Analysis of Ultra-narrow Ferromagnetic Domain Walls

by

Catherine Jenkins

Submitted to the Department of Materials Science and
Engineering in partial fulfillment of the requirements for
the degree of

Master of Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
[June 2004]
April 2004

Author ....
Department of Materials Science and Engineering
April 1, 2004

Certified by ....
Senior Lecturer
Department of Materials Science and Engineering
Thesis Supervisor

Certified by ....
Executive Officer and POSCO Professor of Metallurgy
Department of Materials Science and Engineering
Thesis Reader

Accepted by ....
Carl V Thompson II
Stavros Salapatos Professor of Materials Science and Engineering
Department of Materials Science and Engineering
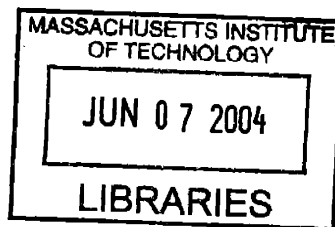
Analysis of ultra-narrow ferromagnetic domain walls

by

Catherine Jenkins

Submitted to the Department of Materials Science and Engineering
on April 1, 2004, in partial fulfillment of the requirements for the
degree of Master of Science in Materials Science and Engineering

Abstract

A mathematical analysis of ultra-narrow ferromagnetic domain walls was undertaken,
with graphical plots coded in the programming language TrueBASIC. An intrinsic inter-
atomic potential stemming from the breakdown of the continuum approximation of matter
is calculated and its contribution to the coercive force of hard materials is depicted. The
interaction of a very narrow domain wall with a similarly narrow planar defect is analyzed.
Time-dependent motion of such walls is modeled for various external driving forces and in
different combinations of material parameters.

This work was completed in parallel with a study of narrow crystallographic magnetic dis-
continuities known as twin boundaries, and was designed to gain an intuition into the con-
trol of high-anisotropy magnetic recording devices. The equations developed here would
be particularly useful as a basis for approaching the calculations of the stability of high-
density storage media.

Thesis Supervisor: David I Paul
Title: Senior Lecturer, Department of Materials Science and Engineering

Thesis Supervisor: Samuel M Allen
Title: Executive Officer and POSCO Professor of Metallurgy

This work would not have existed without the ever-renewing patience of David Paul.

To my parents, who were unfailing in their support of a project they didn't understand simply because they care.

To Jose, more than he knows. To Greyskull, Dangerhaus, and kittens, and for roommates who go on freedom crusades when I run out of time. To everyone from Worcester, who would know who they were if they ever read this, which they won't. To Lauren, for offering to sew my wedding dress. To Eric, for intending to visit, and to Foster, who finally made it. To Wendy, for the snacks.

To Professor Roylance, whose fault it isn't that plastics are the devil. To Toby, who wrote me a nice letter. To Sam, Bob, and Jorge, for the teaching over lunch. To all the frustrated souls on -c help who patiently fixed all of my problems and who collectively could take over the world if they got organized. This production was brought to you in part by support from Weekly Wednesdays and Black Diamond brand ice axes.

Table of Contents

List of Figures

Chapter 1

Introduction and Background

Ferromagnetic domain walls are of continuing importance to magnetic recording media and high-performance films. Mathematical analysis of narrow ferromagnetic domain walls was undertaken in order to quantify the degree of approximation inherent in the continuum model of a crystal lattice. In a very narrow domain wall of width $\pi(\delta/(l))$ equal to about nine (see Chapter 2, where δ is a normalized halfwidth equal to the square root of the ratio of the exchange energy to the anisotropy and l is the lattice spacing), the idealized picture of matter as having a uniform potential field begins to break down. This is mathematically represented by a change from integration to summation over the atomic energies. The geometry has one-dimensional functional variance in space but models a three-dimensional crystal by assuming a plane wall with magnetization distribution as follows:

The dependence of the angle of magnetization on position is conformationally analogous to a travelling soliton, of which helpful reviews can be found in Fogel et al. [1] and Currie et al. [2].

1.1 Of Philosophical Interest

Of primary concern in this exercise have been 180° ferromagnetic domain walls, where there is no crystallographic divide between regions of different anisotropy directions and the equations are thus determined only by the boundary conditions. It was shown in Quattrochi and Paul [3] that the contribution to the energy from the change in anisotropy direction is large when compared with the intrinsic energy or the energy from the material

defect, and so is neglected in order to isolate the effect of the narrowness of the wall. The possibility of multiple walls has also been discounted, since a domain wall, unlike a dislocation, can be stable without a complement of opposite "handedness" (see Nabarro [4]).

Quantum mechanics and quantum mechanical operators are largely ignored in this treatment, despite being the favorite of some very successful theoreticians (Winter [5], Krumhansl and Schrieffer [6], Fogel et al. [7]). For the time-dependent analysis, true perturbation theory is circumvented as in previous work by Paul [8] on the grounds that an external field would cause decisive rotation of the magnetizations and render a complex treatment unnecessary.

In defining the dynamic physical problem, one encounters the dilemma of whether to give the discrete lattice an intrinsic stiffness or to define the wall as possessing an intrinsic mass. Doering [9] proposed that the mass of a wall was inversely proportional to its thickness, which suggests that at least for a narrow wall, the mass therein dominates any inertial forces form the lattice. A revision of Doering's theory would not significantly affect the results here. It should also be noted that since the inertial term is dependent upon the kinetic energy, there is an implicit temperature dependence in the equations presented.

The interatomic potential was chosen to be sinusoidal for simplicity as Egami and Graham [10], while presumably maintaining to a large extent the detail seen in models employing the double-well potential (e.g. Krumhansl and Schrieffer [6]). Extensive use of the *Table of Integrals, Series, and Products* by Gradshteyn and Ryzhik [11] has been made in the calculations.

Below is a table of material parameters used in all graphs and numerical calculations. The unit system is cgs, in deference to magneticians. The relevant conversions are: 10 ergs/cm^3 = 1 Joule/m^3, 10^4 Oersteds [Oe] =Tesla [T].

Hypothetical materials Future 1 and Future 2 have parameters selected on the basis of overall averages of other hard ferromagnetic materials in terms of Curie temperature to determine the exchange and lattice spacing. Kittel [12] showed that the magnitude of the exchange was linearly related to T_C, and iron was chosen as the reference energy. However, an elevated value of the anisotropy energy will be needed to show the full range of these calculations, and so values of two and three times the highest commonly reported were chosen to balance the need for reasonably low coercivities in order to be processable by current methods and the mathematical requirement of a high enough anisotropy o demonstrate the attributes of the narrow wall.

The K parameter for the anisotropy, in units of energy density, is slightly misleading. According to Neumann's principle, materials properties (including magnetocrystalline anisotropy) have to have at least the symmetry of the material itself. This is written symbolically as a series representing the symmetry, of which the K_n are the fitting parameters. However, when considering only uniaxial anisotropy, as in this work, the numbers reported record not an absolute value but the difference in energy between the easy and hard directions.

Table 1.1: Material Properties, actual and projected

	SmCo$_5$	Sm$_2$Co$_{17}$	Nd$_2$Fe$_{14}$B	Fe	Future 1	Future 2
A [erg/cm]	1.9e-6	2.1e-6	1.1e-6	2.0e-6	1.5e-6	1.3e-6
K [erg/cm^3]	1.7e8	3.3e7	4.9e7	4.8e5	5.0e8	3.0e8
spacing [10^{-8} cm]	5	8.4	8.8	2.9	7.0	5.0
M	700	1030	700	1200	800	800

A is the coefficient of the exchange energy, linearly related to the Curie temperature of the material, typically on the order 2×10^{-6}. K is more properly K1, the anisotropy energy in the easy direction, to some degree forcing each material to be cubic by declaring the K independent of direction. The units of M are such that $MH\cos\Theta$ has units of energy density in the Hamiltonian to be defined.

The coefficient of kinetic energy is taken for all materials to be 6×10^9, as a compromise between the papers of Enz [13]and Doering [9].

1.2 Structure of the Thesis

This thesis will consist of three main divisions: first, the intrinsic interatomic potential of a narrow domain wall is calculated to quantify the effect of a periodic atomic impedance to wall motion; second, the interaction of such a narrow domain wall with a similarly narrow planar defect is analyzed to find consistent static solutions leading to a coercive force; finally, an *ad hoc* viscosity is added to account for damping in the time-dependence of the motion, and then said motion is calculated and plotted.

Also included are a basic introduction to the mathematics used in this paper and fully commented copies of the TrueBASIC code.

Chapter 2

Properties due to Effective Discreteness

Beginning with a Hamiltonian that includes terms for the exchange and anisotropy, a solution is obtained in a manner similar to Paul [8]. Substituting this solution back in to the energy formulation of each atom and summing gives a mathematical series, one term of which can be physically interpreted as the energy arising from the discreteness of the lattice. This is where the moving domain wall feels the effect of individual atoms instead of the bulk continuum potential that has heretofore been so successful in materials with much wider walls.

Egami and Graham [10] have noted that "the potential barrier can be expressed as a single sinusoidal function of the position of the wall, with the periodicity of the interlayer spacing," and determined that true energy minimization always leads to the a reasonably narrow wall being centered between atoms, rather than evenly on top of those sources of potential.

2.1 Energy

Time-dependent terms are for the moment ignored, and as such we are left with an energy Hamiltonian on each atom equal to

$$\Gamma_i = A\left(\frac{\Delta\theta}{\Delta x_i}\right)^2 + K(\sin\theta_i)^2 \tag{2.1}$$

A is the coefficient of the nearest-neighbor exchange, and K is the magnitude of the anisotropy. A differential change of magnetization angle with respect to position is inappropriate here, since the limit cannot be taken with the distance between points approaching zero. The careful reader will also have noticed that the units on both terms are a volumetric energy density, where certain other authors report energies in areal density

[erg/cm^2]. This was intentional, as it becomes more relevant in our case to consider the total volume between atoms as well when an areal density would only have accounted for the value at each atom and not through the space between.

Finding the minimum of the similar Euler equation with respect to θ by setting the variation in Γ equal to 0 gives

$$\theta_i = \text{asin}\left(\text{sech}\frac{x_i - x_0}{\delta}\right), \tag{2.2}$$

where the derived characteristic length $\delta = \sqrt{\frac{A}{K}}$. The wall width, or the region over which the majority of the magnetic rotation takes place, can be taken as $\pi\delta$. The shift factor x_0 accounts for a wall centered close to an atom other than at the origin. This solution can not blindly be taken as true, as one must be careful in computer coding to use the correct value of the inverse sine, but the function produces a curve of intuitively correct shape. It is also possible to use the equivalent solution presented in Bishop and Lewis [14], of the form $\text{atan}(\exp(\pm\omega x))$, where ω is some frequency under a Lorentz transformation. Such a transform was deemed a needless complexity for this work, where domain walls are always assumed to be moving at much less than the speed of light. In another simplification, this work will consider only domain walls with boundary conditions enforcing a rotation between zero and $180°$ as previously mentioned, but a simple scaling factor is all that would be required to bring the solution to complete generality,

$$\theta_i = \Theta_A + \frac{\Theta_B - \Theta_A}{\pi} \cdot \text{asin}\left(\text{sech}\frac{x_i - x_0}{\delta}\right) \tag{2.3}$$

The center of the domain wall is taken to be at $x = 0$ with no loss of generality in the reference frame being considered. Substituting our solution for the differential equation back in to the Hamiltonian, it reduces to

$$\Gamma_i = 2K\left(\operatorname{sech}\frac{x_i - x_0}{\delta}\right)^2 \tag{2.4}$$

In summing over all space we employ Poisson's sum formula which states

$$\sum_i \Gamma_i = \sum_n 2K\left(\operatorname{sech}\left(\frac{(n+\varepsilon)l}{\delta}\right)\right)^2 = \sum_s \int_{-\infty}^{\infty} \Gamma(n)\cos(2\pi sn)\,dn \tag{2.5}$$

The offset ε is added to allow for a non-integer equilibrium position of the domain wall, and is retained throughout the calculations as the basis by which to take the derivative in converting between energies and forces. With a change of position variables to a temporary $p = (n+\varepsilon)l$, $dp = l\,dn$, carefully noting the cancellation of odd terms in an integration over all space, we obtain an expression for the total energy of the domain wall as

$$\sum_s \frac{4\pi^2 sK\delta^2}{l^2 \sinh\left(\frac{\pi^2 s\delta}{l}\right)} \cdot \cos 2\pi s\varepsilon \tag{2.6}$$

of which the $s = 0$ term gives the expected continuum contribution, calculated previously, or $\dfrac{4\sqrt{AK}}{l}$, and the $s = 1$ term gives

$$\frac{4\pi^2 K\delta^2}{l^2 \sinh\left(\frac{\pi^2 \delta}{l}\right)} \cdot \cos 2\pi\varepsilon \tag{2.7}$$

which can be interpreted as the energy arising from the intrinsic potential that a domain wall must overcome in order to move. The $s > 1$ terms are vanishingly small, with the linear dependence on s of argument of the sinh in the denominator. It is apparent from the implicit dependence of δ on K that it should be possible and informative to plot the intrinsic wall energies as direct functions of K, for which see Figure 2.1. While $\delta^2 = \dfrac{A}{K}$, and it is possible to simplify the term to a dependence on A alone, the dependence on K is

15

retained, again in the argument of the hyperbolic function. It is also evident from the pair-
ing of δ and the lattice spacing l that it is the relative and not the absolute wall width
which has any bearing on the transition between a material that is well approximated by
the continuum model and a material that requires intrinsic terms.
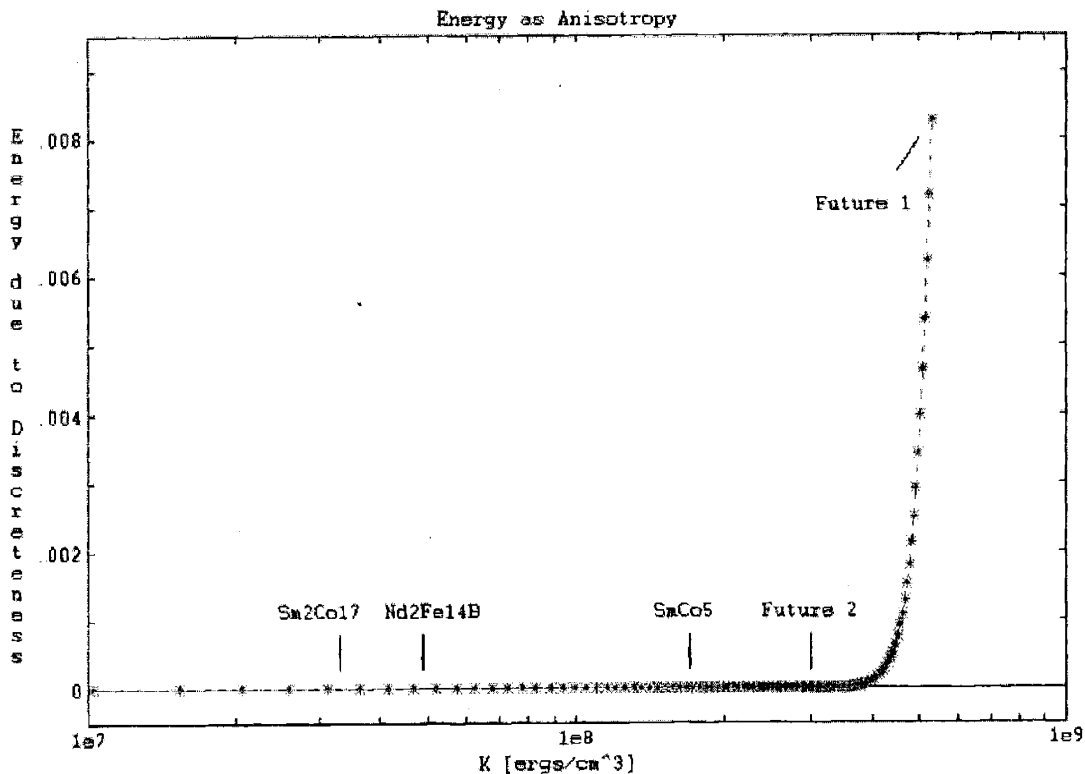
Figure 2.1: Maximum energy inherent in an atomically periodic (as opposed to uniform)
lattice potential as a function of the anisotropy energy K.

Several interesting aspects of this can already be extracted, namely that for a domain
wall any wider than about four lattice constants, as determined for the relevant material,
the contribution from the discreteness becomes largely negligible (See Figure 2.2)

16

Max total energy vs wallwidths at varying external field

+ H=0 [Oe]   *   H=5   ◇   H=50
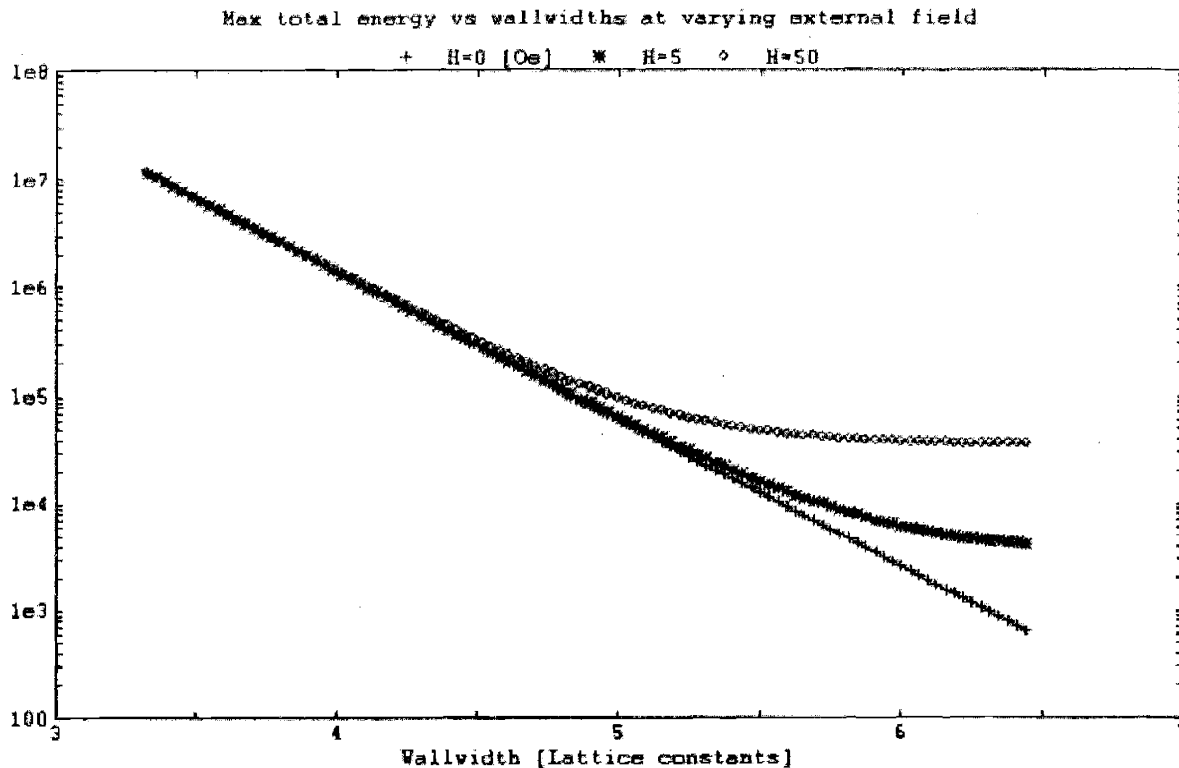
Figure 2.2: The cutoff between energies considered "high" and negligible is taken where the lines converge, or where the magnitude of the field becomes less important than the potential inherent in the lattice. The Hamiltonian is on the ordinate axis and includes terms for applied field, anisotropy, and exchange, but excludes time-dependence.

2.2 Coercivity

We determine a coercivity due to the discreteness and, below a wall width of a few lattice constants, find it to be substantial (See Figure 2.3). Coercivity is defined here to be the maximum field in which there are static solutions to the problem, e.g., the position of the wall does not tend to infinity. This is equivalent to finding the field required to make the Zeeman term of $M \bullet H$ equal in magnitude to the intrinsic lattice energy. It can also be thought of as the maximum depth of the potential well due to the atoms. Any external field of magnitude less than the coercivity will instead have the effect of starting the wall oscillating within the atomic potential well.

For a similar treatment of dislocations in metals, see also Nabarro [4].

Analogous to Eqn. 1.1, the Hamiltonian is now defined as

$$\Gamma_i = A\left(\frac{d\theta}{dx_i}\right)^2 + K(\sin\theta_i)^2 - HM_i\cos\theta_i. \qquad (2.8)$$

Summing over all space in a 2π periodic medium and taking the derivative with respect to the dimensionless ε representing the position in the argument of the cosine, one obtains a minimum H-field due to discreteness of

$$H_c = \frac{8\pi^3 K\left(\frac{\delta}{l}\right)^2}{M\sinh\left(\frac{\pi^2\delta}{l}\right)}. \qquad (2.9)$$

The fit of the experimental data to the following empirical equation is remarkable. In some highly abstract sense, the coercivity can be thought of as showing the mass, or inertial strength of the wall. Kittel and Galt [15] showed this to be inversely proportional to the thickness of a wall in a continuum model, although here we see that the dependence has transitioned to the form $1/r^2$.

$$H_c = 10^6 \times \exp\left(\frac{-4\pi}{3} \cdot \frac{\delta}{l}\right) \qquad (2.10)$$

For comparison, consider the poor fitting of the energies to a line based on what was once considered to be the sole dominant parameter, the absolute wall width, without correcting for the lattice parameter of the material (Figure 2.4). An intriguing feature of this plot is the relative magnitudes of neodymium--iron--boron and SmCo5. The samarium cobalts have long been considered ferromagnetically "harder" than Nd2Fe14B, and so we confirm that the relative wall width over lattice spacing is indeed the dominant parameter in some materials. In others, domain wall pinning is the primary mechanism of coercivity, not nucleation as above, and no theory is proposed for these.

Real Materials: Least Squares fitting

Future 1, K=50 MJ/m^3

Future 2, K=30 MJ/m^3

Nd2Fe14B K=4.9 MJ/m^3

SmCo5 K=17 MJ/m^3

$$H\_crit = \frac{4*PI^2*K*(delta/spacing)^2}{M*SINH(PI^2*(delta/spacing))}$$

Sm2Co17 K=3.3 MJ/m^3

Wallwidth delta/spacing
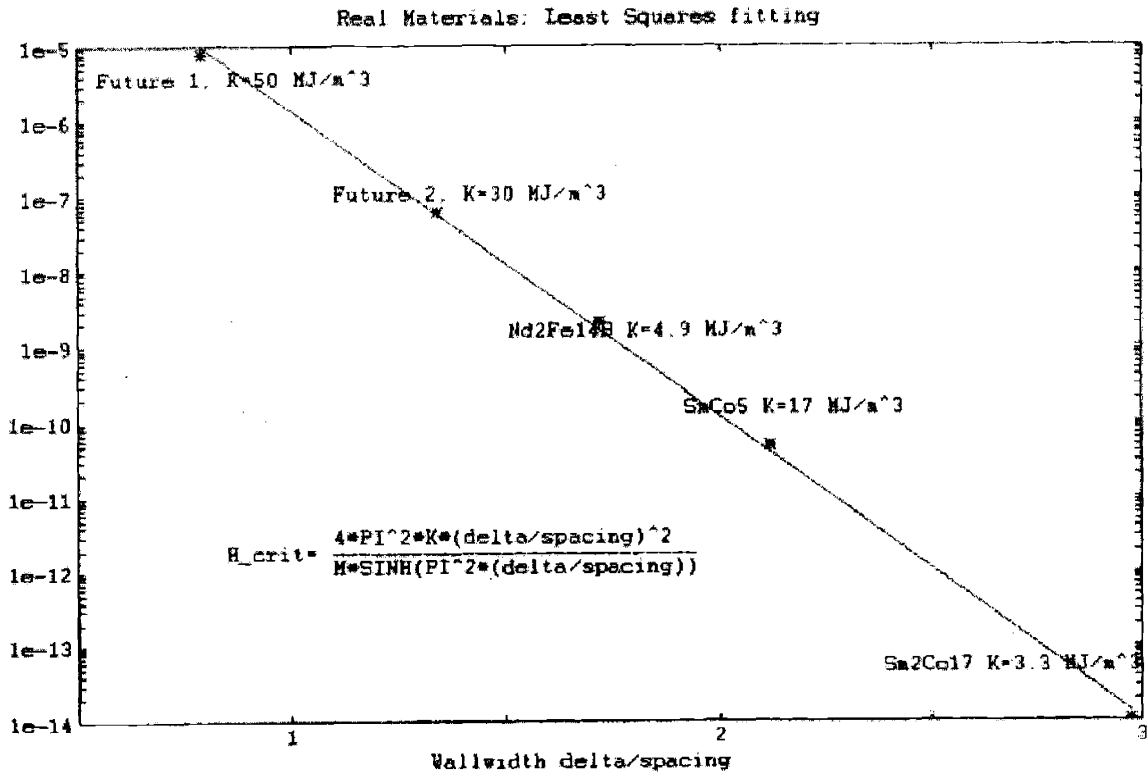
Figure 2.3: Intrinsic coercivity as a function of wall width. Note the remarkable adherence to simple fitting equation (2.9).

Figure 2.4: Intrinsic coercivity plotted for comparison against the absolute wall width. It is informative to notice that the absolute wall width of neodymium--iron--boron is in fact less than that of SmCo$_5$, since it is considerably less effectively "hard".

Figure 2.5: Relative magnitude of the coercivity and the total wall energy with contributions from the boundary conditions and the intrinsic term, plotted against the relative wall width which has been shown to be the relevant abscissa.

Coercive Force

Future 1, K=50 MJ/m^3

Future 2, K=30 MJ/m^3

Nd2Fe14B K=4.9 MJ/m^3

SmCo5 K=17 MJ/m^3

Sm2Co17 K=3.3 MJ/m^3

Coercive force

delta [A]

Ratio of Energy from Discreteness to Rest Energy

Future 1, K=50 MJ/m^3=5*10^8 erg/cm^3

Future 2, K=30 MJ/m^3=3*10^8 erg/cm^3

Nd2Fe14B K=4.9 MJ/m^3=4.9*10^7 erg/cm^3

SmCo5 K=17 MJ/m^3=1.7*10^8 erg/cm^3

Sm2Co17 K=3.3 MJ/m^3=3.3*10^7

[Dimensionless]

Relative Wallwidth

20
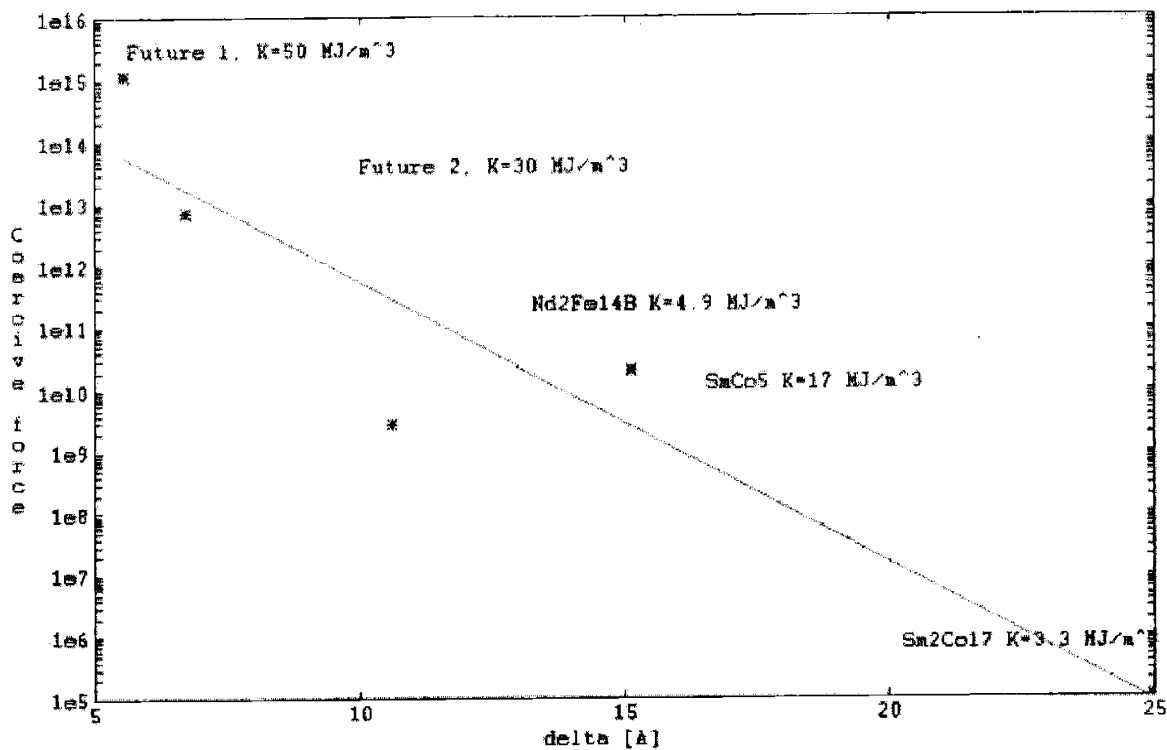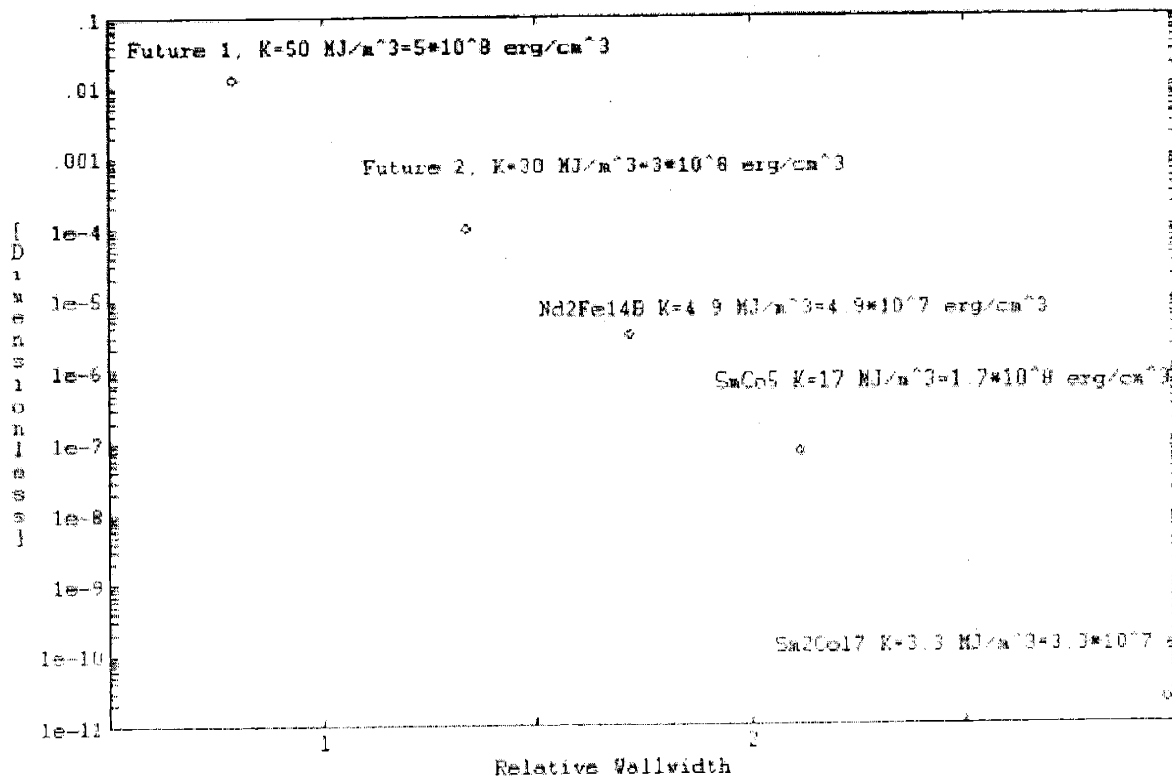
Chapter 3

Defect Interaction

In previous chapters we only considered domain walls moving in a perfect medium. In this part we discuss the static solutions to a narrow wall in a medium with a planar defect of a certain width, and compare these solutions to those obtained using a continuum picture.

Only planar inhomogeneities are analyzed here, and not precipitation defects proposed by Kersten to reduce total wall energy [16]. In brief, Kersten states that there is some critical defect size above which the additional energy from the interface with the defect is balance by the reduction in energy coincident with the decreased total length of the domain wall. It can be assumed that there will be a functionally similar critical size faced by defects in a medium containing very narrow magnetic walls.

3.1 Geometry and Mathematics of the Problem

Following Paul and Quattrochi [3], space is divided into three regions, two infinite perfect grains of anisotropy angles perpendicular to the x-axis and a defect region of inconsistent properties. In the reference provided, a more generalized formulation was used, with anisotropy axes at Θ_A and $\pi - \Theta_A$, where it was shown that the contribution for marked change in anisotropy directions, the energy change over a defect dwarfed the change in direction of anisotropy energy. Therefore, in order to examine the effect of magnetic changes, we have considered only walls with a complete reversal, or no change in anisotropy angle. In addition to complicating the influence of the narrowness of the wall, a change in anisotropy angle any amount other than 180° requires yet another boundary in the crystal where the favored direction changes. When the shift is a reversal, the fact that the magnetization vector has neither head nor tail works to the advantage of the modeler by allowing the conditions to be set only at infinity. There is no conceptual difference,

only more algebra. One small approximation that is made in both the previous argument and this entire thesis, which is that anisotropy can change abruptly at an edge, when physically it must change smoothly and continuously. However continuity is maintained by imposing apparent interface boundary conditions.

Chemical composition is not necessarily constant over the entire crystal, although the magnetic properties are constant across the perfect grains. The defect can be considered to be an absent monolayer, a slight dislocation, or a diffusion grain boundary. The size of the unit cell is presumed constant.
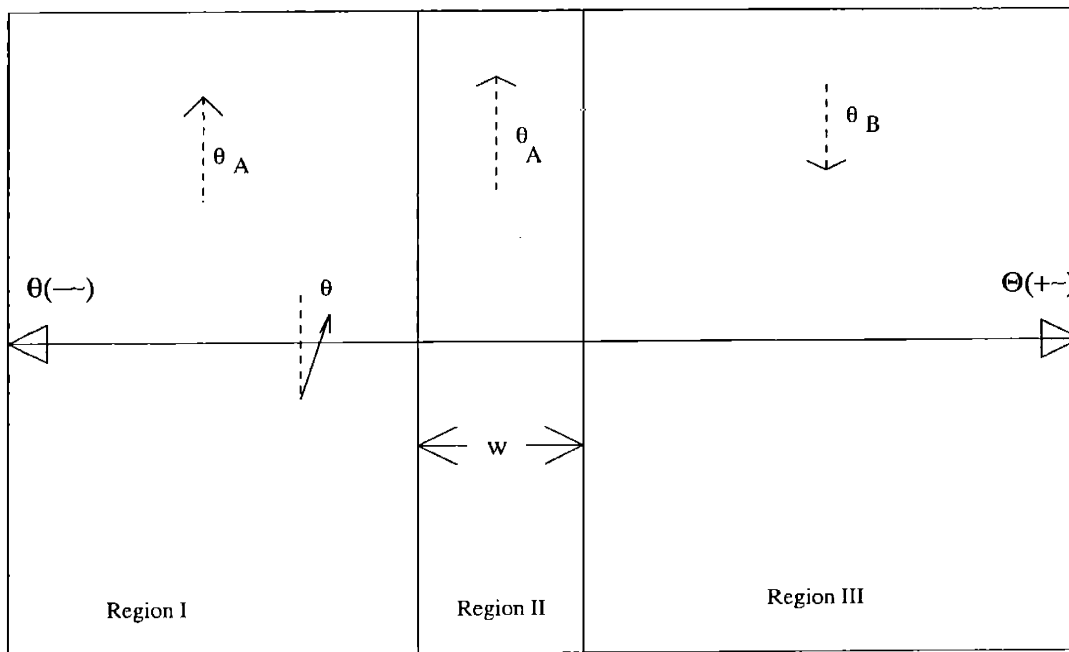
Figure 3.1: Regions I and III are considered the perfect grains, differing only in the boundary value of the direction of magnetizationd, but having the same anisotropy energy, exchange energy, and magnitude of magnetizations. The field is applied parallel to the anisotropy axis.

The Hamiltonian is again formulated as in Chapter 2, with the subscripts on the materials parameters indicating the value in RI, II, or III.

$$\Gamma_i = A_j\left(\frac{\Delta\theta}{\Delta x_i}\right)^2 + K_j(\sin\theta_i)^2 - HM_j\cos\theta_i \qquad (3.1)$$

Three integral equations are now formulated, with

$$\Gamma_{RI} = \sum_{-\infty}^{x^{left}_1} \Gamma_i \qquad (3.2)$$

$$\Gamma_{RII} = \sum_{x^{right}_1}^{x^{left}_2} \Gamma_i \qquad (3.3)$$

$$\Gamma_{RIII} = \sum_{x^{right}_2}^{\infty} \Gamma_i \qquad (3.4)$$

Continuity is applied at the left and right sides of the defect, x_1 and x_2, so that the value of both θ and $A_j\theta'$ are equal across each boundary. Manipulation of the resultant equations confirms the expression presented in full generality in [3],

$$b(\cos\theta_1)^2 + ha(\cos\theta_1) - b(\cos\theta_2)^2 - ha(\cos\theta_2) = 0. \qquad (3.5)$$

The dimensionless parameters are the same as in that reference, or $a = \left(\frac{A_1 M_1}{A_2 M_2}\right) - 1$, $b = \left(\frac{A_1 K_1}{A_2 K_2}\right) - 1$, and normalized field $h = \frac{M_2 H}{K_2}$. The integral equation that must be solved is

$$\int_{x_1}^{x} \frac{dx}{\delta_2} = \int_{\theta_1}^{\theta} d\theta \left\{ (\sin\theta)^2 - h(\cos\theta) + b(\sin\theta_1)^2 - ha(\cos\theta_1) + h(a+1) \right\}^{\frac{-1}{2}}$$

The relative defect width $w = \frac{x_2 - x_1}{\delta_2}$ is found when x is set equal to x_2 in the integral, and is normalized by the projected domain wall halfwidth in the defect region, $\delta_2 = \sqrt{\frac{A_2}{K_2}}.$

3.2 Results and Plots

Code was written (see Appendix B) to find combinations of θ_1, θ_2, and h consistent with a static solution to the problem. The width of the defect having such properties was then calculated, and a width in lattice constants was inferred. A complete set of solutions is not necessarily found by this method, but any finer resolution of the iterations would take prohibitively long on the computers used, and would add little information to the trends to the trends already observed.
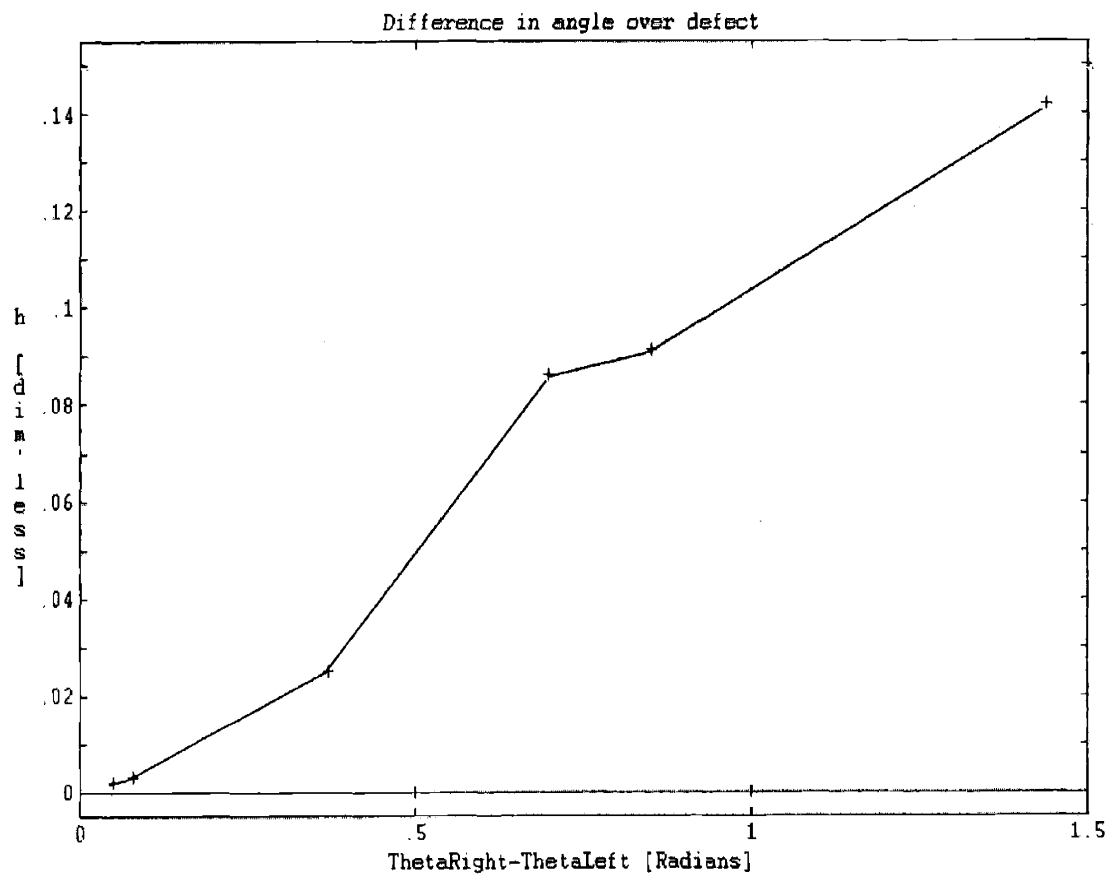
Figure 3.2: Note the monotonically increasing change in magnetization angle across the defect with increased applied field (normalized). This shows that for increasing magnetic field energies, the material concentrates more rotation in an already imperfect region, maximizing the volume of the medium that can be considered perfect. a=0.1, b=0.294.

24

Figure 3.2 shows the correlation between increasing change in magnetization angle over the defect with increasing applied field for a and b values corresponding to approximately a ten per cent change across the defect region. For each increase in Zeeman energy, the material concentrates more of the rotation (which can be thought of as a huge increase in the exchange correlation energy from the misalignment of neighboring atoms) in the defect region.

Figure 3.3 uses the material parameters of $SmCo_5$ and the same values of the defect strength indicating a ten percent change in the magnetic properties. Note that the estimation by integration of the energy of interaction between a domain wall and a defect is always lower by about a factor of two. This is most true at the narrowest defect widths, seeming to taper as the effect of individual atoms is smoothed out over a wider imperfection in the crystal.

It is not a trivial solution when in high strength defects $a = b = 0.5$, the energy calculation based on summation produces a smaller result than that based on integration, although this is only true for defectwidths of less than one lattice constant. Exchange is a correlation term that affects not just the atom in question but its neighbors, which means that even the narrowest defect will jog the equilibrium of both the domain wall shape and its position: at minimum, three atoms are forced to an unstable state, which has more of a direct effect on the terms in the summation.
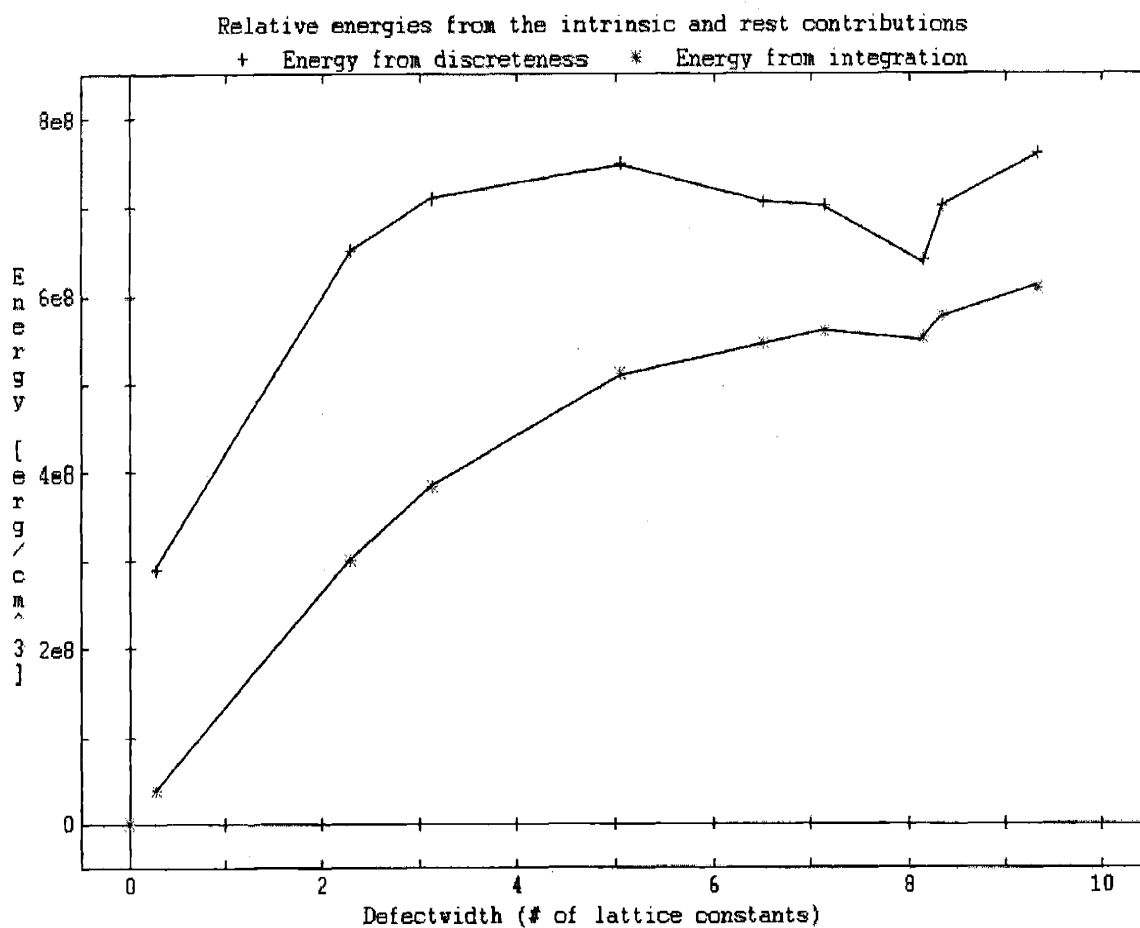
Figure 3.3: Again for a=0.1, b=0.294, the relative heights of the energy calculations from integration and summation are shown. The estimation of the domain wall energy from discreteness is always larger in a relatively weak defect such as this than from integration.

Chapter 4

Time Dependence

A viscosity term is now added to the energy Hamiltonian and the slight perturbation in the argument of the θ is defined to have a time-varying part. For a thorough treatment of the mathematics in this section refer to Titchmarsh [17] or Watson [18], and for a basic coverage of the main points, see Appendix A.

The form of the unperturbed domain wall can be recognized mathematically as a soliton wave, explained succinctly in Trullinger et al. [1,7]. In our work, the most important aspect of a true soliton wave is that it retains its shape during propagation. Nakajima [19] described the properties of solitons and the small perturbations thereof.

4.1 Formulation

The definition over all space of the energy now includes a viscosity term, which can physically be thought of as representing eddy currents (Kittel and Galt [15]), demagnetization fields from time-lagged precession of the spins (Boutron [20]), and/or microdefect interactions (Kittel [12]).

The equation of small-amplitude motion of the $180°$ wall problem has previously been described in a linear model by Kittel and Galt [15], of the form

$$m\ddot{z} + \beta\dot{z} + \alpha z = 2IH \tag{4.1}$$

where m is the mass of the wall, β is the viscosity, and α is a reasonably arbitrary lattice stiffness. The right-hand side of the equation is the pressure on the wall, and H is the applied field. As an aside, wall energy in this reference is also reported in areal density.

This is equivalent to an electrical circuit of net resistance m, total inductance β, and effective capacitance α. In large enough fields the inertial term becomes negligible and the equation reduces to a differential equation of first order. However, in our formulation

this results in an inappropriate oversimplification of the form of the response of the domain wall in the presence of such atomic potential changes.

Our total Hamiltonian in a perfect lattice becomes

$$A\left(\frac{\partial\theta}{\partial x_i}\right)^2 + K(\sin\theta_i)^2 - MH(\cos\theta_i) - \eta\left(\frac{\partial\theta_i}{\partial t}\right) - C\left(\frac{\partial\theta_i}{\partial t}\right)^2 = \Gamma_i \qquad (4.2)$$

where θ is no longer a direct function of $x + \varepsilon$ but is instead

$$\theta \approx \theta_0 + \varepsilon\theta' \qquad (4.3)$$

as explained in the Appendix, where the second term is in total equal to ϕ. In addition, one must be careful with the units on the kinetic energy term, as θ is dimensionless where a length is expected. This is avoided by redefining the coefficient C to include a character-istic length (in units of length squared) of the wall. The coefficient of the prime term includes a Legendre polynomial as a function only of $\left(-\tanh\frac{x}{\delta}\right)$ and a time-dependent solution only of time, T(t).

$$A\left(\frac{d\theta_0}{dx_i} + \frac{d\phi}{dx_i}\right)^2 + K(\sin(\theta_0 + \phi))^2 - MH\cos(\theta_0 + \phi) - \eta\frac{d\phi}{dt} - C\left(\frac{d\phi}{dt}\right)^2 \qquad (4.4)$$

The intrinsic potential is included in the manipulations of the term Θ_0. Substituting the full expression for the angle dependence into the Hamiltonian, one arrives at an inte-gral equation, and proceeds using

$$\phi = P_{total}T_{total} \qquad (4.5)$$

$$P_1^{\ 1} = \frac{l}{\delta} \cdot \mathrm{sech}\frac{\varepsilon l}{\delta}. \qquad (4.6)$$

(Higher order polynomials $P_1^{\ i\mu}$ are ignored as representing radiative spin waves, which are encompassed in the viscosity term.) Manipulating and multiplying each term in

the perturbed Hamiltonian by $P_1^{\ 1}$, then integrating over all space (in a process called projecting out), one is left with only terms of the first kind, the orthogonality of the higher order polynomials with each other having cancelled each one. The assumption is made that ϕ is of a size that the small angle sine approximation can be made. The second-order differential in time is the result:

$$\ddot{T} + \frac{\eta}{2C} \cdot \dot{T} = \frac{\pi^2 A}{Cl^2 \sinh\frac{\pi^2 \delta}{l}} + \frac{HM}{2C}. \tag{4.7}$$

This is solved using the boundary conditions of $T = \dot{T} = 0$ at t=0, and obeys the canonical solution

$$T = s\exp\left[\frac{-t}{n}\right] + \frac{s}{n}t - \frac{s}{n} \tag{4.8}$$

where $n = \eta/(2C)$, $p = \delta/l$, and the coefficient s is equal to $\frac{2\pi^2 Kp^2}{\eta \sinh(\pi^2 p)} + \frac{HM}{\eta}$.

From this expression one sees that at some time large enough, the assumption that the perturbation ε is small becomes invalid. However, the solution maintains its physical realism when it is remembered that this equation of motion is for a perfect crystal, and in real matter with a dislocation density of $10^4/cm^2$, there would be only a finite small distance a domain wall would travel unhindered. Figure 4.1 confirms that the size of the perturbation is small for all relevant times on the scale of a material response to a magnetic field by comparing the size of ε/θ_0 at x=0.
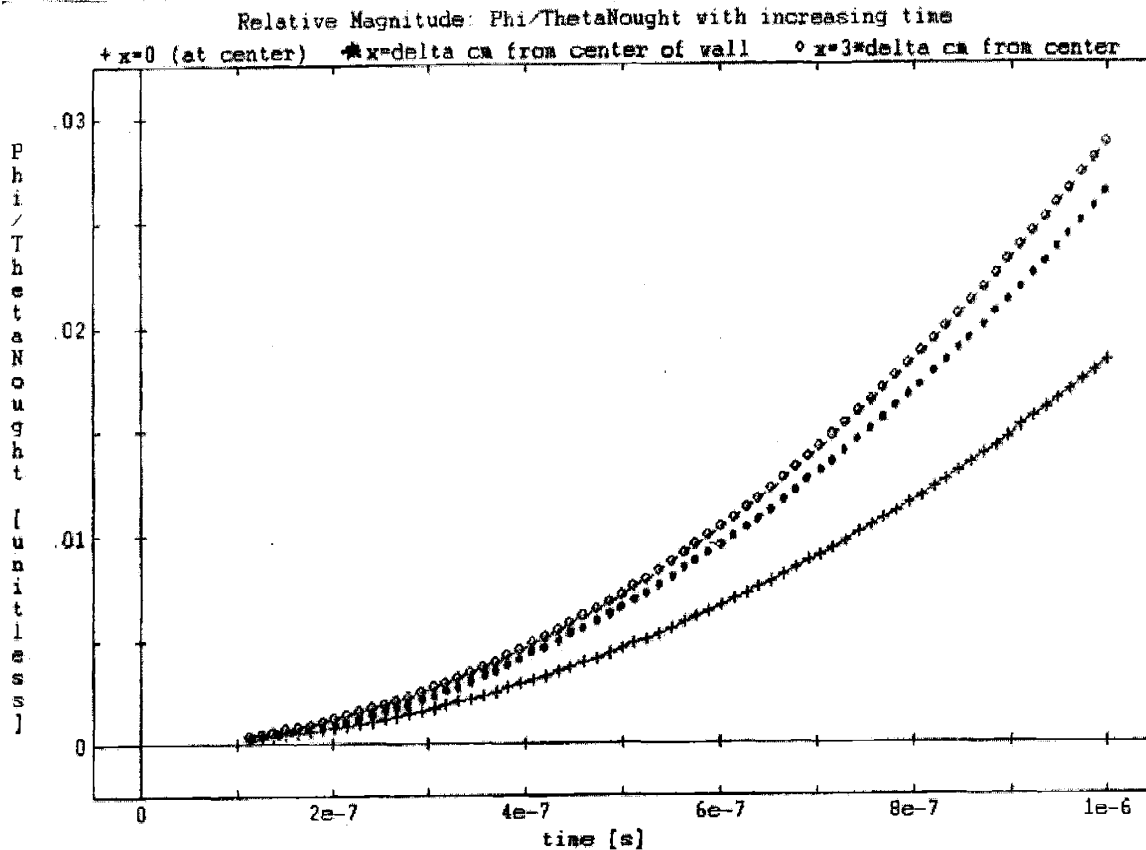
Figure 4.1: Relative magnitude of the perturbation compared with the size of the equilibrium angle distribution at a given point. This confirms our assumption that it is small and allows us to make some necessary simplifications in the integration.

4.2 Results

Figure 4.2 shows the evolution of the shape of the wall at different times. The reference frame is always centered at the midpoint of the wall which is defined to be symmetric.

Figure 4.3 models the effect of an increasing intrinsic viscosity (representing eddy currents, spin waves, etc.) on the motion of the wall in a given time. Only the T part of the perturbation is plotted, as the maximum distortion from the equilibrium distribution with time is felt at the midpoint of the wall because of the hyperbolic secant coefficient which falls off to zero away from the center.

Figure 4.4 has been run at great patience from the author on a stepsize requiring several hours to calculate, and yet the general shape is the same, which we postulate to show the effect of individual atoms on the motion of the domain wall. Again only the motion at the center of the wall is shown as it is the most significant. A viscosity any larger begins to smooth out the visible effects from each single atom and instead merely show the cumulative effects of damped motion.
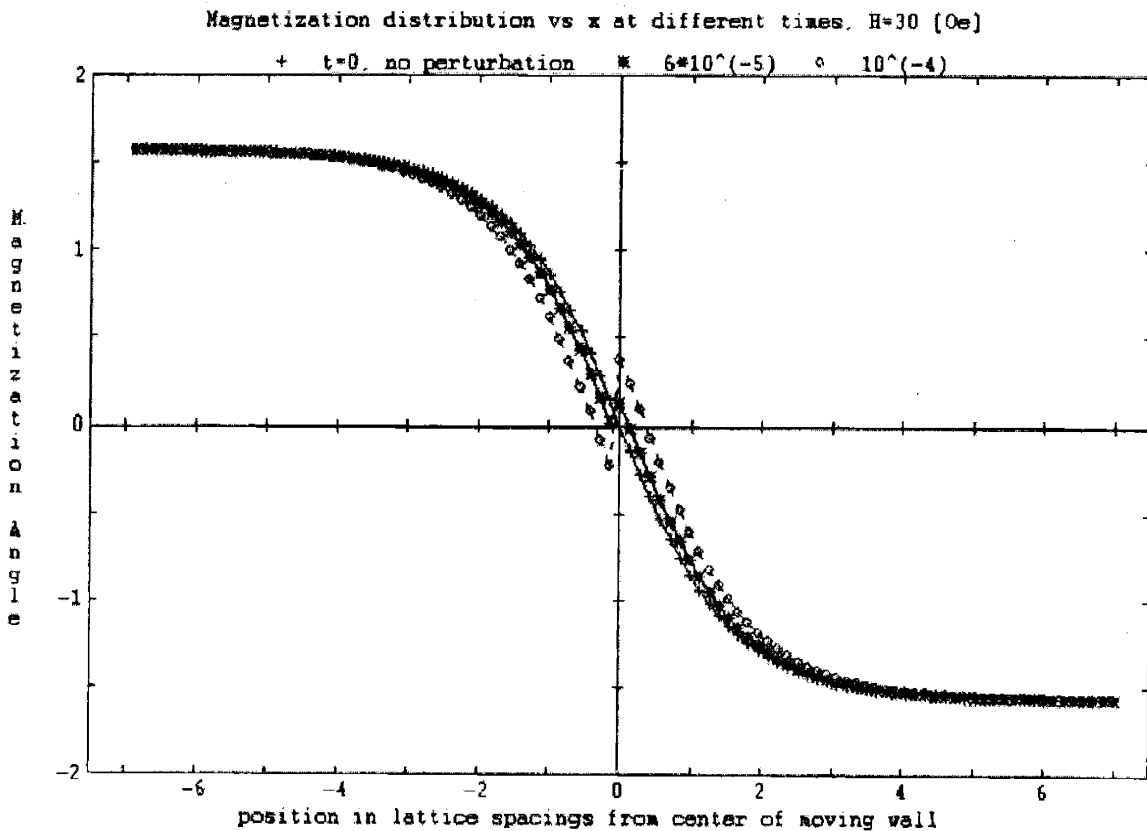
Figure 4.2: Shows the evolution of the shape of the wall at times 0 seconds, 6×10^{-5} sec., and 10^{-4} sec. Notice the instability around the center.

Figure 4.3: Showing the effect of an increasing viscosity on the ability of the domain wall to move in response to an applied field. Zero viscosity is not shown since there is no static solution; an infinitesimally small field applied to a domain wall in a perfect crystal will set it moving forever.

Figure 4.4: Proposed to show the effects of individual atoms on the movement of the domain wall in a medium with viscosity near zero, increasing the effect of the intrinsic potential on the equation of motion of the wall.

Time dependent part vs time at x=0 at H=20
+ eta1=10^(-5)   *   eta2=10^(-4)   ◇   eta3=10^(-3)

Time dependent part using VISCOSITY eta=1.4*10^(-11), H=20

32
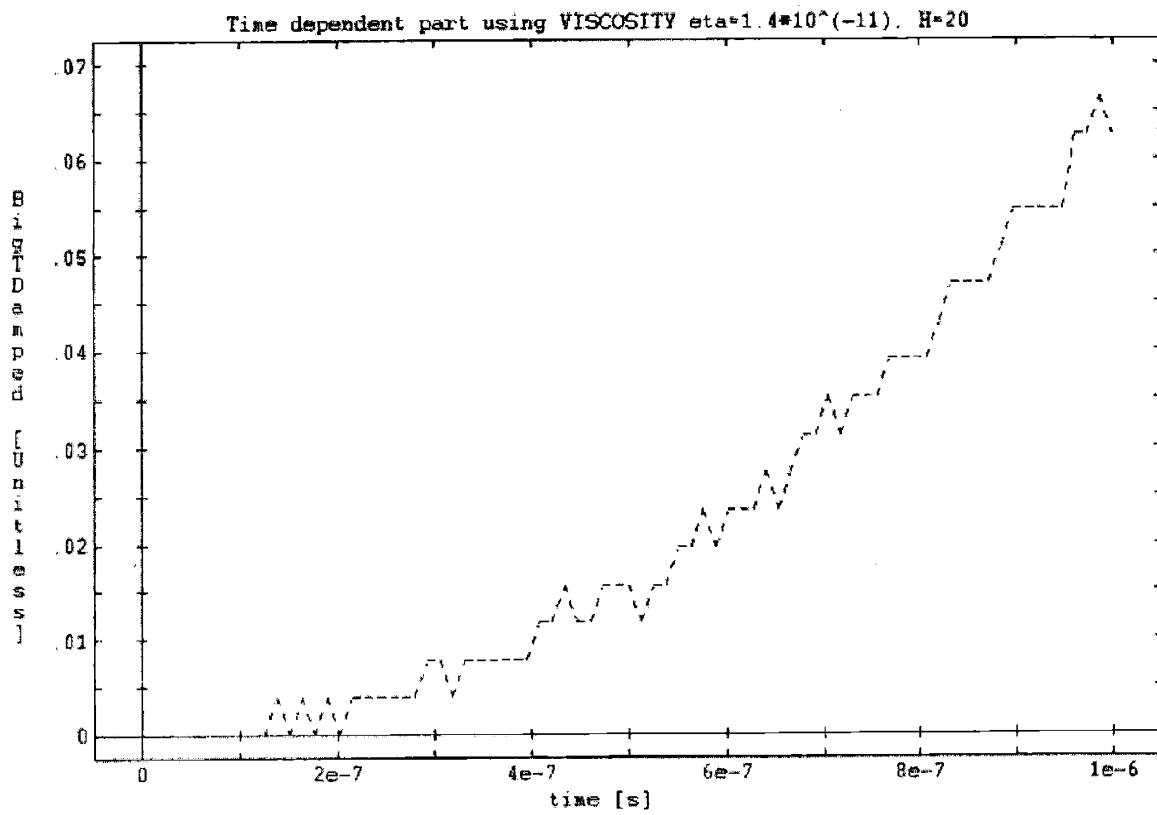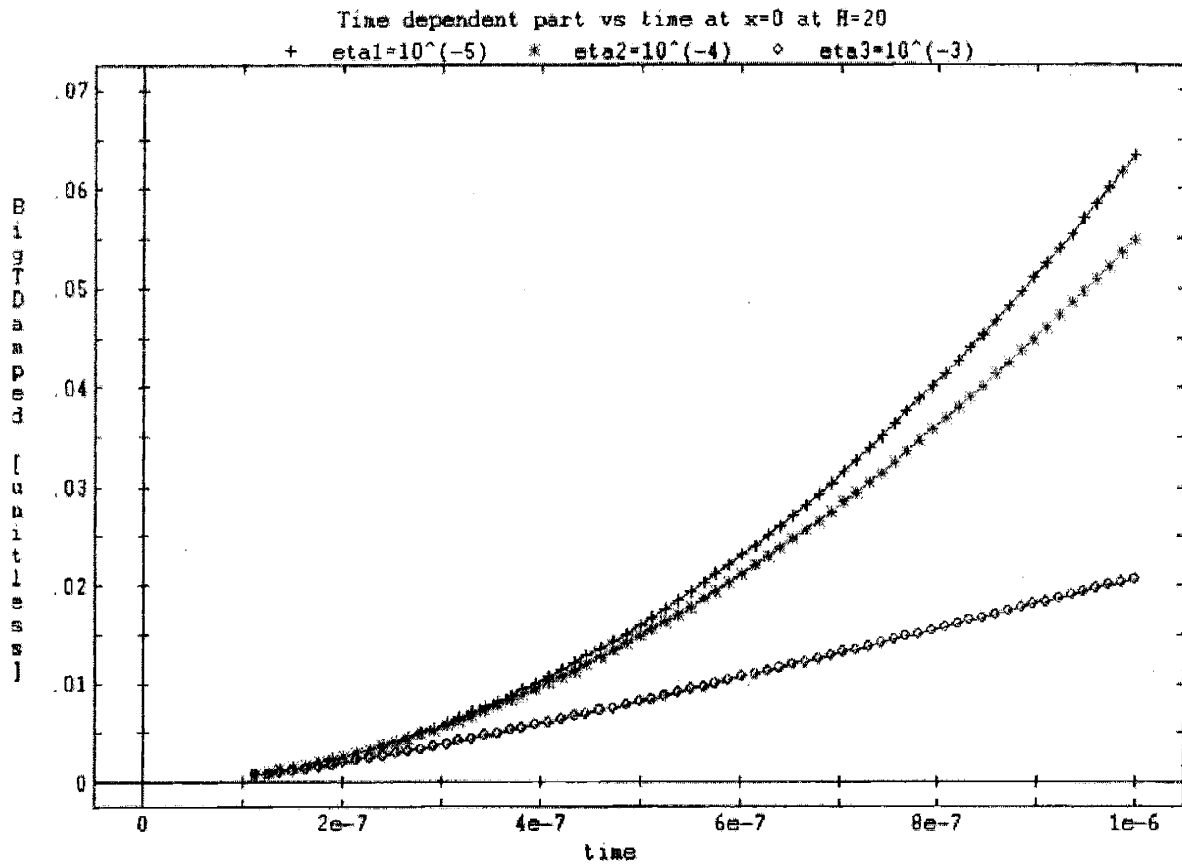
Chapter 5

Conclusions and Future Direction

Currently, materials that require such narrow-wall analysis can not be processed. However, with the advancing technology of hard magnetic materials, it will not be too far in the future before these mathematical results have the opportunity to prove useful.

It has been seen that as processing catches up with the symbolic ability to achieve ideal crystals and ultra-narrow walls, the continuum approximation of matter will break down, and will do so significantly (orders of magnitude). Defects of known widths and strengths will be able to pin narrow walls reliably under appropriate field conditions, which will be extremely useful for high-density magnetic recording or other applications.

A magnetization profile over time, to an external field with a shape other than harmonic, should be solvable with these equations. Multiple walls must be considered for their interactions with each other and the magnetization history of the material as each passes through even a perfect crystal.

Future work includes the extended calculations, most probably with numerical (non-analytic) approximations, correlated to domain walls of rotations other than 180 degrees. A more thorough but less intuitive result might also be reached using the full power of quantum mechanics or spin-polarized computation.

References

[1] *Dynamics of sine-Gordon solitons in the presence of perturbations*, MB Fogel, SE Trullinger, AR Bishop, JA Krumhansl, Phys. Rev. B **15**(3). 1977, pp1578-1592.

[2] *Numerical simulation of sine-Gordon soliton dynamics in the presence of perturbations*, JF Currie et al, Phys. Rev. B, **15**(12), 1977.

[3] *Magnetization distribution and coercivity in multigrain materials*, DI Paul and D Quattrochi, J Appl Phys, **91**(10), 2002.

[4] *Dislocations in a simple cubic lattice*, FRN Nabarro, 1946.

[5] *Bloch wall excitation, application to nuclear resonance in a Bloch wall*, JM Winter, Phys Rev, **124**(2), 1961, pp 452-459.

[6] *Dynamics and statistical mechanics of a one-dimensional model Hamiltonian for structural phase transitions*, JA Krumhansl and Schrieffer, Phys Rev B, **11**(9), 1975.

[7] *Dynamic polarizability of the sine-Gordon soliton*, MB Fogel, SE Trullinger, AR Bishop, Physics Letters, **59A**(2), 1976.

[8] *Soliton theory and the dynamics of a ferromagnetic domain wall*, DI Paul, J Phys C, vol 12, 1977.

[9] *Über die trägheit der Wände zwischen Weißchen Bezirken*, W Döring, Z Naturforsch, **3a**(373), 1948.

[10] *Domain walls in ferromagnetic Dy and Tb*, T Egami and CD Graham, Jr, J Appl Phys, **42**(4), 1971, pp 1299-1300.

[11] Table of Integrals, Series, and Products, Gradshteyn and Ryzhik, 1965, 5th ed.

[12] *Physical Theory of Ferromagnetic Domains*, C Kittel, Reviews of Modern Physics, **21**(4), 1949.

[13] *Die Dynamik der Blochschen Wand*, U Enz, Helvetia Physica Acta, vol 37, 1964.

[14] *A theory of intrinsic coercivity in narrow magnetic domain wall materials*, AR Bishop and Lewis, J Phys C, **12**, 1979.

[15] Advances in Solid-State Physics, vol 3 *Ferromagnetic Domain Theory*, C Kittel and J Galt, , pp 534-564.

[16] *Grundlagen einer neuen Theorie der Koerzitivkraft*, M Kersten, Liepzig, 1943.

[17] The Theory of Functions 2nd ed., EC Titchmarsh, Oxford, 1947

[18] Methods in Theoretical Physics, P Morse & H Feshbach, McGraw-Hill, New York, 1953.

[19] *Nonequilibrium stationary coupling of solitons*, K Nakajima, Y Sawada, Y Onodera, J Appl Phys, **46**, pp5272-5279, 1979.

[20] *Facteur de Transmission d'une paroi de Bloch pour des ondes de spin de vecteur d'onde normal a la paroi*, F Boutron, Academie des Sciences, 1961, pp 3955-3957.

Appendix A

Review of Relevant Mathematics

A.1 Legendre polynomials

Legendre polynomials are more expertly addressed in Boutron [20], Morse and Feshbach [18], or Titchmarsh [18], and the truly mathematically inclined are directed there. However, a three-minute overview is presented for the casual reader.

The premise is that of stability and time evolution. A static solution θ_0 is discussed here, as a function directly of $x + \varepsilon$, and occasionally of time as well. Somewhat akin to the separation of variables procedure in solving differential equations, it is decided that

$$\theta(x + \varepsilon) \approx \theta_0(x) + \varepsilon\theta_0',$$

(1.1)

for small enough values of the offset ε. Here, the second term of the righthand side of the equation is written as the first term of the Legendre polynomial multiplied by whatever time-dependent part solves the relevant mathematical formulation.

$$\phi = P_1^{\ 1}(u)T_1,$$

(1.2)

where P(u) represents our mythical Legendre polynomial in its entirety, although the u in our case must equal $-\tanh\dfrac{x}{\delta}$, in order to conform to the terms in the Legendre equation. We assume that $P_1^{\ 1}$ is the only important term in the series, and that the $P_1^{\ i\mu}$ represent radiative spin waves which contribute to the viscosity of the lattice, already accounted for here by the *ad hoc* viscosity term and are of no additional concern to our current problem. Conveniently, the polynomials are all orthogonal, which means that the integral over all space of their product is zero if you multiply any two of them together that have different sub- and superscripts.

Substituting the above equations into the Hamiltonian for the domain wall, we obtain the integral equation similarly to before, and multiply each series by the $P_1^{\ 1}$ in a procedure called *projecting out*. This exploits the orthogonality of our terms and leaves only terms of the first kind. (Incidentally, what we work with in this thesis are true Legendre polynomials, but there exist associated Legendre polynomials, denoted $Q_1^{\ 1}$, which conform to a different expression. Again, see Watson (**) for a more technical treatment). The terms are then integrated as usual to produce an equation of the first and second time derivatives of T_1. This is then solved, using boundary conditions at the crystal edges, for an expression for the time dependent part of the perturbation.

A.2 Hyperbolic functions

The trigonometric functions cosine and sine have counterparts in the hyperbolic functions cosh() and sinh(), as do the tangent, secant, etc. Where the trigonometric functions consider only the unit circle, hyperbolic functions parametrize the standard hyperbola of $x^2 - y^2 = 1$. Hyperbolic identities are often similar, although not identical, to the trigonometric fundamentals. Inverse hyperbolic functions are obtained by reflecting the relevant operator about the y=x line. Major functions have form as follows:
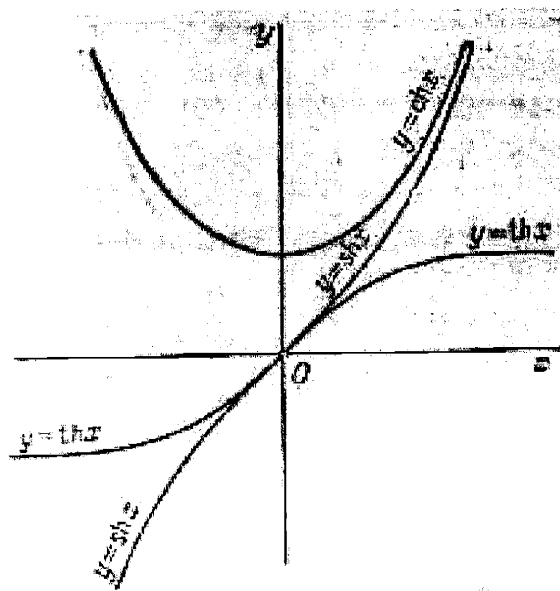
Appendix B

Code

Included are the following files, compiled using the TrueBASIC Silver Edition on an IBM ThinkPad laptop.

Table 2.1: Code inclusions.

Filename	Final Revision Date	# of pages	Used to generate which figures
Energy-using-scigr--30-06-03.TRU	1-Jul-2003	5	2.1
Htotal-vs-delta--5-11-03.TRU	5-Nov-2003	2	2.2
Real-lattice-const--12-11-03.TRU	4-Feb-2004	9	2.3, 2.4
Coercivity--04-02-04.TRU	9-Feb-2004	3	2.5
Defect-quattrochi--06-02-04.TRU	8-Feb-2004	7	3.2, 3.3
Theta-vs-x--28-10-03.TRU	3-Mar-2004	5	4.1-4

```
!Catherine Jenkins, elidor@mit.edu
!SM thesis work under David Paul, dipaul@mit.edu

! Started 26th June 2003
! Last mod 1st July 2003

! Program to calculate and graph the variation
! in wall energy for a wall of halfwidth delta as
! a function of the perturbation alpha (dimensionless)
! anisotropy energy K in ergs/cm^3 and the
! exchange energy A in ergs/cm.

! For analytic derivations of energies found here,
! refer to CAJ notes of 22-26/6

! Defaults to a lattice constant of 2*10^-8 cm,
! exchange energy of 2*10^-6 ergs/cm, and anisotropy
! of 2*10^8 ergs/cm^3.

library "sglib.trc"
library "framelib.trc"
library "sgfunc.trc"

OPTION NOLET                              .

! A dummy variable used for future loops to
! test logically for whether a command has
! been executed. Clearly unimportant.
done=0

DO
        INPUT prompt "Defaults ok? (y or n)": default$

        IF default$="y" THEN
                spacing=2*10^(-8)
                A=2*10^(-6)
                K=2*10^(+8)
                done=1
                PRINT
                PRINT "spacing [cm]="; spacing
                PRINT "A [erg/cm]"; A
                PRINT "K [erg/cm^3]"; K

        ELSEIF default$="n" THEN
                INPUT prompt "spacing=": spacing
                INPUT prompt "A=": A
                INPUT prompt "K=": K
                done=1
        ELSE
                done=0
        END IF
LOOP while done=0

! The Hamiltonian is summed over all space but in
! practice only the s=0 and s=1 terms are visible.
! The variables alpha, s, and delta are introduced.
! An approximation for small alpha and s is used to
! overcome the exception evaluating SINH(0)

alpha=0.1
```

```
s=0
delta=(A/K)^0.5

DEF FNHamiltonian (delta, spacing, s, alpha)= &
& (2*PI^2*delta^2*spacing^(-2)) * s * &
& COS(2*PI*s*alpha) / &
& SINH(PI^2*delta*s/spacing) * (2*K)

DEF FNHamilException (A, K, spacing)=4*(A*K)^(0.5)/spacing

PRINT
restenergy=FNHamilException (A, K, spacing)
PRINT "The s=0, alpha=0 term is"; restenergy

! The rest energy is the Hamiltonian evaluated
! at s, alpha=0, with the physical meaning that
! it is in its lowest energy state at zero perturbation.

! The barrier height is the s=1 term evaluated at some
! small alpha plus the s=0 term at some small alpha,
! minus the rest energy, although in this case the two
! cancel and it can be written as just one term.

PRINT
PRINT "Domain wall halfwidth in cm is";delta
PRINT "Domain wall halfwidth in lattice constants is"; delta/spacing

PRINT
barrier = FNHamiltonian (delta, spacing, 1, alpha)
PRINT "The energy barrier at alpha=";alpha;&
&"and s=1 is";barrier

PRINT
PRINT "Continue. "

!----------------------------------------------------

GET KEY null
CLEAR
SET MODE "Graphics"

!Defining matrices for alpha and the energy Hamiltonian.
DIM x(100), y(100)

FOR dog=1 to 100
        x(dog)=dog/100         ! a desperate way of declaring alpha proporti
        y(dog)=FNHamiltonian(delta, spacing, 1, dog/100)
NEXT dog

!Syntax DataGraph(x(), y(), pstyle, lstyle, col$)
CALL SetText("Energy Change With Static Offset", "alpha", "Energy Barrier")
CALL DataGraph(x(),y(), 6,4, "black black red")

GET KEY null

!----------------------------------------------------
! The following graph watches the change in
! the value of the s=1 term as delta is varied
! directly as l. Necessary values of A and K are stated.
! Initial values were defined in the first section.
```

```
! It is important to remember that delta does not
! automatically adjust with changing values of its
! defining parameters, A and K.

CLEAR

DIM w(100), v(100)

alpha=0.1
xmin=0.1
xmax=6

FOR foo=1 to 100
        delta= spacing*(xmin +foo/100*(xmax - xmin))
        w(foo)=delta
        v(foo)=delta !FNHamiltonian (delta, spacing, 1, alpha)
NEXT foo

CALL SetText("Energy vs Domain Wall Halfwidth","Halfwidth [cm]","s=1 term")
CALL SetGraphType("LOGY")
CALL DataGraph(w(),v(),2,2,"black black blue")

GET KEY null

CLEAR

! The next graph shows the variation in the energy
! as the anisotropy K is varied directly, which
! produces largely the same results as varying delta,
! given the elementary relation between the two.

xmin=1/1000
xmax=1
Kmin=5*10^6
Kmax=5.3*10^8

DIM t(100), u(100), r(100), q(100)

FOR krotus=1 to 100
        xax=Kmin+(Kmax-Kmin)*krotus/100
        t(krotus)=xax
        yax=SQR(A/xax)
        u(krotus)=FNHamiltonian (yax, spacing, 1, alpha)
NEXT krotus

CALL SetText("Energy as Anisotropy","K [ergs/cm^3]","Energy due to Discrete
CALL SetGraphType("XY")
CALL SetGraphType("LOGX")
CALL DataGraph(t(),u(),3,2,"black black green")

DIM bear(2), var(2), bunny(2), jimbo(2)

        bear(1)= 4.9*10^7
        var(1)=.0003
        bear(2)=4.9*10^7
        var(2)=.0008
        PLOT TEXT, at 0.35,0.2: "Nd2Fe14B"

        CALL AddDataGraph(bear,var,0,1,"black")
```

```
        bear(1)= 4.5*10^8
        var(1)=.0075
        bear(2)=5*10^8
        var(2)=.008
        PLOT TEXT, at 0.75,0.73: "Future 1"

        CALL AddDataGraph(bear,var,0,1,"black")

        bear(1)=3.3*10^7
        var(1)=.0003
        bear(2)=3.3*10^7
        var(2)=.0008
        PLOT TEXT, at 0.25,0.2: "Sm2Co17"

        CALL AddDataGraph(bear,var,0,1,"black")

        bear(1)= 1.7*10^8
        var(1)=.0003
        bear(2)=1.7*10^8
        var(2)=.0008
        PLOT TEXT, at 0.6,0.2: "SmCo5"

        CALL AddDataGraph(bear,var,0,1,"black")

        bear(1)= 3.0*10^8
        var(1)=.0003
        bear(2)=3.0*10^8
        var(2)=.0008
        PLOT TEXT, at 0.7,0.2: "Future 2"

        CALL AddDataGraph(bear,var,0,1,"black")

GET KEY null

xmin=.5
xmax=2

FOR florey=1 to 100
        delta=spacing*(xmin+florey/100*(xmax-xmin))
        one=FNHamiltonian (delta, spacing, 1, alpha)
        zero=FNHamilException (K*delta^2, K, spacing)
        r(florey)=PI*delta/spacing
        q(florey)=one/zero
NEXT florey

CALL SetText("Relative magnitudes of s=1/s=0 terms vs width","Wallwidth [la
&"Correction From Discreteness")
CALL SetGraphType("XY")
CALL DataGraph(r(), q(), 2,4,"black black magenta")

GET KEY null
!----------------------------------------------------

!This graph plots the s=1 term of the domain wall,
! allowing for the center to be shifted x_0 away
! from a lattice constant in addition to the
! perturbation alpha against the s=1 term unmodified.

CLEAR
```

44

```
DIM n(100), p(100)

FOR beer=1 to 100
        x_0=spacing*beer/100
        n(beer)=x_0
        p(beer)=(COS(2*PI*x_0/spacing)*COS(2*PI*alpha)+SIN(2*PI*x_0/spacing
&*SIN(2*PI*alpha))/COS(2*PI*alpha)
NEXT beer

CALL SetText("s=1 energy term vs shift x_0 in addition to alpha",&
&"x_0 in lattice constants","s=1 term")
CALL DataGraph(n(),p(), 8,3,"black black cyan")

GET KEY null

END
```

```
!Catherine Jenkins, elidor@mit.edu
! SM Thesis work under David Paul, dipaul@mit.edu

! Plots H_total including intrinsic resistance,
! applied external field, anisotropy, kinetic,
! and exchange energies on the atomic scale.
! Going to try two things, 1) using the new values
! of A and K to define delta, and 2) declaring the
! number of lattice constants without respecting
! their obedience to A and K

!Started 5/Nov/03
!Last Mod 5/Nov/03

OPTION NOLET

library "sglib.trc"
library "sgfunc.trc"
library "framelib.trc"

A=2*10^(-6)          !ergs/cm
K=5*10^8          !ergs/vol
C=10^(-10)           !ergs*s^2/vol
l=2*10^(-8)            !cm, lattic spacing
M=5*10^(2)

DEF FNHtotal (epsilon, delta, t, H)=&
&         4*PI^2*A*cos(2*PI*epsilon)/(l^2*SINH(PI^2*delta/l))+&
&         M*H/COSH(epsilon*l/delta)+&
&         M*H*tanh(epsilon*l/delta)/cosh(epsilon*l/delta)&
&         *exp((-1)*(M*H*PI/(8*c))^(0.5)*t) +&
&         M*H*PI*exp((-2)*(M*H*PI/(8*C))^(0.5)*t)/&
&         (8*(cosh(epsilon*l/delta))^2)

!----------------------------------------------------
! Plotting Htotal vs epsilon for a few lattice
! constants with a few arbitrary delta/l ratios.
! All at t=0 for now.

DIM d(3,100), e(3,100), legend$(3)
steps=100
epsmin=-5
epsmax=5
H0=10

FOR pink=1 to steps
        epsilon=epsmin+pink*(epsmax-epsmin)/steps
        d(1,pink)=epsilon
        d(2,pink)=epsilon
        d(3,pink)=epsilon
        prvni=(A/K)^(0.5)
        e(1,pink)=FNHtotal(epsilon,prvni,0,H0)
        druhy=2.65*l
        e(2,pink)=FNHtotal(epsilon,druhy,0,H0)
        treti=2.5*l
        e(3,pink)=FNHtotal(epsilon,treti,0,H0)
NEXT pink

legend$(1)="SQRT(A/K)"
legend$(2)="2.7*l"
```

```
legend$(3)="2.5*l"
colors$="black black magenta cyan"
CALL SetText("Htotal vs epsilon for different wall widths",&
&           "epsilon in lattice spacings","Htotal")
CALL ManyDataGraph(d,e,2,legend$,colors$)

GET KEY null

!-----------------------------------------------
! Htotal,max vs the ratio delta over l to see where
! is reasonable, where counts as "narrow", etc
! Plotted just at epsilon=0. On the grounds that
! PI*delta is the width, more or less, and that
! walls would have a hard time switching in *less*
! that one lattice constant, let the smallest
! ratio be delta/l=1/PI.

! At epsilon=t=0 and varying H

DIM f(3,100), g(3,100), label$(3)

iterations=100
ratiomax=2
H1=0
H2=5
H3=50

FOR count=1 to steps
        ratio=1.05+count*ratiomax/(2*steps)
        delta=ratio*l
        f(1,count)=PI*ratio
        g(1,count)=FNHtotal(0,delta,0,H1)
        f(2,count)=PI*ratio
        g(2,count)=FNHtotal(0,delta,0,H2)
        f(3,count)=PI*ratio
        g(3,count)=FNHtotal(0,delta,0,H3)
NEXT count

label$(1)="H=0 [Oe]"
label$(2)="H=5"
label$(3)="H=50"
colors$="black black red blue"
CALL SetText("Max total energy vs wallwidths at varying external field",&
&           "Wallwidth [Lattice constants]","Maximum Energy [ergs/volume]")
CALL SetGraphType("LOGY")
CALL ManyDataGraph(f,g,4,label$,colors$)

GET KEY null

END
```

47

```
!Catherine Jenkins, elidor@mit.edu
! SM thesis work under David Paul, dipaul@mit.edu

! Started 12/11/03
! Last Mod 04/02/04
! Longer scale from the intrinsic resistance to
! movement graph. Plot critical wallwidths vs real
! lattic parameter for Fe, NdFeB, SmCo.

OPTION NOLET

library "sgfunc.trc"
library "sglib.trc"
library "framelib.trc"

A=2*10^(-6)          !ergs/cm
K=5*10^8          ! More even than SmCo5 but for future...
M=500                  ! About half that of iron, cgs units
C=6*10^(-9)

K1=5*10^8          !Future
K2=1.7*10^8          !SmCo5
K3=4.9*10^7          !Nd2Fe14B
K4=4.8*10^5          !Fe
K5=5.3*10^6          !Cobalt
K6=(4.5)*10^4          !Nickel: maybe going to be a slight
                  !problem. Don't know what to do with
                  ! negative sign. Should look up somewhere
                  ! other than E du Tremolet de Lacheisserie
K7=3.3*10^7          !Sm2Co17
K8=3*10^8          !Future 2
A1=(800/1043)*2.00*10^(-6)!Fxn of curie temp. As kittel.
A2=1.91*10^(-6)               !(Tc,mat1/1043[K]=Amat1/2.00*10^(-6)
A3=1.12*10^(-6)
A4=2.00*10^(-6)                    !Reference
A5=(1388/1043)*2.00*10^(-6)
A6=(627/1043)*2.00*10^(-6)
A7=(1073/1043)*2.00*10^(-6)
A8=(700/1043)*2.00*10^(-6)
spacing1=7.0*10^(-8)          ![cm]
spacing2=5.0*10^(-8)
spacing3=8.8*10^(-8)
spacing4=2.9*10^(-8)
spacing5=2.5*10^(-8)
spacing6=3.5*10^(-8)
spacing7=8.4*10^(-8)
spacing8=5.0*10^(-8)


!-----------------------------------------------------
! Solve for the field to overcome the intrinsic,
! anisotropy, and exchange energies. Ignore kinetic
! on the assumption that it is still to start with.
! All with relation to delta/spacing ratio and as
! a function of the relevant properties for a given
! material (viz E du Tremolet de Lacheisserie, 2002)

DEF FNField(delta, A, K, M,spacing)=4*PI^2*A/&
&          (M*spacing^2*SINH(PI^2*delta/spacing))

!DIM j(4,70), n(4,70), title$(4)
```

```
DIM j(3,70), n(3,70), title$(3)
Amin=1*10^(-10)
Amax=1*10^(-6)
amount=70

! I happen to know the magnetisation of iron to be
! 1200, and it would be good if future materials
! would have slightly lower than our current top-of
!-the-line of around 500. Ish.

FOR book=1 to amount
        Anow=Amin+(Amax-Amin)*book/amount
        delta1=(Anow/K1)^(0.5)
        delta2=(Anow/K2)^(0.5)
        delta3=(Anow/K3)^(0.5)
        delta4=(Anow/K4)^(0.5)
        j(1,book)=PI*delta1/spacing1
        n(1,book)=FNField(delta1,Anow,K1,400,spacing1)
        j(2,book)=PI*delta2/spacing2
        n(2,book)=FNField(delta2,Anow,K2,M,spacing2)
        j(3,book)=PI*delta3/spacing3
        n(3,book)=FNField(delta3,Anow,K3,M,spacing3)
!        j(4,book)=PI*delta4/spacing4
!        n(4,book)=FNField(delta4,Anow,K4,1200,spacing4)
NEXT book

title$(1)="Future hard materials a~0.7nm"
title$(2)="SmCo5 a=0.5nm"
title$(3)="Nd2Fe14B a=0.88nm"
!title$(4)="Iron a=0.29nm"
colors$="black black green magenta"
CALL SetText("Minumum field to overcome different deltas for real materials
&        "Relative Wallwidth: PI*delta/spacing","Field [Oe]")
CALL ManyDataGraph(j,n,1,title$, colors$)

DIM x(2), y(2)
x(1)=0
x(2)=20
y(1)=10^7
y(2)=10^7
!
CALL AddDataGraph(x,y,0,1,"black")

GET KEY null

!---------------------------------------------------
! Define new fxn of delta/spacing ratio along ordinate,
! field to overcome discreteness along yax, and as a
! fxn of K along the curve.

DEF FNEasyField (ratio, anisotropy, magnetisation)=&
&        4*PI^2*anisotropy*ratio^2/&
&        (magnetisation*SINH(PI^2*ratio))

!------------------------------------------------------------
!As the first graph, but on a much longer scale

DIM s(3,40),t(3,40)

Amin=(0.1)*10^(-6)
```

```
Amax=(2.3)*10^(-6)
amount=40

FOR book=0 to (amount-1)
        Anow=Amin+(Amax-Amin)*book/amount
        delta1=(Anow/K1)^(0.5)
        delta2=(Anow/K2)^(0.5)
        delta3=(Anow/K3)^(0.5)
        s(1,book+1)=PI*delta1/spacing1
        t(1,book+1)=FNField(delta1,Anow,K1,0.8*M,spacing1)
        s(2,book+1)=PI*delta2/spacing2
        t(2,book+1)=FNField(delta2,Anow,K2,M,spacing2)
        s(3,book+1)=PI*delta3/spacing3
        t(3,book+1)=FNField(delta3,Anow,K3,M,spacing3)
NEXT book

title$(1)="Future hard materials a~0.7nm"
title$(2)="SmCo5 a=0.5nm"
title$(3)="Nd2Fe14B a=0.88nm"
!title$(4)="Iron a=0.29nm"
colors$="black black blue red"
CALL SetGraphType("LOGY")
CALL SetText("Minumum field to overcome different intrinsic deltas for real
&       "Relative Wallwidth: PI*delta/spacing","Field [Oe]")
CALL ManyDataGraph(s,t,2,title$, colors$)

!Add a line to show minimum reasonable field anyone
!would bother applying, at 10 Oe
!
!x(1)=0
!x(2)=20
!y(1)=1
!y(2)=1
!
!CALL AddDataGraph(x,y,0,1,"yellow")

! Adding three points to show where real materials
! currently are on each graph. At some point I suppose
! I should bother to learn the syntax for MAT READ
!
!DIM u(3), v(3)
!delta1=(A1/K1)^(0.5)                    ! Future
!u(1)=delta1*PI/spacing1
!v(1)=FNField(delta1,A1,K1,0.8*M,spacing1)
!delta2=(A2/K2)^(0.5)                    ! SmCo5
!u(2)=delta2*PI/spacing2
!v(2)=FNField(delta2,A2,K2,M,spacing2)
!delta3=(A3/K3)^(0.5)                    !Nd2Fe14B
!u(3)=delta3*PI/spacing3
!v(3)=FNField(delta3,A3,K3,M,spacing3)
!
!CALL AddDataGraph(u,v,3,0,"black")

GET KEY null


!-----------------------------------------------------
! More comprehensive materials data comparison graph
! incl. elemental cobalt, nickel, iron, and Sm2Co17.
! Specifically showing sensitivity in K, so add "future
! materials" with different constants. Key:
```

```
!1 Future 1
!2 SmCo5
!3 Nd2Fe14B
!4 Iron                                    .
!5 Cobalt
!6 Nickel
!7 Sm2Co17
!8 Future 2

M1=800
M2=700
M3=700                                     .
M4=1200
M5=1370
M6=485
M7=1030
M8=800

DIM b(5,30),e(5,30), label$(5)

Amin=(0.4)*10^(-6)
Amax=(2.3)*10^(-6)
amount=30

FOR book=0 to (amount-1)
        Anow=Amin+(Amax-Amin)*book/amount
        delta1=(Anow/K1)^(0.5)
        delta2=(Anow/K2)^(0.5)
        delta3=(Anow/K3)^(0.5)
        delta4=(Anow/K4)^(0.5)
        delta5=(Anow/K5)^(0.5)
        delta6=(Anow/K6)^(0.5)
        delta7=(Anow/K7)^(0.5)
        delta8=(Anow/K8)^(0.5)
        b(1,book+1)=PI*delta1/spacing1
        e(1,book+1)=FNField(delta1,Anow,K1,M1,spacing1)
        b(2,book+1)=PI*delta2/spacing2
        e(2,book+1)=FNField(delta2,Anow,K2,M2,spacing2)
        b(3,book+1)=PI*delta3/spacing3
        e(3,book+1)=FNField(delta3,Anow,K3,M3,spacing3)
!        b(4,book+1)=PI*delta4/spacing4
!        e(4,book+1)=FNField(delta4,Anow,K4,M4,spacing4)
!        b(5,book+1)=PI*delta5/spacing5
!        e(5,book+1)=FNField(delta5,Anow,K5,M5,spacing5)
!        b(6,book+1)=PI*delta6/spacing6
!        e(6,book+1)=FNField(delta6,Anow,K6*100,M6,spacing6)
        b(4,book+1)=PI*delta7/spacing7
        e(4,book+1)=FNField(delta7,Anow,K7,M7,spacing7)
        b(5,book+1)=PI*delta8/spacing8
        e(5,book+1)=FNField(delta8,Anow,K8,M8,spacing8)
NEXT book

label$(1)="Future hard materials"
label$(2)="SmCo5"
label$(3)="Nd2Fe14B"
!label$(4)="Iron"
!label$(5)="Cobalt"
!label$(6)="Nickel"
label$(4)="Sm2CO17"
label$(5)="Future (K=3*10^8)"
```

```
legend$="black black magenta green"
CALL SetGraphType("LOGY")
CALL SetText("Comparison of real Materials",&
&          "Relative Wallwidth: PI*delta/spacing","Intrinsic Coercivity")
CALL ManyDataGraph(b,e,2,label$, legend$)

!Add a line to show minimum reasonable field anyone
!would bother applying, at 1 Oe

!DIM x(2),y(2)

x(1)=0
x(2)=20
y(1)=1
y(2)=1

CALL AddDataGraph(x,y,0,1,"yellow")

! Adding three points to show where real materials
! currently are on each graph. At some point I suppose
! I should bother to learn the syntax for MAT READ

DIM f(5), g(5)
delta1=(A1/K1)^(0.5)                      ! Future 1
f(1)=delta1*PI/spacing1
g(1)=FNField(delta1,A1,K1,M1,spacing1)
delta2=(A2/K2)^(0.5)                      ! SmCo5
f(2)=delta2*PI/spacing2
g(2)=FNField(delta2,A2,K2,M2,spacing2)
delta3=(A3/K3)^(0.5)                      !Nd2Fe14B
f(3)=delta3*PI/spacing3
g(3)=FNField(delta3,A3,K3,M3,spacing3)
!delta4=(A4/K4)^(0.5)                     ! Iron
!f(4)=delta4*PI/spacing4
!g(4)=FNField(delta4,A4,K4,M4,spacing4)
!delta5=(A5/K5)^(0.5)                     ! Cobalt
!f(5)=delta5*PI/spacing5
!g(5)=FNField(delta5,A5,K5,M5,spacing5)
!delta6=(A6/K6)^(0.5)                     !Nickel
!f(6)=delta6*PI/spacing6
!g(6)=FNField(delta6,A6,K6,M6,spacing6)
delta7=(A7/K7)^(0.5)                      ! Sm2Co17
f(4)=delta7*PI/spacing7
g(4)=FNField(delta7,A7,K7,M7,spacing7)
delta8=(A8/K8)^(0.5)                      ! Future 2
f(5)=delta8*PI/spacing8
g(5)=FNField(delta8,A8,K8,M8,spacing8)

CALL AddDataGraph(f,g,3,0,"black")
CALL AddLSGraph(f,g,1,"black")

GET KEY null

!----------------------------------------------------
!Same as above but against absolute delta and not
! PI*delta/spacing
!1 Future 1
!2 SmCo5
!3 Nd2Fe14B
!4 Iron
```

```
!5 Cobalt
!6 Nickel
!7 Sm2Co17
!8 Future 2

Amin=(0.4)*10^(-6)
Amax=(2.3)*10^(-6)
amount=30

FOR book=0 to (amount-1)
        Anow=Amin+(Amax-Amin)*book/amount
        delta1=(Anow/K1)^(0.5)
        delta2=(Anow/K2)^(0.5)
        delta3=(Anow/K3)^(0.5)
        delta4=(Anow/K4)^(0.5)
        delta5=(Anow/K5)^(0.5)
        delta6=(Anow/K6)^(0.5)
        delta7=(Anow/K7)^(0.5)
        delta8=(Anow/K8)^(0.5)
        b(1,book+1)=delta1*10^8
        e(1,book+1)=FNField(delta1,Anow,K1,M1,spacing1)
        b(2,book+1)=delta2*10^8
        e(2,book+1)=FNField(delta2,Anow,K2,M2,spacing2)
        b(3,book+1)=delta3*10^8
        e(3,book+1)=FNField(delta3,Anow,K3,M3,spacing3)
!        b(4,book+1)=delta4*10^8
!        e(4,book+1)=FNField(delta4,Anow,K4,M4,spacing4)
!        b(5,book+1)=delta5*10^8
!        e(5,book+1)=FNField(delta5,Anow,K5,M5,spacing5)
!        b(6,book+1)=delta6*10^8
!        e(6,book+1)=FNField(delta6,Anow,K6*100,M6,spacing6)
        b(4,book+1)=delta7*10^8
        e(4,book+1)=FNField(delta7,Anow,K7,M7,spacing7)
        b(5,book+1)=delta8*10^8
        e(5,book+1)=FNField(delta8,Anow,K8,M8,spacing8)
NEXT book

label$(1)="Future hard materials"
label$(2)="SmCo5"
label$(3)="Nd2Fe14B"
!label$(4)="Iron"
!label$(5)="Cobalt"
!label$(6)="Nickel"
label$(4)="Sm2CO17"
label$(5)="Future (K=3*10^8)"
legend$="black black cyan magenta"
CALL SetGraphType("LOGY")
CALL SetText("Comparison of real Materials",&
&         "Absolute Wallwidth [Angstroms]","Intrinsic Coercivity")
CALL ManyDataGraph(b,e,2,label$, legend$)

!Add a line to show minimum reasonable field anyone
!would bother applying, at 1 Oe

x(1)=0
x(2)=40
y(1)=1
y(2)=1

CALL AddDataGraph(x,y,0,1,"yellow")
```

53

```
! Adding three points to show where real materials
! currently are on each graph. At some point I suppose
! I should bother to learn the syntax for MAT READ

DIM ff(5), gg(5)

delta1=(A1/K1)^(0.5)                      ! Future 1
ff(1)=delta1*10^8
gg(1)=FNField(delta1,A1,K1,M1,spacing1)
delta2=(A2/K2)^(0.5)                      ! SmCo5
ff(2)=delta2*10^8
gg(2)=FNField(delta2,A2,K2,M2,spacing2)
delta3=(A3/K3)^(0.5)                      !Nd2Fe14B
ff(3)=delta3*10^8
gg(3)=FNField(delta3,A3,K3,M3,spacing3)
!delta4=(A4/K4)^(0.5)                     ! Iron
!ff(4)=delta4*10^8
!gg(4)=FNField(delta4,A4,K4,M4,spacing4)
!delta5=(A5/K5)^(0.5)                     ! Cobalt
!ff(5)=delta5*10^8
!gg(5)=FNField(delta5,A5,K5,M5,spacing5)
!delta6=(A6/K6)^(0.5)                     !Nickel
!ff(6)=delta6*10^8
!gg(6)=FNField(delta6,A6,K6,M6,spacing6)
delta7=(A7/K7)^(0.5)                      ! Sm2Co17
ff(4)=delta7*10^8
gg(4)=FNField(delta7,A7,K7,M7,spacing7)
delta8=(A8/K8)^(0.5)                      ! Future 2
ff(5)=delta8*10^8
gg(5)=FNField(delta8,A8,K8,M8,spacing8)

CALL AddDataGraph(ff,gg,3,0,"black")
CALL AddLSGraph(ff,gg,1,"black")

GET KEY null

!-----------------------------------------------------
! Addendum to earlier printing:

! Referring to earlier definition of single f and g
! when *relative* was called for.

delta1=(A1/K1)^(0.5)                      ! Future 1
f(1)=delta1/spacing1
g(1)=FNField(delta1,A1,K1,M1,spacing1)*spacing1/(4*(A1*K1)^(0.5))
delta2=(A2/K2)^(0.5)                      ! SmCo5
f(2)=delta2/spacing2
g(2)=FNField(delta2,A2,K2,M2,spacing2)*spacing2/(4*(A2*K2)^(0.5))
delta3=(A3/K3)^(0.5)                      !Nd2Fe14B
f(3)=delta3/spacing3
g(3)=FNField(delta3,A3,K3,M3,spacing3)*spacing3/(4*(A3*K3)^(0.5))
!delta4=(A4/K4)^(0.5)                     ! Iron
!f(4)=delta4*PI/spacing4
!g(4)=FNField(delta4,A4,K4,M4,spacing4)
!delta5=(A5/K5)^(0.5)                     ! Cobalt
!f(5)=delta5*PI/spacing5
!g(5)=FNField(delta5,A5,K5,M5,spacing5)
!delta6=(A6/K6)^(0.5)                     !Nickel
!f(6)=delta6*PI/spacing6
```

54

```
!g(6)=FNField(delta6,A6,K6,M6,spacing6)
delta7=(A7/K7)^(0.5)                    ! Sm2Co17
f(4)=delta7/spacing7
g(4)=FNField(delta7,A7,K7,M7,spacing7)*spacing7/(4*(A7*K7)^(0.5))
delta8=(A8/K8)^(0.5)                    ! Future 2
f(5)=delta8/spacing8
g(5)=FNField(delta8,A8,K8,M8,spacing8)*spacing8/(4*(A8*K8)^(0.5))

CALL SetText("Real Materials; Least Squares fitting",&
&        "Wallwidth delta/spacing", &
&        "Energy due to Discreteness[Oe]")
CALL Datagraph(f,g,3,0,"black")
CALL AddLSGraph(f,g,1,"red")

PLOT TEXT, AT 0.1,0.9: "Future 1, K=50 MJ/m^3"
PLOT TEXT, AT 0.3,0.75: "Future 2, K=30 MJ/m^3"
PLOT TEXT, AT 0.45,0.57: "Nd2Fe14B K=4.9 MJ/m^3"
PLOT TEXT, AT 0.6,0.47: "SmCo5 K=17 MJ/m^3"
PLOT TEXT, AT 0.77, 0.13: "Sm2Co17 K=3.3 MJ/m^3"

PLOT TEXT, AT 0.3,0.3: "4*PI^2*K*(delta/spacing)^2"
PLOT TEXT, AT 0.21,0.28:"H_crit= ---------------------------"
PLOT TEXT, AT 0.3, 0.26:"M*SINH(PI^2*(delta/spacing))"

!---------------------------------------------------------
GET KEY null

delta1=(A1/K1)^(0.5)                    ! Future 1
ff(1)=delta1/spacing1
gg(1)=FNField(delta1,A1,K1,M1,spacing1)/delta1

delta2=(A2/K2)^(0.5)                    ! SmCo5
ff(2)=delta2/spacing2
gg(2)=FNField(delta2,A2,K2,M2,spacing2)/delta2

delta3=(A3/K3)^(0.5)                    !Nd2Fe14B
ff(3)=delta3/spacing3
gg(3)=FNField(delta3,A3,K3,M3,spacing3)/delta3

delta7=(A7/K7)^(0.5)                    ! Sm2Co17
ff(4)=delta7/spacing7
gg(4)=FNField(delta7,A7,K7,M7,spacing7)/delta7

delta8=(A8/K8)^(0.5)                    ! Future 2
ff(5)=delta8/spacing8
gg(5)=FNField(delta8,A8,K8,M8,spacing8)/delta8

CALL SetText("Relative Magnitude of Coercivity due to Discreteness vs Total
&        "Relative Wallwidth delta/spacing", &
&        "Discreteness / Wall Energy")
CALL Datagraph(ff,gg,3,0,"black")
CALL AddLSGraph(ff,gg,1,"blue")

PLOT TEXT, AT 0.1,0.9: "Future 1, K=50 MJ/m^3"
PLOT TEXT, AT 0.3,0.75: "Future 2, K=30 MJ/m^3"
PLOT TEXT, AT 0.45,0.57: "Nd2Fe14B K=4.9 MJ/m^3"
PLOT TEXT, AT 0.6,0.47: "SmCo5 K=17 MJ/m^3"
PLOT TEXT, AT 0.77, 0.13: "Sm2Co17 K=3.3 MJ/m^3"

!---------------------------------------------------------
```

```
GET KEY null

delta1=(A1/K1)^(0.5)                  ! Future 1
ff(1)=delta1*10^8
gg(1)=FNField(delta1,A1,K1,M1,spacing1)*2*PI*M1/delta1

delta2=(A2/K2)^(0.5)                  ! SmCo5
ff(2)=delta2*10^8
gg(2)=FNField(delta2,A2,K2,M2,spacing2)*2*PI*M2/delta2

delta3=(A3/K3)^(0.5)                  !Nd2Fe14B
ff(3)=delta3*10^8
gg(3)=FNField(delta3,A3,K3,M3,spacing3)*2*PI*M3/delta3

delta7=(A7/K7)^(0.5)                  ! Sm2Co17
ff(4)=delta7*10^8
gg(4)=FNField(delta7,A7,K7,M7,spacing7)*2*PI*M7/delta7

delta8=(A8/K8)^(0.5)                  ! Future 2
ff(5)=delta8*10^8
gg(5)=FNField(delta8,A8,K8,M8,spacing8)*2*PI*M8/delta8

CALL AddDataGraph(ff,gg,3,0,"black")
CALL AddLSGraph(ff,gg,1,"black")

CALL SetText("Coercive Force","delta",&
&        "Coercive force")
CALL Datagraph(ff,gg,3,0,"black")
CALL AddLSGraph(ff,gg,1,"green")

PLOT TEXT, AT 0.1,0.9: "Future 1, K=50 MJ/m^3"
PLOT TEXT, AT 0.3,0.75: "Future 2, K=30 MJ/m^3"
PLOT TEXT, AT 0.45,0.57: "Nd2Fe14B K=4.9 MJ/m^3"
PLOT TEXT, AT 0.6,0.47: "SmCo5 K=17 MJ/m^3"
PLOT TEXT, AT 0.77, 0.13: "Sm2Co17 K=3.3 MJ/m^3"

END
```

56

```
!Catherine Jenkins, elidor@mit.edu
! SM thesis work under David Paul, dipaul@mit.edu

OPTION NOLET
library "sgfunc.trc"
library "sglib.trc"
library "framelib.trc"

K1=5*10^8          !Future 1
K2=1.7*10^8          !SmCo5
K3=4.9*10^7          !Nd2Fe14B
K4=3.3*10^7          !Sm2Co17
K5=3*10^8          !Future 2

A1=(800/1043)*2.00*10^(-6)!Fxn of curie temp. As kittel.
A2=1.91*10^(-6)                    !(Tc,matl/1043[K]=Amatl/2.00*10^(-6)
A3=1.12*10^(-6)
A4=(1073/1043)*2.00*10^(-6)
A5=(700/1043)*2.00*10^(-6)! With a medium but arbitrary Curie temp

spacing1=7.0*10^(-8)          ![cm]
spacing2=5.0*10^(-8)
spacing3=8.8*10^(-8)
spacing4=8.4*10^(-8)
spacing5=5.0*10^(-8)

M1=800          !Magnetisation of iron=1200. Numbers chosen fairly randomly
M2=700
M3=700
M4=1030
M5=800


!---------------------------------------------


delta1=(A1/K1)^(0.5)
delta2=(A2/K2)^(0.5)
delta3=(A3/K3)^(0.5)
delta4=(A4/K4)^(0.5)
delta5=(A5/K5)^(0.5)

ratio1=delta1/spacing1
ratio2=delta2/spacing2
ratio3=delta3/spacing3
ratio4=delta4/spacing4
ratio5=delta5/spacing5


!---------------------------------------------
! The problem here is that currently there are choices as
! to defining the coercive force at all. Friedberg and Paul (10)
! define it as the "maximum height of the energy barrier", but
! conversations with Paul (see notes 2/2/04-5/2/04) suggest the
! coercive force to be merely the maximum DERIVATIVE of the
! s=1 term of the energy barrier. This cannot be true, since
! a true derivative would also include the s=0 terms, unless
! we are only considering the coercive force DUE TO DISCRETENESS,
! in which case the s=0 argument is null but the fact remains
! that using the height of the energy barrier makes more sense.
!---------------------------------------------
```

```
!*******************************************
! GRAPH #1, Coded in RED
!-------------------------------------------
! This graph plots the values of wall energy from discreteness
! using the equation 8*PI^2*K*ratio^2/SINH(PI^2*ratio). The
! x-axis is the *relative* wallwidth

DIM b(5), c(5)

b(1)=ratio1
b(2)=ratio2
b(3)=ratio3
b(4)=ratio4
b(5)=ratio5

Discrete1=8*PI^2*K1*ratio1^2/SINH(PI^2*ratio1)
Discrete2=8*PI^2*K2*ratio2^2/SINH(PI^2*ratio2)
Discrete3=8*PI^2*K3*ratio3^2/SINH(PI^2*ratio3)
Discrete4=8*PI^2*K4*ratio4^2/SINH(PI^2*ratio4)
Discrete5=8*PI^2*K5*ratio5^2/SINH(PI^2*ratio5)

c(1)=Discrete1
c(2)=Discrete2
c(3)=Discrete3
c(4)=Discrete4
c(5)=Discrete5

CALL SetGraphType("LOGY")
CALL SetText("Energy due to discreteness (s=+/-1 terms)",&
&        "Relative wallwidth","[ergs/cm^3]")
CALL DataGraph(b,c,4,0,"black")
CALL AddLSGraph(b,c,3,"red")

PLOT TEXT, AT 0.1,0.9: "Future 1, K=50 MJ/m^3=5*10^8 erg/cm^3"
PLOT TEXT, AT 0.3,0.75: "Future 2, K=30 MJ/m^3=3*10^8 erg/cm^3"
PLOT TEXT, AT 0.45,0.57: "Nd2Fe14B K=4.9 MJ/m^3=4.9*10^7 erg/cm^3"
PLOT TEXT, AT 0.6,0.48: "SmCo5 K=17 MJ/m^3=1.7*10^8 erg/cm^3"
PLOT TEXT, AT 0.65, 0.15: "Sm2Co17 K=3.3 MJ/m^3=3.3*10^7 erg/cm^3"

GET KEY null

!*********************************************
!GRAPH #2, coded BLUE
! This is the (reasonably indisputable) rest wall energy

b(1)=ratio1
b(2)=ratio2
b(3)=ratio3
b(4)=ratio4
b(5)=ratio5

Rest1=4*(A1*K1)^(0.5)/spacing1
Rest2=4*(A2*K2)^(0.5)/spacing2
Rest3=4*(A3*K3)^(0.5)/spacing3
Rest4=4*(A4*K4)^(0.5)/spacing4
Rest5=4*(A5*K5)^(0.5)/spacing5

c(1)=Rest1
c(2)=Rest2
c(3)=Rest3
```

```
c(4)=Rest4
c(5)=Rest5

CALL SetText("Rest Energy","Relative Wallwidth","[erg/cm^3]")
CALL DataGraph(b,c,4,0,"black")
CALL AddLSGraph(b,c,3,"blue")

PLOT TEXT, AT 0.1,0.55: "Future 1, K=50 MJ/m^3=5*10^8 erg/cm^3"
PLOT TEXT, AT 0.3,0.65: "Future 2, K=30 MJ/m^3=3*10^8 erg/cm^3"
PLOT TEXT, AT 0.35,0.25: "Nd2Fe14B K=4.9 MJ/m^3=4.9*10^7 erg/cm^3"
PLOT TEXT, AT 0.55,0.50: "SmCo5 K=17 MJ/m^3=1.7*10^8 erg/cm^3"
PLOT TEXT, AT 0.65, 0.35: "Sm2Co17 K=3.3 MJ/m^3=3.3*10^7 erg/cm^3"

GET KEY null

!***************************************************
!GRAPH #3, coded GREEN
! This is the ratio of the discreteness to the rest mass

c(1)=Discrete1/Rest1
c(2)=Discrete2/Rest2
c(3)=Discrete3/Rest3
c(4)=Discrete4/Rest4
c(5)=Discrete5/Rest5

CALL SetText("Ratio of Energy from Discreteness to Rest Energy",&
&        "Relative Wallwidth","[Dimensionless]")
CALL DataGraph(b,c,4,0,"black")
CALL AddLSGraph(b,c,3,"green")

PLOT TEXT, AT 0.1,0.9: "Future 1, K=50 MJ/m^3=5*10^8 erg/cm^3"
PLOT TEXT, AT 0.3,0.75: "Future 2, K=30 MJ/m^3=3*10^8 erg/cm^3"
PLOT TEXT, AT 0.45,0.57: "Nd2Fe14B K=4.9 MJ/m^3=4.9*10^7 erg/cm^3"
PLOT TEXT, AT 0.6,0.48: "SmCo5 K=17 MJ/m^3=1.7*10^8 erg/cm^3"
PLOT TEXT, AT 0.65, 0.15: "Sm2Co17 K=3.3 MJ/m^3=3.3*10^7 erg/cm^3"

END
```

```
!Catherine Jenkins, elidor@mit.edu
!SM thesis work under David Paul, dipaul@mit.edu

OPTION NOLET

library "sgfunc.trc"
library "sglib.trc"
library "framelib.trc"

!The idea here is to solve for values of Theta1 and
! Theta2 (the magnetisation directions at the boundaries
! of the defect) consistent with an applied little-h
!and Eqn 4, Ref (11).

INPUT prompt "Use a 180degree wall in SmCo5 (Y or N)? ":jamaica$
SELECT CASE jamaica$

CASE "Y","y"
                Material$="SmCo5"
                A1=1.91*10^(-6)
                K1=1.7*10^8
                M1=700
                spacing=5*10^(-8)
                CapThetaA=0
        CASE ELSE
                PRINT "What Material?"
                INPUT Material$

                PRINT "With what value of exchange energy (A)?"
                INPUT A1

                PRINT "And the anisotropy in the perfect medium (K)?"
                INPUT K1

                PRINT "Magnetization:"
                INPUT M1

                PRINT "Lattice spacing (constant over defect)"
                INPUT spacing

                PRINT "CapThetaA (Anisotropy Direction in Region I)"
                INPUT CapThetaA
END SELECT

PRINT "Okay. Materials parameters inputted."
PRINT ""
CapThetaB=PI-CapThetaA          !We are in RADIANS.

INPUT prompt "Dimensionless number defaults (a=0.1, b=0.294) okay? (Y or N)
SELECT CASE choice$
CASE "Y", "y"
        a=0.1
        b=0.294
CASE ELSE
        PRINT "Dimensionless a (A1*M1/(A2*M2)-1)"
        INPUT a

        PRINT "Dimensionless b (A1*K1/(A2*K2)-1"
        INPUT b
END SELECT
```

```
!Figure A2 and K2 here for width later:
A2=A1/(a+1)
K2=(A1*K1)/(A2*(b+1))
delta2=SQR(A2/K2)

!It isn't technically possible to have Theta1
!exactly equal to ThetaA without the wall being
!of infinite width.

DEF FNThetaSolve (Theta1, Theta2, CapThetaA, a,b,h)=&
&          b*(COS(Theta1-CapThetaA))^2+&
&          h*a*COS(Theta1-CapThetaA)-&
&          (b+1)*(COS(CapThetaA-PI+Theta2))^2+&
&          (COS(Theta2-CapThetaA))^2-&
&          h*a*COS(Theta2-CapThetaA)-&
&          h*(a+1)*(1-COS(PI-2*CapThetaA))

!*****************************************************
! Three different loops to avoid the problem of needing
! higher dimensions.

PRINT "Okay. Continue."

GET KEY null
CLEAR
Outputfile$="Output-"&Date$&"-"&Material$&".TXT"
!Filename1$="File1of2-" & Date$ & "-" & Material$ & ".TXT"
OPEN #1: name Outputfile$, create new
ERASE #1

PRINT #1:Date$
PRINT #1:"CapThetaA=";CapThetaA
PRINT #1:"A1";A1
PRINT #1:"K1";K1
PRINT #1:"M1";M1
PRINT #1:"a";a
PRINT #1:"b";b
PRINT #1:"spacing";spacing
PRINT #1:""
PRINT #1:"Count,        Theta1,         Theta2,        h"
DIM e(200), f(200), g(200)
count=1

PRINT "Thinking..."
PRINT ""

FOR h=0.001 to 0.35 step 0.001
        FOR Theta1=(CapThetaA+PI/100) to CapThetaB step PI/300
                FOR Theta2=Theta1 to (CapThetaB-PI/100) step PI/300
                        IF ABS(FNThetaSolve (Theta1, Theta2, CapThetaA, a,b
                                PRINT #1:Count,        Theta1,         Theta
                                PRINT "Theta1=";Theta1;"; Theta2=";Theta2;"
                                e(count)=Theta1
                                f(count)=Theta2
                                g(count)=h
                                count=count+1
                        END IF
                NEXT Theta2
        NEXT Theta1
```

61

```
NEXT h

!CLOSE #1
!SOUND 550, 2
PRINT "Done.";(count-1);"solutions." !Copy from ANGLES.TXT to retain."
GET KEY null
CLEAR

!-------------------------------------------------
DIM i(200)

FOR salamander=1 to count
        i(salamander)=f(salamander)-e(salamander)
NEXT salamander

CALL SetText("Difference in angle over the domain",&
&       "Theta2-Theta1","h")
CALL DataGraph(i,g,5,0,"black")

GET KEY null

CLEAR

!*************************************************
!Integrating Eqn 5 (11) using Simpson's rule, with the
!Theta1 and Theta2 generated earlier, renaming them
!slightly to allow for the fact that they now have to
! be a matrix.

DEF FNWidthSolve (foo, h) = &
&       ((sin(foo-CapThetaA))^2-h*cos(foo-CapThetaA)+&
&       b*(sin(Theta1-CapThetaA))^2-&
&       h*a*cos(Theta1-CapThetaA)+h*(a+1))^(-0.5)

DIM Width(200), Defectwidth(200), Lattices(200)

PRINT #1:""
!OPEN #2: name "Widths.TXT", create old
!ERASE #2
!PRINT #2: "Theta 1, Theta 2, Norm. field, norm. width, abs. width, # of co

FOR water=1 to (count-1)
!        PRINT water

        Theta1=e(water)
        Theta2=f(water)
        h=g(water)
        Doremi=FNWidthSolve(Theta2,h)
        Fasola=FNWidthSolve(Theta1,h)
        Estimate=Doremi-Fasola

        Steps=200
        Stepsize=(Theta2-Theta1)/Steps

        !Compute the sum of odd index terms, as in (15) subroutine SIMP
        Space=Theta1-Stepsize
!        PRINT "Steps:";Steps

        FOR krotus=1 to (Steps-1) step 2
                Space=Space+2*Stepsize
```

```
                    Estimate=Estimate+4*FNWidthSolve(Space,h)
          NEXT krotus
!          PRINT "And the odd terms:";Estimate

          !Compute the sum of the even index terms
          Space=Theta1
          FOR jack=2 to (Steps-2) step 2
                    Space=Space+2*Stepsize
                    Estimate=Estimate+2*FNWidthSolve(Space,h)
          NEXT jack
!          PRINT "And the even points:";Estimate

          Estimate=Estimate*Stepsize/3

!          PRINT "Overall defect width:"; Estimate
          Width(water)=Estimate
          Defectwidth(water)=Width(water)*delta2/spacing
          Lattices(water)=INT(Defectwidth(water))
!          PRINT #2: Theta1;";";Theta2;";";h;";";Estimate;";";DefectWidth(wat
NEXT water
!PRINT
!CLOSE #2

!PRINT "Done printing to file Widths.txt. Rename it if you want to keep it.
PRINT "Line 236"
!PRINT "Press a key to continue to calculating energies using these values.
GET KEY null
!*********************************************
DIM Angle(200), ThetaPrime(200)

DEF FNAngle(oink)=CapThetaA + ((PI-2*CapThetaA)/PI)*&
&          ASIN(1/COSH(((oink-1)+num)*spacing/delta2))
DEF FNThetaPrime(oink)=((PI-2*CapThetaA)/(PI*delta2))*&
&          (1/COSH(((oink-1)+num)*spacing/delta2))

DEF FNThetaTemp (num)=((PI-2*CapThetaA)/PI)*&
&          ASIN(1/COSH(num*spacing/delta2))+CapThetaA

DEF FNIntegratedEnergy (position, num)=K2*(1/COSH((position+num)*spacing/de
&          K2*(SIN(ASIN(1/COSH((position+num)*spacing/delta2))-CapThetaA))^2-
&          M1*Hfield*COS(ASIN(1/COSH((position+num)*spacing/delta2))-CapTheta

Filename2$="File2of2-" & Date$ & "-" & Material$ & ".TXT"

DIM Sigma(200), Seahorse(200)

FOR xxx=1 to (count-1)
          Theta1=e(xxx)
          Theta2=f(xxx)
          !PRINT "Theta1=";theta1
          !PRINT "Theta2=";Theta2
          h=g(xxx)
          !PRINT "h=";h
          HField=h*K2/M1
          w=DefectWidth(xxx)
          !PRINT "Real # of lattice constants ";w
          number=Lattices(xxx)          !integer Number of lattice consts wide
          !PRINT "INT(above)";number
                              ! eg, 1.33 means sum over 0 and 1
                              ! terms. s
```

```
PRINT #1: xxx;"th combination"
PRINT #1: "Theta1=";Theta1;", Theta2=";Theta2;", norm field=";h
PRINT #1: "Defect width relative in number of lattice constants";w

!Trying to find center in eqn to fit Theta
num=0
difference=10^(-7)

DO WHILE abs(difference) >= 10^(-5)
        num=num+difference
        thetanow=FNThetaTemp(num)
        difference=Theta1-ThetaNow
LOOP

!A check to make sure num isn't that far off
ThetaCheck=FNThetaTemp (w+num)
IF Theta2>PI/2 THEN
        ThetaCheck=PI-ThetaCheck
        PRINT "Thetacheck after mod";thetacheck
END IF

SUM=0 ! Just to reinitialise
! Finding Theta and ThetaPrime at each relevant lattice constant,
! to be used in the summation
FOR oink=1 to (number+1)
        DelTheta=FNThetaPrime(oink)
        Rotation=FNAngle(oink)
        Energy=A2*Deltheta^2+K2*(sin(Rotation-CapThetaA))^2&
                -M1*HField*COS(Rotation-CapThetaA)
        Sum=Sum+Energy
NEXT oink
PRINT #1: "Summed energy over integral # of constants:";sum
!PRINT "The energy for the narrow wall from Theta=";Theta1;&
         " to Theta=";Theta2;" in field ";Hfield;" is ";sum

Sigma(xxx)=sum

Integral=0 ! To reinitialise for safety
! Integration in a continuous medium, for comparison, using Simpson
!Bounds are x1=0 and x2=w. Use infinitesimal dTheta/dX here not del
Guess=FNIntegratedEnergy(0,num)+FNIntegratedEnergy(w,num)
Iterations=100
Stepsize=w/steps
Position=-stepsize

!odd
FOR meow=1 to (Iterations-1) step 2
        position=position+2*stepsize
        Guess=Guess+4.0*FNIntegratedEnergy(position, num)
NEXT meow

!even
position=0
FOR meow=1 to (Iterations-2) step 2
        position=position+2*stepsize
        Guess=Guess+2.0*FNIntegratedEnergy(position, num)
NEXT meow

Guess=Guess*stepsize/3.0
```

```
            Seahorse(xxx)=guess

            !Energy from a continuous wall which somehow has the same dimless #
            ! other params as my discrete walls but without the necessity of su
            !PRINT "The energy assuming you had integrated:";Guess
            PRINT #1: "The energy assuming you had integrated";Guess

            ! Difference between the two
            !PRINT "Energy in Region II solely from discreteness:";(sum-guess)
            PRINT #1:"Energy in region 2 solely from discreteness [ergs/cm^3]:"
            PRINT #1:""
            !PRINT

    NEXT xxx

    CLOSE #1

    !*****************************************************
    !Graphs.

    DIM x(2), y(2)

    CALL SetText("Normalised consistent field vs Defect Width",&
    &         "h (dim'less)","defectwidth (lattice constants)")
    CALL DataGraph(g,DefectWidth,3,0,"red")

    x(1)=(-1)
    x(2)=20
    y(1)=SQR(A1/K1)/spacing
    y(2)=y(1)

    CALL AddDataGraph(x,y,0,1,"black")

    GET KEY null

    !-------------------------------------------
    DIM breath(2,200), heart(2,200), legend$(2)

    FOR penguin=1 to count
            breath(1,penguin)=DefectWidth(penguin)
            heart(1,penguin)=Sigma(penguin)
            breath(2,penguin)=DefectWidth(penguin)
            heart(2,penguin)=Seahorse(penguin)
    NEXT penguin

    legend$(1)="Energy from discreteness"
    legend$(2)="Energy from integration"
    colors$="black black blue green"
    CALL SetText("Relative energies from the intrinsic and rest contributions",
    &         "Defectwidth (# of lattice constants)","Energy [erg/cm^3]")
    CALL ManyDataGraph(breath,heart,0,legend$,colors$)

    CALL AddDataGraph(y,x,0,1,"black")

    GET KEY null
    !---------------------------------------------------

    DIM michigan(200), pennstate(200)

    FOR clock=1 to (count-1)
```

```
                michigan(clock)=DefectWidth(clock)
                pennstate(clock)=Seahorse(clock)/Sigma(clock)
NEXT clock

CALL SetText("Relative energies from intrinsic and rest terms",&
&          "DefectWidth (# of lattice constants)","Ratio: Integrated/Summed e
CALL DataGraph(michigan, pennstate, 2,0,"black")

GET KEY null

!-------------------------------------------------
DIM emu(2,200), roo(2,200)

FOR koala=1 to count
        emu(1,koala)=DefectWidth(koala)
        roo(1,koala)=e(koala)
        emu(2,koala)=DefectWidth(koala)
        roo(2,koala)=f(koala)
NEXT koala

legend$(1)="Left side"
legend$(2)="Right side"
colors$="black black magenta blue"
CALL SetText("Angle as wallwidth","Defectwidth (lattice constants)","Angle
CALL ManyDataGraph(emu,roo,1,legend$,color$)

CALL AddDataGraph(y,x,0,1,"black")

GET KEY null
!-------------------------------------------------

FOR koala=1 to count
        emu(1,koala)=g(koala)
        roo(1,koala)=e(koala)
        emu(2,koala)=g(koala)
        roo(2,koala)=f(koala)
NEXT koala

legend$(1)="Left side"
legend$(2)="Right side"
colors$="black black magenta blue"
CALL SetText("Angle as normalised field","Normalised field","Angle (radians
CALL ManyDataGraph(emu,roo,0,legend$,color$)

CALL AddDataGraph(y,x,0,1,"black")

GET KEY null

END
```

```
! Catherine Jenkins, elidor@mit.edu
! SM thesis work under David Paul, dipaul@mit.edu

! started 28/10/03
! last mod 3/3/04

library "sglib.trc"
library "framelib.trc"
library "sgfunc.trc"

OPTION NOLET

!Can be assumed for now that only P_1_1 is
!important, so Theta_total reduces to two terms
! and not an infinite sum.

! See solution for T(t) in CAJ 28/10/03.

!Variables
A=10^(-6)            !exchange, energy/length
K=5*10^8                    !anisotropy, energy/volume
C=10^(-10)           !coefficient of KE term as Kittel and Galt
M=10^(-3)            !Magnetization
spacing=2*10^(-8)
delta=(A/K)^(0.5)

!VERY IMPORTANT: Theta is a function not directly of
!absolute position but of relative number of lattice
!spacings, e.g., Theta_Nought(2) is the magnetization
! at 4*10^(-8) cm from the center of the domain wall

DEF FNTheta_Nought (x)=asin(1/cosh(x))

!Phi is spatial*temporal, or sech(x)*exponents
DEF FNPhi (x,H,t)=(1/cosh(x))*(spacing/delta)*&
&        ((2/PI)*(exp(((M*H*PI/(8*C))^(0.5))*t))+        &
&        (2/PI)*(exp((-1)*((M*H*PI/(8*C))^(0.5))*t))-4/PI)

DEF FNBigT (H,t)=(2/PI)*(exp(((M*H*PI/(8*C))^(0.5))*t))+                &
&        (2/PI)*(exp((-t)*((M*H*PI/(8*C))^(0.5)))))-4/PI

!------------------------------------------------------------
xmin=(-7)
xmax=7
t1=0
t2=6*10^(-5)
t3=1*10^(-4)
temp=0
H0=30

! This graph plots Theta vs x for different t at H=1000

DIM e(3,100), f(3,100), label$(3)
iterations=100

!PRINT "cosh(xmin)=";cosh(xmin)
!PRINT "sech(xmin)=";1/cosh(xmin)
!PRINT "sin(-1)sech(xmin)";asin(1/cosh(xmin))
!GET KEY null
```

67

```
! AN *enormous* kludge here, setting the magnetization equal
! to the difference between the variant anisotropy and the
!calculated theta. Not sure how to cope once it has moved.
! to not be centered around x_0

FOR foo=1 to iterations
        temp=xmin+(xmax-xmin)*foo/iterations
        e(1,foo)=temp
!        PRINT e(1,foo);"=x"
!        PRINT FNTheta_Nought(temp)
        IF temp<0 THEN f(1,foo)=PI/2-FNTheta_Nought(temp)-FNPhi(temp,H0,t1)
&            ELSE f(1,foo)=-PI/2+FNTheta_Nought(temp)+FNPhi(temp,H0,t1)
!        PRINT f(1,foo);"=theta"
        e(2,foo)=temp
        IF temp<0 THEN           f(2,foo)=PI/2-FNTheta_Nought(temp)&
&              -FNPhi(temp,H0,t2) &
&              ELSE f(2,foo)=-PI/2+FNTheta_Nought(temp)&
&              +FNPhi(temp,H0,t2)
        e(3,foo)=temp
        IF temp<0 THEN f(3,foo)=PI/2-FNTheta_Nought(temp)&
&              -FNPhi(temp,H0,t3)&
&              ELSE f(3,foo)=-PI/2+FNTheta_Nought(temp)&
&              +FNPhi(temp,H0,t3)
NEXT foo

!GET KEY null

label$(1)="t=0, no perturbation"
label$(2)="t2"
label$(3)="t3"
colors$="black black red blue"
CALL SetText("Magnetization distribution vs x at different times, H=30 [Oe]
&          ,"position in lattice spacings","Magnetization Angle")
CALL ManyDataGraph(e,f,5,label$,colors$)

GET KEY null

!------------------------------------------------------------
xmin=(-7)
xmax=7
H0=0
H1=10
H2=100
temp=0

! This graph plots Theta vs x for different H at t=0

DIM g(3,100), j(3,100), words$(3)
cat=40

! Same kludge as before with the angle

FOR dog=1 to cat
        temp=xmin+(xmax-xmin)*dog/cat
        g(1,dog)=temp
        IF temp<0 THEN j(1,dog)=PI/2-FNTheta_Nought(temp)-FNPhi(temp,H0,0)&
&              ELSE j(1,dog)=-PI/2+FNTheta_Nought(temp)+FNPhi(temp,H0,0)
!        PRINT f(1,dog);"=theta"
        g(2,dog)=temp
        IF temp<0 THEN           j(2,dog)=PI/2-FNTheta_Nought(temp)&
```

```
&                      -FNPhi(temp,H1,0)  &
&                 ELSE j(2,dog)=-PI/2+FNTheta_Nought(temp)&
&                      +FNPhi(temp,H1,0)
          g(3,dog)=temp
          IF temp<0 THEN j(3,dog)=PI/2-FNTheta_Nought(temp)&
&                      -FNPhi(temp,H2,0)&
&                 ELSE j(3,dog)=-PI/2+FNTheta_Nought(temp)&
&                      +FNPhi(temp,H2,0)
NEXT dog

!GET KEY null

words$(1)="No field"
words$(2)="10 [oe]"
words$(3)="100"
colors$="black black green cyan"
CALL SetText("Magnetization distribution vs x at different fields, t=t_noug
&          ,"position in lattice spacings","Magnetization Angle")
CALL ManyDataGraph(e,f,5,words$,colors$)

GET KEY null

!------------------------------------------------------------
!****** This plots BigT vs time for different H
!
steps=70
DIM d(3,70), b(3,70), legends$(3), single(70), xax(70)


tmin=10^(-7)
tmax=10^(-6)
H1=0
H2=1
H3=20
temp=0


p=delta/spacing
eta=10^(-5)          ! This is a pretty random guess
H1=0
H2=20
H3=100

DEF FNBigTDamped(eta,H,time)= exp((-1)*eta*time/(2*C))*&
&     ((4*C*PI^2*K*p^2/((eta^2)*SINH(p*PI^2))+H*M*2*C/eta^2))+&
&     ((2*PI^2*K*p^2/(eta*SINH(p*PI^2)))+H*M/eta)*time-&
&     ((4*C*PI^2*K*p^2/((eta^2)*SINH(p*PI^2)))+2*H*M*C/eta^2)

FOR foo=1 to steps
        temp=tmin+(tmax-tmin)*foo/steps
        d(1,foo)=temp
        b(1,foo)=FNBigTDamped(eta,H1,temp)
        d(2,foo)=temp
        b(2,foo)=FNBigTDamped(eta,H2,temp)
        d(3,foo)=temp
        b(3,foo)=FNBigTDamped(eta,H3,temp)
NEXT foo

legends$(1)="H1=0 [Oe]"
legends$(2)="H2=20"
legends$(3)="H3=100"
```

69

```
colors$="black black green yellow"
!CALL SetGraphType("LOGX")
CALL SetText("Time dependent part vs time at x=center of wall, eta=10^(-5)"
&         "time [sec]","BigTDamped [unitless]")
CALL ManyDataGraph(d,b,6,legends$,colors$)
GET KEY null


! Varying eta *****
eta1=10^(-5)
eta2=10^(-4)
eta3=10^(-3)

FOR foo=1 to steps
        temp=tmin+(tmax-tmin)*foo/steps
        d(1,foo)=temp
        b(1,foo)=FNBigTDamped(eta1,H2,temp)
        d(2,foo)=temp
        b(2,foo)=FNBigTDamped(eta2,H2,temp)
        d(3,foo)=temp
        b(3,foo)=FNBigTDamped(eta3,H2,temp)
NEXT foo

legends$(1)="eta1=10^(-5)"
legends$(2)="eta2=10^(-4)"
legends$(3)="eta3=10^(-3)"
colors$="black black magenta green"
!CALL SetGraphType("LOGX")
CALL SetText("Time dependent part vs time at x=0 at H=20",&
&         "time","BigTDamped [unitless]")
CALL ManyDataGraph(d,b,6,legends$,colors$)
GET KEY null



!*****
etanow=14*10^(-12)

FOR bang=1 to steps
        temp=tmin+(tmax-tmin)*bang/steps
        xax(bang)=temp
        single(bang)=FNBigTDamped(etanow,H2,temp)
NEXT bang

CALL SetText("Time dependent part using VISCOSITY eta=1.4*10^(-11), H=20",&
&         "time [s]","BigTDamped [Unitless]")
CALL DataGraph(xax,single,1,2,"black")
GET KEY null

!!!!!!!!*!!!*!!!!!!!!!!!*!!!!!!!!!!!!!*!!!!!!!!!!*!!!!!!!!!!!!!!
!!!!!!*!!*!!!!!!!!!!!!!!!!!!*!!!!!!!!*!!!!!!!!!!!!!*!!!!!!!!!!!!!
! Graphing relative size of PHI (=P_1_1(x)*T(t)) and ThetaNought
! at increasing time at x=1,2,and 3 wallwidths away from center
! For some eta.

eta=10^(-6)
H=10

FOR whammo=1 to steps
        temp=tmin+(tmax-tmin)*whammo/steps
        Phi_temp_1=(spacing/delta)*(1/COSH(0))*FNBigTDamped(eta,H,temp)
        ThetaNought_1=ASIN(1/COSH(0))
```

70

```
        Phi_temp_2=(spacing/delta)*(1/COSH(1))*FNBigTDamped(eta,H,temp)
        ThetaNought_2=ASIN(1/COSH(1))
        Phi_temp_3=(spacing/delta)*(1/COSH(3))*FNBigTDamped(eta,H,temp)
        ThetaNought_3=ASIN(1/COSH(3))

        Relative1=Phi_Temp_1/ThetaNought_1
        Relative2=Phi_Temp_2/ThetaNought_2
        Relative3=Phi_Temp_3/ThetaNought_3

        d(1,whammo)=temp
        b(1,whammo)=Relative1
        d(2,whammo)=temp
        b(2,whammo)=Relative2
        d(3,whammo)=temp
        b(3,whammo)=Relative3
NEXT whammo

legends$(1)="x=0 (at center)"
legends$(2)="x=delta cm from center of wall"
legends$(3)="x=3*delta cm from center"
colors$="black black blue yellow"
!CALL SetGraphType("LOGX")
CALL SetText("Relative Magnitude: Phi/ThetaNought with increasing time",&
&         "time [s]","Phi/ThetaNought [unitless]")
CALL ManyDataGraph(d,b,2,legends$,colors$)
GET KEY null
END
```