

## AN ABSTRACT OF THE THESIS OF

David Sergio Matusevich for the degree of Doctor of Philosophy in Physics  
presented on November 5, 2002.

Title: Magneto-crystalline Anisotropy Calculation in Thin Films with Defects

**Redacted for Privacy**

Abstract approved: \_\_\_\_\_

Henri J. F. Jansen

The code is developed for the calculation of the magneto-crystalline anisotropy (MAE) in thin films using a classical Heisenberg hamiltonian with a correction developed by Van Vleck. A Metropolis style Monte Carlo algorithm was used with adequate corrections to accelerate the calculation. The MAE was calculated for the case of a thin film with an increasing number of defects on the top layer for the cases where the defects were distributed randomly and when they assumed ordered positions. The results obtained agree qualitatively with the results provided by the literature and with the theory.

Magneto-crystalline Anisotropy Calculation in Thin Films with Defects

by

David Sergio Matusevich

A THESIS

submitted to

Oregon State University

in partial fulfillment of  
the requirements for the  
degree of


Doctor of Philosophy

Completed November 5, 2002  
Commencement June 2003


Doctor of Philosophy thesis of David Sergio Matusevich presented on  
November 5, 2002

APPROVED:


Redacted for Privacy

  
Major Professor, representing Physics

Redacted for Privacy

  
Chair of the Department of Physics

Redacted for Privacy

  
Dean of the Graduate School

I understand that my thesis will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my thesis to any reader upon request.

Redacted for Privacy

  
David Sergio Matusevich, Author

## ACKNOWLEDGMENT

I'd like to thank the people of Corvallis for their warm welcome and for the help they gave me to adapt to the American culture.

To my Major Professor, Dr Henri F. K. Jansen for his continuous encouragement and patience.

To my office mates Günter Schneider, Haiyan Wang, and Jung-Hwang Song, who helped me with all my computer problems.

And finally to my parents, who were pleasantly surprised the day I finished High School.

Financial support was provided by ONR under grant N00014-9410326

# TABLE OF CONTENTS

	<u>Page</u>
1 INTRODUCTION .....	1
1.1 Introduction .....	1
1.2 Motivation .....	2
2 MAGNETOCRYSTALLINE ANISOTROPY: AN OVERVIEW .....	3
2.1 Introduction .....	3
2.2 Phenomenology of Magnetic Anisotropy .....	6
2.3 Some theoretical results .....	9
2.3.1 The Ising Model .....	10
2.3.1.1 The 1-dimensional Ising Model .....	11
2.3.1.2 The 2-dimensional Problem: The Onsager Solution ....	12
2.3.2 The Classical Heisenberg Model .....	14
2.4 Van Vleck Approximation .....	16
2.5 Theoretical Results .....	17
3 SOME RESULTS IN $T = 0$ .....	20
3.1 The System .....	20
3.2 The $T = 0$ approximations .....	20
4 THE MONTE CARLO SIMULATION .....	28
4.1 Introduction to Sampling Monte Carlo Methods .....	28
4.1.1 The Relationship Between Experiment, Theory, and Simulation	29
4.1.2 The Monte Carlo Method and Markov Chains .....	31
4.1.3 The Metropolis Algorithm .....	32
4.2 Solving The Ising Model with Monte Carlo .....	33
4.2.1 The Algorithm .....	33
4.2.2 Boundary Conditions and Related Concerns .....	36

## TABLE OF CONTENTS (Continued)

	<u>Page</u>
4.3 Solving the Heisenberg Model with Monte Carlo.....	38
4.3.1 Changes Made to Speed the Program .....	40
4.3.2 Other Points of Interest .....	45
4.3.3 Changes Made to Accommodate Defects .....	47
 5 MONTE CARLO TESTS.....	 48
5.1 Testing Stability .....	48
5.2 How Reliable is it to Take Just One Example of Random Coverage? .....	50
5.3 Comparison to $T = 0$ Results.....	51
5.4 Critical Behavior.....	52
 6 RESULTS OBTAINED WITH THE CODE DEVELOPED.....	 55
6.1 Critical Behavior.....	55
6.2 Magneto-Crystalline Anisotropy Energy in the Case of a Complete Top Layer.....	55
6.3 Magneto-Crystalline Anisotropy Energy Calculations when the Top Surface is Incomplete.....	58
 7 DISCUSSION, CONCLUSIONS AND FUTURE DIRECTION OF DE- VELOPMENT.....	 68
7.1 Discussion .....	68
7.2 Conclusions .....	70
7.3 Expected Direction of Future Development. ....	70
7.3.1 Parallel Computations .....	70
7.3.2 2nd Neighbors and Other Terms in the Hamiltonian .....	71
7.3.3 Roughness .....	72

## TABLE OF CONTENTS (Continued)

	<u>Page</u>
7.3.4 Substitutions . . . . .	72
7.3.5 Other Geometries . . . . .	72
 BIBLIOGRAPHY . . . . .	 75
 APPENDIX . . . . .	 77
APPENDIX A Notes regarding critical exponents. . . . .	78
APPENDIX B Dipole-dipole coupling in the case of periodic boundary con- ditions . . . . .	82
APPENDIX C The Program and Subroutines . . . . .	83

# LIST OF FIGURES

<u>Figure</u>	<u>Page</u>
2.1 Integration area. . . . .	5
2.2 Schematic of a Toque Magnetometer. . . . .	8
2.3 Spontaneous magnetization in the two dimensional Ising model. . . . .	13
3.1 The system at $T = 0$ . . . . .	21
3.2 Magnetization in the direction of the field. . . . .	24
3.3 Energy versus $\theta_H$ . $a = 0.01 \times J$ . . . . .	25
3.4 $\theta_S$ versus $\theta_H$ . $a = 0.01 \times J$ . . . . .	26
3.5 Examples . . . . .	27
4.1 Schematic view of the relationship between theory, simulation and experiment. . . . .	30
4.2 Flow diagram of the Metropolis algorithm. . . . .	34
4.3 Monte Carlo results for the spontaneous magnetization in the two dimensional Ising model. . . . .	35
4.4 Three examples of boundary conditions: periodic, screw-periodic and open. . . . .	37
4.5 Three methods of generating random vectors:(a) Cartesian, (b) Spherical and (c) Mixed. . . . .	41
4.6 Total energy versus the azimuth angle of the applied field . . . . .	44
4.7 Limiting the possible deflections in the angle of the spin. . . . .	46
5.1 Energy vs. Time for different values of the limit $\ell$ . . . . .	49
5.2 Average time required for a single step relative to the limit $\ell$ . . . . .	50
5.3 Energy vs. angle of the applied field for $T = J/1000$ . . . . .	52
5.4 Magnetization in the direction of the field. . . . .	53
5.5 Magnetization vs. Temperature with no external field. . . . .	54



## LIST OF FIGURES (Continued)

<u>Figure</u>	<u>Page</u>
6.1 Critical behavior. ....	56
6.2 Zero field magnetization versus temperature for different values of the anisotropy parameter. ....	57
6.3 Magnetic anisotropy energy versus temperature (complete top layer). 58	
6.4 Low temperature behavior for the MAE (complete top layer). ....	59
6.5 Magnetic anisotropy energy versus remanent magnetization (com- plete top layer). ....	60
6.6 Behavior of the energy under changes in the random coverage of the top layer ( $T = 0.01$ ). ....	61
6.7 Behavior of the energy under changes in the random coverage of the top layer ( $T = 0.05$ ). ....	62
6.8 Behavior of the energy under changes in the random coverage of the top layer with a 20% coverage ( $T = 0.01$ ). ....	63
6.9 Magnetic anisotropy for the case where there is a single hole or spin on the top layer. ....	64
6.10 MAE vs. Temperature for different coverages of the top layer. ....	65
6.11 MAE as a function of the temperature for different surface coverages 66	
6.12 MAE as a function of the temperature for different random surface coverages ....	66
6.13 Magnetic anisotropy energy as a function of the missing lines. ....	67
6.14 Magnetic anisotropy energy as a function of the coverage of the top layer for different values of the anisotropy parameter. ....	67
7.1 Surface roughness as a function of the number of ridges and the area. 73	
7.2 From simple cubic crystals to FCC and BCC. ....	74

## LIST OF TABLES

<u>Table</u>		<u>Page</u>
2.1	Values of $K_1$ and $K_2$ for iron and nickel at room temperature. . . . .	7
2.2	Critical exponents for the Ising and Heisenberg models. . . . .	14
2.3	Critical exponents (Experimental results). . . . .	14
4.1	Magnitude of the different components of the Heisenberg Hamiltonian. . . . .	44

# MAGNETO-CRYSTALLINE ANISOTROPY CALCULATION IN THIN FILMS WITH DEFECTS

## 1. INTRODUCTION

### 1.1. Introduction

Magnetism is an electronically driven phenomenon that, although weak compared to electrostatic effects, is subtle in its manifestations. It's origins are quantum-mechanical and are based on the existence of the electronic spin and the Pauli exclusion principle. It leads to a number of short and long range forces, with both classical and quantum-mechanical effects, useful for a variety of engineering and technical applications. In particular with the growing need to store and retrieve information, the body of research in science and technology has experienced an explosive growth, and central to those pursuits is the study of magnetism as applied to surfaces, interfaces and, in particular, *thin films*.

Apart from the application driven pressures, we can mention three major advances in this field:

- The development of new sample preparation techniques (*Molecular Beam Epitaxy, Metal-Organic Chemical Vapor Deposition, sputtering, lithography, etc.*) which are increasingly less expensive and which now permit the manufacture of single purpose devices to very accurate specifications.

- The availability of better sample characterization techniques, based mostly on centrally located facilities. These techniques are based on x-ray, ultra-violet, visible and infra-red photons from synchrotron and laser sources, neutrons from reactors and electrons of a number of energies from electron microscopes and scattering experiments.
- The increasing availability of fast, operationally inexpensive and numerically intensive computers which have permitted the calculation of a large variety of problems related to realistic systems.

## 1.2. Motivation

Magnetocrystalline anisotropy of thin films has gained considerable attention because of their potential application as high-density recording media. In the past decade the areal density of magnetic recording media has doubled yearly in laboratory conditions, due to the development of high-sensitivity GMR heads and advanced recording media. As the density continues to increase, the thermal stability limit for the current copper alloys used in the commercial media will be reached in the near future and in order to continue this growth trend new metal alloys are being developed and a better understanding of the physics behind the anisotropy effects is required.

## 2. MAGNETOCRYSTALLINE ANISOTROPY: AN OVERVIEW

In this chapter we will present definitions and two popular models for magnetism (Ising and Heisenberg). We will also discuss modifications to accommodate anisotropy.

### 2.1. Introduction

**Definition 1 (Spontaneous Magnetization)** *We define spontaneous magnetization as the magnetization present in a sample when no external fields are applied.*

When a demagnetized ferromagnetic substance is put in an increasing magnetic field, its magnetization increases until it finally reaches *saturation magnetization*, when the apparent magnetization of the sample is equal to the spontaneous magnetization. Such process is called technical magnetization because it is essentially achieved by aligning the fields of the domains of the sample, and can be distinguished from spontaneous magnetization, because when the field is reduced, the magnetization also drops.

After the magnetization reaches the saturation value, it increases in value proportionally to the intensity of the applied field. This effect is disturbed by the thermal agitation of the ions and is very small even under moderately strong magnetic fields, except at temperatures just below the Curie Point.

In 1926 Honda and Kaya [9] measured magnetization curves for single crystals of iron and discovered that the shape of the magnetization curve is dependent on the direction of the applied field with respect to the crystallographic axis. In fact there are directions along which the magnetic saturation occurs at lower fields than in others. These directions are called the directions of easy magnetization or *easy*

*directions*. This means that the magnetization is stable when pointing in any of these directions. Conversely, there are orientations of the external field in which the saturation is harder, that is, we require a higher field to find saturation. The directions in which the saturation is the hardest are called the directions of hard magnetization or simply *hard directions*. The dependance of the internal energy on the direction of magnetization is called *magnetocrystalline anisotropy*.

**Definition 2 (MAE)** *We define the Magnetocrystalline Anisotropy Energy as the difference between the saturation energy of a sample when measured in the easy and hard directions:  $MAE = E^H|_{sat} - E^E|_{sat}$*

If we recall that  $\mathbf{m} = -\frac{\partial A}{\partial \mathbf{H}}$  where  $A$  is the Helmholtz free energy and  $\mathbf{H}$  is the external field, then if we choose the field in the direction of one of the easy axis,

$$A(H_{SAT})|_e - A(0)|_e = - \int_0^{H_{SAT}} m(H_e) dH_e$$

The same equation is valid for the case we take the field in the hard direction:

$$A(H_{SAT})|_h - A(0)|_h = - \int_0^{H_{SAT}} m(H_h) dH_h$$

If we subtract the last two equations, and we remember that  $A(0)|_e = A(0)|_h$  (as there is no field present) then

$$A(H_{SAT})_h - A(H_{SAT})_e = \int_0^{H_{SAT}} m(H_e) dH_e - \int_0^{H_{SAT}} m(H_h) dH_h$$

Finally we can gather both integrals into one if we define an integration surface  $S$  limited on top by the magnetization curve in the easy axis and on the

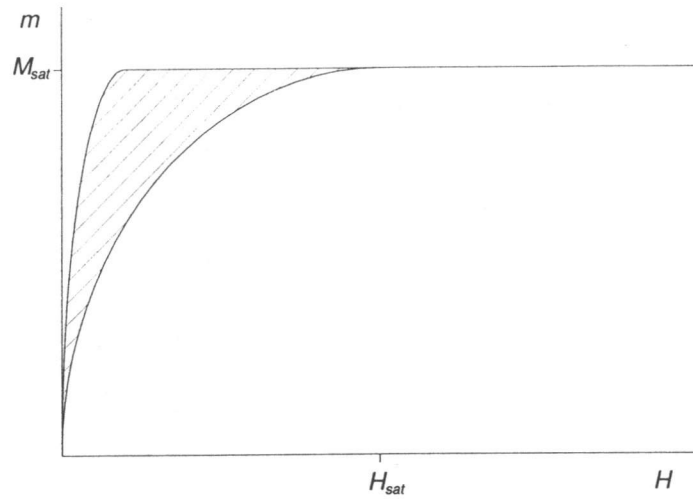


FIGURE 2.1. Integration area.

bottom by the magnetization curve for the hard axis and  $C$  the border of this surface:

$$MAE = \Delta A = \int_{\partial A} \mathbf{m}(\mathbf{H}) \cdot d\mathbf{H} \quad (2.1)$$

where  $\partial A$  is the path defined by the saturation curves.

The saturation field used in the integration is chosen to be the one with highest value, since theoretically the magnetization would be the same for both directions at that point. In practice this is not true and a compromise must be reached: We choose for this field of *technical saturation* the field when the magnetization in the easy and hard directions are experimentally indistinguishable.

## 2.2. Phenomenology of Magnetic Anisotropy

Magnetic Anisotropy is defined as the dependence of the internal energy on the direction of the spontaneous magnetization. The term in the internal energy that reflects this dependence is generally called the Magnetocrystalline Anisotropy Energy or  $E_a$ . Although this term is related to the MAE, it's not the same, as it depends on the orientation of the applied magnetic field. This term has the symmetry of the crystal. It can be affected by the application of heat or mechanical stresses to the crystal, but we are not going to deal with that in this work.

The anisotropy energy can be expressed in terms of the direction cosines ( $\alpha_1, \alpha_2, \alpha_3$ ) of the internal magnetization with respect to the three cube edges. In cubic crystals like iron and nickel, due to the high symmetry of the crystal the energy  $E_a$  can be expressed in a fairly simple way: Suppose that we expand this term in a polynomial series in  $\alpha_1, \alpha_2$  and  $\alpha_3$ . Terms that include odd powers of the cosines must vanish by symmetry. Also the expression must be invariant to the exchange of any 2  $\alpha_i$ , so terms of the form  $\alpha_1^i \alpha_2^j \alpha_3^k$  must have the same coefficient, for any permutation of  $i, j$  and  $k$ . The first term ( $\alpha_1^2 + \alpha_2^2 + \alpha_3^2$ ) is always equal to 1. The next terms (of order 4 and 6) can be manipulated to give:

$$E_a = K_1(\alpha_1^2 \alpha_2^2 + \alpha_2^2 \alpha_3^2 + \alpha_3^2 \alpha_1^2) + K_2 \alpha_1^2 \alpha_2^2 \alpha_3^2 + \dots \quad (2.2)$$

$K_1$  and  $K_2$  are the anisotropy constants.

It is interesting to note that in the case of a cubic lattice, a  $K_1 > 0$  means that the first term of 2.2 is minimized at the  $[100]$  direction, while if  $K_1 < 0$  it does so at  $[111]$ .



	Iron	Nickel
$K_1$	$4.8 \times 10^4 J/m^3$	$-4.5 \times 10^4 J/m^3$
$K_2$	$5 \times 10^3 J/m^3$	$2.34 \times 10^3 J/m^3$

TABLE 2.1. Values of  $K_1$  and  $K_2$  for iron and nickel at room temperature.

### Measuring the Magnetic Anisotropy: The Torque Magnetometer.

The most common apparatus for measuring the MAE is the *torque magnetometer*. Basically it consists of an elastic string used to suspend a sample of the material between the poles of a rotatable magnet. When a magnetic field is applied to the specimen, it tends to rotate to align its internal magnetization with the external field, and the torque can be measured if the elastic constant of the string is known. If we rotate the magnet, the torque can be measured as a function of the crystallographic directions of the sample. This is called the *magnetic torque curve*, from which we can deduce the magnetic anisotropy energy.

The torque exerted by a unit volume of the specimen is

$$\tau_\phi = -\frac{\partial E_a}{\partial \phi} \quad (2.3)$$

where  $\phi$  is the angle of deflection of the sample due to the internal magnetization measured in the plane of the specimen. If we confine the magnetization to the (001) plane in a cubic crystal, we can write  $\alpha_1 = \cos \phi$ ,  $\alpha_2 = \sin \phi$  and  $\alpha_3 = 0$ . Then the first term of 2.2 becomes:

$$E_a = \frac{1}{4} K_1 \sin^2 2\phi \quad (2.4)$$

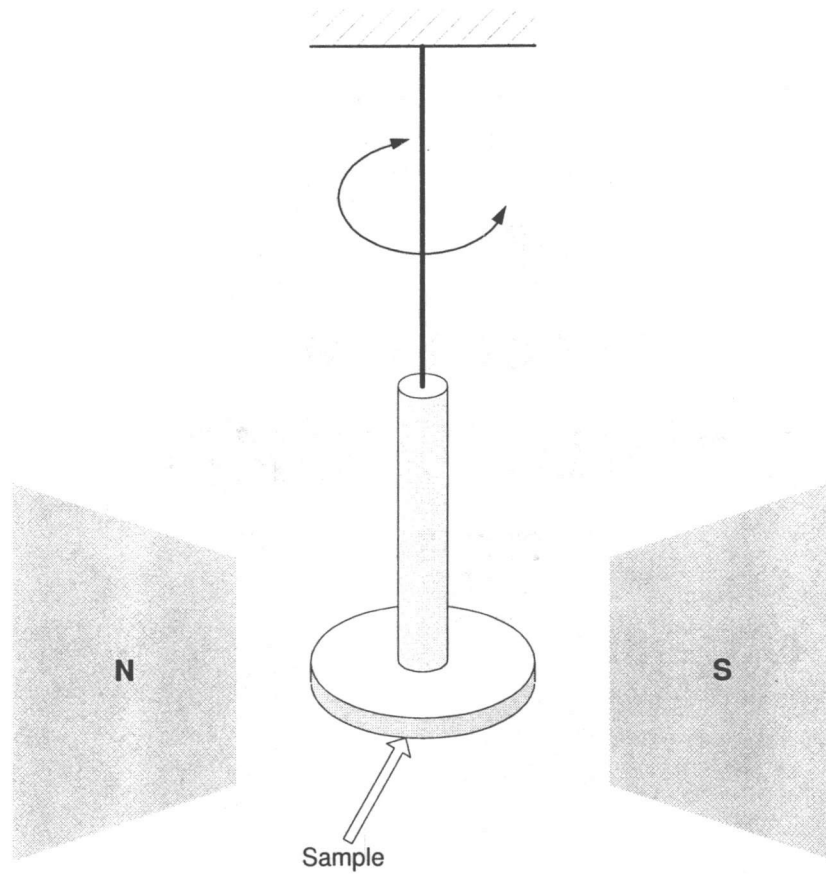


FIGURE 2.2. Schematic of a Torque Magnetometer.

From 2.3 we have:

$$\tau_{\phi} = -\frac{1}{2}K_1 \sin 4\phi \quad (2.5)$$

By comparing the results of the experiment with expression 2.5, we can find a value for  $K_1$ . Higher order terms are obtained by means of a Fourier analysis. Magnetic anisotropy can also be measured by means of ferromagnetic resonance. The resonance frequency depends on the external magnetic field, which exerts a torque on the precessing spin system. Since the magnetic anisotropy also causes a torque on the spin system if it points in a direction other than the easy directions,

the resonance frequency is expected to depend on the anisotropy. For cubic anisotropy and in the case the magnetization is nearly parallel to the easy direction, the anisotropy 2.2 can be expressed by:

$$\begin{aligned} E_a &= K_1[\theta^4 \sin^2 \phi \cos^2 \phi + (1 - \frac{1}{2}\theta^2)\theta^2] \\ &\approx K_1\theta^2 \end{aligned} \quad (2.6)$$

This is equivalent to the presence of a magnetic field  $H_a$  (*anisotropy field*) parallel to the easy direction. If  $M_s$  is the magnetization,  $E = -M_s H_a \cos \theta = A + \frac{1}{2}M_s H_a \theta^2$  then

$$H_a = \frac{2K_1}{M_s} \quad (2.7)$$

for the  $\langle 100 \rangle$  directions. A similar calculation yields that when the magnetization is near parallel to the  $\langle 111 \rangle$  directions,

$$H_a = \frac{4K_1}{3M_s} \quad (2.8)$$

Therefore if the field is rotated from  $\langle 100 \rangle$  to  $\langle 111 \rangle$ , the shift of the resonance is changed by

$$\Delta H = \frac{10K_1}{3M_s} \quad (2.9)$$

If we find the dependence of the resonance field on the crystallographic direction, we can easily estimate the value of  $K_1$ . This method has the advantage of not only enabling us to measure the anisotropy of very small samples, but also offers information on the magnitude of the local anisotropy.

### 2.3. Some theoretical results

In this section we will present the most popular of the models for magnetism, the *Ising model* [8], the extension made by *Heisenberg*, and the *Van Vleck* approximation to anisotropy.

### 2.3.1. The Ising Model

The Ising model is an attempt to simulate the structure of a *magnetic domain*<sup>1</sup> in a ferromagnetic substance. It's main virtue lies in the fact that a 2-dimensional Ising model yields the only non-trivial example of a phase transition that can be worked out with mathematical rigor.

The system to be considered is an array of  $N$  fixed points (*lattice sites*) that form an  $n$ -dimensional periodic lattice. The geometrical structure is not important as the model works for all of them. Associated with each site is spin variable  $s_i$ , ( $i = 1 \cdots N$ ) which can only have values of  $+1$  or  $-1$ . If a spin variable has a positive value, it is said to have (*spin up*) and if it is negative, (*spin down*). For a given set of numbers  $\{s_i\}$  the energy of the system is defined to be;

$$E_I\{s_i\} = - \sum_{i,j} J_{i,j} s_i s_j - H \sum_i^N s_i$$

where the interaction energies  $J_{i,j}$  and the external magnetic field  $\mathbf{H}$  are given constants.

For simplicity we will only sum over first neighbors in the first sums, and we will specialize to the case of isotropic interaction so all the interaction energies are equal to  $J$ . Thus the energy 2.3.1 is simplified to

$$E_I\{s_i\} = -J \sum_{\langle i,j \rangle} s_i s_j - H \sum_i^N s_i \quad (2.10)$$

If  $J > 0$  then the model describes ferromagnetism and when  $J < 0$ , antiferromagnetism. In the future we will only consider  $J > 0$ .

---

<sup>1</sup>A magnetic domain is a group of atoms or molecules that act as a unit. The magnetization of a domain is the sum of the magnetic moments of each of its components all of which are pointing in the same direction.

### 2.3.1.1. The 1-dimensional Ising Model

In the one-dimensional problem we have a chain of  $N$  spins, each interacting only with its two nearest neighbors and the external field. The energy is then reduced to:

$$E_I = -J \sum_i s_i s_{i+1} - H \sum_i s_i \quad (2.11)$$

If we impose periodic boundary conditions, we transform our chain into a ring. The partition function of this system is;

$$Q_I(H, T) = \sum_{s_1} \sum_{s_2} \cdots \sum_{s_N} \exp \left[ \beta \sum_{i=1}^n (J s_i s_{i+1} + H s_k) \right] \quad (2.12)$$

Following Kramers and Wannier [11], we can express 2.12 in terms of matrices:

$$Q_I(H, T) = \sum_{s_1} \sum_{s_2} \cdots \sum_{s_N} \exp \left\{ \beta \sum_{i=1}^n \left[ J s_i s_{i+1} + \frac{1}{2} H (s_k + s_{k+1}) \right] \right\} \quad (2.13)$$

We define a  $2 \times 2$  *transfer matrix*  $\mathbf{P}$  such that its matrix elements are defined by:

$$\langle s | \mathbf{P} | s' \rangle = e^{\beta [J s s' + \frac{1}{2} H (s + s')]}$$

and therefore an explicit representation of  $\mathbf{P}$  is:

$$\mathbf{P} = \begin{pmatrix} e^{\beta(J+H)} & e^{-\beta J} \\ e^{-\beta J} & e^{\beta(J-H)} \end{pmatrix} \quad (2.14)$$

With these definitions and a bit of operator algebra we arrive to:

$$Q_I(H, T) = \sum_{s_1} \langle s_1 | \mathbf{P}^N | s_1 \rangle = \text{Tr} \mathbf{P}^N = \lambda_1^N + \lambda_2^N \quad (2.15)$$

where  $\lambda_i$  are the eigenvalues of  $\mathbf{P}$  with  $\lambda_1 \geq \lambda_2$ . Using simple algebra we arrive at the solution for the two eigenvalues:

$$\lambda_i = e^{\beta J} \left[ \cosh(\beta H) \pm \sqrt{\sinh^2(\beta H) + e^{-4\beta J}} \right] \quad (2.16)$$

The Helmholtz free energy per spin is then given by

$$\frac{1}{N}A_i(H, T) = -J - kT \log \left[ \cosh(\beta H) + \sqrt{\sinh^2(\beta H) + e^{-4\beta J}} \right] \quad (2.17)$$

and the magnetization per spin by

$$\frac{1}{N}M_I(H, T) = \frac{\sinh(\beta H)}{\sqrt{\sinh^2(\beta H) + e^{-4\beta J}}} \quad (2.18)$$

Note that for all  $T > 0$   $M_I(0, T) = 0$  and there is no phase transition. This result led Ising to believe that this model had no practical application, as it is known that magnetic materials have a remanent magnetization at  $M = 0$  when the temperature is below the *Curie Temperature*

### 2.3.1.2. The 2-dimensional Problem: The Onsager Solution

In 1944 Onsager solved the two dimensional problem, in the absence of magnetic fields. The complete derivation is too long to include in this work but he concluded that the Helmholtz free energy per spin is:

$$\beta a_I(0, T) = -\log(2 \cosh 2\beta J) - \frac{1}{2\pi} \int_0^\pi d\phi \log \frac{1}{2} \left( 1 + \sqrt{1 - \kappa^2 \sin^2 \phi} \right) \quad (2.19)$$

where  $\kappa = 2/\cosh 2\phi \coth 2\phi$ .

If we define the Curie Temperature  $T_C$  such that

$$2 \tanh^2 \frac{2J}{kT_C} = 1$$

then  $kT_C = (2.269185)J$ . At this temperature all the thermodynamic functions that are calculated from 2.19 have a singularity of some kind. If we examine the spontaneous magnetization, calculated as the derivative of the free energy with

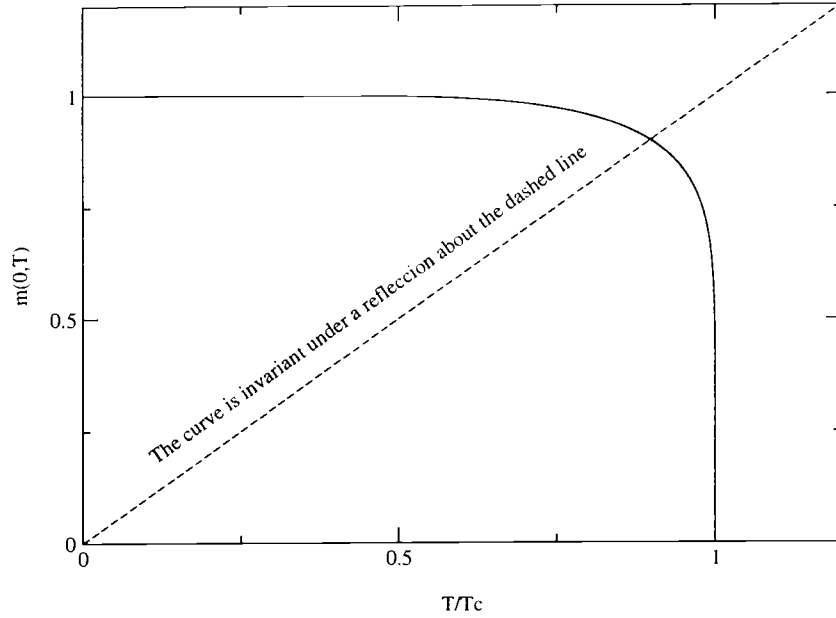


FIGURE 2.3. Spontaneous magnetization in the two dimensional Ising model.

respect to  $H$  at  $H = 0$  [15]:

$$m_i(0, T) = \begin{cases} 0 & \text{if } T > T_C, \\ \{1 - [\sinh(2\beta J)]^{-4}\}^{\frac{1}{8}} & \text{if } T < T_C. \end{cases} \quad (2.20)$$

where  $\beta = \frac{1}{kT}$

This shows that at  $T = T_C$  there is indeed a phase transition: As  $T \rightarrow T_C$ ,  $1 - [\sinh(2\beta J)]^{-4} \rightarrow 0$  and  $m_I \rightarrow 0$  as  $[\beta - \beta_C]^{\frac{1}{8}}$ . So we conclude that the critical exponent  $\beta = \frac{1}{8}$ . Other critical exponents for this model, as well as for the 3 dimensional model(not yet solved analytically) and for the Heisenberg model (2.3.2) can be found in table 2.2

In table 2.2 we present the values for the 6 critical exponents calculated using the 2 models discussed. The Ising model in 3D was solved using Monte Carlo [22] and the Heisenberg model, using a high-T expansion.

	Related Variables [5]	Ising ( $d = 2$ )	Ising ( $d = 3$ ) [22]	Heisenberg ( $d = 3$ )
$\alpha$	$C(T)$ specific heat	0 (log)	$0.119 \pm .006$	$-0.08 \pm .04$
$\beta$	$M(T)$ magnetization	1/8	$0.326 \pm .004$	$0.38 \pm .03$
$\gamma$	$\chi(T)$ susceptibility	7/4	$1.239 \pm .003$	$1.38 \pm .02$
$\delta$	$B(M)$ external field	15	$4.8 \pm .05$	$4.63 \pm .29$
$\eta$	$G_c^{(2)}(R)$ correlation function	1/4	$0.024 \pm .007$	$0.07 \pm .06$
$\nu$	$\xi(T)$ correlation length	1	$0.627 \pm .002$	$0.715 \pm .02$

TABLE 2.2. Critical exponents for the Ising and Heisenberg models.

	$^4\text{He}$ (2D)	Fe	Ni
$\alpha$	$-0.014 \pm .016$	$-0.03 \pm .12$	$-0.04 \pm .12$
$\beta$	$0.34 \pm .01$	$0.37 \pm .01$	$0.358 \pm .003$
$\gamma$	$1.33 \pm .03$	$1.33 \pm .015$	$1.33 \pm .02$
$\delta$	$3.95 \pm .15$	$4.3 \pm .1$	$4.29 \pm .05$
$\eta$	$0.021 \pm .05$	$0.07 \pm .04$	$0.041 \pm .01$
$\nu$	$0.672 \pm .001$	$0.69 \pm .02$	$0.64 \pm .1$

TABLE 2.3. Critical exponents (Experimental results).

### 2.3.2. The Classical Heisenberg Model

In the Ising model the magnetic dipoles can only point in two directions: up and down. But if they are allowed greater flexibility of orientations, we obtain qualitatively different results. In the classical Heisenberg model the dipoles can point in any direction. They are in fact 3 dimensional classical spins,  $\mathbf{s}_i$ . The energy for this model is the same as that of Ising, with the difference that now we have vector products instead of simple scalar products:



$$E_H\{s_i\} = -J \sum_{\langle i,j \rangle} \mathbf{s}_i \cdot \mathbf{s}_j - \mathbf{H} \sum_i^N \mathbf{s}_i \quad (2.21)$$

where we have assumed nearest neighbors interactions and isotropy of interactions.

There are no theoretical results yet for this model, but it has been studied using Monte Carlo calculations by several authors [22], [7], and there are results using mean field theory and renormalization theory. For two dimensional lattices Mermin and Wagner [21] showed that, for the Heisenberg model, there is no spontaneous magnetization at  $T > 0$ . Therefore there is no Curie temperature, although the possibility of other types of singularities is not excluded. For three dimensional lattices there is a Curie temperature, but the low-temperature behavior is radically different from that of the Ising model (Ashcroft and Mermin [2]). There are no small localized perturbations from the ground state and the changes in spontaneous magnetization and heat capacity from their zero-temperature values are proportional to  $T^{\frac{3}{2}}$ . At  $T < T_C$  there is a divergence in the magnetic susceptibility as  $\mathbf{H} \rightarrow 0$ . In this model the low-temperature heat capacity has the form  $Nk_B + aT$ , where  $a$  is a constant. Results for the high temperature expansions like the *linked-cluster* or the *connected-graph* methods are presented in the table 2.2. These methods and other High T series are explained in [6].

Tables 2.2 and 2.3 (see [7] for a complete list of references) show the values obtained for the critical exponents obtained theoretically for a two-dimensional and three-dimensional Ising models and for the three dimensional Heisenberg model, and we can compare them to the experimental results in iron, nickel and for a 2 dimensional gas. Nickel and iron are very close (within standard deviation of each other) to the results for Ising and Heisenberg in three dimensions. For the definition of the critical exponents and some of their properties, the reader is directed to APPENDIX A and to [5].

## 2.4. Van Vleck Approximation

The forces between the atoms which are responsible for ferromagnetism are the exchange interactions  $-J\mathbf{s}_i \cdot \mathbf{s}_j$ . As these products are clearly invariant under rotations, this forces cannot lead to anisotropy. To explain the fact that in real life magnetization depends on the direction, it is necessary to superpose some other coupling as a perturbation the larger exchange interactions. The first approximation to this kind of coupling is the dipole-dipole interaction.

If all the dipoles are perfectly parallel, it has been proven that their mutual dipole energy doesn't give any anisotropy in cubic crystals (see APPENDIX B). However, when the dipoles are not all mutually parallel, this is no longer true. In fact Van Vleck [10] has showed that when a perturbation calculation is carried to the second order, there is a dependance on the direction except in the case of complete parallelism. This latter case is an ideal one and only achieved at  $T = 0$ .

If we follow the derivation by Van Vleck we arrive at an interaction term:

$$H_{VV} = \sum_{i>j} C_{ij} [\mathbf{s}_i \cdot \mathbf{s}_j - 3(\mathbf{s}_i \cdot \hat{\mathbf{r}}_{ij})(\hat{\mathbf{r}}_{ij} \cdot \mathbf{s}_j)] + \sum_{i>j} \gamma_{ij} (\mathbf{s}_i \cdot \hat{\mathbf{r}}_{ij})^2 (\hat{\mathbf{r}}_{ij} \cdot \mathbf{s}_j)^2 \quad (2.22)$$

In the first sum, the term  $C_{ij}\mathbf{s}_i \cdot \mathbf{s}_j$  can be absorbed in the exchange interaction term. In fact, we can consider this as the first two terms in a Taylor expansion of the anisotropy interaction in terms of  $(\mathbf{s}_i \cdot \hat{\mathbf{r}}_{ij})(\hat{\mathbf{r}}_{ij} \cdot \mathbf{s}_j)$

$$H_{VV} = \sum_{k=1}^{\infty} a_k \sum_{i>j} [(\mathbf{s}_i \cdot \hat{\mathbf{r}}_{ij})(\hat{\mathbf{r}}_{ij} \cdot \mathbf{s}_j)]^k \quad (2.23)$$

Note that this interaction Hamiltonian depends not only on the direction of the spins, but also on their relative orientations. For simplicity's sake in the rest of this work we will keep only first neighbor interactions and the first term in the

sum, but more terms can be added without making any significant changes to the algorithms that will be presented. Therefore:

$$H_{VV} = a \sum_{\langle i,j \rangle} [(\mathbf{s}_i \cdot \hat{\mathbf{r}}_{ij})(\hat{\mathbf{r}}_{ij} \cdot \mathbf{s}_j)] \quad (2.24)$$

## 2.5. Theoretical Results

The equation 2.22 yields several interesting results that can be handled analytically:

**Sign of  $K_1$ :** It can be shown that  $K_1$  is negative for a face or body-centered lattice and positive for a simple cubic lattice, if it is due solely to the dipole-dipole member of 2.22. This is in agreement with the magnetometer measurements mentioned in Section 2.2. The dipole-dipole coupling gives a  $K_1$  of the proper sign for Nickel, since the latter has a face centered lattice and negative  $K_1$ . For Iron the observed sign does not agree with the dipole model, but that may be due to the fact that it is probable that iron atoms have a spin close to unity and thus more terms of the approximation must be used. It is impossible to estimate a priori which term is larger, so the best that can be done is to appeal to empirical evidence.

**Magnitude of  $K_1$ :** If the constant  $K_1$  is due to the dipole-dipole interaction, it should be of order  $A^4/10kT_C h^2 \nu^2$  per atom<sup>2</sup>, and of order  $A^4/h^3 \nu^3$  if it is due

---

<sup>2</sup> $A$  is the spin-orbit constant,  $T_C$  is the Curie temperature and  $\nu$  is a constant of the order of magnitude of the separation of the energy levels caused by the interaction of the orbit with the crystalline field. Reasonable estimates are  $A/h\nu \sim 10\text{cm}^{-1}$  and  $kT_C = 10^3\text{cm}$

to the quadrupole-quadrupole effect. This estimate is of order  $K_1 \sim 10^5$  is of the order of the experimental values from table 2.1. To derive the estimate of  $K_1$ , Van Vleck calculated that the constant  $K_1$  is of the order  $C^2/10kT_C$  per atom if it is due to dipole-dipole interactions. The quantity  $C$  is the constant of proportionality in the first term of 2.22. It should be of the same order of magnitude  $A^2/h\nu$  as the spin-orbit coupling in a molecular state with no mean angular momentum. If the dipole constant arose primarily from a pure magnetic interaction rather than from spin-orbit coupling,  $C$  would be too small by a factor of about  $10^{-3}$ . If  $K_1$  is due to the quadrupole-quadrupole coupling, it should be of the same order of magnitude as  $\gamma$  and  $\gamma$  is of order  $A^4/h^3\nu^3$ .

**Temperature dependance of  $K_1$ :** In this regard the Van Vleck model gives only rough qualitative results. It predicts that the anisotropy should vanish faster than the magnetization itself as the temperature approaches  $T_C$ , but the temperature variation is not as drastic as is found experimentally. In the case of iron, the observed anisotropy varies approximately as  $J^{10}$  near the Curie point, while the computed values using the approximation are between  $J^5$  and  $J^6$ . In the case of nickel,  $K_1$  is actually over fifty times larger at  $17^\circ$  than at  $293^\circ$ , whereas Van Vleck's calculations predict very little change in the magnitude of  $K_1$  over this range.

A more complete and detailed description of these results can be found in reference [10].

In 1954 Zener [17] treated the problem of the anisotropy energy for the thermally perturbed spin system for very low temperatures, assuming local parallelism in the spin system, and calculated the constant  $K_1(T)/K_1(0)$ . He found that the

temperature dependance of this anisotropy term for iron. He found that

$$\left[ \frac{K_1(T)}{K_1(0)} \right] \approx 1 - \frac{10}{\alpha} \quad (2.25)$$

where  $1/\alpha = k_B T / MwI$  ( $wI$  is the molecular field and  $M$  is the dipolar moment of the atom). This relationship works remarkably well for iron, as Carr [20] showed for iron. For nickel and cobalt the relationship is quadratic with the temperature, due to the different crystal lattice configurations.

### 3. SOME RESULTS IN $T = 0$ .

In this chapter we present a few simple results derived for the case  $T = 0$ . These results will be useful as tests for the outcome of the simulations that will be described later.

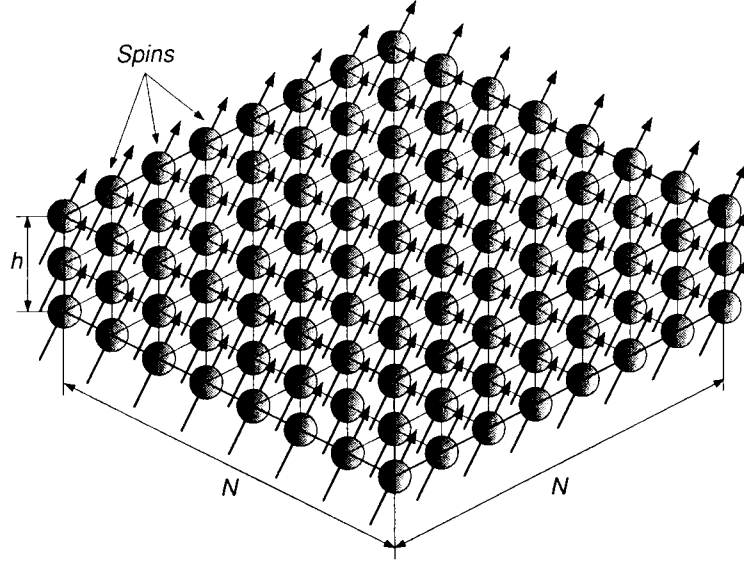
#### 3.1. The System

We will consider a system that consists of an array of  $N \times N \times h$  spins, where  $N \gg h$  to simulate a thin film. Each of these spins has a total magnetization  $s_i = 1$  and this magnetization is free to rotate in any direction.

Since  $N$  is a finite number and we want to approximate the behavior of a real film, we will assume  $\mathbf{s}_{i \pm Njk} = \mathbf{s}_{ijk}$  and  $\mathbf{s}_{ij \pm Nk} = \mathbf{s}_{ijk}$ . This is what we call *periodic boundary conditions* and basically turn a plane into a torus. On the other hand it is important that we have a surface in the  $k$  direction so we simply leave the boundaries open:  $\mathbf{s}_{ijN+1} = 0$  and  $\mathbf{s}_{ij0} = 0$ . This is fundamental as otherwise there is no anisotropy (see APPENDIX B) at zero temperature.

#### 3.2. The $T = 0$ approximations

Let the temperature  $T \rightarrow 0$ . In this case all thermal agitation will tend to vanish and the spins will be frozen in place. If we start from an initial state with spins aligned, then all the spins in the film will stay aligned, no matter where the external field is pointing to. In this case the energy can be found exactly. From equations 2.21 and 2.23, we obtain the total energy per particle. It must be remembered that to avoid double counting  $\langle i, j \rangle$  must sum over only half of the nearest neighbors:

FIGURE 3.1. The system at  $T = 0$ .

$$e = \frac{1}{N^2 h} \left[ -J \sum_{\langle i,j \rangle} \mathbf{S}_i \cdot \mathbf{S}_j - \sum_{i=1}^N \mathbf{S}_i \cdot \mathbf{H} - a \sum_{\langle i,j \rangle} (\mathbf{S}_i \cdot \mathbf{r}_{ij})(\mathbf{r}_{ij} \cdot \mathbf{S}_j) \right] \quad (3.1)$$

As the spins are all parallel,  $\mathbf{S}_i = \mathbf{S}$  for all  $i$ . Then the sums simplify in the following fashion:

$$-J \sum_{\langle i,j \rangle} \mathbf{S}_i \cdot \mathbf{S}_j = -JN^2(3h - 1) \quad (3.2)$$

$$-\sum_{i=1}^N \mathbf{S}_i \cdot \mathbf{H} = -N^2 h \mathbf{H} \cdot \mathbf{S} \quad (3.3)$$

$$-a \sum_{\langle i,j \rangle} (\mathbf{S}_i \cdot \mathbf{r}_{ij})(\mathbf{r}_{ij} \cdot \mathbf{S}_j) = -aN^2(h - 1) + aN^2 \cos^2(\theta_S) \quad (3.4)$$

where  $\theta_S$  is the spins azimuth angle.

The last term arises from the fact that the boundary conditions on the surface are open, as can be seen in APPENDIX B.

The total energy in the case of parallel spins can be written as:

$$e = f(J, a, h) - \mathbf{H} \cdot \mathbf{S} + \frac{a}{h} \cos^2(\theta_S) \quad (3.5)$$

where  $f(J, a, h)$  is a function of the *internal* parameters of the system and of the number of domains, and doesn't depend on the angle of the spins or the applied field.

We aim to find the minima of the energy for a given  $\mathbf{H}$ , as we would like to see what the ground state of the system looks like, so we take the derivatives with respect to  $\theta_S$  and  $\phi_S$ :

$$\frac{\partial e}{\partial \phi_S} = -H \sin(\theta_H) \sin(\theta_S) \sin(\phi_S - \phi_H) \quad (3.6)$$

If we set this derivative to 0 we deduce  $\phi_S = \phi_H$ . This result is important in itself as it is telling us that there is no anisotropy effect in the  $XY$  plane, as can be expected from the symmetry of the system. From

$$\frac{\partial e}{\partial \theta_S} = -H \sin(\theta_S - \theta_H) + \frac{a}{h} \sin(2\theta_S) \quad (3.7)$$

we get a transcendental equation for  $\theta_S$ :  $\sin(2\theta_S) = \frac{hH}{a} \sin(\theta_S - \theta_H)$ . Transcendental equations can't be solved analytically, but they can still render some interesting results. For instance we can find a solution for the inverse:

$$\theta_H = \theta_S - \sin^{-1} \left[ \frac{a}{hH} \sin(2\theta_S) \right] \quad (3.8)$$

There are two cases that are important: Let's imagine that there is no external field. The energy is then:

$$e = f(J, a, h) + \frac{a}{h} \cos^2(\theta_S) \quad (3.9)$$

As the energy doesn't depend on it,  $\phi_S$  is not determined and that means that it has the freedom of being anywhere in the  $XY$  plane. The minimization for  $\theta_S$  gives



$\frac{a}{h} \sin 2\theta_S = 0$ . Then  $\theta_S = 0, \pi$ , maxima or  $\theta_S = \pi/2$  a minimum. If we recall the definitions from the previous chapter, we have found the easy and hard directions.

Let us now study the magnetization in the direction of the applied field, and calculate it's minima.

$$M_{||} = \frac{\mathbf{H} \cdot \mathbf{S}}{|\mathbf{H}|} \quad (3.10)$$

$$= \cos(\theta_H - \theta_S) \quad (3.11)$$

$$\frac{\partial M_{||}}{\partial \theta_H} = \sin(\theta_H - \theta_S) \left[ 1 - \frac{\partial \theta_S}{\partial \theta_H} \right] = 0 \quad (3.12)$$

so we have 2 alternatives: either  $\theta_H = \theta_S$  or  $\frac{\partial \theta_S}{\partial \theta_H} = 1$  or to express it in terms of the variables we know  $\left( \frac{\partial \theta_S}{\partial \theta_H} \right)^{-1} = 1$  Solving the last equation, we get that  $\theta_S = \frac{\pi}{4}, \frac{3\pi}{4}, \dots$

We find that we have extremes for  $\theta_H = \frac{\pi}{4} - \sin^{-1} \left( \frac{a}{hH} \right)$

Values of interest are when  $\frac{a}{hH} = \pm 1$  as they are the limits of existence of the sine. We are also interested in when  $\theta_H = 0, \pi$ . These values are reached when  $H = \frac{\sqrt{2}a}{h}$  and this is the field strength of the critical line where we see no more minima in the range of  $\theta$

Note that if  $H \rightarrow 0$  then  $\theta_S \rightarrow 0, \forall \theta_H$  and that if  $a \rightarrow 0$ , then  $\theta_S \rightarrow \theta_H \forall \theta_H$

If we take a look at the energy corresponding to these values of  $\theta_S$  and  $\phi_S$  as a function of  $\theta_H$ , we observe that it has only one minimum, at  $\pi/2$ .

If we set  $H = 0$ , we can find the easy and hard axis from 3.5, and as we expected the hard axis is in the direction perpendicular to the film and the easy axis is in the plane of the film.

As the temperature is 0, there is no entropy contribution to the energy and therefore the value of MAE obtained from the integration of the magnetization

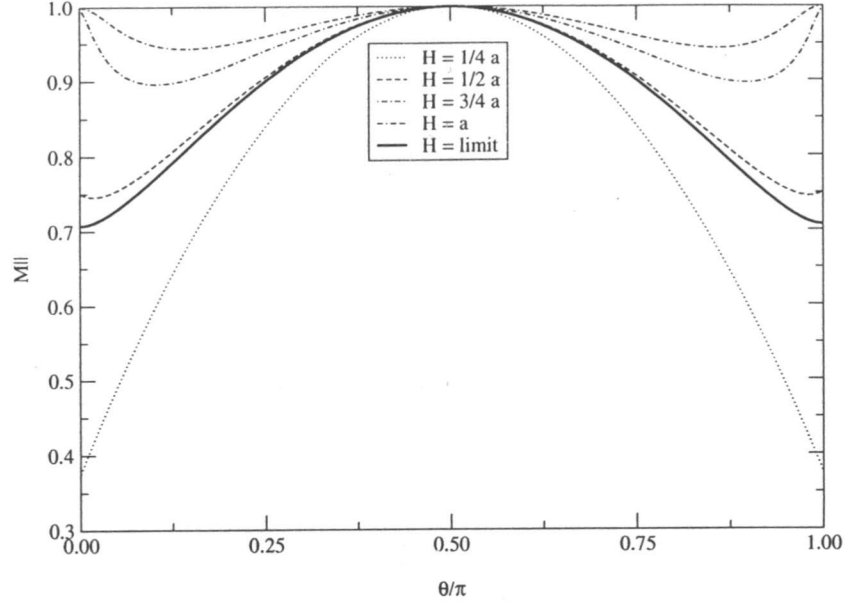


FIGURE 3.2. Magnetization in the direction of the field.

curves is identical to the difference of the Helmholtz free energies in the easy and hard axis measured at saturation.

$$MAE = \frac{a}{h} \quad (3.13)$$

We can also calculate the energy in the case when part of the top layer is missing, forming a step, parallel to the y axis. In this case:

$$e = f(J, a, N, m, h) - \mathbf{H} \cdot \mathbf{S} + \frac{a}{N(h-1) + m} [(N-1) \cos^2(\theta_s) + \sin^2(\theta_s) \sin^2(\phi_s)] \quad (3.14)$$

We can find the extremal axis by setting the external field to 0 and looking for the maxima and minima: We find that again the easy axis is parallel to the plane

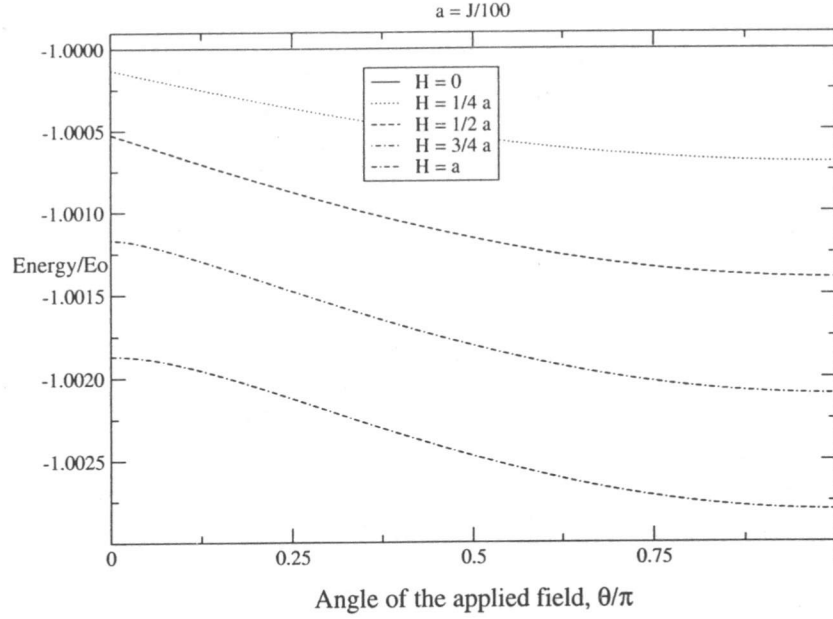


FIGURE 3.3. Energy versus  $\theta_H$ .  $a = 0.01 \times J$ .

of the field and the hard is perpendicular to it, but as we now lose the azimuthal symmetry, the easy axis is also parallel to the step.

$$MAE = \frac{aN}{(h-1)N + m} \quad (3.15)$$

Last we can work on two other particular cases:

- When there is one atom in the top layer: In this case the easy and hard axis don't change (we haven't changed the symmetry) and  $MAE = aN^2/[N^2(h-1) + 1]$

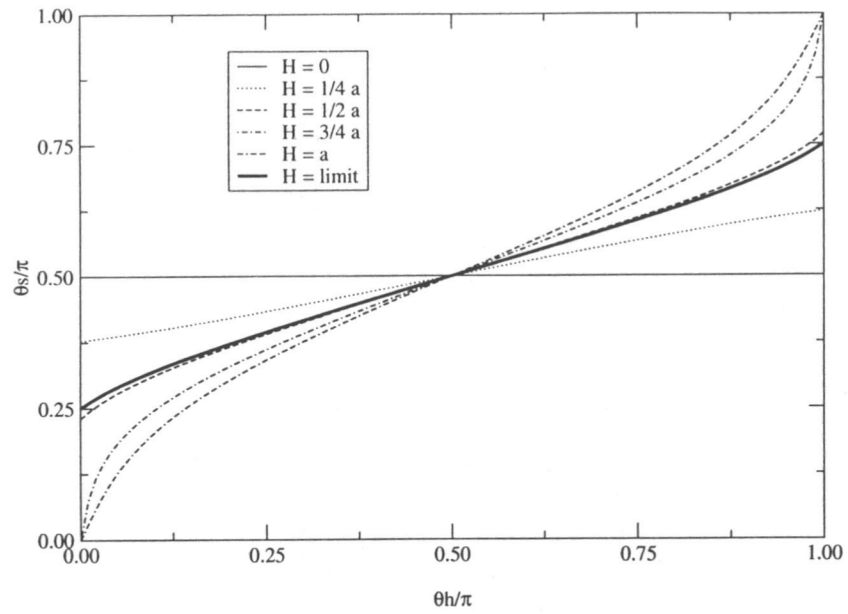


FIGURE 3.4.  $\theta_S$  versus  $\theta_H$ .  $a = 0.01 \times J$ .

- When only one atom is missing on the top layer: Again the extremal axis don't change and  $MAE = a(N^2 - 1)/[N^2h - 1]$

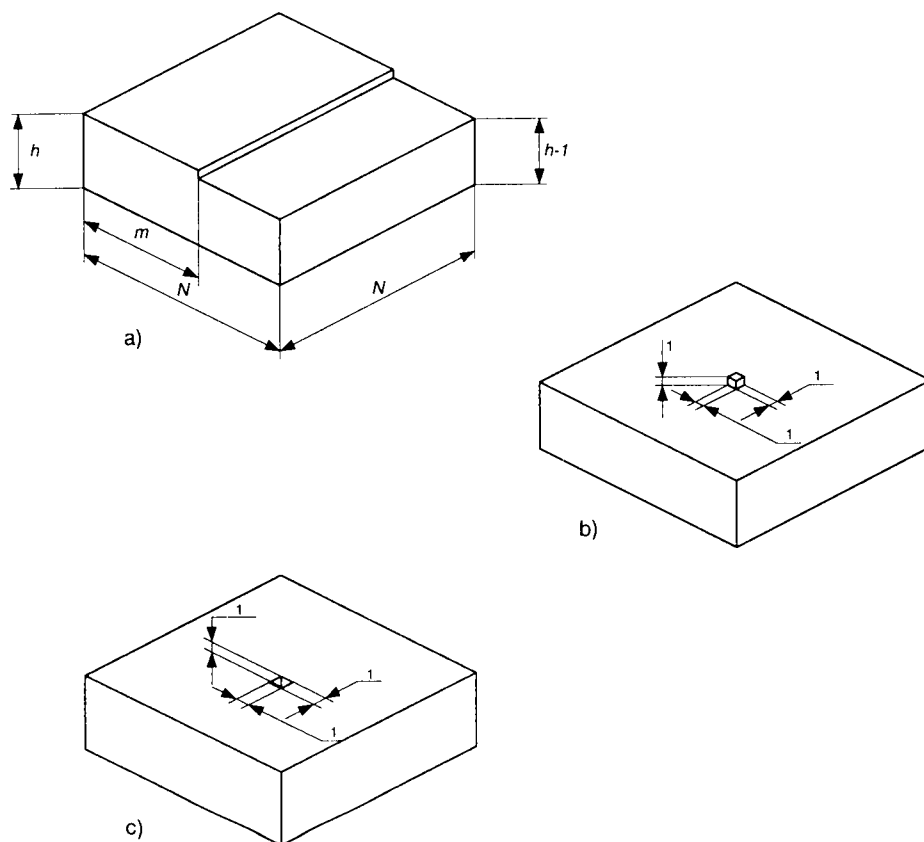


FIGURE 3.5. a) Film with a step; b) Film with a single extra spin on top; c) Film with a hole on the top layer.

## 4. THE MONTE CARLO SIMULATION

### 4.1. Introduction to Sampling Monte Carlo Methods

A Monte Carlo simulation is an attempt to follow the time dependence of a model whose change does not follow a rigorous equation of motion, but rather depends on a series of random events. This sequence of events can be simulated by a sequence of random numbers generated to that effect. If a second, different sequence of random numbers is generated, the simulation should yield different results, although the average values to which it arrives should lie within some *standard deviation* of each other. Examples of system frequently solved with Monte Carlo simulations are the *percolation problem* in which a lattice is gradually filled with particles placed randomly; diffusion and random walks, in which the direction of the next step of a particle is stochastic; diffusion limited aggregation, in which a particle executes a random walk until it encounters a 'seed' to which it sticks and the growth of the seed mass is studied when several random walkers are released simultaneously; etc. Monte Carlo methods are frequently used to estimate the equilibrium properties of a model, as they calculate the thermal averages of systems of many interacting particles.

The accuracy of a Monte Carlo estimate depends on how well the phase space is probed, so the results improve the longer the simulation runs, unlike in other analytic techniques for which the extension to a better accuracy may be too difficult.

The range of physical phenomena that can be explored with Monte Carlo simulations is very broad, as many models can be discretized either naturally or by approximation: The motion of individual atoms can be examined directly using random walks. Growth phenomena involving microscopic particles, since the

masses of colloidal particles are orders of magnitude larger than the atomic masses, the motion of these particles in liquids can be modelled as a Brownian motion. The motion of a fluid may be studied considering *blocks* of fluid as individual particles, but larger than the actual molecules forming the fluid. Large collections of interacting classical particles are in general well suited to Monte Carlo modelling and quantum particles can be studied either by transforming the system into a pseudo-classical model, or by considering the permutation properties independently. Also polymer growth can be studied as the simplest polymer growth model is just a random walk.

#### *4.1.1. The Relationship Between Experiment, Theory, and Simulation*

Simulations were developed originally to study systems so complex that there was no solution in a closed form, like the specific behavior of a system during a phase transition. If a model Hamiltonian is proposed that contains all the essential physical features, then its properties can be calculated and compared to the experimental values. If the simulation doesn't agree with experiment then the theory can be adjusted until the three elements (theory, experiment, and simulation) are in agreement. Once the simulation and the experiment yield similar results, the model can be studied in a detail not possible with experimental techniques, such as *turning off* certain parts of the Hamiltonian to investigate their overall effect, simulating different boundary conditions, etc., yielding a much better understanding of the model used and possibly suggesting new paths of research in both the theory and the experiment.

Other uses of simulations are to mimic the effects of experiments that cannot be tested, such as a reaction meltdown or nuclear war, or that the compounds

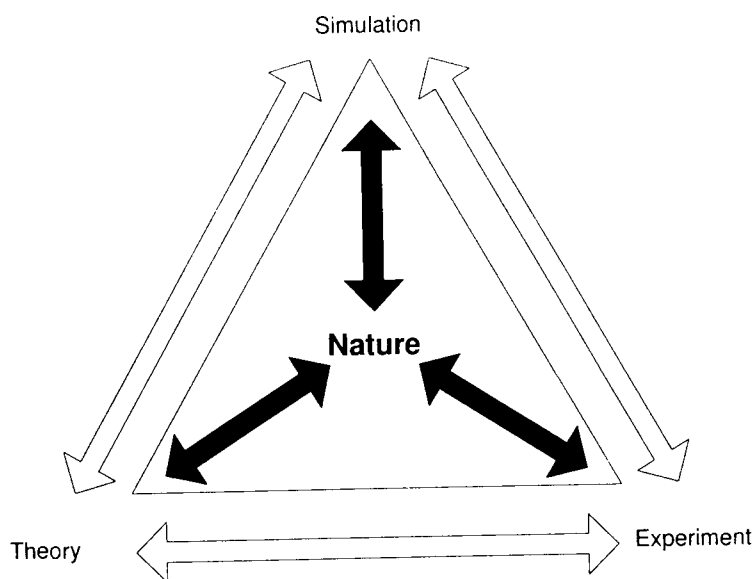


FIGURE 4.1. Schematic view of the relationship between theory, simulation and experiment.

investigated have not yet been found in nature. For instance in models such as diffusion limited aggregation, there is a very large body of simulation results, but experimental results are only now being obtained. Also unstable particles with very short half-life can be studied, etc.

In short Theory, Simulation, and Experiment are the three corners of the triangle shown in fig. 4.1, all with the same importance and advantages (as well as disadvantages) and all three are important in the ultimate goal of promoting a better understanding of Nature.<sup>1</sup>

---

<sup>1</sup>As the Philosopher said: The Creator had many good ideas when he was making the Universe, but making it easy to understand was not one of them.



#### 4.1.2. The Monte Carlo Method and Markov Chains

The Monte Carlo method for minimizing the Gibbs Free Energy involves an element of chance, hence its name. Random numbers are used to select the configurations of the system. To appreciate how this process works, it is useful to understand first the concept of *Markov processes*.

A *Markov* process is a rule for randomly generating a new configuration from the one present at the moment. The important thing about this rule is that it should **only** depend on the present state of the system, and it should not require knowledge of any previous one. This rule can be expressed as a set of probabilities: for each state  $\alpha$  and  $\alpha'$ , there is an associated probability,  $P(\alpha \longrightarrow \alpha')$ , that if the system is now in the state  $\alpha$ , it will be at the state  $\alpha'$  in the next step. These probabilities satisfy the sum rule, which is just the statement that the system will be somewhere in the next step:

$$\sum_{\alpha'} P(\alpha \longrightarrow \alpha') = 1 \quad (4.1)$$

We are interested in producing a Markov chain: a sequence of states generated by a Markov process, in which the frequency of occurrence of the state  $\alpha$  is proportional to the associated Gibbs probability,  $p_\alpha$ . To do this we need two more assumptions on the probabilities  $P(\alpha \longrightarrow \alpha')$ :

- From a given starting point it must be possible to evolve the system to any other configuration, by applying the evolution rule a sufficient number of times.
- The transition probabilities satisfy the *detailed balance* condition:

$$p_\alpha P(\alpha \rightarrow \alpha') = p_{\alpha'} P(\alpha' \rightarrow \alpha) \quad (4.2)$$

One example of a transition probability that obeys these rules for any Gibbs distribution is  $P(\alpha \rightarrow \alpha') = e^{\beta(E_\alpha - E_{\alpha'})}$ , where  $E_\alpha$  is the energy of the state  $\alpha$ .

Once we have chosen  $P$ , we can construct the probability  $W(\alpha, n)$  that the configuration  $\alpha$  will appear at step  $n$  of the Markov chain. It can easily be proven that if

$$W(\alpha, n) = p_\alpha$$

then the  $W(\alpha, n + 1)$  is also equal to  $p_\alpha$ .

It can also be proven that the difference at step  $n$  between the actual probability distribution of states and the Gibbs distribution decreases as we progress along the chain.

#### 4.1.3. The Metropolis Algorithm

The most important and most frequently used algorithm to generate Markov chains is the one developed by Metropolis *et al.* [16]. The change in the energy of the system as it goes from state  $\alpha$  to  $\alpha'$  is calculated. If the change is negative, then the new configuration is automatically accepted. If it is positive, the change is accepted with probability  $e^{-\beta(E_\alpha - E_{\alpha'})}$ . In other words,

$$P(\alpha \rightarrow \alpha') = \begin{cases} A^{-1} & \text{if } E_{\alpha'} < E_\alpha, \\ A^{-1}e^{-\beta(E_\alpha - E_{\alpha'})} & \text{if } E_{\alpha'} > E_\alpha. \end{cases} \quad (4.3)$$

where  $A$  is a normalization constant to insure that equation 4.1 is satisfied. The detailed balance assumption and the accessibility criterions can be proven to be satisfied if state  $\alpha'$  is obtained from state  $\alpha$  in a finite number of steps.

The practical implementation of this algorithm is very simple and straightforward, and this is one of the main reasons of its enduring popularity and success.

First a new configuration  $\alpha'$  is created by some method. Then the energies of both the old and the new states is calculated. If the difference is negative then the new state is accepted. If it is positive a pseudo-random number is generated and used to accept or reject the move with probabilities given by equation 4.3

## 4.2. Solving The Ising Model with Monte Carlo

### 4.2.1. The Algorithm

The recipe typically used to simulate the time evolution of the Ising model is the Metropolis algorithm 4.2, described generally in the previous section. The initial state of the lattice (it could be 1, 2, 3 or  $n$  dimensional) can be chosen to be ordered, when all the spins point in the same direction, or random, that is, the spins are chosen to point up or down depending on some random variable. Either of these initial states has advantages, specially at low temperatures: the ordered state is convenient when the final state one wants to arrive to is magnetized, and the random state, when the final state is presumed to have no magnetization.

After the initial state is chosen, the algorithm chooses one spin of the lattice and calculates the energy change ( $\Delta E$ ) corresponding to altering the state of this spin, as well as the *Boltzmann factor*  $\exp(-\beta\Delta E)$ . If the Boltzmann factor is greater than 1 ( $\Delta E$  negative), it means that the new direction of the spin is favorable from an energy point of view, and the change is accepted. If, on the other hand, it's less than 1, the change is accepted only if it is larger than some random number  $0 < r < 1$ . It is in this step where the thermal effects are found.

The next step is to choose another spin and perform the same operation. The next spin can be chosen either sequentially or randomly. On the average both

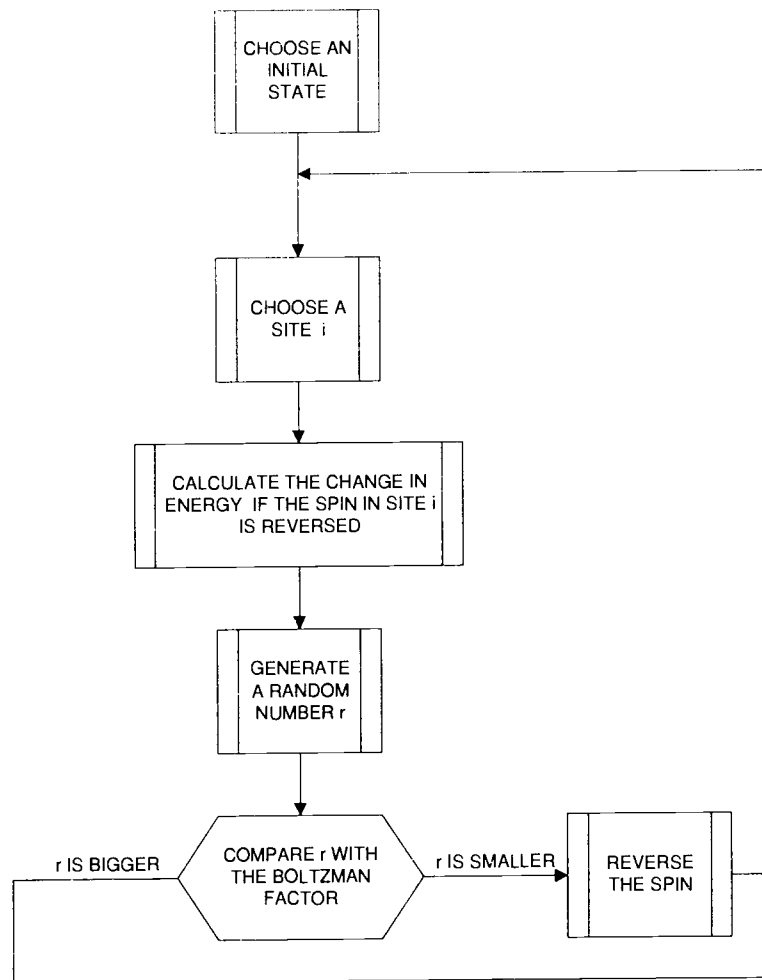


FIGURE 4.2. Flow diagram of the Metropolis algorithm.

methods yield the same results and the user can opt for either of them taking into account factors like processor time, ease of implementation, etc.

*Measurements* on the system should be taken between unrelated states, that means that states should be statistically independent. In other words the second state shouldn't be obtained from the first via a finite sequence of intermediate states. This is impossible to do, as the implementation of the algorithm shows,

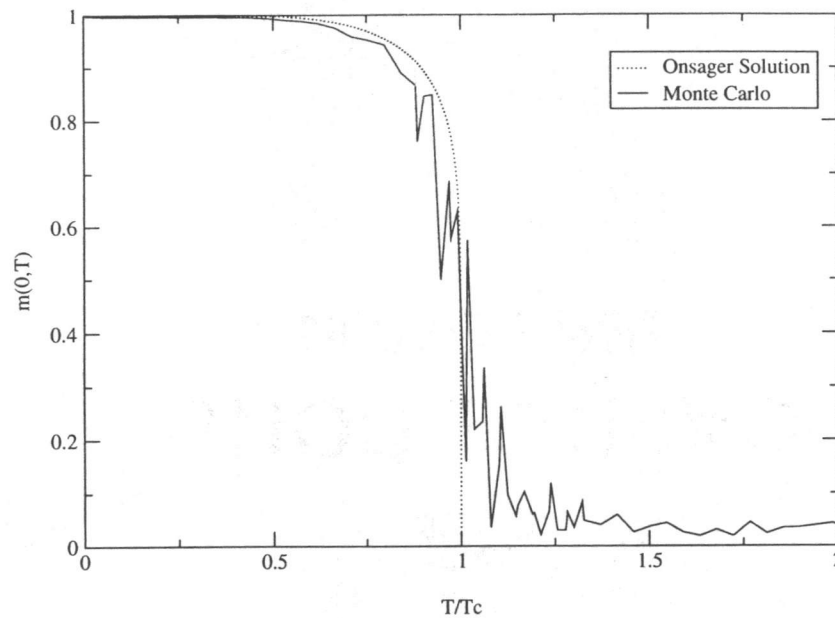


FIGURE 4.3. Monte Carlo results for the spontaneous magnetization in the two dimensional Ising model.

but the states can be rendered *pseudo-independent* if the sequence of states between measurements is large enough. If we call a *Monte Carlo step* the number of iterations of the procedure equal to the number of sites in the system, the sequence mentioned should be at least an order of magnitude larger than said step and, depending on the system, two or three orders of magnitude larger.

Another concern when making measurements is that, depending on the temperature, the system may have a considerably long relaxation time in which it goes from the initial to the final state. This time can be reduced choosing an adequate initial state, as is mentioned above. This consideration is not too critical for the Ising model, as the phase space is relatively small. The special measures taken to minimize the relaxation time for the Heisenberg model will be described in detail later.

In the figure 4.3, we have plotted the spontaneous magnetization of a 2-dimensional square lattice with 100 sites per side, calculated using the algorithm described above. If we compare it with figure 2.3, we can see that they have the same general shape although, as it is to be expected near a region of critical behavior, the computational noise near the critical temperature is quite noticeable. In fact the agreement with Onsager's solution (shown in a dashed line) is quite remarkable, and the critical temperature, is between  $2J$  and  $2.5J$  as it was predicted, even if the algorithm used was quite primitive by today standards and the lattice quite small.

#### 4.2.2. Boundary Conditions and Related Concerns

One of the problems of simulating a system in a computer is how to treat the *edges* of the system. There are several options: Periodic boundary conditions where the system is basically wrapped in a  $d + 1$  torus, free or open boundary conditions, where the sites on the border of the system are not connected to any other site (other than the explicit ones), mixed boundary conditions, where in some direction we have periodic boundary conditions and in the others free, etc.

- **Periodic boundary conditions:** As we mentioned above, periodic boundary conditions wrap the system in a  $d + 1$  torus ( $d$  is the dimensionality of the system). In this case the last spin of each row is *bonded* to the first spin in the row as if they were first neighbors. The same is true between the top and bottom rows. This procedure eliminates all surface effects, but the system is still characterized by a longitude  $L$ , as the maximum correlation length is  $L/2$  and the results differ from the results on an infinite lattice.

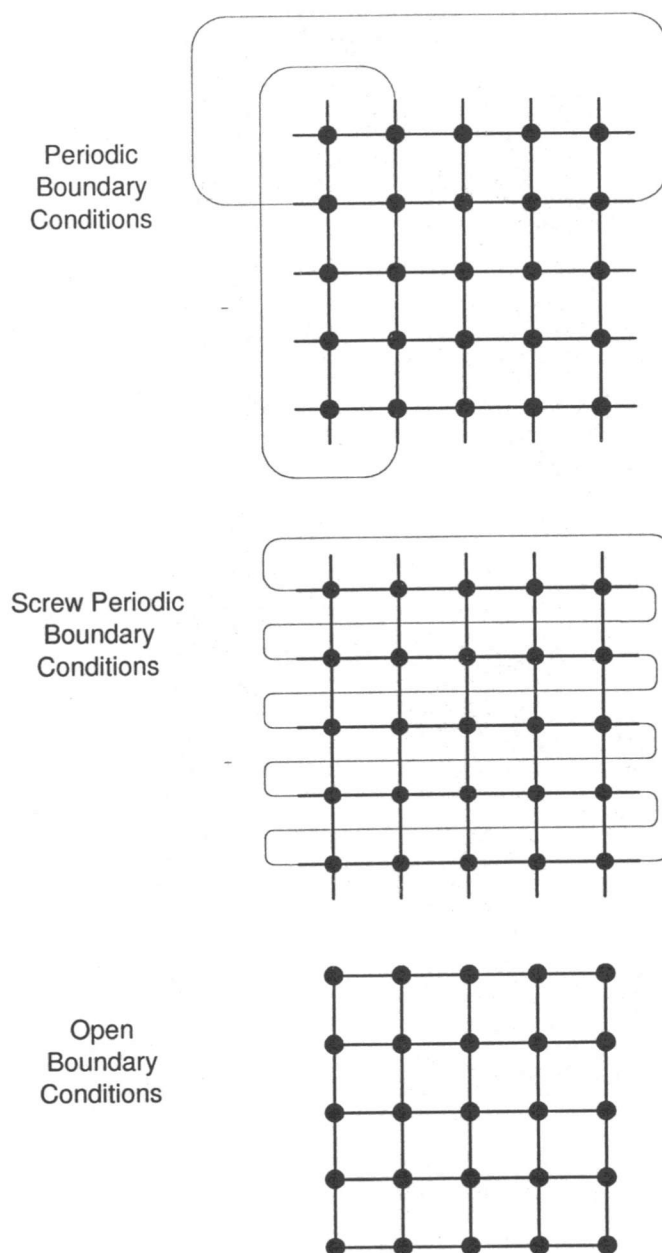


FIGURE 4.4. Three examples of boundary conditions: periodic, screw-periodic and open.

- **Screw periodic boundary conditions:** In this kind of boundary condition the last spin of each row sees the first spin of the next. The main result is that a seam is introduced in the system making the system inhomogeneous. In the limit large number of spins this effect is negligible, but for finite systems there will be a systematic difference with the periodic boundary conditions. This type of boundary condition is particularly efficient when simulating systems with dislocations.
- **Antiperiodic boundary conditions:** These boundary conditions were introduced to simulate systems with vortices. The last spin on each row is connected with the first one anti-ferromagnetically to produce a geometry in which vortices can exist. This is a very specialized boundary condition, and it is only used in a limited number of cases.
- **Free boundary conditions:** When there is no connection between the end rows and any other row we have open or free boundary conditions. In this case we introduce some finite size smearing effect as well as surface and corner effects in the places where we have missing bonds.

Other boundary conditions are possible such as *Mean field boundary conditions* and *Hyperspherical boundary conditions* and the reader is directed to Landau and Binder [7] for more information.

#### 4.3. Solving the Heisenberg Model with Monte Carlo

The main difference between the Ising and Heisenberg models is that, while the Ising spins can take only one of two values, along a single direction, the Heisenberg spins can point in any direction. At first sight this would make the attempt



of solving it using a Metropolis algorithm impossible, as having an infinite set of possible states clearly violates the first of our assumptions on the probabilities. But working with real computers means that we are dealing with finite precision numbers and this makes the state space finite (albeit very large).

The first of the issues we have to deal with is the generation of random vectors. Two possibilities are immediately evident: We could generate 3 random numbers and normalize the resulting vector to 1 or generate only 2 and use them as the two directions of a spherical unitary vector. The problem with just generating a cartesian vector is that even if the random number generator has a very uniform distribution, the vectors will be concentrated in the region where at least 2 of their components are similar (see fig 4.5). This effect is due to the normalization process, as the absolute value function doesn't have an uniform distribution. Another problem with this method is that we have to generate 3 random numbers, and good generators consume a significant amount of CPU time. The second option is more efficient, as generating an unitary vector in spherical coordinates requires only 2 parameters, but the price we pay is that the vectors are concentrated near the axis of the parameter sphere, as can be observed in figure 4.5.

An elegant solution for this problem is generating two numbers in mixed coordinates:  $-1 \leq z \leq 1$  and  $0 \leq \phi \leq 2\pi$ . The third parameter  $\rho$  can be easily obtained from the normalization:  $\rho = \sqrt{1 - z^2}$ . Then from these numbers we can easily calculate  $x$  and  $y$ :

$$x = \rho \cos \phi$$

$$y = \rho \sin \phi$$

This method gives a very uniform distribution of vectors on the surface of the sphere, as can be seen in the example of figure 4.5. It must be remarked that

in the 3 examples of said figure, the random number sequences were exactly the same, but the resulting distributions are completely different in shape.

A second question that arises from having 3-dimensional unitary vectors is whether to use a spherical or cartesian notation as the norm for the rest of the program. While spherical notation is very compact and practical when dealing with paper and pencil problem (specially when dealing with rotations and figuring the mean orientation of the sample), using it in a computer algorithm has the disadvantage that it requires the use of trigonometric functions and these slow down the operation of the program, in particular when a large number of vectors must be added together.

#### *4.3.1. Changes Made to Speed the Program*

Due to the size of the sampling space the time it takes to the sample to go from the initial to the final state (relaxation time) turns out to be very long, specially when we deal with medium to low temperatures <sup>2</sup>. One way to deal with this problem is to choose carefully the initial state so that it is as close as possible to the final state. There are several ways of ensuring that the algorithm starts from the ideal initial state. Probably the most popular would be to start every spin in the direction of the external field, that is start the system in a perfectly magnetized state. The problem is that we risk *locking* the system in this state of saturation. Another possible initial state is to start the system in a random configuration, with no internal magnetization whatsoever. This is the logical choice when we

---

<sup>2</sup>We will consider low temperatures to be at least an order of magnitude smaller than ambient temperature.

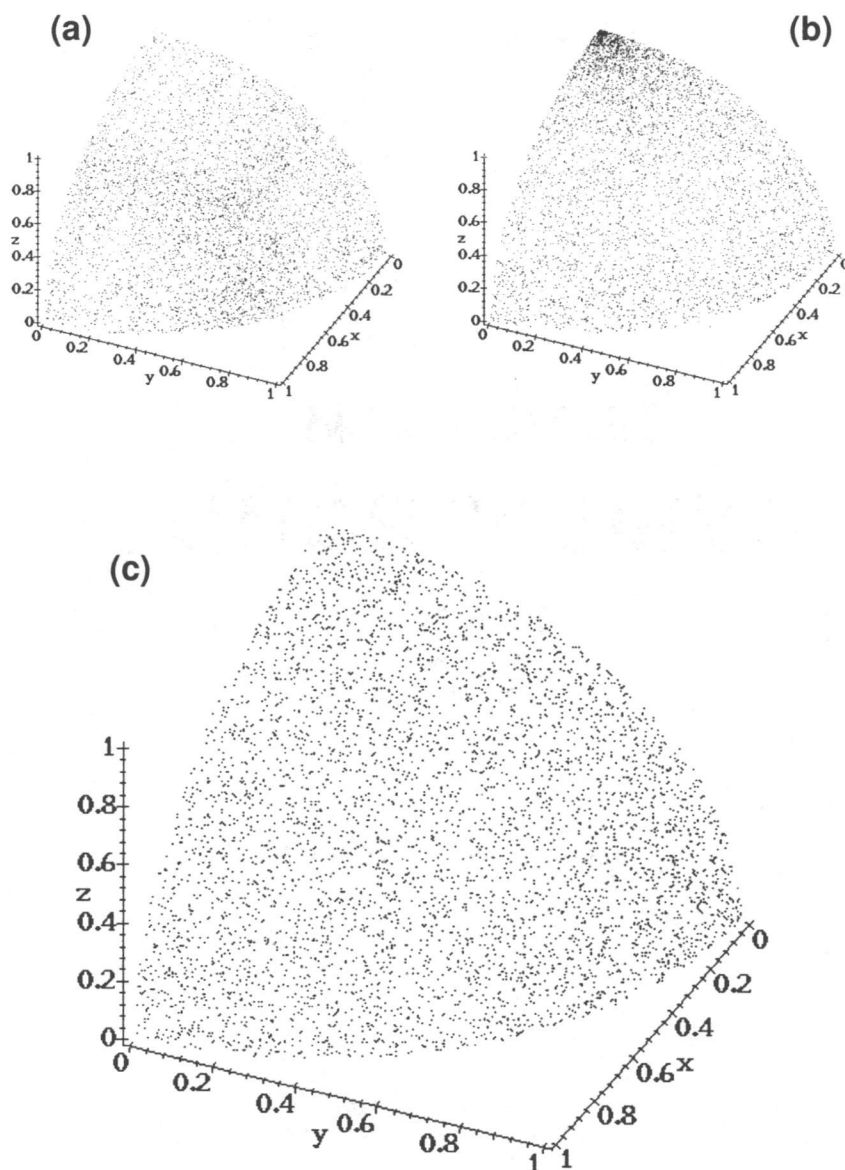


FIGURE 4.5. Three methods of generating random vectors: (a) Cartesian, (b) Spherical and (c) Mixed.

want to study the magnetization curve, but it is not very useful otherwise. Again we have chosen a mixed approach: In the initialization of the program the spins are all aligned, pointing in the direction of the easy axis. Then using the Newton method, we minimize the energy for a given to the direction of the magnetization. Because of the competition between the anisotropy term and the field term, the minimum energy direction will can be in between the easy axis and the direction of the external field. This gives a good starting point to a first run of the program where we let the system evolve for a considerable period of time, randomizing the sample. After this long relaxation time, we use the state we arrived to as the initial state for the next run of the algorithm. Subsequent measurements use the final state of the system, after a run through the minimization algorithm, as the initial state.

Another difficulty that presents itself in the systems we will consider is the relative size of the terms in the Hamiltonian. If we examine the values found in nature for iron (see table 4.1), we will notice that  $J \sum_{\langle i,j \rangle} \mathbf{S}_i \mathbf{S}_j$  is the dominant term, and is at least 4 orders of magnitude larger than the anisotropy term. This produces a separation of the time scales of the evolution of the Hamiltonian, one for the main term, an another for the anisotropy term. Fortunately the fast term is invariant under rotations, so the Newton minimization only takes the anisotropy and the external field into consideration. For this reason we accelerate the convergence of the system by minimizing the energy before each main Monte Carlo step. If  $\mathbf{R}$  is the operator that rotates a vector,

$$\begin{aligned}
\mathbf{R} \left( \sum_{\langle i,j \rangle} \mathbf{s}_i \cdot \mathbf{s}_j \right) \mathbf{R}^{-1} &= \sum_{\langle i,j \rangle} \mathbf{R} (\mathbf{s}_i \cdot \mathbf{s}_j) \mathbf{R}^{-1} \\
&= \sum_{\langle i,j \rangle} (\mathbf{R} \mathbf{s}_i \mathbf{R}^{-1}) \cdot (\mathbf{R} \mathbf{s}_j \mathbf{R}^{-1}) \\
&= \sum_{\langle i,j \rangle} \mathbf{s}'_i \cdot \mathbf{s}'_j
\end{aligned}$$

As the recipe for the calculation of the partition function, and hence to the determination of the temperature dependance of the system calls for the integration of the energy over all possible orientations, it is clear that a rigid rotation of the spins will not affect the final result. Therefore when we minimize the energy by rotating the spins we will only feel the influence of the anisotropy term. This will in turn accelerate the convergence of the algorithm and will insure that the effect of the Van Vleck term will be taken into account, regardless of the difference of relative sizes between it and the main exchange term, that would otherwise be reflected in different time scales for their respective convergence.

In figure 4.6 we show the dependance of the energy with the angle of the magnetization. We must notice that for the  $T = 0$  there is a maximum at  $\theta = 0$  and a minimum at  $\theta = \pi/2$  corresponding to the easy and hard axis. It can be seen that as the temperature increases these extremum change their position. Also the difference in energy between the maximum and the minimum decreases with growing temperature, so we must be careful when minimizing the energy, not to reach false minima.

To reduce even more the time it takes to the system to arrive to a steady state, we use a method suggested by Landau in a personal communication. We need to maximize the number of accepted transitions to help the system evolve faster. One way to do this would be raising the temperature, thus increasing the

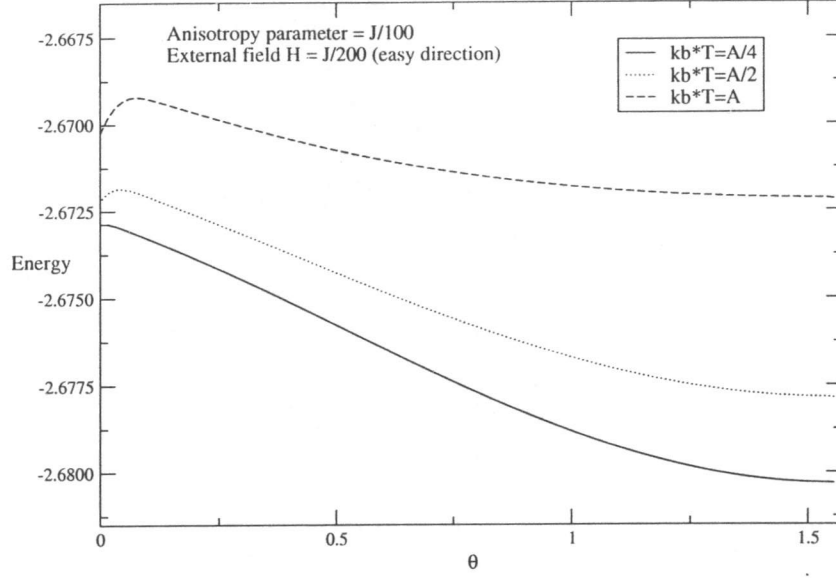


FIGURE 4.6. Total energy versus the azimuth angle of the applied field

Parameter	Magnitude [eV]	Magnitude relative to $J$
$k_B T_C$	$10^{-1}$	1
$k_B T_{ROOM}$	$2.5 \times 10^{-2}$	.25
$\mu_B H_{SAT}$	$2 \times 10^{-2}$	$2 \times 10^{-3}$
$A$	$10^{-6} \rightarrow 10^{-5}$	$10^{-5} \rightarrow 10^{-4}$

TABLE 4.1. Magnitude of the different components of the Heisenberg Hamiltonian.

thermal excitation, but one of the goals of this work is to study the temperature dependence of the MAE, and this would defeat the purpose. Instead we realize that if the difference  $|\mathbf{S}_i - \mathbf{S}'_i|$  is small, the change in energy will also be small, and therefore more likely to be accepted as a valid transition, even if it doesn't lead to a decrease in the total energy of the system. To exploit this fact we can limit the angle of rotation between the old and new vectors. We present an example in figure 4.7 (for simplicity's sake we use an initial vector in the  $Z$  direction, but any

orientation is possible). In a normal situation the final state vector  $\mathbf{S}_i$  will be in any position in the sphere of radius 1, as is shown in figure 4.7 (a), and the angle  $\gamma$  will have values between 0 and  $\pi$ . Therefore the maximum value of  $|\mathbf{S}_i - \mathbf{S}'_i|$  is 2. If we limit the angle  $\gamma$  between the vectors, as in figure 4.7 (b), the maximum difference will be smaller. In the example we have limited the angle to the interval  $[0, \pi/2]$  and so the maximum difference is obtained with spin  $\mathbf{S}''_i$  and will be equal to  $\sqrt{2}$ , but any limit angle is possible, although the selection must be careful, as too small a limit will make the program to adapt to large deflections.

#### *4.3.2. Other Points of Interest*

Two more modifications were made to the standard Metropolis algorithm: one subroutine to find the technical saturation point and another to find the principal axis of symmetry.

It is easy to explain the need to find the easy and hard axis every time the sample changes. A priori no assumption can be made regarding their direction, although it is evident that the easy axis is in the plane of the sample and the hard direction is perpendicular to it. On the other hand when the defects start accumulating on the top layer of the system, we can't assume that they will stay in the same place as the symmetry is broken in a random manner.

While the saturation point at zero temperature can be found exactly, we have to be more creative when the temperature rises. One option is to choose an ad hoc saturation field so large that we can insure that the saturation condition has been reached. This approach has two disadvantages. First is that we lose a lot of definition on the magnetization curves as we pretty much know that most of the

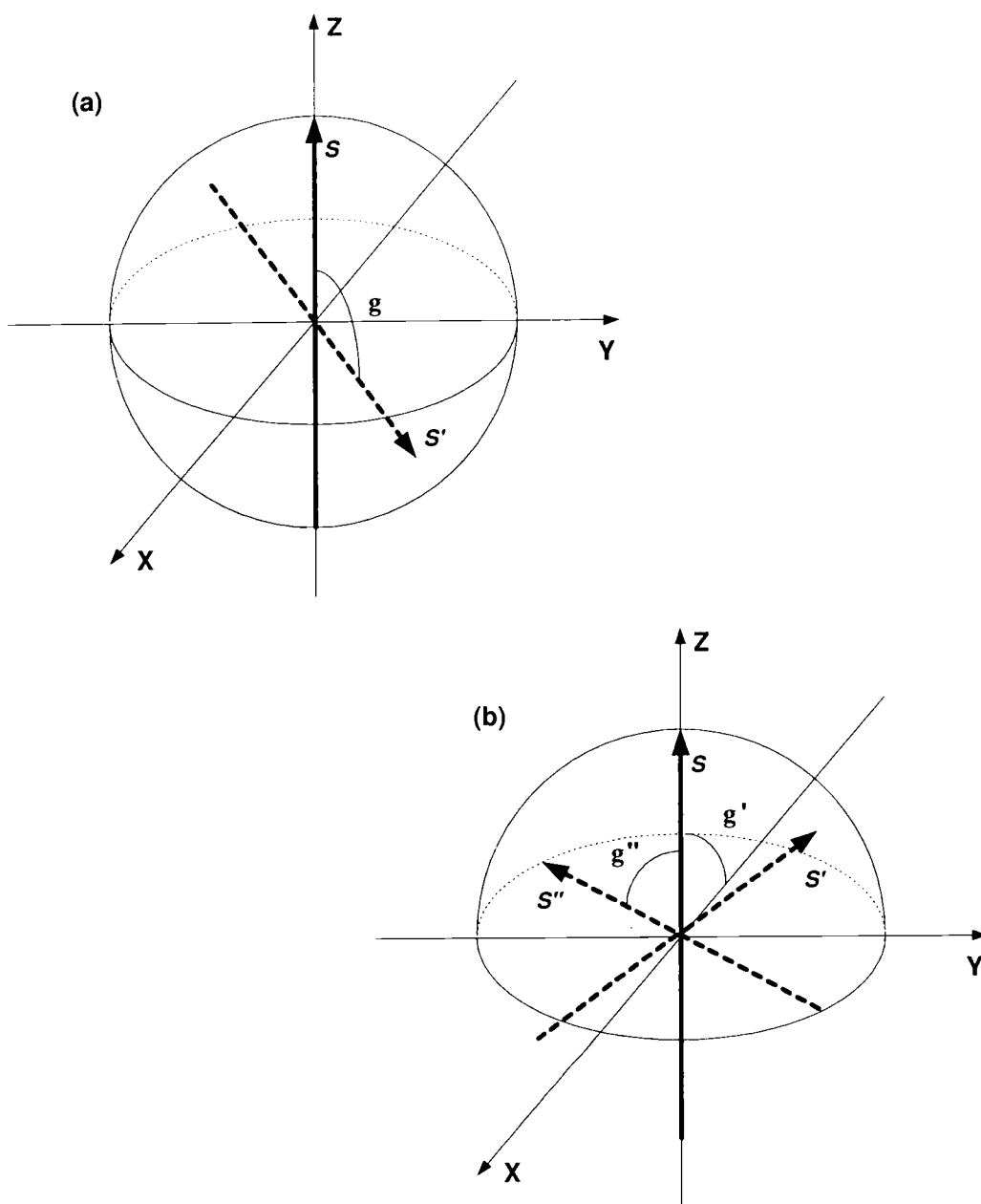


FIGURE 4.7. Limiting the possible deflections in the angle of the spin.



computational effort will be wasted in regions of the curves with no interest to the calculation. Second is that entropy related effects will not be taken into account.

To solve this we opted for a bisection type scheme to find the place where the two curves are reasonably close (a percentage of the maximum magnetization). This approach is conveniently fast as we only take 10 iterations (more would be redundant) to reach a satisfactory precision on the saturation point.

#### *4.3.3. Changes Made to Accommodate Defects*

One last set of modifications were made to the classical Metropolis algorithm, and it was the capability to add defects or holes to the sample in any position. This was accomplished simply defining a logical array that declared if each position was occupied or not. If the position was empty, the length of the spin was set to zero, but no other action was taken. It is notable that this approach can also be useful if we want to deal with substitutional defects, as when certain atoms are replaced with impurities, just by setting the length to a number different than 0 and 1.

## 5. MONTE CARLO TESTS

In this chapter we will describe some of the tests performed to insure that the simulation performs adequately. These tests include tests to the program itself and to the model. The results from Chapter 3 are used primarily as a basis of comparison.

### 5.1. Testing Stability

The first test we performed was regarding the number of Monte Carlo steps needed to achieve a stationary solution, and how limiting the angle of rotation influenced this number.

As mentioned in Chapter 4 the system needs some time to evolve from its initial state to the final or stationary state, which will be defined as a state that is invariant in time, within some standard deviation. It is clear that, due to the thermal excitation of the system, there will be fluctuations even in a stationary state. Therefore it is not practical to expect any final state to be perfectly constant. To solve this problem the program must calculate the value of some parameter of the system (usually the total energy), and average it over a number of steps. As the system approaches the steady state, the mean value will tend to a constant and the standard deviation will be of the order of the thermal excitation.

In practice the easiest way to insure a steady state is to find the worst case scenario (for instance when the thermal excitation is very small) and find approximately the number of steps until it reaches the final state. This number is then used for every other calculation.

In the figure 5.1 we depict a run of the algorithm for a very low temperature (in fact the lowest temperature used in this work). The total energy was calculated

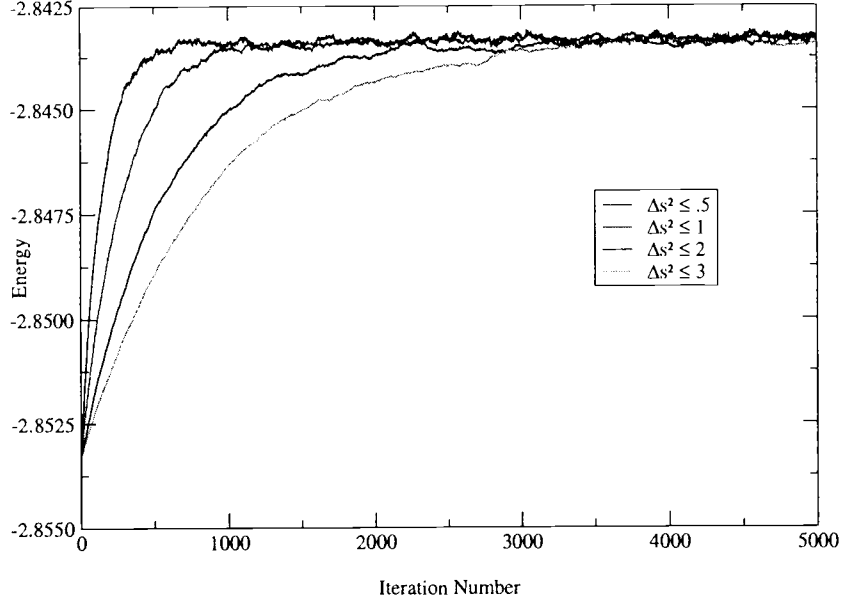


FIGURE 5.1. Energy vs. Time for different values of the limit  $\ell$ .

for every step and it is shown versus the step number. This number is congruent with time, and can be interpreted as such. As we mentioned in section 4.3.1, the maximum angle of deflection for individual spins during a step was limited:  $|\mathbf{s}_i - \mathbf{s}'_i| \leq \ell$ , where  $\ell$  is a number between 0 and 2. This was accomplished generating vectors and testing the difference against the existing spin. Although this seems to be inefficient, it is the only way of incorporating the factor  $\ell$  without losing the uniformity of the spin distribution. Also shown in figure 5.1 is the energy vs. time curve depicted for several values of the constant  $\ell$ .

As can be easily perceived, the smaller the limit  $\ell$  is, the sooner the steady state is achieved. There is however another factor that should be considered. While many more changes are accepted by the Monte Carlo part of the algorithm, a large number of possible vectors are rejected by the limiting part. As the generation of

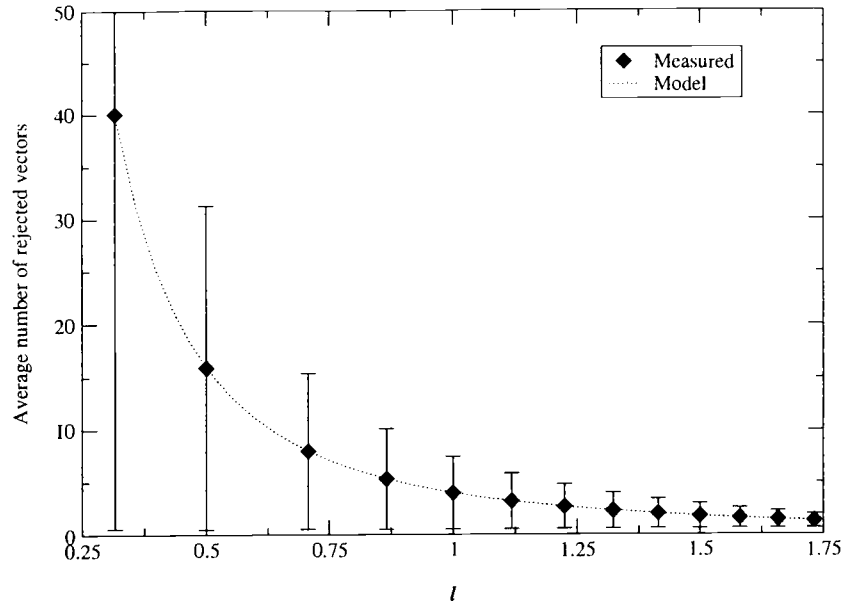


FIGURE 5.2. Average time required for a single step relative to the limit  $\ell$ .

these random vectors consumes a significant amount of CPU time, the overall effect is that the same number of Monte Carlo steps take longer to complete, the smaller  $\ell$  is (see figure 5.2). The average number of discarded vectors is easy to estimate if we realize that it is proportional to the relative surfaces of the sphere and the segment of sphere described by the limit  $\ell$ . This ratio has a very simple expression:  $N = 4/\ell^2$ . Therefore it is a balancing act to find a limit that insures the arrival to a steady state without rendering the process too slow to be practical.

## 5.2. How Reliable is it to Take Just One Example of Random Coverage?

The energy of the system depends mostly in the number of bonds between the spins. Each spin on the bulk of the material has 6 bonds (nearest neighbors) and those on the surface only 5. This missing bond is the one responsible for

the existence of the magnetocrystalline anisotropy as was shown in APPENDIX B. Therefore when more bonds are missing, as is the case when we have holes or defects, the MAE will be a function of the average number of nearest neighbors present. This was discussed by Taniguchi [19] who using Van Vleck's hamiltonian, found that the anisotropy constant is a function of  $n^2$ , the density of spins. This result has a direct influence in our problem as, in random configurations of spins on the top layer, the density does not change even if the particular distribution does. This was further verified experimentally as is described in section 6.3

### 5.3. Comparison to $T = 0$ Results

One of the first tests of our program was to see how similar to the  $T = 0$  results, were the ones obtained for very low temperatures. We chose for this test a temperature of  $k_B T = J/1000$  and we calculated the magnetization in the direction of the field and the energy, both graphed versus the angle of the applied field, to compare them with their counterparts (figs 3.2 and 3.3) from chapter 3. We found that the shapes of the curves were remarkably similar, reproducing most of the characteristics of the  $T = 0$  graphs.

Also it is important to note that the values of MAE tend to the predicted  $T = 0$  values predicted when the temperature  $T \rightarrow 0$  as can be seen in fig 6.10. This behavior was also verified in the case of one atom on the top layer and in the case of a single hole. The results are presented in figure 6.9, where the energies are measured as multiples of the MAE at 0 temperature, to make visualization easier.

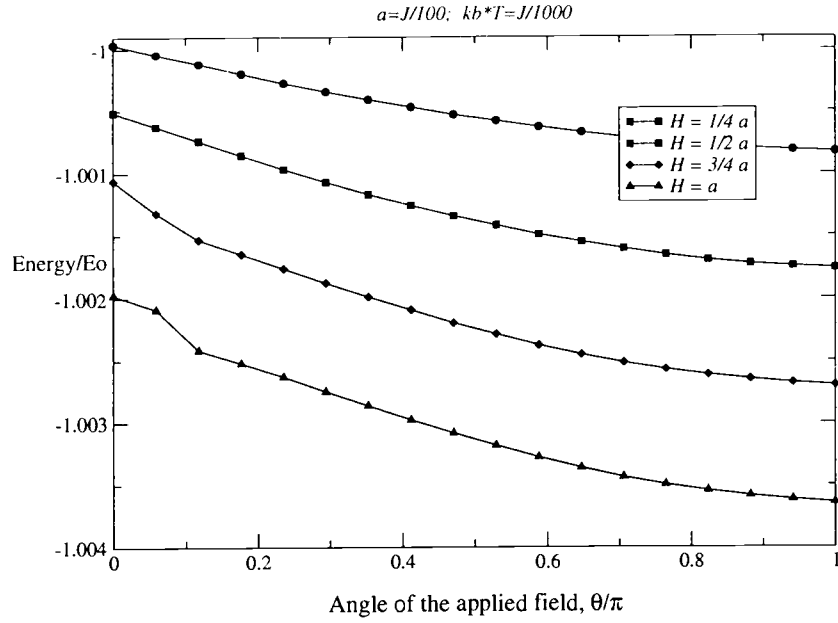


FIGURE 5.3. Energy vs. angle of the applied field for  $T = J/1000$ .

#### 5.4. Critical Behavior

As a further test to the model we calculated the Magnetization vs. Temperature curve at 0 external field. The shape of this curve is perfectly congruent to the existence of a phase transition from ferromagnetism to paramagnetism, as well as the existence of a Curie temperature. However no critical exponents can be calculated as it would have required further modifications to the program, slowing it even more (see [7]). As an effect of the finite size of the sample, Monte Carlo sampling of response functions systematically underestimates them. This effect comes from a result in probability theory that in estimating the variance  $s^2$  of a probability distribution using  $n$  independent samples, the expectation value  $E(s^2)$  of the variance is systematically lower than the true variance  $\sigma^2$  of the distribution, by a factor  $(1 - 1/n)$ . The corrections necessary for an accurate prediction of the

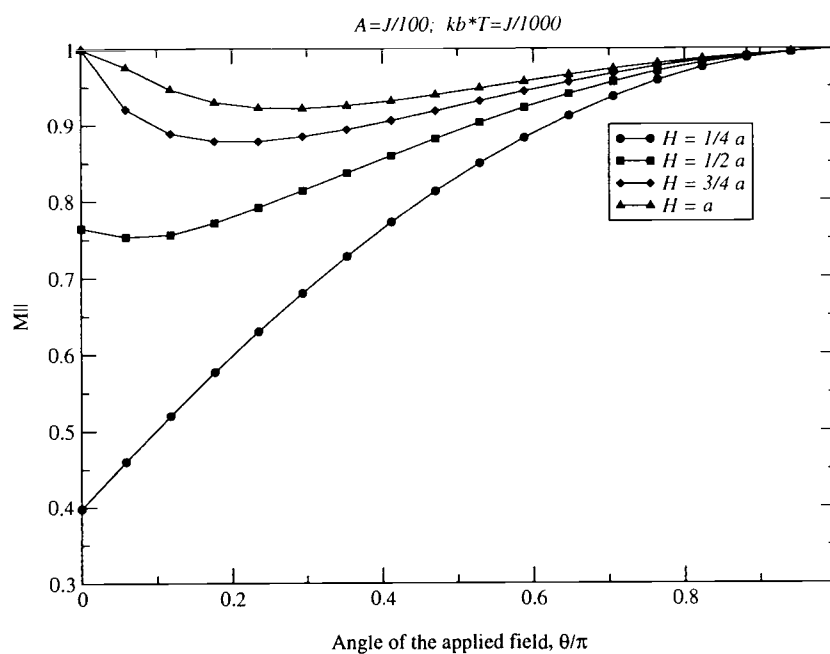


FIGURE 5.4. Magnetization in the direction of the field.

critical exponents and temperatures would require the calculation of correlation times for each system and for several sizes of each of them, significantly slowing down the acquisition of data, although this might be an interesting line of research for the future.

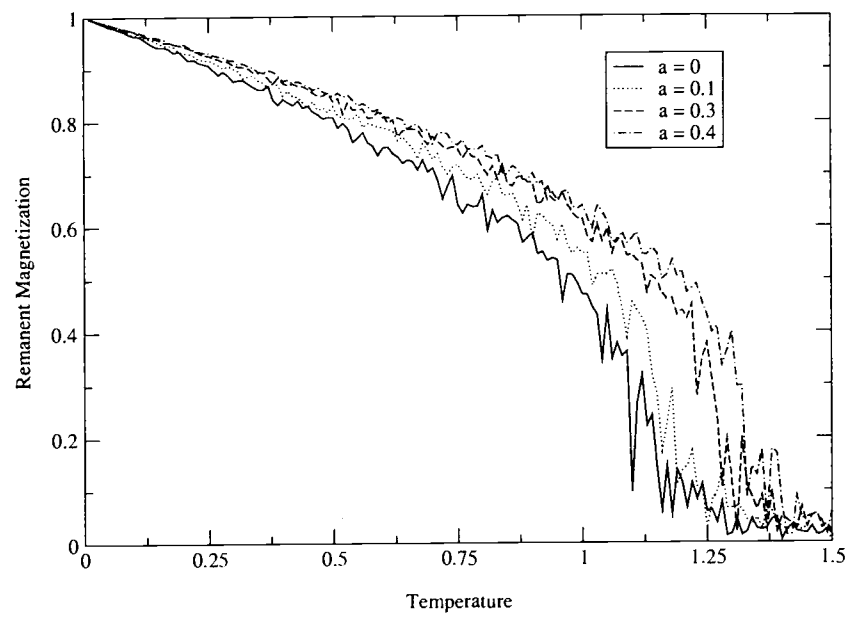


FIGURE 5.5. Magnetization vs. Temperature with no external field.



## 6. RESULTS OBTAINED WITH THE CODE DEVELOPED.

### 6.1. Critical Behavior

Even though the critical exponents cannot be calculated from the data obtained by our Monte Carlo simulation, some conclusions can be drawn:

Looking at figure 6.1 it is remarkable how similar in shape it is to experimental measurements of the remanent magnetization. All the characteristic of a real material are present, even the measurement noise near the critical temperature.

If we try to fit a power law to the points near the critical temperature, we arrive to the expression that (near  $T_C$ )

$$m \simeq 1.09 \times |1.21 - T|^{.54} \quad (6.1)$$

This value of  $\beta = 0.54$  is more than 40% larger than the one that can be found in table 2.2, but is at least of the same order of magnitude, which is the best we can ask if we take into account that the code was not optimized for the calculation of critical exponents. The critical temperature calculated from this data is  $kT_C = 1.20803$  while according to Weiss [13]  $k\Theta_C = 1.85$  for a simple cubic system.

Another conclusion that can be drawn is that the critical temperature increases with the anisotropy parameter, as can be seen in figure 6.2, but the actual dependence was not investigated any further.

### 6.2. Magneto-Crystalline Anisotropy Energy in the Case of a Complete Top Layer

To have an idea of the behavior of the thin film, we calculated the change in Magneto-crystalline anisotropy energy as a function of the temperature. This behavior can be seen in figure 6.3 and in figure 6.4, a detail at very low temperatures.

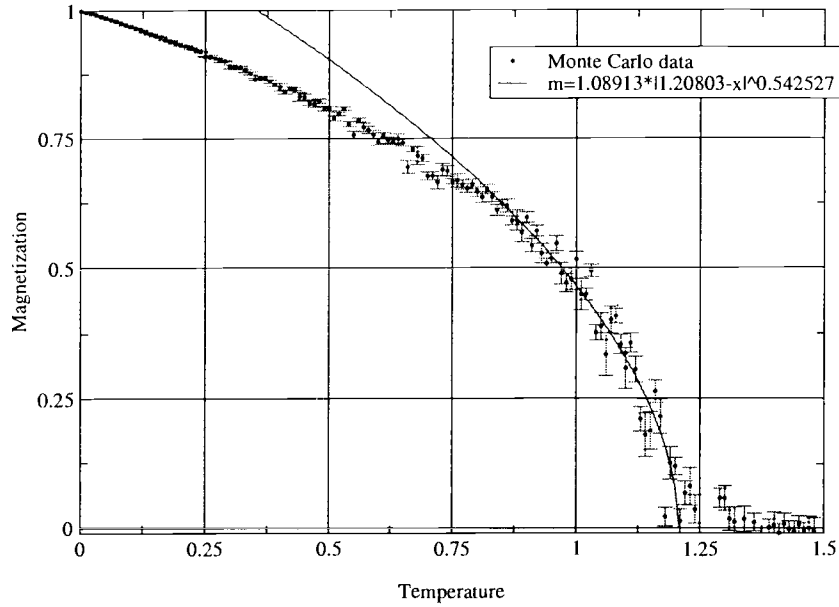


FIGURE 6.1. Critical behavior.

It can be appreciated that the MAE depends mostly linearly on the temperature, at least in the low temperature range and that it is mostly quadratic for the rest of the calculated range. We should remark that this behavior is congruent with the described by Carr [20], where for low temperatures the anisotropy energy increases linearly.

In the case that no external fields are present, the spins lay in the average, along the direction of the easy axis; this behavior is not significantly changed by increasing the temperature as without the external field to break the symmetry all sites are fundamentally equivalent. Vedmedenko [23] studied the magnetic microstructure of a monolayer of classical magnetic moments on a triangular lattice of about 10000 magnetic spins and with open boundary conditions. The Hamiltonian used was similar to ours, but included two extra terms  $K_1 \sum_i \sin^2 \theta \pm K_2 \sum_i \sin^4 \theta$ .

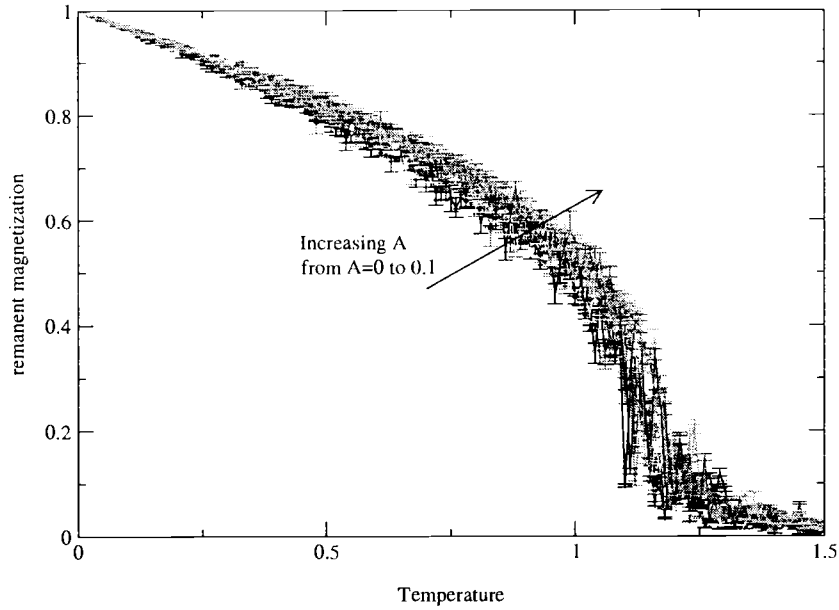


FIGURE 6.2. Zero field magnetization versus temperature for different values of the anisotropy parameter.

In this configuration exhibits a magnetic microstructure that is not present in our case. This microstructure depends on the constants  $K_1$  and  $K_2$  and it is shown that the domain walls broaden in the case  $K_1, K_2 \rightarrow 0$ . It is then reasonable to assume that when the Hamiltonian is reduced to the one we used, that no walls would be present and that the structure would be similar to ours.

In fig 6.5 we show the behavior of the anisotropy energy as a function of the remanent magnetization of the sample. It can be seen that MAE depends quadratically with the magnetization. This result can be understood if we consider that in the range of temperatures considered (well below the Curie temperature) the behavior of the magnetization is linear and the MAE is quadratic. From this result we can draw the inference that the saturation field is also linear with the temperature. This can be derived by considering the MAE as being roughly the area of

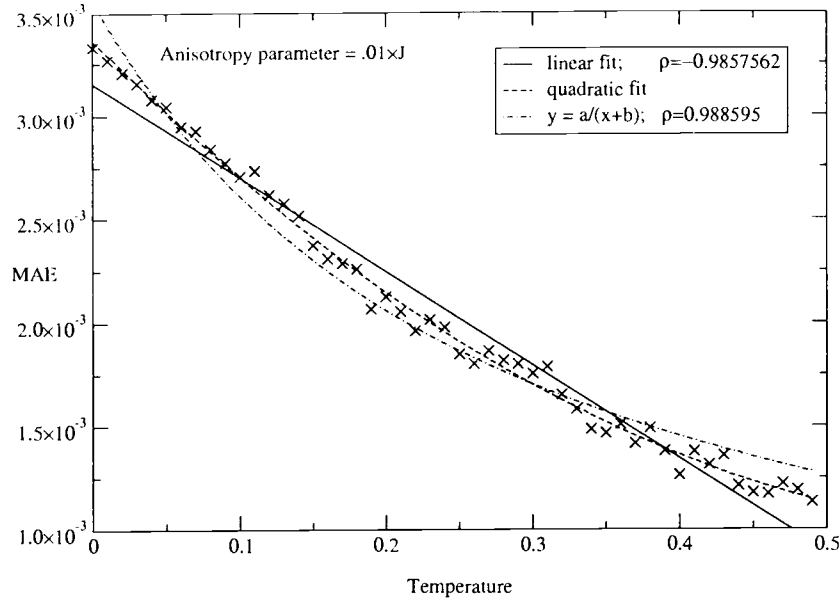


FIGURE 6.3. Magnetic anisotropy energy versus temperature (complete top layer).

a right triangle with sides given by the saturation field and the magnetization of the sample. As the saturation magnetization is about equal to the remanent magnetization along the easy axis,  $MAE \simeq H_S \times (M_R)/2 \propto T^2$  and we can conclude that

$$H_S \propto T \quad (6.2)$$

This conclusion seems to be confirmed by the simulation data, but we have no hard evidence to that effect.

### 6.3. Magneto-Crystalline Anisotropy Energy Calculations when the Top Surface is Incomplete

The first result that we noticed was that the direction of the easy and hard axis does not change with respect to the directions calculated for the complete top

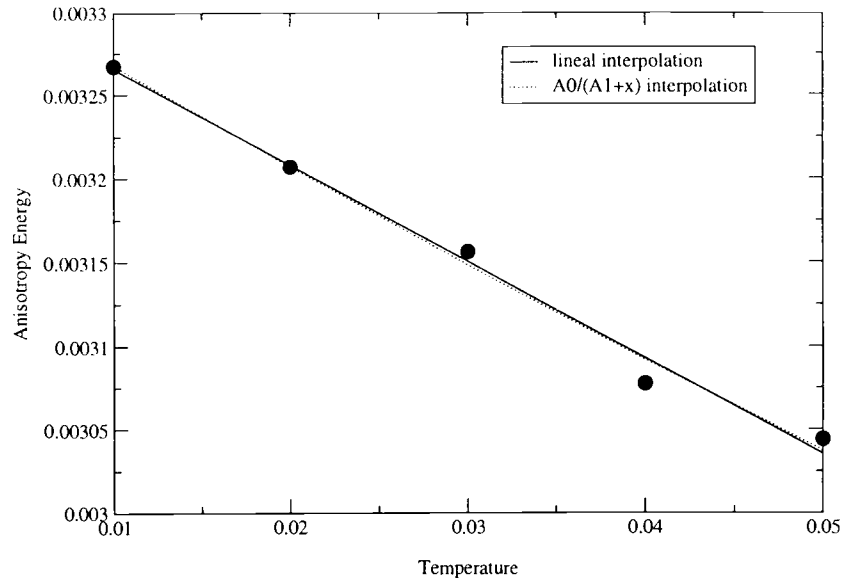


FIGURE 6.4. Low temperature behavior for the MAE (complete top layer).

layer, no matter where the defects were located. The spins will still point along the same direction if no external fields are present. This is due to the predominance of the bulk material over the influence of the top surface. The only effect we notice would be to make the easy axis a bit less "easy", making pulling the spins out of the plane a bit easier. This is then the reason for the smaller magneto-crystalline anisotropy energy observed, and its dependence with the number of atoms on the top layer.

The next step we took was to make sure that there was an equivalence between different random configurations. We did this for 2 different temperatures and coverages. We found that the MAE didn't change significantly and that the standard deviation was smaller than the error derived from the thermal agitation. In the figures 6.6 and 6.7 we show the behavior of the MAE when we change the position of spins on the top layer. We took measurements for 20 different configura-

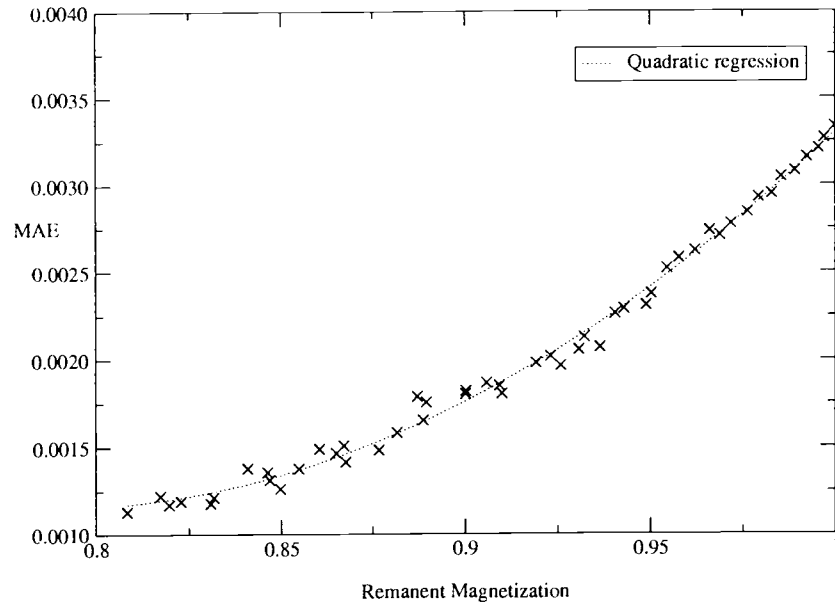


FIGURE 6.5. Magnetic anisotropy energy versus remanent magnetization (complete top layer).

rations and averaged the resulting energies. Two coverages were considered, 80% and 20% 6.8 and 2 different temperatures. In the case when the surface coverage was of 80%, the standard deviation of the mean was between .07 and .25% of the average value, significantly lower than the error due to the thermal excitation. This error is of about 4.3% for  $T = .01$ , and grows with temperature, as is expected from thermal excitations, so we can safely say that there is no need to calculate the energies for different random patterns and that only one sample will suffice.

This step verifies that the magneto-crystalline energy depends primarily on the average number of nearest neighbors the spins on the top layer have, and not in the particular positions. However it should be noted that these calculations were performed when the spins on the top layer were deposited in a random manner. The behavior changes when the atoms are arranged regularly. It is evident that

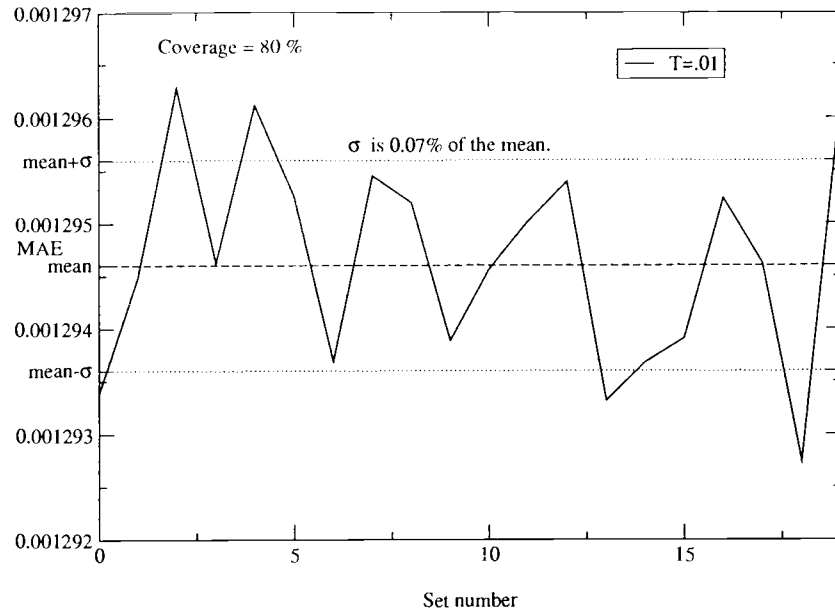


FIGURE 6.6. Behavior of the energy under changes in the random coverage of the top layer ( $T = 0.01$ ).

in the case when the coverage is less than 50% , there are configurations where the spins have only one neighbor, the one on the bottom layer. Other types of ordering will also change the average number of bonds, such as when all the spins are grouped in one section of the layer, or forming particular distributions, such as ridges or other regular coverages. In these cases it is expected that the MAE will depart from Taniguchi's result [19], and this is verified in the case of a regular distribution, although these changes are still within the expected error due to the temperature.

To verify the computations of MAE in the case where we have only one site occupied or only one site missing in the top layer, we performed a computation of MAE versus temperature for both cases. The results are shown in figure 6.9, where the energies are expressed in units of  $\text{MAE}(T = 0)$ , to avoid the differences

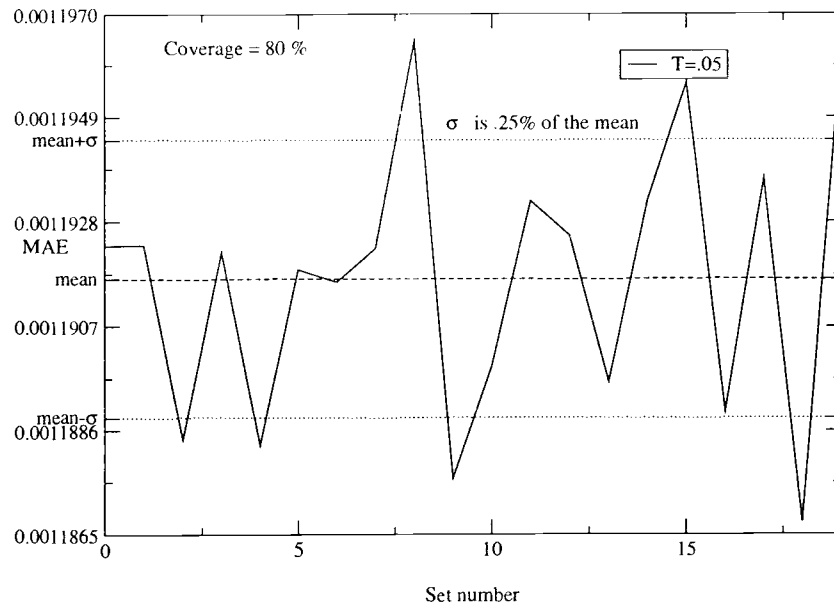


FIGURE 6.7. Behavior of the energy under changes in the random coverage of the top layer ( $T = 0.05$ ).

that are solely the result of a different number of complete layers in the system. As can be seen the MAE neatly converges to its  $T = 0$  result confirming the validity of our assumptions. The results of chapter 3 were also verified for the case when a significant portion of the surface was missing. These results, shown in figure 6.10, show that the energies calculated at  $T = 0$  by means of our approximation, are consistent with the Monte Carlo results to a very good degree.

An examination of figures 6.11 and 6.12 leads to the conclusion that, as it is to be expected, the magneto-crystalline anisotropy energy grows as a function of the portion of the surface that is missing, as can be expected. These results are also confirmed in figures 6.13 and 6.14. Here the Magneto-crystalline anisotropy energy was calculated as a function of the ratio defects to spins on the top layer. The results agree with the predictions made for the 0 temperature case where the



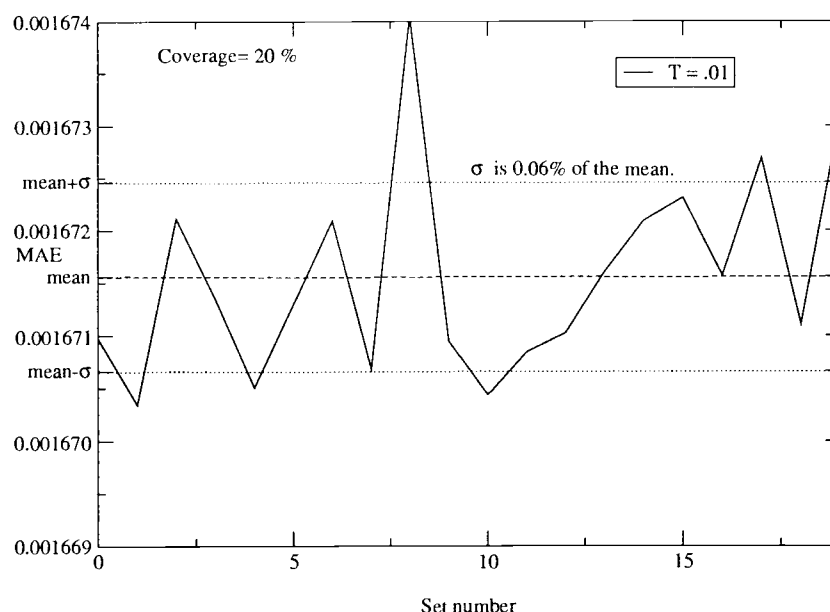


FIGURE 6.8. Behavior of the energy under changes in the random coverage of the top layer with a 20% coverage ( $T = 0.01$ ).

MAE was inversely proportional to this ratio. In the limit of very low temperatures the values predicted are also congruent with those found by the simulation.

We have found no adequate explanation for the noticeable steps that can be seen in those graphs. Our estimate is that they originate in the determination of the saturation point and that if we estimated it with better precision they would disappear. Another possibility is that there is a flaw in the integration subroutines, needing more data points for a better approximation of the energies<sup>1</sup>. The overall shape of the curves would tend to agree with the predicted results for 0-temperature

---

<sup>1</sup>We discovered after all the data for these graphs was collected that there was a small systematic error in the calculation of the energy integrals. However this did not change the fundamental shape of the curves, adding only a constant number to all the results.

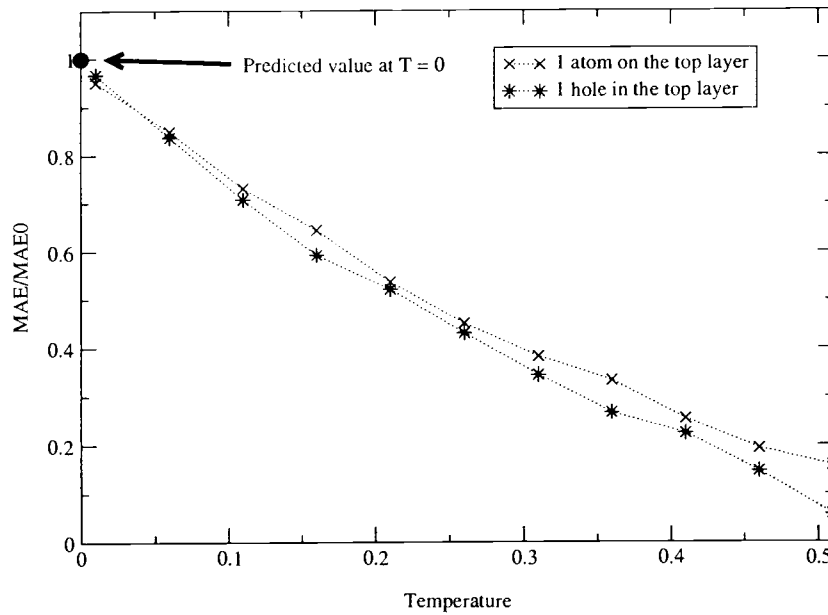


FIGURE 6.9. Magnetic anisotropy for the case where there is a single hole or spin on the top layer.

that, as can be seen in Chapter 3 is of the form:

$$MAE \propto \frac{1}{Ax + B} \quad (6.3)$$

where  $A$  and  $B$  depend on the dimensions of the sample and, possibly, on the temperature.

It has been shown that the anisotropy field (and hence the magneto-crystalline anisotropy in thin **Co-Ni-Fe-N** films increases with decreasing film thickness (see Kim *et al* [24]. Although they present no results for fractional thickness, that is when the top layer is increasing so that the thickness of the film increases from  $h$  to  $h + 1$  layers, their result seems to verify the behavior we see in this case. The same result was presented by Bottoni [25], but for **Co Cr Pt Ta/Cr V** thin films. The reader can also verify this experimental result in the study Wang *et al* [26]

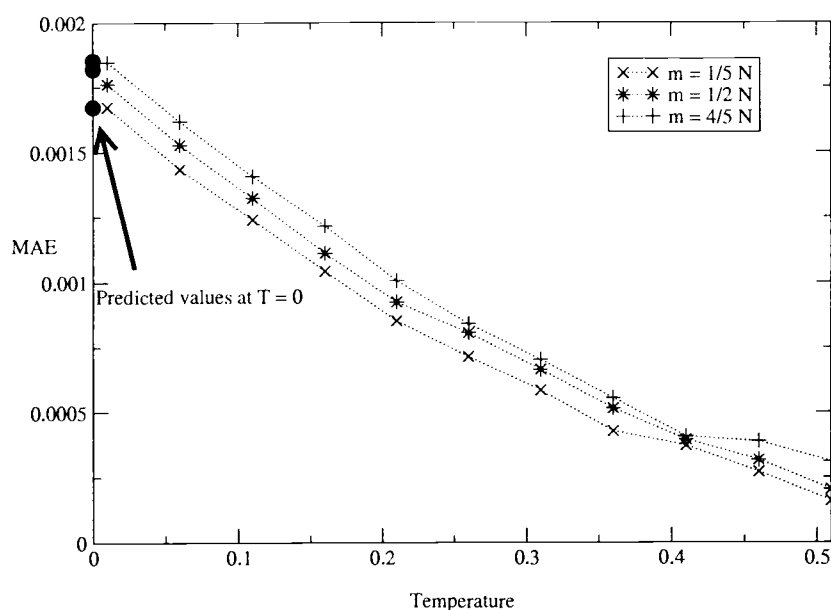


FIGURE 6.10. MAE vs. Temperature for different coverages of the top layer.

made of  $\text{La}_{0.88}\text{Sr}_{0.1}\text{MnO}_3$  films. In a study of films ranging from a thickness of  $100\text{\AA}$  (ultra thin) to  $2500\text{\AA}$  they found a remarkable increase of the anisotropy energy, that also verifies the notion that a fractional increase on the number of layers should reduce the magneto-crystalline energy.

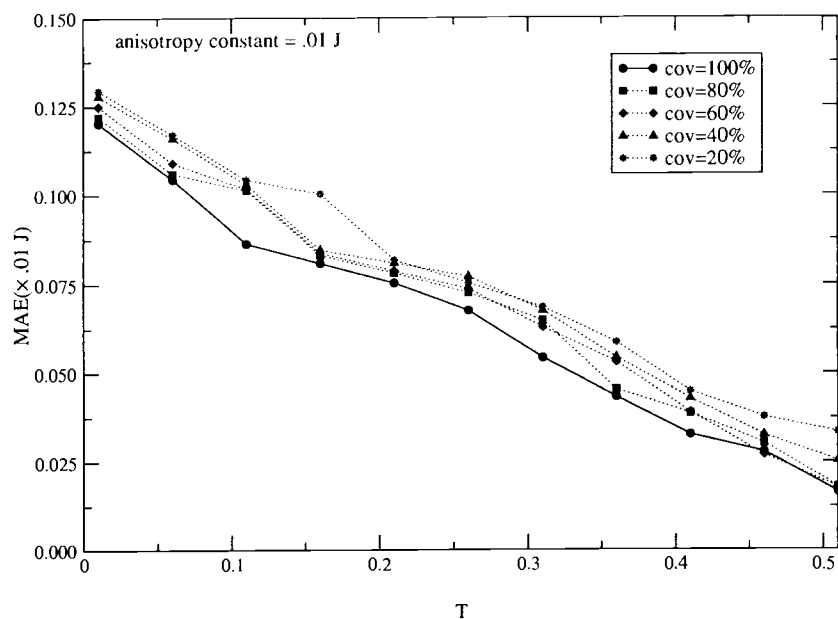


FIGURE 6.11. MAE as a function of the temperature for different surface coverages

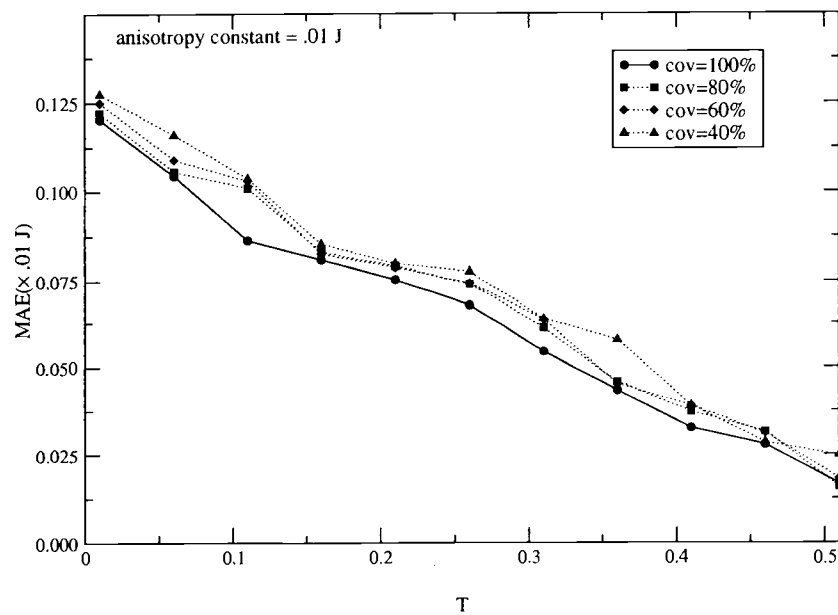


FIGURE 6.12. MAE as a function of the temperature for different random surface coverages

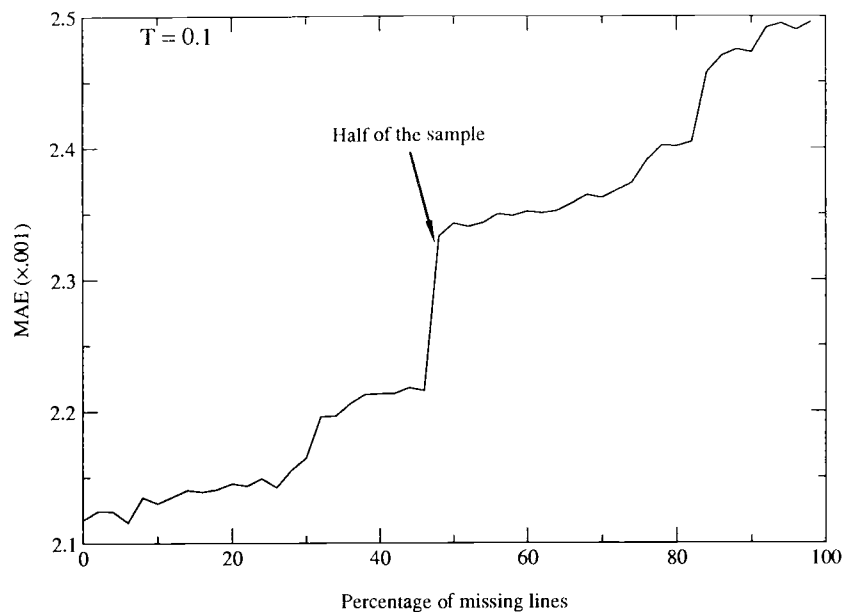


FIGURE 6.13. Magnetic anisotropy energy as a function of the missing lines.

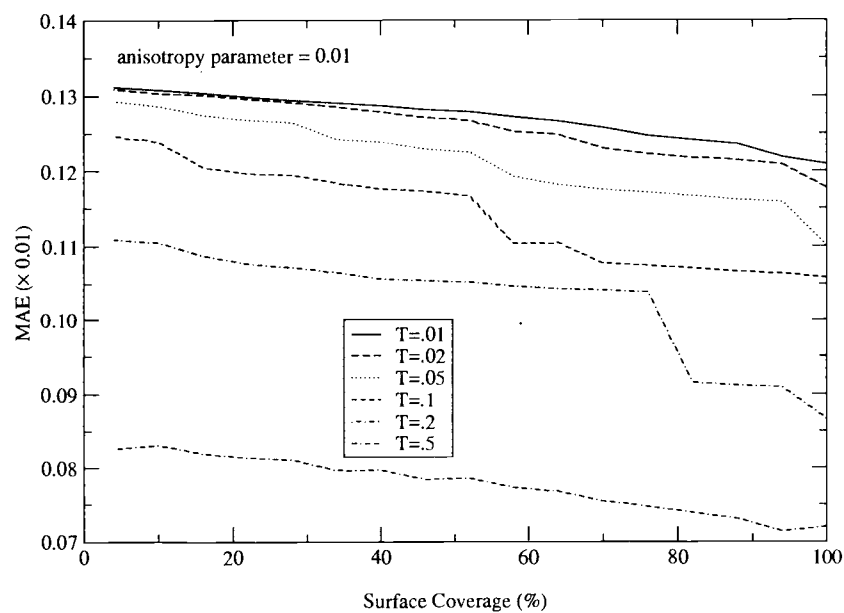


FIGURE 6.14. Magnetic anisotropy energy as a function of the coverage of the top layer for different values of the anisotropy parameter.

## 7. DISCUSSION, CONCLUSIONS AND FUTURE DIRECTION OF DEVELOPMENT

### 7.1. Discussion

In this chapter we will discuss the meaning of the results shown in Chapter 6.

The first important outcome of our research is the fact that the Curie Temperature of the film rises as the anisotropy constant increases, even if we couldn't find the exact functional form of this dependence. We have not found this result in the literature but we expect it to be of importance for the development of new magnetic materials.

Then we noticed the quadratic dependence of the MAE on the temperature and on the remanent magnetization. In this aspect we extend the result derived by Carr [20] who predicted that, at low temperatures, the temperature dependence of the MAE should be linear. This is indeed true for our system too but as we get closer to the Curie temperature, the functional relationship tends more and more to a parabola. This would indicate that the magnetic anisotropy is still present at higher temperatures than expected by previous calculations. Another conclusion that can be derived is that the technical saturation field is also proportional to the temperature.

The reason we didn't find any evidence of the existence of domains is twofold. First we have to remember that the system is not large enough to really develop them, and a larger sample must be explored. Also the periodic boundary conditions on the  $XY$  preclude the existence of any domains, as there is translational symmetry and therefore, as any site must be equivalent to any other, the spins are on the average parallel.

When the top layer was incomplete we verified the results that state that the particular disposition of the atoms shouldn't influence the calculation of the anisotropy energy. This result due to Taniguchi [19], was obtained for the case of a single layer of spins of two different classes, but it is applicable here if we consider one of the atom classes to have no magnetic moment. We determined that the difference between configurations is smaller than the thermal excitation error and hence too small to matter.

We also verified our own calculations regarding the anisotropy energy for the cases of a hole, a single atom and a step. In all these cases, the magneto-crystalline anisotropy tends to the correct value when the temperature tends to zero. That means that the spins tend to be aligned as the temperature decreases, as is to be expected, and as we presumed for the zero temperature calculations. These results were also verified for the calculation of the parallel magnetization ( $M_{||} = \mathbf{M} \cdot \hat{H}$ ) where we see the same kind of behavior we observed for zero temperature, with minima appearing at the correct values if the applied field was above a certain limit, determined by our equations.

Also we determined that the MAE grows as a function of the coverage of the top layer as our 0-temperature results predicted. For very low temperatures the simulation results are also consistent with the theory and confirm our assumptions. These results were obtained for the case where the disposition of the atoms was random and where they were ordered. As was expected there was a very small difference between them, although it is noticeable. This would support the hypothesis that a film with more surface defects is more anisotropic and would aid in the development of materials with higher anisotropy.

## 7.2. Conclusions

We have successfully developed the code for the treatment of thin films with defects and for the computation of the Magneto-crystalline Anisotropy Energy, using the Classical Heisenberg model with a Van Vleck anisotropy perturbation.

The critical behavior of the model is consistent with that of real materials, and it is shown that the critical temperature and exponent increases with the anisotropy parameter of the Hamiltonian, although this couldn't be quantified precisely, due to the poor precision of the Monte Carlo results near the critical point. As was mentioned before, the program itself should be modified extensively before more accurate computations of these parameters can be made.

The MAE is shown to be dependant on the temperature and the  $T = 0$  model, proposed in chapter 3, has been proven correct to a very good measure. Also it has been shown that it depends on the *number* of missing sites on the top layer, not in the particular disposition of said sites, except for the case when they are all grouped together, as the average distance is minimized in that case and that predisposes the system to a different kind of ordering on the surface.

## 7.3. Expected Direction of Future Development.

In this section we will discuss some of the possible directions that research using the algorithm and code developed in this work may take.

### 7.3.1. *Parallel Computations*

One of the ways of making the algorithm run significantly faster is to modify it to work in a cluster of parallel computers. This will have the advantage of



making larger system viable in terms of memory and speed. As it is now it takes an average of 8 hours to obtain one single measurement of the magneto-crystalline anisotropy energy, for a system of 15000 spins. It is logical to expect this time to be significantly reduced if instead of one workstation the load is distributed among a set of 4 or 16 working in parallel, each taking up a portion of the work.

### *7.3.2. 2nd Neighbors and Other Terms in the Hamiltonian*

As was mentioned for simplicity reasons we only included nearest neighbor interaction and only the first term in the Van Vleck multipole expansion. An easy way to obtain more accurate results is to add the second nearest neighbors to the energy expression. The disadvantage of this approach is the inclusion of more external parameters and making the process somewhat more involved. But the benefit would be seen when we treat samples with a very sparsely populated upper layer, as there would be a better chance that some interaction will be seen between the sites that are occupied.

Before adding any more terms to the Hamiltonian the program should be modified a bit. We have used a layer of vacancies to simulate the open boundary condition on the  $Z$  axis, for reasons of convenience. But if we are to consider interactions of longer range, then this is not enough of a buffer to avoid undesirable interactions between the top and bottom layers. Therefore it will be required to make the open boundary condition more explicit in the future.

To add more terms of the multipole expansion one has to remember that they decay in magnitude as powers of the interatomic distance, so it may be necessary to take a longer evolution time to see their actual effect on the calculated energies. This may prove to be too much for the program, as the other terms are orders

of magnitude larger and it is possible that the influence of these perturbations is smaller than the numeric capabilities of the average computer.

### *7.3.3. Roughness*

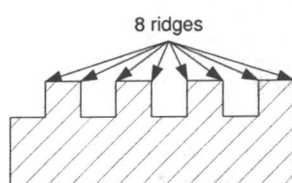
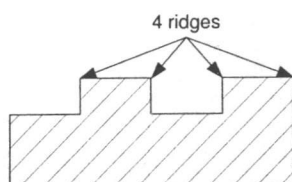
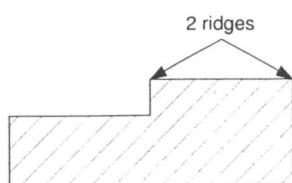
One interesting topic to research is that of the dependance of the magneto-crystalline anisotropy energy with the surface roughness. We can define the roughness parameter as being proportional to the number of ridges on the surface and inversely proportional to the surface covered. As is shown in figure 7.1 for the same coverage there can be different ways of covering the surface, and vice-versa. A preliminary calculation (not shown) indicates that the MAE would increase with the number of ridges (for constant surface coverage), but more work should be dedicated to this topic before advancing any conclusions.

### *7.3.4. Substitutions*

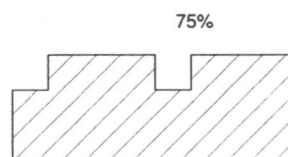
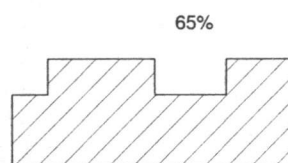
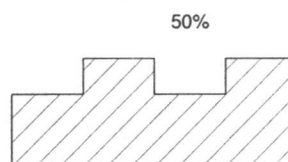
In all previous chapters the only defects considered were empty sites, or holes, in the top layer (surface) of the film. After the modifications suggested in section 7.3.2 it will be possible to treat substitutions, that is spins whose absolute values are not 1 or 0, without much trouble. The main difference will be that the matrix `def` (see APPENDIX C) will point to spins of some absolute value different from 0 (hole) or 1.

### *7.3.5. Other Geometries*

Apart from the simple cubic geometry used for this work, other crystal geometries are possible, just by a judicious definition of the defects matrix `def`



The upper layer coverage is constant, but the surface roughness increases with the number of ridges.



The number of ridges is constant, but the surface roughness decreases as the upper layer coverage increases.

FIGURE 7.1. Surface roughness as a function of the number of ridges and the area.

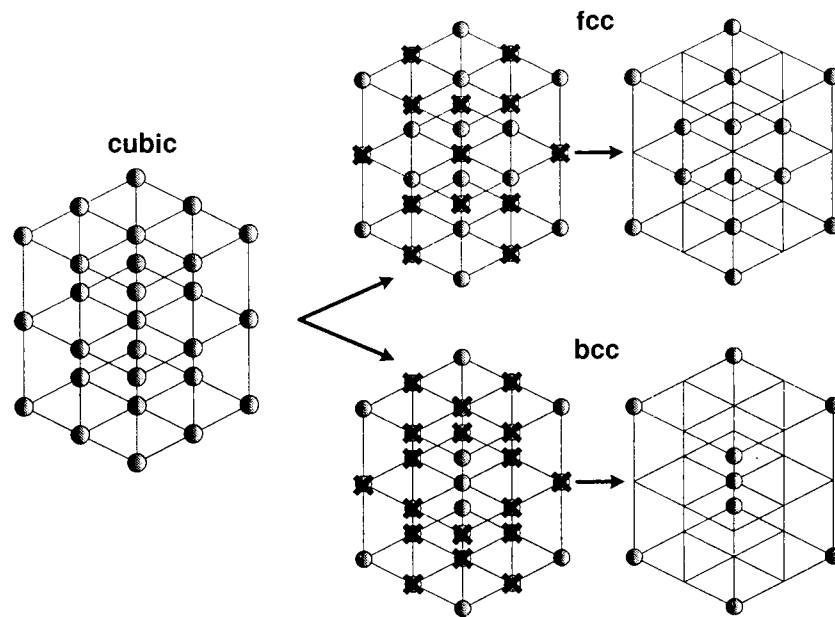


FIGURE 7.2. From simple cubic crystals to FCC and BCC.

(see APPENDIX C). We can characterize each site of the lattice by 3 integer numbers that denote its position in the  $(X, Y, Z)$  directions. Let's call these numbers  $(m_1, m_2, m_3)$ . If we mark as defects the ones in which  $m_1 + m_2 + m_3$  is odd we obtain an fcc crystal. The bcc crystal is a bit more complicated. First we have to eliminate all sites that have at least one of  $m_1, m_2$  or  $m_3$  even. Then for all of the remaining sites we have to add the one that is directly diagonal to it. What we have done basically is to consider a cubic lattice with a basis, a common way of viewing bcc lattices.

## BIBLIOGRAPHY

- [1] George T. Rado and Harry Suhl (Editors). *Magnetism, Volume 1*. Academy Press, 1963
- [2] Neil W. Ashcroft and N. David Mermin. *Solid State Physics*. Saunders College Publishing, 1976
- [3] Malcolm McCaig. *Permanent Magnets in Theory and Practice*. Halsted Press, 1977
- [4] G. M. Kalvius and Robert S. Tebble (Editors). *Experimental Magnetism*. John Wiley & Sons, 1979
- [5] J. J. Binney, N. J. Dowrick, A. J. Fisher and M. E. J. Newman. *The Theory of Critical Phenomena*. Oxford Science Publications, 1993
- [6] D. A. Lavis and G. M. Bell. *Statistical Mechanics of Lattice Systems. Vols. 1 and 2. Sec. Ed.* Springer, Texts and Monographs in Physics, 1999
- [7] D. P. Landau and K. Binder. *A Guide to Monte Carlo Simulations in Statistical Mechanics*. Cambridge University Press, 2000
- [8] E. Ising. *Z. Phys.* **31**, 253 (1925)
- [9] K. Honda and S. Kaya. *Sci. Rept. Tohoku Imp. Univ.* **15**, 721 (1926)
- [10] J. H. Van Vleck. *Phys. Rev.* **52**, 1178 (1937)
- [11] H. A. Kramers and G. H. Wannier. *Phys. Rev.* **60**, 252 (1941)
- [12] L. Onsager. *Phys. Rev* **65**, 117 (1944)
- [13] P. R. Weiss *Phys. Rev.* **74**, 1493 (1948)
- [14] L. Nel. *Compt. Rend.* **228**, 664 (1949)
- [15] C. N. Yang. *Phys. Rev* **85**, 809 (1952)
- [16] N. Metropolis, A. Rosenbluth, M. Rosenbluth, Teller A., Teller E. *J. Chem. Phys.* **21**, 1087 (1953)
- [17] C. Zener *Phys. Rev.* **96**, 1335 (1954)
- [18] C. P. Bean. *J. Appl. Phys.* **26**, 1381 (1955)
- [19] Taniguchi, S. *Sci. Rep. Tôhoku Univ.* **10**, 842 (1956)
- [20] W. J. Carr Jr. *Phys. Rev.* **29**, 436 (1958)

- [21] N. D. Mermin and H. Wagner. *Phys. Rev. Lett.* **17**, 1133 (1966)
- [22] A. M. Ferremberg and D. P. Landau. *Phys. Rev B* **44**, 5081 (1991)
- [23] E. Y. Vedmedenko, H. P. Oepen and J. Kischner. *J. Appl. Phys.* **89**, 7145 (2001)
- [24] Y. M. Kim, S. H. Han, H. J. Kim, D. Choi, K. H. Kim and J. Kim *J. Appl. Phys.* **91**, 8462 (2002)
- [25] G. Bottoni, D. Gandolfo and A. Cecchetti *J. Appl. Phys.* **91**, 8602 (2002)
- [26] Z. Wang, H. Kronmüller, O. I. Lebedev, G. M. Gross, F. S. Razavi, H. U. Habermeier and B. G. Shen. *Phys. Rev. B* **65**, 054411 (2002)

## APPENDIX

## APPENDIX A. Notes regarding critical exponents.

We will briefly discuss phase transitions and their relationship to critical exponents.

We say that a system has had a *phase transition* when it undergoes a precipitous change in one or more of its properties. The classical example is water that solidifies at  $273.15^\circ\text{K}$  and boils at  $373.15^\circ\text{K}$ <sup>1</sup>. These are examples of first order phase transitions, as the initial and final states are distinct, occur at separate regions of the thermodynamic configuration space and involve latent heat. In many first order transitions, such as the water-vapor transition, there is a region in which, if the parameters are just right, the latent heat vanishes. These values define the **Critical point**. For example, for the water-vapor transition  $\rho_c = 0.0323\text{g cm}^{-3}$ ,  $T_c = 647^\circ\text{K}$ . At  $T > T_c$  the water and the vapor cease to be separate entities. and the phase transition ceases to be of first order.

In a second order or *continuous* phase transition, the transition occurs between contiguous states in the thermodynamic configuration space. These phase transitions don't involve latent heat and are characterized by *order parameters*, thermodynamic quantities that exhibit divergent fluctuations. An example of this kind of transition is the ferromagnetic-paramagnetic transition that occurs at the **Curie Temperature**,  $T_C$  ( $T_C = 1043^\circ\text{K}$  for iron).

---

<sup>1</sup>These temperatures are measured at a pressure of one standard atmosphere



### A.1. Critical Exponents

The vanishing of the latent heat does not ensure that the specific heat changes as a smooth function of the temperature or that it is even finite. In fact it often diverges in the neighborhood of  $T_c$  as  $c \sim |T - T_c|^{-\alpha}$ . This number  $\alpha$ , that characterizes this part of the phase transition is called a *critical exponent*, and is one of a set of such exponents that describe the singular behavior of some interesting quantities in a continuous transition.

As a matter of fact  $\alpha$  can be defined in a more mathematically precise manner: Let  $\frac{d^n c}{dT^n}$  be the lowest derivative of  $c$  that diverges as a power of  $|T - T_c|$  in the limit  $T \rightarrow T_c$ , and let it diverge as  $|T - T_c|^{-k}$ , then  $\alpha = k - n$ .

Other thermodynamic quantities diverge at a continuous phase transition: For example in a ferromagnetic-paramagnetic transition the susceptibility  $\chi_T \equiv \partial m / \partial \mathbf{B} \sim (T - T_c)^\gamma$  and  $m_0$  tends to zero as  $m_0 \sim (T_c - T)^\beta$ . At  $T_c$  itself,  $m$  becomes proportional to a power of  $B$  specifically  $m \sim B^{1/\delta}$ . All these quantities have parallels in the liquid-vapor transition and indeed  $\beta$ ,  $\gamma$  and  $\delta$  are all critical exponents.

### A.2. The order parameter

We will define the *order parameter*,  $\phi$ , as a quantity whose thermal average on one side of the phase transition vanishes and moves away from zero on the other side.

The order parameter may fluctuate on both sides of the phase transition, but we are only interested in its thermal average, that is its value is averaged over a long period of equilibrium at constant temperature.

For example in a liquid-gas transition we will commonly use as the order parameter the difference  $\phi(\mathbf{x}) = \rho(\mathbf{x}) - \rho_{gas}(\mathbf{x})$ . Above  $T_c$  the order parameter fluctuates around zero, but below it fluctuates around a positive number. In a ferromagnetic-paramagnetic transition, the most adequate order parameter is the instantaneous mean magnetization in a small region near  $\mathbf{x}$ . Thus the order parameter is now a vector field instead of a simple function.

### A.3. Correlation functions

We define the *two-point correlation function* as the thermal average of the product of the order parameters evaluated at those two points.

$$G^{(2)} = \langle \phi(\mathbf{x})\phi(\mathbf{y}) \rangle \quad (\text{A1})$$

In general the two-point correlation function often depends only on the difference  $\mathbf{x} - \mathbf{y}$ , so it is customary to write it as:

$$G^{(2)} = \langle \phi(\mathbf{0})\phi(\mathbf{r}) \rangle \quad (\text{A2})$$

where  $\mathbf{r} = \mathbf{x} - \mathbf{y}$ .

As well below  $T_c$   $G^{(2)}$  becomes large for all values of the argument (due to the increased correlation length) we define the *connected two-point correlation function*:

$$G_c^{(2)} = \langle \phi(\mathbf{0})\phi(\mathbf{r}) \rangle - |\langle \phi \rangle|^2 \quad (\text{A3})$$

Above  $T_c$   $\langle \phi \rangle$  is zero so  $G_c^{(2)}$  is identical to  $G^{(2)}$  and both measure the degree of coordination of the order parameter at different points. Below  $T_c$   $G_c^{(2)}$  reflects only the fluctuations in the order parameter.

Experiments show that for  $\mathbf{r} \neq 0$ ,  $G_c^{(2)}(\mathbf{r})$  is small for both small and large  $T/T_c$ . Furthermore, when  $T = T_c$  and if  $r$  is large compared with intermolecular distances, we can write it in an asymptotic form:

$$G_c^{(2)}(r) \sim \frac{1}{r^{d-2+\eta}} \quad (\text{A4})$$

where  $d$  is the dimensionality of the system and  $\eta$  is another critical exponent. Away from  $T_c$  we have

$$G_c^{(2)}(r) \sim \exp(-r/\xi) \quad (\text{A5})$$

for  $r$  large and  $0 \neq |T - T_c|/T_c \ll 1$ . The characteristic length  $\xi$  is called the *correlation length*. The correlation length diverges as the system approaches  $T_c$  and it has been found empirically that:

$$\xi \sim |T - T_c|^{-\nu} \quad (\text{A6})$$

where  $\nu$  is yet another critical exponent.

#### A.4. *Universality*

It is surprising that systems as different as the liquid-gas transition and the ferromagnetic-paramagnetic transition have critical exponents that are identical within experimental error. This phenomenon, where very dissimilar systems exhibit the same critical exponents is called *universality*. One of the chief goals of the theory of phase transitions is to explain how different systems with very different physics yield the same critical exponents, for there is a paradox here: On one hand inter-atomic forces are responsible for the existence of phase transitions; on the other the details of these forces cannot play any role in determining the critical exponents, since these stay the same when the atoms, and therefore the interactions between them, change.

## APPENDIX B. Dipole-dipole coupling in the case of periodic boundary conditions

Let's consider an operator  $R$  that rotates a vector a solid angle  $\Omega$ . Let us examine how the different parts of the hamiltonian 2.22 change if we rotate all the spins by the same solid angle. Furthermore let's suppose that there is no external field present.

$$H\{\mathbf{s}_i\} = -J \sum_{\langle i,j \rangle} \mathbf{s}_i \cdot \mathbf{s}_j - A \sum_{\langle i,j \rangle} (\mathbf{s}_i \cdot \hat{r}_{ij})(\hat{r}_{ij} \cdot \mathbf{s}_j) \quad (\text{B1})$$

If we look at the first term of the equation, it is evident that

$$\begin{aligned} \mathbf{s}'_i \cdot \mathbf{s}'_j &= (R\mathbf{s}_i) \cdot (R\mathbf{s}_j) \\ &= \mathbf{s}_i R^{-1} \cdot R\mathbf{s}_j \\ &= \mathbf{s}_i \cdot \mathbf{s}_j \end{aligned}$$

it is invariant under rotations. On the other hand we can't say anything about the second term, except in the case when all the spins are parallel. Then:

$$(\mathbf{s}'_i \cdot \hat{r}_{ij})(\hat{r}_{ij} \cdot \mathbf{s}'_j) = (\mathbf{s}' \cdot \hat{r}_{ij})^2$$

if we remember that we are using a simple cubic lattice and summing over nearest neighbors, then

$$\sum_{\langle i,j \rangle} (\mathbf{s}_i \cdot \hat{r}_{ij})(\hat{r}_{ij} \cdot \mathbf{s}_j) = \sum_{\langle i,j \rangle} [(s'_x)^2 + (s'_y)^2 + (s'_z)^2] = n \quad (\text{B2})$$

so no matter where the spins are pointing, the result is always the same: the total number of spins. This is not true if in any direction we have open boundary conditions as one of the sums will not add to one.

## APPENDIX C. The Program and Subroutines

### *C.1. The Main Program*

In this part of the program all variables are initialized, the main symmetry axis are found as well as the technical saturation point.

#### *C.1.1. Variables*

- **pi**:  $\pi$ .
- **ndim**: Number of sites in the  $X$  and  $Y$  directions.
- **h**: Number of sites in the  $Z$  direction.
- **ndef**: Total number of spins.
- **jj**: Number of missing lines in the upper layer.
- **co**: Logical variable that indicates if there is need of a long thermalization time or not.
- **MaPl**: Matrix ( $ndim \times ndim \times h \times 3$ ) where the spins are stored.
- **yplane**: Four dimensional vector that contains the external parameters (field and anisotropy constant) when the This matrix will be used when the field is along the easy axis.
- **easy**: Direction of easy magnetization  $(\theta, \phi)$ .
- **MaPe**: Matrix ( $ndim \times ndim \times h \times 3$ ) where the spins are stored. This matrix will be used when the field is along the hard axis. field is along the easy axis.

- **yperp**: Four dimensional vector that contains the external parameters (field and anisotropy constant) when the field is along the hard axis.
- **hard**: Direction of hard magnetization  $(\theta, \phi)$ .
- **def**: Logical matrix ( $ndim \times ndim \times h$ ) that
- **dd**: Auxiliary vector for the definition of the holes. Contains the information about the presence of a spin or a hole.
- **T, beta**: Temperature;  $\beta = 1/T$ .
- **any**: Anisotropy constant.
- **B0**: Magnitude of the external field.
- **B0high, B0med, B0low**: Fields used in the search for the technical saturation field. The field found is stored in B0med.
- **X**: Initial direction.
- **mdif**: Difference in the magnetization on the easy and hard axis  $(\Delta M, \sigma_{\Delta M})$ .
- **edif**: Difference in the energy on the easy and hard axis  $(\Delta E, \sigma_{\Delta E})$ .
- **integral**: MAE.
- **m3**: Magnetization.
- **i, j, k, l, du, point**: Counters.
- **lim, te1, te2, vector**: Auxiliary variables

### C.1.2. Subroutines called

**init**( $n_{defects}$ , **dd**, **defects**, **Matrix**, **X**) Initializes a  $(n \times n \times h \times 3)$  matrix that contains all the spins of the system.

**Input**

- $n_{defects}$ : total number of spins.
- **dd**: auxiliary vector that contains the positions of the defects.
- **X**: initial direction of the spins.

**Output**

- **defects**: logical  $(n \times n \times h)$  matrix. If the site is occupied the corresponding value of the matrix is `.true.`, and `.false.` if it isn't.
- **Matrix**:  $(n \times n \times h \times 3)$  matrix that contains the spins of the system.

**fimin**(**Matrix**, **y**) Subroutine that finds the minimum of the energy by rotating all the spins together.

**Input**

- **Matrix**: System matrix.
- **y**: External parameters.

**Output**

- **Matrix**: Minimized (rotated) system matrix

**mag**(**Matrix**, **m**) Calculates the total magnetization (per spin) of the system.

**Input**

- **Matrix**: System matrix

**Output**

- **m**: 3 dimensional vector that contains the total magnetization per spin of the system.

**sph**(**V**) Transforms a cartesian vector into a spherical one.

**Input**

- **V**: Cartesian vector.

**Output**

- **V**: Spherical vector.

**func(Matrix<sub>E</sub>, Matrix<sub>H</sub>, y<sub>E</sub>, y<sub>H</sub>,  $\beta$ , defects, n<sub>defects</sub>, co,  $\Delta_m$ ,  $\Delta_e$ )**

Calculates the difference of energy and magnetization between the system in the hard axis and the system in the easy axis.

**Input** • Matrix<sub>E</sub>: System matrix in the easy axis.

- Matrix<sub>H</sub>: System matrix in the hard axis.
- y<sub>E</sub>: External parameters in the easy axis.
- y<sub>H</sub>: External parameters in the hard axis.
- $\beta$ :  $1/T$ , inverse of the temperature.
- defects: Matrix that contains the defects.
- n<sub>defects</sub>: Number of atoms not counting the defects.
- co: Logical variable that if `.true.` takes a long thermalization time and a short one if it is `.false..`

**Output** •  $\Delta_m$ : Difference between the magnetization in the hard and easy axis.

- $\Delta_e$ : Difference between the energy in the hard and easy axis.

**integration(vector,step,integral)** Integrates a function whose values are stored in vector.

**Input** • vector: Vector that contains the values of a function.

- step:  $\Delta x$

**Output** • integral.

### *C.1.3. Functions called*

**drand48** Pseudo-random number calculator



#### C.1.4. Output files

**energy.dat:** Evolution of  $\Delta_e$  with the external field.

**Mparallel.dat:** Evolution of  $\Delta_m$  with the external field.

**magna.dat:** Evolution of the technical saturation point with temperature.

**energy-vs-B0.dat:** Evolution of the energy with the remanent magnetization.

**energy-vs-T.dat:** Evolution of the energy with temperature.

**hard.dat:** Positions of the hard axis versus the number of missing lines.

**easy.dat:** Positions of the easy axis versus the number of missing lines.

#### C.1.5. The Main Program

```

Program Integral
implicit none
integer*4 h,i,j,l,k,ndef,jj,ti,du,ndim
parameter (ndim=50,h=7)
logical co,def(ndim,ndim,h)
Integer*4 dd(ndim*ndim*h,3), point
real*8 drand48, te1, te2
real*8 yplane(4),MaPl(ndim, ndim,h,3),beta,y(4)
real*8 yperp(4), MaPe(ndim, ndim,h,3),easy(2),hard(2)
real*8 T,any,pi,X(2),B0high,B0med,B0low,m3(3)
real*8 B0,mdif(2),edif(2),lim,vector(51),integral
common/cpi/pi
Open(11, File='energy.dat', Status='Old')
Open(12, File='Mparallel.dat', Status='Old')
Open(13, File='magna.dat', Status='Old')
Open(14, File='energy_vs_B0.dat', Status='Old')
Open(15, File='energy_vs_T.dat', Status='Old')
Open(16, File='hard.dat', Status='Old')
Open(17, File='easy.dat', Status='Old')
do i=1,51

```



c Find the easy and hard directions

cc

```

call fimin(Mapl,y)
call mag(Mapl,m3)
call sph(m3)
easy(1)=m3(2)
easy(2)=m3(3)
call fimax(Mape,y)
call mag(Mape,m3)
call sph(m3)
hard(1)=m3(2)
hard(2)=m3(3)
write(*,*) "hard",hard
write(*,*) "easy",easy
write(16,*) jj,hard
write(17,*) jj,easy
do 10101 T=0.01, .51,.05
write(*,*) "Starting T= ",T
if (T.ne.0) then
    beta=1/T
else
    beta=1d30
endif

```

cc

c Find the Technical Saturation point

cc

```

write(*,*) 'Searching for the saturation field'
co=.true.
B0=any
yplane(4)=any
yperp(4)=any
yplane(1)=B0
yperp(1)=B0
yplane(2)=easy(1)
yperp(2)=hard(1)
yplane(3)=easy(2)
yperp(3)=hard(2)
call func(Mapl,MaPe,yplane,yperp,beta,def,ndef,co,Mdif,edif)
lim=abs(mdif(1))+Mdif(2)*2
yperp(3)=0.0d0
yplane(1)=0.0d0
B0low=0

```

```

B0high=2*any/(h-1)
call init(ndef,dd,def,Mapl,easy)
call init(ndef,dd,def,Mape,easy)
do i=1,10
  B0med=(B0low+B0high)/2
  yplane(1)=B0med
  yperp(1)=B0med
  yplane(2)=easy(1)
  yperp(2)=hard(1)
  yplane(3)=easy(2)
  yperp(3)=hard(2)
  call func(Mapl,MaPe,yplane,yperp,beta,def,ndef,co,Mdif,edif)
  If (abs(Mdif(1)).le.lim+Mdif(2).or.Mdif(1).le.0)
    then
      B0high=B0med
      call init(ndef,dd,def,Mape,hard)
    else
      B0low=B0med
    endif
end do
write(*,*) 'Saturation Field = ',B0med
write(13,*) T,B0med
x(1)=0.0d0
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c Calculate the Magnetization curves
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
  call init(ndef,dd,def,Mapl,easy)
  call init(ndef,dd,def,Mape,easy)
  co=.false.
  do ti= 0,20
    B0=B0med/20*ti
    yplane(1)=B0
    yperp(1)=B0
    yplane(2)=easy(1)
    yperp(2)=hard(1)
    yplane(3)=easy(2)
    yperp(3)=hard(2)
    call func(Mapl,MaPe,yplane,yperp,beta,def,ndef,co,Mdif,edif)
    vector(ti+1)=Mdif(1)
    write(12,*) B0,Mdif
    write(11,*) B0,edif
  end do

```

```
call integration(vector,B0med/20,integral)
write(14,*) vector(1),integral
write(15,*) T,integral
write(11,*)
10101 write(12,*)
write(13,*)
write(14,*)
write(15,*)
end
```

## C.2. Initialization Subroutine

This subroutine initializes all the main variables and matrices. It first sets the system matrix with all its spins in the direction indicated by the vector  $x$  and modulus 1. Then it takes the position of the defects from the `dd` vectors and sets up the logical matrix `mdef`, and sets the corresponding spins in the system matrix to 0.

### C.2.1. Variables

**Input:**     • `dd`: Defects auxiliary matrix.

              • `ndef`: Total number of spins.

              • `x`: Initial direction.

**Output**     • `Ma`: System matrix after time evolution.

              • `mdef`: Energy after minimization  $(E, \sigma)$ .

**Internal**    • `ndim`: Number of sites in the  $X$  and  $Y$  directions.

              • `h`: Number of sites in the  $Z$  direction.

              • `m1`, `m2`, `m3`: Position of the site.

              • `i`, `j`, `k`, `l`: Counters.

### C.2.2. The subroutine

```
Subroutine init(ndef,dd,def,Ma,x)
Implicit none
Integer*4 ndim,h
Parameter(ndim=50,h=7)
Integer*4 m1,m2,m3, i, j, l, k
```

```

Integer*4 ndef, dd(ndim*ndim*h,3)
Logical def(ndim,ndim,h)
Real*8 Ma(ndim, ndim,h,3)
Real*8 x(2)
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c Start the Matrix
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
do 50 i=1,ndim
do 49 j=1,ndim
do 49 k=1,h
Ma(i,j,k,3)=cos(x(1))
Ma(i,j,k,1)=sin(x(1))*cos(x(2))
Ma(i,j,k,2)=sin(x(1))*sin(x(2))
End do
End do
End do
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c Initialize the defects
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
do 51 l=1,ndef
m1=dd(1,1)
m2=dd(1,2)
m3=dd(1,3)
def(m1,m2,m3)=.false.
Ma(m1,m2,m3,1)= 0.
Ma(m1,m2,m3,2)= 0.
Ma(m1,m2,m3,3)= 0.
End do
Return
End

```

### *C.3. Calculate the Magnetization and Energy differences.*

This subroutine invokes **matrix** twice to calculate the energy and magnetization in the hard and easy axis and then calculates the difference. First it transforms the **y** vector that is originally expressed in spherical coordinates into cartesian, as is required by **matrix**. After the energies and magnetizations are calculated, the subroutine calculates the differences and their standard deviations.

#### *C.3.1. Variables*

- Input:**
- Ma1: System matrix in the easy axis.
  - Ma2: System matrix in the hard axis.
  - def: Defects matrix.
  - y1: External parameters for the easy axis.
  - y2: External parameters for the hard axis.
  - beta: Inverse temperature.
  - ndef: Total number of spins.
  - x: Initial direction.
  - co: Logical variable that is **.true.** if the Matrix subroutine has not been used yet and **.false.** otherwise.
- Output**
- Ma1: System matrix in the easy axis after evolution.
  - Ma2: System matrix in the hard axis after evolution.
  - Mdif: Difference between the magnetization in the easy axis and the magnetization in the hard axis. The magnetization values refer to the two different systems.



- edif: Difference between the energy in the easy axis system and the energy in the hard axis system.

**Internal**    • ndim: Number of sites in the  $X$  and  $Y$  directions.

- h: Number of sites in the  $Z$  direction.
- B: External field.
- M1, M2: Magnetization in system 1 and 2 respectively.
- m3, m4: Auxiliary magnetization vectors.
- ene1, ene2: Energy in system 1 and 2 respectively.
- ha, ea: Normalized vectors in the hard and easy directions respectively.
- haha, eaea: Square norms.

### *C.3.2. Functions Called*

**dot(x,y):** Scalar product.

**Input:**    • x: 3 dimensional cartesian vector.

              • y: 3 dimensional cartesian vector.

### *C.3.3. The Subroutine*

```
subroutine Func(Ma1,Ma2,Y1,Y2,beta,def,ndef,co,Mdif,edif)
Implicit none
Integer*4 ndef,ndim,h,k
Parameter (ndim=50,h=7)
Logical co,def(ndim,ndim,h)
Real*8 y1(4),Ma1(ndim, ndim,h,3),M1(3,2),ene1(2)
Real*8 y2(4),Ma2(ndim, ndim,h,3),M2(3,2),ene2(2)
Real*8 beta,Mdif(2),edif(2),m3(3),m4(3)
Real*8 ha(3), ea(3), dot, eaea, haha, b
```

```

External dot
b=y1(1)
ea(1)=1
ha(1)=1
ea(2)=y1(2)
ha(2)=y2(2)
ea(3)=y1(3)
ha(3)=y2(3)
call cart(ha)
call cart(ea)
yy1(1)=b*ea(1)
yy2(1)=b*ha(1)
yy1(2)=b*ea(2)
yy2(2)=b*ha(2)
yy1(3)=b*ea(3)
yy2(3)=b*ha(3)
yy1(4)=y1(4)
yy2(4)=y2(4)
call matrix(Ma1,def,ndef,beta,y1,ene1,M1,co)
call matrix(Ma2,def,ndef,beta,y2,ene2,M2,co)
co=.true.
m3(1)=M1(1,1)
m3(2)=M1(2,1)
m3(3)=M1(3,1)
m4(1)=M2(1,1)
m4(2)=M2(2,1)
m4(3)=M2(3,1)
haha=dot(ha,ha)
eaea=dot(ea,ea)
Mdif(2)=0.0d0
Mdif(1)=dot(m3,ea)/sqrt(eaea)-dot(m4,ha)/sqrt(haha)
do k=1,3
  Mdif(2)=Mdif(2)+(ha(k)*M1(k,2))**2/haha
  Mdif(2)=Mdif(2)+(ea(k)*M2(k,2))**2/eaea
end do
Mdif(2)=sqrt(Mdif(2))
edif(1)=ene1(1)-ene2(1)
edif(2)=sqrt(ene1(2)**2+ene2(2)**2)
end

```

#### *C.4. The Main Subroutine*

This is where the Monte Carlo steps take place. The system is first brought to equilibrium making it evolve for a number of steps. If it is the first time it invoked, the thermalization process lasts for 5000 steps, otherwise just 500 is enough. After each step the energy is minimized with respect to the direction of the magnetization. This accelerates the convergence to a stable state, by reducing the influence of the isotropic term. Then 50 more measurements are taken in which the energy and magnetic moments are calculated.

##### *C.4.1. Variables*

**Input:**   • Ma: System matrix.

• def: Defects matrix.

• ndef: Total number of spins.

• beta: Inverse temperature.

• y: External parameters.

• co: Logical variable that indicates if this is the first time that thermalization occurs.

**Output**   • Ma: System matrix after time evolution.

• energ: Energy after minimization  $(E, \sigma)$ .

• Mav: Magnetization of the system  $(\mathbf{M}, \sigma)$ .

**Internal**   • ndim: Number of sites in the  $X$  and  $Y$  directions.

• h: Number of sites in the  $Z$  direction.

- m1, m2, m3: Position of the site to test.
- i, j: Counters.
- jcount: Counter that registers the number of actual changes in the energy.
- endi: Number of thermalization Monte Carlo steps.
- B, any: External parameters.
- fracoc: Fractional occupation.
- Moment: Magnetization of the system.
- del: Change in the energy.

#### *C.4.2. Subroutines called*

**fmin(Matrix, y)** Subroutine that finds the minimum of the energy by rotating all the spins together.

**Input**    • Matrix: System matrix.  
             • y: External parameters.

**Output**    • Matrix: Minimized (rotated) system matrix

**mag(Matrix, m)** Calculates the total magnetization (per spin) of the system.

**Input**    • Matrix: System matrix

**Output**    • m: 3 dimensional vector that contains the total magnetization per spin of the system.

**change(Ma,y,m1,m2,m3, $\beta$ ,del,Moment)** : Changes the spin of the site if it is favorable or if the Boltzmann factor is less than a random number.

**Input**    • Ma: System matrix.

• y: External parameters.

• m1, m2, m3: Position of the particular site to be tested.

•  $\beta$ : Inverse temperature.

**Output**    • Moment: 3 dimensional vector that contains the total magnetization per spin of the system.

• del: Change in energy.

#### *C.4.3. Functions called*

**Energy(Ma,y)** : Calculates the energy of the system.

**Input**    • Matrix: System matrix

**Output**    • y: External parameters.

#### *C.4.4. The subroutine*

```
subroutine matrix (Ma,def,ndef,beta,y,energ,Mav,co)
  Implicit none
  Integer*4 ndim, h
  parameter(ndim=50, h=7)
  Integer*4 m1, m2, m3, i, j
  Integer*4 jcount,ndef,step,endi
  logical def(ndim,ndim,h),co
  Real*8 Ma(ndim, ndim, h, 3), B(3), E, Mav(3,2), Eav(2)
  Real*8 beta, any, del, Moment(3)
  Real*8 Energy, fracoc, energ(2),y(4)
  external function energy
  step=0
  B(1)=Y(1)
  B(2)=Y(2)
  B(3)=Y(3)
```

```

any=Y(4)
fracoc=h*ndim*ndim
fracoc=1.0d0/(fracoc-ndef)
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c bringing to equilibrium
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
    call fimin(ma,y)
    call mag(Ma,moment)
    moment(1)=moment(1)/fracoc
    moment(2)=moment(2)/fracoc
    moment(3)=moment(3)/fracoc
    endi=5000
    if (co.eq..true.) then
        endi=500
    end if
    E=Energy(Ma,y)
    do i=1,endi
        step=step+1
        do m1=1,ndim
            do m2=1,ndim
                do m3=1,h
                    if (def(m1,m2,m3).eq..true.) then
                        call change(Ma,y,m1,m2,m3,beta,del,Moment)
                        E=E+del*fracoc
                    end if
                end do
            end do
        end do
    end do
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c Averaging
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
    Eav(1)=0.d0
    Eav(2)=0.d0
    Mav(1,1)=0.d0
    Mav(2,1)=0.d0
    Mav(3,1)=0.d0
    Mav(1,2)=0.d0
    Mav(2,2)=0.d0
    Mav(3,2)=0.d0
    jcount=0
    do j=1,50

```

```

do i=1,5
  do m1=1,ndim
    do m2=1,ndim
      do m3=1,h
        If (def(m1,m2,m3).eq..true.) then
          call change(Ma,y,m1,m2,m3,beta,del,Moment)
          E=E+del*fracoc
        end if
      end do
    end do
  end do
  end do
  jcount=jcount+1
  Eav(1)=Eav(1) + E
  Eav(2)=Eav(2) + E**2
  Mav(1,1)= Mav(1,1)+Moment(1)
  Mav(1,2)= Mav(1,2)+Moment(1)*Moment(1)
  Mav(2,1)= Mav(2,1)+Moment(2)
  Mav(2,2)= Mav(2,2)+Moment(2)*Moment(2)
  Mav(3,1)= Mav(3,1)+Moment(3)
  Mav(3,2)= Mav(3,2)+Moment(3)*Moment(3)
end do
Eav(1)=Eav(1)/(jcount)
Eav(2)=sqrt((Eav(2)-jcount*Eav(1)**2)/(jcount-1))
Mav(1,1)=Mav(1,1)/(jcount)
Mav(1,2)=sqrt((Mav(1,2)-jcount*Mav(1,1)**2)/(jcount-1))
Mav(2,1)=Mav(2,1)/(jcount)
Mav(2,2)=sqrt((Mav(2,2)-jcount*Mav(2,1)**2)/(jcount-1))
Mav(3,1)=Mav(3,1)/(jcount)
Mav(3,2)=sqrt((Mav(3,2)-jcount*Mav(3,1)**2)/(jcount-1))
energ(1)=Eav(1)
energ(2)=Eav(2)
Mav(1,1)=Mav(1,1)*fracoc
Mav(2,1)=Mav(2,1)*fracoc
Mav(3,1)=Mav(3,1)*fracoc
Mav(1,2)=Mav(1,2)*fracoc
Mav(2,2)=Mav(2,2)*fracoc
Mav(3,2)=Mav(3,2)*fracoc
9901 Format(i5,4f15.8)
Return
End

```

### *C.5. The Change Subroutine*

In this subroutine a vector from the main matrix is rotated by an arbitrary angle and the change in the energy of the system is calculated. If the change reduces the total energy, or if the Boltzmann factor is larger than some random number, the change is accepted. The first thing calculated is the aleatory rotation angle, and it is tested that it lies inside a segment of sphere, as is explained in Section 4.3.1

#### *C.5.1. Variables*

**Input:**     • M: System matrix.

- y: External parameters.
- m1, m2, m3: Position of the spin to be tested.
- beta: Inverse temperature.
- moment: Magnetization.

**Output**     • Ma: System matrix after a one spin change.

- del: Change in the energy.
- moment: Magnetization after the change

**Internal**     • ndim: Number of sites in the  $X$  and  $Y$  directions.

- h: Number of sites in the  $Z$  direction.
- mn1, mn2, mn3, mp1, mp2, mp3: Position of the neighboring sites.
- x1, x2, x3: The spin to be changed
- v1, v2, v3: Random change vector.



- de1, de2, de3: Difference between the original and the new spins.
- radius: Absolute value of the random vector.
- phi: Random angle.
- b1, b2, b3: External field.
- any: Anisotropy constant.
- W: Boltzmann factor.

### *C.5.2. Functions Called*

**drand48** Random number generator.

### *C.5.3. The Subroutine*

```

Subroutine change(Ma,y,m1,m2,m3,beta,DEL,Moment)
Implicit none
Integer*4 ndim, h
Parameter(ndim=50, h= 7)
Integer*4 m1,m2,m3,mn1,mn2,mp1,mp2,mn3,mp3
Real*8 Ma(ndim,ndim,h,3), y(4),radius,any
Real*8 drand48, pi, DEL,de1,de2,de3,W,Moment(3),phi,beta
Real*8 b1,b2,b3,x1,x2,x3,v1,v2,v3
common/cpi/pi
x1=Ma(m1,m2,m3,1)
x2=Ma(m1,m2,m3,2)
x3=Ma(m1,m2,m3,3)
de1=1.0d0
de2=1.0d0
de3=1.0d0
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c Find a vector in the permitted region
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
do while (de1*de1+de2*de2+de3*de3.ge.2.0d0)
    v3=2*drand48()-1
    radius=sqrt(1-v3*v3)

```



```

cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
  if ( DEL .le. 0 ) then
    W=1.0d0
  else
    W=exp(-DEL*beta)-drand48()
  end if
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c Change the spin if the Boltzmann factor is favorable
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
  if (w.ge.0) then
    Moment(1)=Moment(1)+de1
    Moment(2)=Moment(2)+de2
    Moment(3)=Moment(3)+de3
    Ma(m1,m2,m3,1)=V1
    Ma(m1,m2,m3,2)=v2
    Ma(m1,m2,m3,3)=v3
  else
    DEL=0.0d0
  end if
  return
end

```

### *C.6. Magnetization of the System*

This subroutine averages all the spins in the system to calculate the magnetization per spin.

#### *C.6.1. Variables*

**Input:**   • Ma: System matrix.

**Output**   • m: Magnetization.

**Internal**   • ndim: Number of sites in the  $X$  and  $Y$  directions.

- h: Number of sites in the  $Z$  direction.
- i, j, k, l: Auxiliary counters.
- co: Counter.
- V: Auxiliary vector.
- mod: Auxiliary vector's absolute value.

#### *C.6.2. The subroutine*

```

Subroutine Mag(Ma,m)
Implicit none
Integer*4 ndim,h,i,j,k,l,co
Parameter (ndim=50,h=7)
Real*8 Ma(ndim,ndim,h,3),V(3),m(3),mod
m(1)=0.d0
m(2)=0.d0
m(3)=0.d0
co=0
do i=1,ndim
  do j=1,ndim
    do k=1,h

```

```
      do l=1,3
        V(l)=Ma(i,j,k,l)
      End do
      mod=sqrt(v(1)**2+v(2)**2+v(3)**2)
      if (mod.gt.1.0d-10) co=co+1
      m(1)=m(1)+v(1)
      m(2)=m(2)+v(2)
      m(3)=m(3)+v(3)
    end do
  end do
  m(1)=m(1)/co
  m(2)=m(2)/co
  m(3)=m(3)/co
  Return
End
```

### C.7. Energy of the System

Calculates the total energy per spin of the system.

**Input:**   • Ma: System matrix.

          • y: External parameters.

**Output**   • Energy: Energy of the system.

**Internal**   • ndim: Number of sites in the  $X$  and  $Y$  directions.

          • h: Number of sites in the  $Z$  direction.

          • i, j, k, l: Counters.

          • inext, lnext, knext: Auxiliary counters.

          • co: Counter (number of spins).

          • V: Auxiliary vector

          • mod: Auxiliary vector V's absolute value.

          • E, e1: Intermediate energies.

#### C.7.1. The function.

```
Real*8 Function Energy(Ma,y)
Implicit none
Integer*4 ndim, h
Parameter (ndim=50,h=7)
Integer*4 i, j, l, inext, lnext
Integer*4 k, knext, co
Real*8 E, e1,mod,v(3)
Real*8 Ma(ndim,ndim,h,3),y(4)
E=0
co=0
do i=1,ndim
  do l=1,ndim
```

```

do k=1,h
  do j=1,3
    V(j)=Ma(i,l,k,j)
  end do
  mod=sqrt(v(1)**2+v(2)**2+v(3)**2)
  if (mod.gt.1.0d-10) co=co+1
  inext=i+1
  if ( inext .gt.ndim ) inext=1
  lnext=l+1
  if ( lnext .gt.ndim ) lnext=1
  knext=k+1
  if ( knext .gt.h ) knext=1
  e1=0
  do j=1,3
    e1=e1-V(j)*Ma(inext,l,k,j)
    e1=e1-V(j)*Ma(i,lnext,k,j)
    e1=e1-V(j)*Ma(i,l,knext,j)
    e1=e1-y(j)*V(j)
  end do
  E=E+e1-y(4)*V(1)*Ma(inext,l,k,1)
  E=E-y(4)*V(2)*Ma(i,lnext,k,2)
  E=E-y(4)*V(3)*Ma(i,l,knext,3)
end do
end do
energy=e/co
Return
End

```

### *C.8. The Minimization Subroutine*

This subroutine minimizes the energy of the system with respect to the direction of the total magnetization. It uses a simple Newton type scheme in the two dimensional space of the angles.

#### *C.8.1. Variables*

**Input:**    • M: System matrix.

             • ex: External parameters.

**Output**    • M: Matrix rotated to the position that minimizes the energy.

**Internal**   • ndim: Number of sites in the  $X$  and  $Y$  directions.

             • h: Number of sites in the  $Z$  direction.

             • small: Measure of smallness.

             • i: Counters.

             • imax: Maximum number of iterations.

             • fnew: Change in the energy

             • x: Initial direction

             • xnew: Direction of the Gradient

             • xti, xphy: Increment in the variables.

             • del: Increment.

             • grad: Gradient.

             • m3: Magnetization.



### C.8.2. Functions called

**delta(M, dir, ex:** Calculate the change in energy in the direction **dir**

**Input:**

- M: System matrix.
- dir: Direction of change.
- ex: External parameters.

**grad(M,ex,del)** Calculate the gradient

**Input:**

- M: System matrix.
- ex: External parameters.
- del: Increment.

### C.8.3. The subroutine

```

Subroutine fimin(M,ex)
  Implicit none
  Integer*4 imax,ndim,h
  Real*8 small
  Parameter (ndim=50,h=7 ,imax=500, small=1.d-7)
  Integer*4 i
  Real*8 M(ndim,ndim,h,3), ex(4), pi
  Real*8 Fnew, xnew(2), x(2), delta, del, grad, m3(3)
  Real*8 gr(2), mod, ti, phi, xti(2), xphi(2)
  External delta, grad
  Common/cpi/pi
  x(1)=0.0d0
  x(2)=0.0d0
  call mag(M,m3)
  del=pi/40.0d0
  do i=1,imax
    xti(1)=del
    xti(2)=0.0d0
    xphi(1)=0.0d0
    xphi(2)=del
  
```

```

gr(1)=delta(M,xti,ex)/del
gr(2)=delta(M,xphi,ex)/del
mod=grad(M,ex,del)
if (mod.gt.small) then
  xnew(1)=-del*gr(1)/mod
  xnew(2)=-del*gr(2)/mod
  fnew=delta(M,xnew,ex)
  if (fnew.ge.0) then
    del=del/2.0d0
  else
    call rotate(M,xnew)
    x(1)=x(1)+xnew(1)
    x(2)=x(2)+xnew(2)
    call orient(x(1),x(2))
  end if
  if (del.lt.small) goto 110
else
  goto 110
end if
end do
110 Continue
Return
End

```

### *C.9. The Maximization Subroutine*

This is fundamentally the same subroutine as `fimin`, but adapted to finding a maximum instead of a minimum.

#### *C.9.1. Variables*

**Input:**   • `M`: System matrix.

          • `ex`: External parameters.

**Output**   • `M`: Matrix rotated to the position that minimizes the energy.

**Internal**   • `ndim`: Number of sites in the  $X$  and  $Y$  directions.

          • `h`: Number of sites in the  $Z$  direction.

          • `small`: Measure of smallness.

          • `i`: Counters.

          • `imax`: Maximum number of iterations.

          • `fnew`: Change in the energy

          • `x`: Initial direction

          • `xnew`: Direction of the Gradient

          • `xti`, `xphy`: Increment in the variables.

          • `del`: Increment.

          • `grad`: Gradient.

          • `m3`: Magnetization.

### C.9.2. Functions called

**delta(M, dir, ex:** Calculate the change in energy in the direction **dir**

**Input:**   • M: System matrix.

      • dir: Direction of change.

      • ex: External parameters.

**grad(M,ex,del)** Calculate the gradient

**Input:**   • M: System matrix.

      • ex: External parameters.

      • del: Increment.

### C.9.3. The subroutine

```

subroutine fimax(M,ex)
  Implicit none
  Integer*4 imax,ndim,h
  Real*8 small
  Parameter (ndim=50,h=7 ,imax=500, small=1.d-7)
  Integer*4 i
  Real*8 M(ndim,ndim,h,3), ex(4), pi
  Real*8 Fnew, xnew(2), x(2), delta, del, grad, m3(3)
  Real*8 gr(2), mod, ti, phi, xti(2), xphi(2)
  External delta, grad
  x(1)=0.0d0
  x(2)=0.0d0
  comp=energy(M,ex)
  del=pi/40.0d0
  do i=1,imax
    xti(1)=del
    xti(2)=0.0d0
    xphi(1)=0.0d0
    xphi(2)=del
    gr(1)=delta(M,xti,ex)/del
  
```

```
gr(2)=delta(M,xphi,ex)/del
mod=grad(M,ex,del)
if (mod.gt.small) then
  xnew(1)=del*gr(1)/mod
  xnew(2)=del*gr(2)/mod
  fnew=delta(M,xnew,ex)
  if (fnew.le.0) then
    del=del/2.0d0
  else
    Call rotate(M,xnew)
    x(1)=x(1)+xnew(1)
    x(2)=x(2)+xnew(2)
    call orient(x(1),x(2))
  endif
  if (del.lt.small) goto 110
else
  goto 110
endif
end do
continue
return
end
```

### *C.10. Rotate a Matrix*

This subroutine rotates the system matrix as a whole.

#### *C.10.1. Variables*

**Input:**    • M: System matrix.

          • x: Angles of rotation.

**Output**    • M: Rotated system matrix.

**Internal**    • ndim: Number of sites in the  $X$  and  $Y$  directions.

          • h: Number of sites in the  $Z$  direction.

          • i, j, k: Auxiliary counters.

          • s: Auxiliary vector.

#### *C.10.2. Subroutines called*

**rot(x, y):** Rotate a single vector

**Input:**    • x: Vector to be rotated.

          • y: Angles of rotation.

**Output**    • x: Rotated vector.

#### *C.10.3. The subroutine*

```
Subroutine rotate(M,x)
  Implicit none
  Integer*4 h,i,j,k,ndim
  Parameter(ndim=50,h=7)
```

```
Real*8 x(2),M(ndim,ndim,h,3),s(3)
do i=1,h
  do j=1,ndim
    do k=1,ndim
      s(1)=M(k,j,i,1)
      s(2)=M(k,j,i,2)
      s(3)=M(k,j,i,3)
      call rot(s,x)
      M(k,j,i,1)=s(1)
      M(k,j,i,2)=s(2)
      M(k,j,i,3)=s(3)
    End do
  End do
End do
Return
End
```

### *C.11. Change in Energy by a Rotation*

Calculates the change of energy when there is a rotation of the whole matrix.

#### *C.11.1. Variables*

**Input:**     • M: System matrix.

              • ex: External parameters.

              • r: Rotation vector.

**Output**     • delta: Energy of the system.

**Internal**    • ndim: Number of sites in the  $X$  and  $Y$  directions.

              • h: Number of sites in the  $Z$  direction.

              • small: Measure of smallness.

              • i, j, k, l: Counters.

              • inext, jnext, knext: Auxiliary counters.

              • co: Counter (number of spins).

              • V, Vi, Vj, Vk: Auxiliary vectors

              • B: External field.

              • mod: Auxiliary vector V's absolute value.

              • E, el: Intermediate energies.

#### *C.11.2. Functions called*

**dot(x,y):** Scalar product.



**Input:**   • x: 3 dimensional cartesian vector.

          • y: 3 dimensional cartesian vector.

### *C.11.3. The function.*

```

Real*8 Function delta(M,r,ex)
Implicit none
Integer*4 i, j, l, ndim, inext, jnext,h
Integer*4 k, knext,co
Parameter(ndim=50,h=7)
Real*8 M(ndim,ndim,h,3), e2, e1, delta, small
Parameter( small=1.0d-10)
Real*8 v(3),vi(3),vj(3),vk(3),r(2),ex(4),dot,b(3)
Real*8 rv(3),rvi(3),rvj(3),rvk(3),d1(3)
Real*8 mod
External dot
delta=0.0d0
e1=0.d0
e2=0.d0
co=0
do i=1,ndim
  do j=1,ndim
    do k=1,h
      inext=i+1
      if ( inext .gt.ndim ) inext=1
      jnext=j+1
      If ( jnext .gt.ndim ) jnext=1
      knext=k+1
      if ( knext .gt.h ) knext=1
      do l=1,3
        B(l)=ex(l)
        v(l)=M(i,j,k,l)
        rv(l)=v(l)
        vi(l)=M(inext,j,k,l)
        rvi(l)=vi(l)
        vj(l)=M(i,jnext,k,l)
        rvj(l)=vj(l)
        vk(l)=M(i,j,knext,l)
        rvk(l)=vk(l)
      end do
    end do
  end do
end do

```

```
mod=sqrt(v(1)**2+v(2)**2+v(3)**2)
if (mod.gt.1.0d-10) co=co+1
call rot(rv,r)
call rot(rvi,r)
call rot(rvj,r)
call rot(rvk,r)
call diff(v,rv,d1)
e1=e1+dot(d1,B)
e2=e2+ex(4)*(v(1)*vi(1)-rv(1)*rvi(1))
e2=e2+ex(4)*(v(2)*vj(2)-rv(2)*rvj(2))
e2=e2+ex(4)*(v(3)*vk(3)-rv(3)*rvk(3))
end do
end do
delta=(e1+e2)/co
Return
End
```

### *C.12. Calculate the Gradient of the Energy by a Rotation*

It is necessary during the minimization (maximization) subroutine to calculate the gradient of the energy, with respect to the angles  $\theta$  and  $\phi$ .

#### *C.12.1. Variables*

**Input:**    • M: System matrix.

             • ex: External parameters.

             • del: Distance increment (in radians).

**Output**    • grad: Gradient's absolute value.

**Internal**   • ndim: Number of sites in the  $X$  and  $Y$  directions.

             • h: Number of sites in the  $Z$  direction.

             • xti: Increment in  $\theta$ .

             • xphi: Increment in  $\phi$ .

             • mod: Gradient's absolute value.

             • gr: Gradient.

#### *C.12.2. Functions Called*

**delta** Calculates the change in energy.

#### *C.12.3. The Function*

```
Real*8 function grad(M,ex,del)
Implicit none
```

```
Integer*4 h, ndim
Parameter(ndim=50,h=7)
Real*8 ex(4),M(ndim,ndim,h,3),xti(2),xphi(2),mod
Real*8 delta, gr(2), del
External function delta
xti(1)=del
xti(2)=0.0d0
xphi(1)=0.0d0
xphi(2)=del
gr(1)=delta(M,xti,ex)/del
gr(2)=delta(M,xphi,ex)/del
mod=sqrt(gr(1)**2+gr(2)**2)
grad=mod
Return
End
```

### *C.13. Integration Subroutine*

This subroutine integrates a function evaluated at regular intervals using the Simpson rule.

#### *C.13.1. Variables*

**Input:**   • data: Vector that contains the values of the function to be integrated, measured at regular intervals.

- int:  $\Delta x$ .

**Output**   • inte: Integral.

**Internal**   • ndim: Number of sites in the  $X$  and  $Y$  directions.

- h: Number of sites in the  $Z$  direction.
- i: Auxiliary counter.
- fim, fi, fip: Auxiliary variables.

#### *C.13.2. The subroutine*

```
Subroutine Integration(data,int,inte)
Implicit none
Integer*4 i
Real*8 inte,fim,fi,fip,int,data(51)
inte=0.0d0
fim=data(1)
do i=2,10,2
  fi=data(i)
  fip=data(i+1)
  inte=inte+fim+4*fi+fip
  fim=fip
end do
inte=inte*int/3
```

Return  
End

*C.14. Miscellaneous Subroutines.*

*C.14.1. Spherical to cartesian coordinates.*

**Input:**     • V: 3-dimensional vector in cartesian coordinates.

**Output:**    • V: 3-dimensional vector in spherical coordinates.

**Internal:**   • x, y, z: Auxiliary variables.

The Subroutine:

```

Subroutine cart(v)
  Implicit none
  Real*8 x,y,z,v(3)
  z=v(1)*cos(v(2))
  x=v(1)*sin(v(2))*cos(v(3))
  y=v(1)*sin(v(2))*sin(v(3))
  v(1)=x
  v(2)=y
  v(3)=z
  Return
End

```

*C.14.2. Cartesian to spherical coordinates.*

**Input:**     • V: 3-dimensional vector in spherical coordinates.

**Output:**    • V: 3-dimensional vector in cartesian coordinates.

**Internal:**   • mod, theta, phi: Auxiliary variables.

The subroutine calls `orient` to check that the angles are in the proper quadrant.

The Subroutine:

```

Subroutine sph(v)
Implicit none
Real*8 mod, theta, phi, v(3),pi
Common/cpi/pi
mod=sqrt(v(1)**2+v(2)**2+v(3)**2)
if (mod.lt.1d-10) then
    theta=0.d0
    phi=0.0d0
else
    theta=acos(v(3)/mod)
    if (abs(theta).lt.1.0d-10) then
        phi=0.0d0
        theta=0.0d0
    else
        if (abs(theta-pi).lt.1.0d-10) then
            phi=0.0d0
            theta=pi
        else
            phi=atan2(v(2),v(1))
        end if
    end if
end if
v(1)=mod
v(2)=theta
v(3)=phi
call orient(v(2),v(3))
Return
End

```

*C.14.3. Calculate the sum of 2 cartesian vector*

**Input:**     • x: 3-dimensional vector in cartesian coordinates.

             • y: 3-dimensional vector in cartesian coordinates.

**Output:**    • s: 3-dimensional vector in spherical coordinates.

**Internal:**   • i: Counter.

The Subroutine:



```

Subroutine sum(x,y,s)
Implicit none
Real*8 x(3),y(3),s(3)
Integer*4 i
do i=1,3
  s(i)=x(i)+y(i)
End do
Return
End

```

*C.14.4. Calculate the difference between 2 cartesian vectors*

**Input:**    • x: 3-dimensional vector in cartesian coordinates.

             • y: 3-dimensional vector in cartesian coordinates.

**Output:**    • s: 3-dimensional vector in spherical coordinates.

**Internal:**    • i: Counter.

The Subroutine:

```

Subroutine diff(x,y,s)
Implicit none
Real*8 x(3),y(3),s(3)
Integer*4 i
do i=1,3
  s(i)=x(i)-y(i)
end do
Return
End

```

*C.14.5. Check that the angular part of a spherical vector is in the correct direction*

**Input:**    • theta: .

             • phi: Azimuth angle.

**Output:**    • theta: .

- phi: Azimuth angle.

The Subroutine:

```

Subroutine orient(theta,phi)
Real*8 theta,phi,pi
Common/cpi/pi
if (theta.lt.0) then
    theta=-theta
    phi=phi+pi
end if
if (theta.gt.pi) then
    theta=2*pi-theta
    phi=phi+pi
end if
if (phi.lt.0) phi=2*pi+phi
if (phi.gt.2*pi) phi=phi-2*pi
if (abs(theta).lt.1.0d-10) then
    theta=0.0d0
    phi=0.0d0
end if
if (abs(theta-pi).lt.1.0d-10) then
    theta=pi
    phi=0.0d0
end if
Return
End

```

#### *C.14.6. Rotate a 3 dimensional vector*

**Input:**    • x: 3-dimensional vector in cartesian coordinates.

- r: 2-dimensional vector that contains the angles of rotation.

**Output:**    • x: rotated 3-dimensional vector in cartesian coordinates.

**Internal:**    • small: Parameter that is a measure of how small a number has to be to be considered 0.

The subroutine calls the auxiliary subroutines **cart**, **sph** and **orient**

The Subroutine:

```

Subroutine rot(x,r)
Implicit none
Real*8 x(3),r(2),pi,small
Common/cpi/pi
Parameter (small=1.0d-10)
call sph(x)
if (abs(x(1)).gt.small) then
  x(2)=x(2)+r(1)
  x(3)=x(3)+r(2)
else
  x(1)=0.0d0
  x(2)=0.0d0
  x(3)=0.0d0
end if
call orient(x(2),x(3))
call cart(x)
Return
End

```

*C.14.7. Calculate the scalar product between 2 cartesian vectors (Function)*

**Input:**    • x: 3-dimensional vector in cartesian coordinates.

             • y: 3-dimensional vector in cartesian coordinates.

**Output:**    • dot: Scalar product.

**Internal:**    • i: Counter.

             • d: Auxiliary variable.

The Function:

```

Real*8 Function dot(x,y)
Implicit none
Real*8 d,dot,x(3),y(3)

```

```
Integer*4 i
d=0.0d0
do i=1,3
  d=d+x(i)*y(i)
end do
dot=d
Return
End
```