the $\|\vec{B}_0\|$, $\|\vec{B}_{PE}\|$ and $\|\vec{B}_{FRQ}\|$ distributions are shown in Figures 3.45, 3.46 and 3.47, respectively.

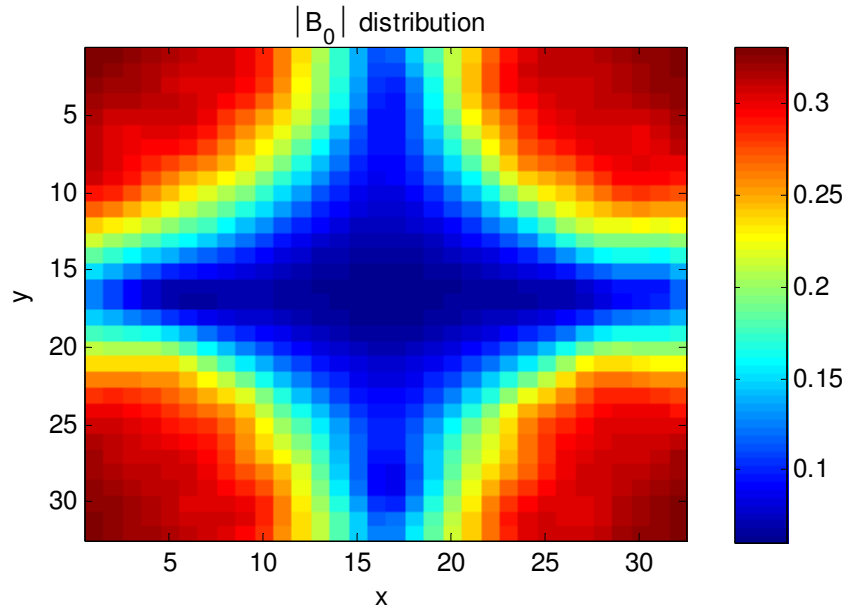


Figure 3.45: $\|\vec{B}_0\|$ distribution when the object is 32x32 mm^2

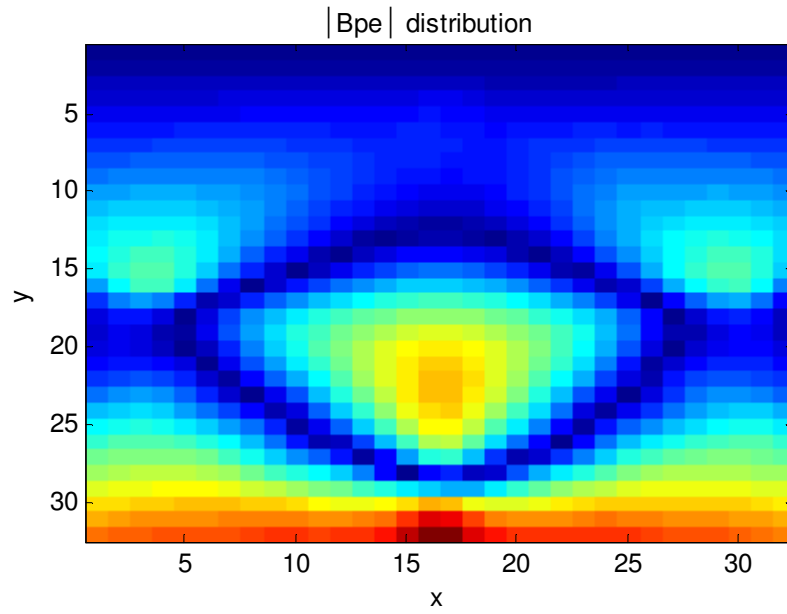


Figure 3.46: $\|\vec{B}_{PE}\|$ distribution when the object is 32x32 mm^2

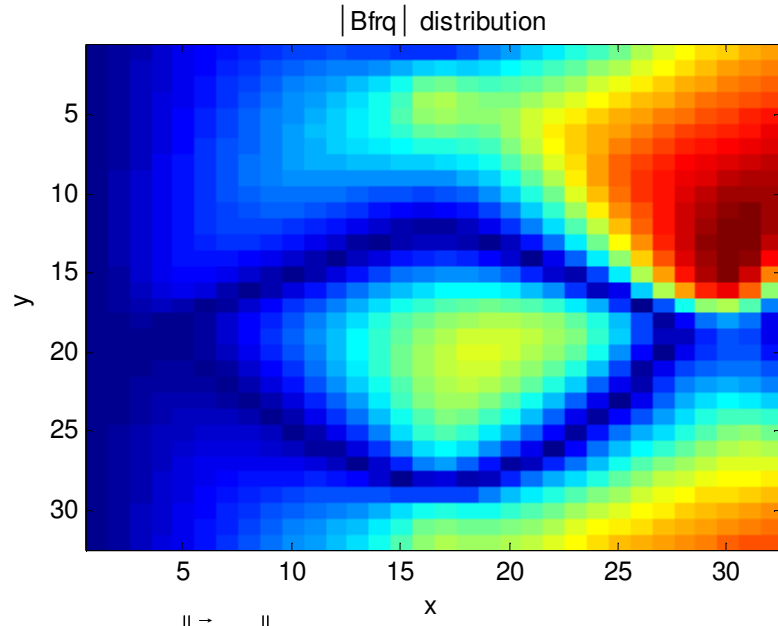


Figure 3.47: $\|\vec{B}_{FRQ}\|$ distribution when the object is $32 \times 32 \text{ mm}^2$

The $\|\vec{M}^0\|$ distribution and the reconstructed image are shown in Figures 3.48 and 3.49.

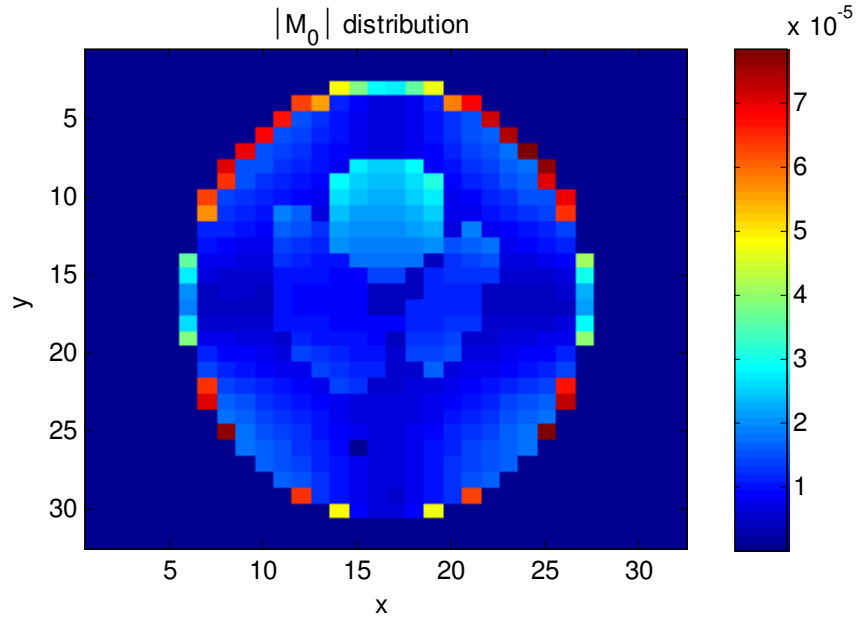


Figure 3.48: $\|\vec{M}^0\|$ distribution when the object is $32 \times 32 \text{ mm}^2$

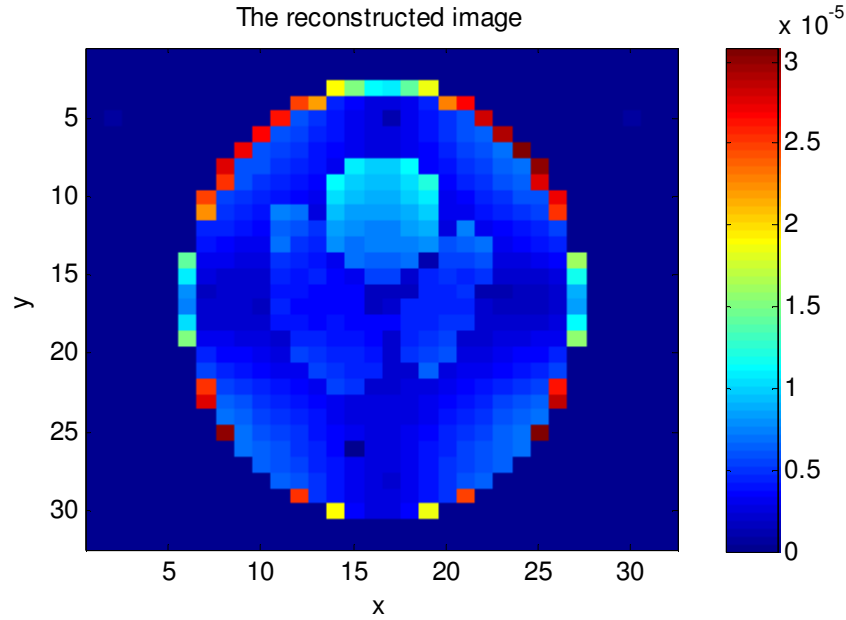


Figure 3.48: The reconstructed image of $32 \times 32 \text{ mm}^2$ object

As shown in the Figures 3.46 and 3.47, the gradient magnetic field distributions do not vary uniformly throughout the object. The classical MRI model requires uniform distribution of gradient fields to be able to reconstruct images. This requirement arises from the fact that uniform distribution of gradient fields guarantee linearly independent measurements and uniform distribution of frequency content of the acquired data. However, since the developed technique reconstructs images of initial magnetization in a least square sense, it is able to calculate initial magnetization with small errors. The error distribution of the reconstructed image at non-zero initial magnetization pixels (calculated using equation (3.125)) is shown in Figure 3.49. The error distribution at zero initial magnetization pixels (calculated using equation (3.124)) is shown in Figure 3.50. The error distribution shown in Figure 3.50 is scaled so that the maximum error is unity.

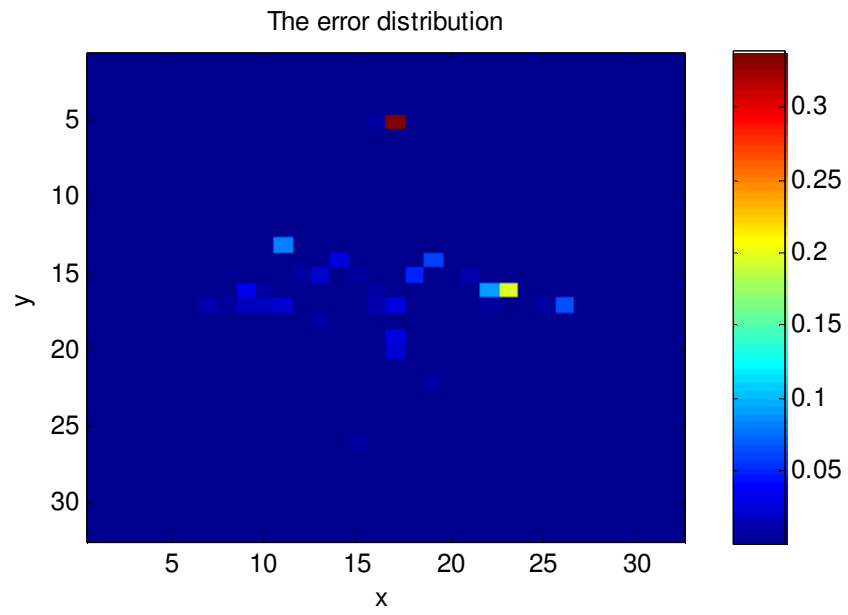


Figure 3.49: The error distribution at non-zero initial magnetization pixels of the reconstructed image of $32 \times 32 \text{ mm}^2$ object

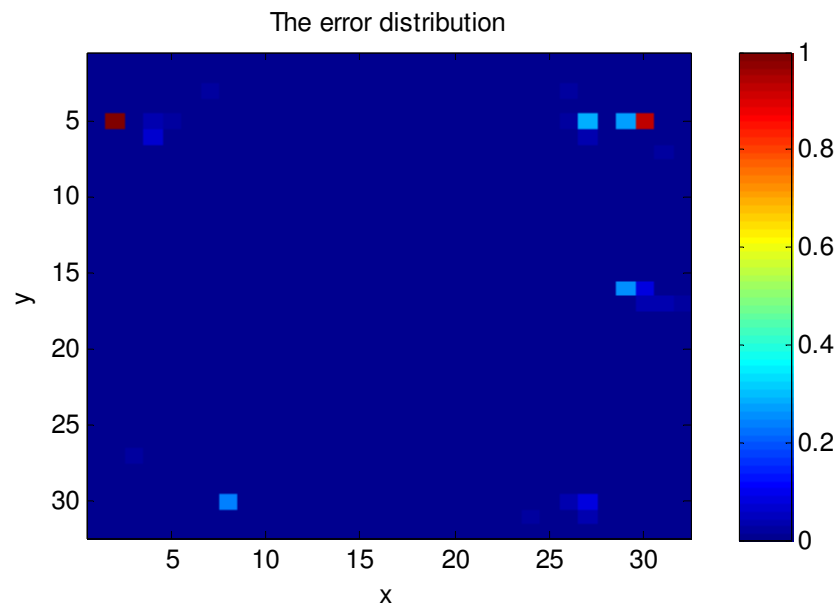


Figure 3.50: The scaled error distribution at zero initial magnetization pixels of the reconstructed image of $32 \times 32 \text{ mm}^2$ object

The obtained results are satisfactory. They signify that a new imaging modality has been developed for MRI in inhomogeneous magnetic fields. The performance of the technique is good enough to reconstruct images from measurements with 10 dB SNR. Moreover, by making use of the inhomogeneity of main magnetic field it allows to reconstruct 3D images using two gradient fields.

3.5. Conclusion

In this part of the work, an MRI simulator in inhomogeneous magnetic fields is constructed and an image reconstruction technique for MRI in inhomogeneous magnetic fields is developed. For the construction of the simulator, the conventional MRI formulation, which has been developed for homogeneous magnetic fields, is extended to inhomogeneous magnetic fields by defining a novel local reference frame. To be able to reconstruct images under the influence of inhomogeneous magnetic fields, a new model of MRI in inhomogeneous magnetic fields is developed. A methodology for reconstructing images of the physical parameters, making use of the developed model, is introduced. The obtained results have been presented as a proof of the concept.

Magnetic field distributions are generally inhomogeneous. Consequently, homogeneous magnetic fields are just a special case of inhomogeneous magnetic fields. Using this fact, the constructed simulator has been verified first by using homogeneous magnetic fields. The images of the objects are reconstructed by the conventional image reconstruction technique using the measurements obtained from the simulator. This experiment has been accepted as a proof of correct operation of the simulator. The same magnetic field distributions have been used in the developed model of MRI. The obtained results of the model along with the signals obtained from the simulator have also been used to reconstruct images using the developed image reconstruction technique. The images reconstructed using both the conventional and the developed techniques have been compared to investigate the correctness of both the model developed and the proposed image reconstruction

technique. Moreover, performances of the developed image reconstruction technique and the conventional technique have been compared for noisy measurement data. The obtained results demonstrate that the developed image reconstruction technique and MRI model perform well enough to be used in MRI in inhomogeneous magnetic fields.

Although the simulator and the MRI model make use of different approaches, if the initial magnetization is known by the MRI model, both the simulator and the MRI model would generate the same magnetization distribution at the same time instant of pulse sequence. The simulator and the MRI model are cross checked using this fact, during the tests of inhomogeneous magnetic fields distribution.

The same objects used in the homogeneous magnetic fields case have been used for tests in the inhomogeneous magnetic fields case. The reconstructed images and the error distribution throughout the image are exhibited to demonstrate the capabilities of the developed image reconstruction technique. Moreover, performance of the reconstruction algorithm is tested for noisy measurements. The obtained results show that the technique can successfully reconstruct images under this condition.

The computation cost of the MRI model is very high to be used in nearly real time applications. However, it is possible to save the relation between initial magnetization and measurements for a fixed pulse sequence with fixed parameters to be used with measurements obtained for different objects.

To sum up, the construction of an MRI simulator in inhomogeneous magnetic fields and the development of image reconstruction technique for MRI in inhomogeneous magnetic fields have been accomplished. The developed technique and the constructed simulator are both novel.

3.6. References

- [1]. F. Bloch, “Nuclear Induction”, Physics Review, vol. 70, pp. 460-474, 1946.
- [2]. P.C. Lauterbour, “Image Formation by Induced Local Interactions: Examples Employing Nuclear Magnetic Resonance”, Nature, vol. 242, pp. 190-191, 1973.
- [3]. Wikipedia, “Magnetic Resonance Imaging”, <http://en.wikipedia.org/wiki/MRI>, 10.08.2006.
- [4]. Z. P. Liang and P. C. Lauterbur, “Principles of magnetic resonance imaging: A signal processing perspective”, SPIE Optical Engineering Press, 2000.
- [5]. H. Chang, “A technique for Accurate Magnetic Resonance Imaging in the Presence of Field Inhomogeneities”, IEEE Trans. on Medical Imaging, vol. 11, no 3, pp. 319-329, 1992.
- [6]. Y. M. Kadah and X. Hu, “Algebraic Reconstruction for Magnetic Resonance Imaging under B_0 Inhomogeneity”, IEEE Trans. on Medical Imaging, vol. 17, no 3, pp. 362-370, 1998.
- [7]. Y. Luo, R.A. de Graaf, L. DelaBarre, A. Tannus, and M. Gardwood, “BISTRO: An Outer-Volume Suppression Method That Tolerates RF Field Inhomogeneity”, Magnetic Resonance in Medicine, vol. 45, pp. 1095-1102, 2001.
- [8]. F. Lin, Y. Chen, J. W. Belliveau, and L. L. Wald, “Removing Signal Intensity Inhomogeneity from Surface Coil MRI Using Discrete Wavelet Transform and Wavelet Packet”, in Proceedings of the Conference on Engineering in Medicine and Biology Society, vol. 3, pp. 2793- 2796, 2001.
- [9]. E. A. Vokurka, N. A Thacker, and A. Jackson, “A fast model independent method for automatic correction of intensity nonuniformity in MRI data”, Journal of Magnetic Resonance Imaging, vol. 10, pp. 550-562, 1999.
- [10]. A. A. Samoilenko, D.Y. Artemov, and L.A. Sibeldina, “Formation of Sensitive Layer in Experiments on NMR Subsurface Imaging of Solids”, JEPT Letter, vol. 47, pp. 417-419, 1988.
- [11]. J. B. Miller and A. N. Garroway, “Planar Imaging by NMR”, 35th Experimental NMR Conference, pp. 186, 1994.

- [12]. G. Eidmann, R. Savelsberg, P. Blümmler and B. Blümich, "The NMR MOUSE, a Mobile Universal Surface Explorer," *Journal of Magnetic Resonance*, vol. 122, pp. 104-109, 1996.
- [13]. B. Blümich, P. Blümmler, G. Eidmann, A. Guthausen, R. Haken, U. Schmitz, K. Saito and G. Zimmer, "The NMR-mouse: Construction, Excitation, and Applications," *Magnetic Resonance Imaging*, vol. 16, pp. 479-484, 1998.
- [14]. S. Mailboom and D. Gill, "Modified Spin-Echo Method for Measuring Nuclear Relaxation Times", *Review of Scientific Instruments*, vol. 29, pp. 681-691, 1958.
- [15]. R. Haken and B. Blümich, "Anisotropy in Tendon Investigated in Vivo by a Portable NMR Scanner, the NMR-MOUSE," *Journal of Magnetic Resonance*, vol. 144, no. 2, pp. 195-199, 2000.
- [16]. P. J. Prado, B. Blümich, and U. Schmitz, "One-Dimensional Imaging with a Palm-Size Probe", *Journal of Magnetic Resonance*, vol. 144, pp. 200–206, 2000.
- [17]. F. Casanova and B. Blümich, "Two-dimensional imaging with a single-sided NMR probe" , *Journal of Magnetic Resonance*, vol. 163, no. 1, pp. 38-45, 2003,.
- [18]. J. Perlo, F. Casanova and B. Blümich, "3D imaging with a single-sided sensor: an open tomography", *Journal of Magnetic Resonance*, vol. 166, no. 2, pp. 228-235, 2004.
- [19]. C. Eppstein, "Magnetic resonance imaging in inhomogeneous fields", *Inverse Problems*, vol. 20, pp. 753-780, 2004.
- [20]. D. A. Thayer, "Imaging Techniques and Hardware for Inhomogeneous MRI", MS Thesis, Brigham Young Univ. Dept. of Elec. and Comp. Eng., 2004.
- [21]. P. P. Hsu and J. H. Chen, "A Simulation of MR Imaging Sequences Using the Numerical Solutions of Bloch Equations", *Proceedings of the 16th Annual International Conference of the IEEE Engineering in Medicine and Biology Society. Part. 1*, Baltimore, pp. 574-575, 1994.
- [22]. D. A. Yoder, Y. Zhao, C. B. Paschal, and J. M. Fitzpatrick, "MRI simulator with object-specific field map calculations", *Magnetic Resonance Imaging*, vol. 22, no. 3, pp. 315-328, 2004.
- [23]. A. R. Brenner, J. Kürsch, and T. G. Noll, "Distributed Large-Scale simulation of MR imaging", *MAGMA (Magnetic Resonance Materials in Physics, Biology, and Medicine)*, vol. 4, no. 2, pp. 291-292, 1996.

- [24]. H. B. Cattin, F. Bellet, J. Montagnat, and C. Odet, "Magnetic resonance imaging (MRI) simulation on a grid computing architecture", *Cluster Computing and the Grid*, pp. 582–587, 2003.
- [25]. C. Cai, Z. Chen, S. Cai, and J. Zhong, "A simulation algorithm based on Bloch equations and product operator matrix: application to dipolar and scalar couplings", *Journal of Magnetic Resonance*, In Press, 2004.
- [26]. Y. Zur, "An algorithm to calculate the NMR signal of a multi spin-echo sequence with relaxation and spin-diffusion", *Journal of Magnetic Resonance*, vol. 171, no. 1, pp. 97-106, 2004.
- [27]. W. B. Blanton, "BlochLib: a fast NMR C++ tool kit", *Journal of Magnetic Resonance*, vol. 162, pp. 269–283, 2003.
- [28]. J. Bittoun, J. Taquin, and M. Sauzade, "A computer algorithm for the simulation of any nuclear magnetic resonance (NMR) imaging method", *Magnetic Resonance Imaging*, vol. 3, pp. 363-376, 1984.
- [29]. R. M. Summers, L. Axel, and S. Israel, "A computer simulation of nuclear magnetic resonance imaging", *Magnetic Resonance in Medicine*, vol. 3, pp. 363-376, 1986.
- [30]. M. B. E. Olsson, R. Wirestam, and B. R. R. Persson, "A computer simulation program for MR imaging: Application to RF and static magnetic field imperfections", *Magnetic Resonance in Medicine*, vol. 34, pp. 612-617, 1995.
- [31]. K. S. Kwan, A. C. Evans, and G. B. Pike, "MRI Simulation-Based Evaluation of Image-Processing and Classification Methods", *IEEE Transactions on Medical Imaging*, vol. 18, no. 11, pp. 1085-1097, 1999.
- [32]. H. Benoit-Cattin, G. Collewet, B. Belaroussi, H. Saint-Jalmes and C. Odet, "The SIMRI Project: a Versatile and Interactive MRI Simulator", *Journal of Magnetic Resonance*, vol. 173, pp. 97–115, 2005.
- [33]. W. H. Press, S. A. Teukolsky, W. T. Vetterling and B. P. Flannery, "Numerical Recipes in C: The Art Scientific Computing", 2nd edition, Cambridge University Press, USA, 1992.
- [34] F. Balibanu, Kidist Hailu, R. Eymael, D. E. Demco, and B. Blumich, "Nuclear Magnetic Resonance in Inhomogeneous Magnetic Fields", *Journal of Magnetic Resonance*, vol. 145, pp. 246-258, 2000.

CHAPTER 4

CONCLUSION

In this thesis, design of a permanent magnet, construction of an MRI simulator and development of an image reconstruction technique in inhomogeneous magnetic fields have been studied.

The design of a permanent magnet has been considered as a problem of permanent magnet surface optimization and optimization of placement of more than one magnet to obtain a magnetic field distribution with the desired properties. For this purpose, real coded and binary genetic algorithms and an optimization routine which does not require gradient or Hessian of fitness function have been utilized. These algorithms have been combined to form a method of permanent magnet design to obtain a desired magnetic field distribution. For each part of the problem, variables of fitness function that changes the magnetic field distribution have been determined and optimized using the methods developed during this work. The unit magnets in the “placement of more than one magnet” part of the problem have the resultant surface of the “surface optimization” part.

The task of designing permanent magnet structures generating a magnetic field with the desired properties in a region of interest has been accomplished. The resultant magnetic field distribution has been used in the tests of both the MRI simulator and the image reconstruction technique in inhomogeneous magnetic fields.

The construction of the simulator is based upon the extension of conventional MRI formulation to inhomogeneous magnetic fields by defining a novel local reference

frame. To be able to reconstruct images under the influence of inhomogeneous magnetic fields, a new model of MRI in inhomogeneous magnetic fields has been developed. A methodology for reconstructing images of the physical parameters, employing of the developed model, is introduced.

The constructed simulator has been verified by using homogeneous magnetic fields. The measurements obtained from the simulator are used to reconstruct images of the objects using the conventional image reconstruction technique. Ability of the conventional image reconstruction technique to reconstruct images from the outputs of the simulator has been accepted as a proof of correct operation of the simulator. The same magnetic field distributions have been used in the developed model of MRI. The obtained results of the model along with the signals obtained from the simulator have also been used to reconstruct images using the developed image reconstruction technique. Capabilities of the developed image reconstruction technique in inhomogeneous magnetic fields have been demonstrated. Moreover, performance of the reconstruction algorithm has been tested under noisy measurements.

As future work, one may use the result of the first part of permanent magnet design as reference and try to optimize geometries on the surface which are in elliptic cylinders shape. This kind of optimization is predicted to converge very fast. For the second part, one can change the fitness function to obtain different unit magnet arrangements using the method developed.

In the future, 3D image reconstruction in inhomogeneous magnetic fields should be tested. The simulated MRI system is assumed to have two gradient coils which is a drawback for MRI functioning in inhomogeneous magnetic fields. Since the gradient in the main magnetic field is an asset for linearly independent measurements, the main magnetic field can be varied by rearranging its sources to obtain linearly independent measurements. In the future, rearrangement of sources of the main magnetic field instead of the gradient fields is planned to be implemented. On the other hand, all of the tests are performed for the RF field which is orthogonal to the main magnetic field at all voxels of the object. Consequently, a proper RF coil

should be designed so that its magnetic field distribution is orthogonal to the main magnetic field.

PUBLICATIONS

Publications Prior to M.Sc. Study

- [1] **H.Yigitler**, A.O. Yilmaz and B. M. Eyuboglu, “An Approach to geometrical design of Permanent Magnets for Biomedical Applications”, Biomed2004, 11th International Biomedical Science and Technology Days, Ankara- Turkey, p.24, 2004.

Publications During M.Sc. Study

- [2] **H.Yigitler** and B. M. Eyuboglu, “Homogen Olmayan Manyetik Alanlarda Manyetik Rezonans Görüntüleme İçin Kalıcı Mıknatıs Tasarımı”, Proceedings of URSI-TÜRKİYE'2006 3rd National Congress, Ankara - Turkey, pp.183-185, 2006.
- [3] V.E. Arpınar, **H.Yigitler** and B. M. Eyuboglu, “Homogen Olmayan Manyetik Alanlarda Manyetik Rezonans Görüntüleme Sisteminin, Üç Boyutlu Uzayda Nümerik Modellemesi”, Proceedings of URSI-TÜRKİYE'2006 3rd National Congress, Ankara - Turkey, pp.186-188, 2006.