Low internal magnetic fields in anisotropic superconductors

Allan J. Greer Jr.
College of William & Mary - Arts & Sciences

Follow this and additional works at: https://scholarworks.wm.edu/etd
Part of the Condensed Matter Physics Commons

Recommended Citation
Greer, Allan J. Jr., "Low internal magnetic fields in anisotropic superconductors" (1994). Dissertations, Theses, and Masters Projects. Paper 1539623852.
https://dx.doi.org/doi:10.21220/s2-87bx-2q20
INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.
Low internal magnetic fields in anisotropic superconductors

Greer, Allan John, Jr., Ph.D.
The College of William and Mary, 1994
LOW INTERNAL MAGNETIC FIELDS IN ANISOTROPIC SUPERCONDUCTORS

A Dissertation
Presented to
the Faculty of the Department of Physics
The College of William and Mary in Virginia

In Partial Fulfillment
of the Requirements for the Degree of
Doctor of Philosophy

By
Allan J. Greer, Jr.
April, 1994
APPROVAL SHEET

This dissertation is submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy.

\[\text{Allan J. Greer, Jr.}\]

Approved, April, 1994

\[\text{W.J. Kossler, Chairman}\]

\[\text{K.G. Petziager}\]

\[\text{H.E. Schone}\]

\[\text{M.K. Hinders}\]

Applied Science

\[\text{C.E. Stronach}\]

Virginia State University
Contents

Acknowledgements vi
List of Tables viii
List of Figures xvii
Abstract xviii

1 Introduction 1

2 Fundamentals of $\mu$SR 4
   2.1 A Little History .......................... 4
   2.2 What is $\mu$SR? ......................... 5
   2.3 Pion and Muon Production .......... 7
   2.4 Muon Decay ............................. 10
   2.5 Thermalization ......................... 12

3 The $\mu$SR Technique 14
   3.1 Time Differential $\mu$SR .............. 14
   3.2 Rotation .................................. 17
   3.3 Time Evolution of the Muon Polarization   . 19
   3.4 The Transverse Field (TF) $\mu$SR Technique 21
   3.5 Longitudinal Field (LF) $\mu$SR .......... 26
   3.6 Zero Field (ZF) $\mu$SR ................. 28
6.5.2 \( b_i^2 \, n(b) \) ......................................................... 123
6.5.3 \( b_z \, n(b) \) .......................................................... 125
6.5.4 \( b_y \, n(b) \) .......................................................... 128
6.5.5 \( b_x \, n(b) \) .......................................................... 132
6.5.6 Other Moments ..................................................... 132
6.6 The Direction of \( B \) ...................................................... 132

7 Conclusions ................................................................. 138

A Field Calculation Program ........................................... 141

B Simulation Calculation Program .................................... 153

C Simulation Analysis Program ....................................... 171

D TRIUMF Data Analysis ............................................... 181
   D.1 The TRIUMF Experimental Arrangement ................. 181
   D.2 The TRIUMF Data .............................................. 182
   D.3 Analysis .......................................................... 183
   D.4 Further Obstacles .............................................. 185

E TRIUMF Data Analysis Program .................................... 188

Bibliography ............................................................... 231

Vita ................................................................................. 237
Acknowledgements

This work has taken many years to complete, and it could not have been accomplished without the help and guidance of the following people:

First and foremost, I wish to thank my advisor, Dr. Jack Kossler, for his constant dedication and excitement about my work. Without his knowledge and insight this thesis would not have been completed.

Second, I would like to thank the rest of my graduate committee for their time in reading the thesis and offering suggestions. In addition: Dr. Ken Petzinger for various theoretical discussions, and for trying to keep me mathematically honest; Dr. Harlan Schone for illuminating troublesome concepts and ideas; Dr. Mark Hinders for agreeing to take this responsibility on without knowing fully what I had done; and Dr. Carey Stronach for being a friend while I was at TRIUMF, as well as for agreeing to make the trip to Williamsburg and serve on the committee.

Third, I would like to thank the faculty of the Physics Department of William and Mary for their kindness and friendship over the last (?) years. I believe I obtained an excellent education from them.

Fourth, I would like to thank the “support” people of the department. Ed Lawrence, Paula Perry, Dianne Fannin, and Sylvia Stout all deserve my deepest gratitude for their making my life here a lot easier.

I would also like to thank the WPSL Machine Shop personnel for their help on those rare occasions when I would bring a hair-brained idea to them with a thumb-nail sketch. To Dr. John Bensel, Mr. Kirk Jacobs, and (of course) Mr. Melvin Woods, I say that it has been a pleasure sharing the basement with you.

My friends here at William and Mary are also deserving of thanks. This includes
graduate students past and present in physics, computer science, American studies, and others. A special thanks goes to Daniele Gaetano, president of Hurricane Graphics, Inc., for developing his program VORTEX just when I needed 3-D plots and contour plots. All of the 3-D and contour plots in this thesis were made using this program.

The people of TRIUMF in Vancouver, Canada, deserve my thanks for allowing me to come out there in July, 1993, to participate in $\mu$SR experiments for two weeks. It was an experience I will not soon forget.

Lastly, I would like to thank my family – my parents, sisters, grandparents, aunts and uncles – for their support while I toiled away at something which was often very strange and foreign to them. It is always nice to know that you've got good, solid backing on the home front.
List of Tables

6.1 These are the results of applying the moment method to simple field distributions directed along $z$, $45^\circ$ to each axis, and again along $z$, for $B = 295.2 \, G$. The left side shows the components within the FLL, and the right shows the averaged values resulting from the analysis. . . . 136

6.2 These are the results of applying the moment method to the different simulation data. The top results are for the $B = 100 \, G$, $\lambda = 256.5 \, nm$, $\theta = 45^\circ$, and $\Gamma = 25$ case. The middle results are for the alternate simulation data: $B = 250 \, G$, $\lambda = 120 \, nm$, $\theta = 45^\circ$, and $\Gamma = 25$. The last data are for $B = 1000 \, G$, $\lambda = 256.5 \, nm$, $\theta = 45^\circ$, and $\Gamma = 25$. . . . 137
List of Figures

2.1 A section of the blueprint from the 1990-1991 proposed Brookhaven National Laboratory muon beam line. Shown is the production target T, the bending magnet B1, the channel, the wedge bending magnet B2, and the sample target TGT. Not shown is a separator/spin rotator, which was not a part of this beam line but is a part of most muon beam lines. .................................................. 6

2.2 A schematic of positive pion decay into a positive muon and its associated neutrino (in the rest frame of the pion). Momenta p and spin directions S are labeled. ................................. 8

2.3 Muon decay in the muon rest frame. This situation depicts the maximum positron energy case, where the positron and the neutrinos go off in opposite directions. Case a is not observed in nature, while case b is. 11

3.1 A typical experimental setup for $\mu$SR experiments. Muons enter at left and encounter the scope detector S, the target m, and possibly the veto counter V. B and F are backward and forward positron detectors, and the coils produce an external magnetic field. .................. 15

3.2 A schematic of a typical $\mu$SR detection system configured to logically reject bad events as discussed in the text. Taken from Ref. [6]. .... 16

3.3 The torque produced on a magnetic moment when experiencing a magnetic field $\mathbf{B}$. .................................................... 18

3.4 The torque on a spinning magnetic moment causes it to precess about the direction of the magnetic field. ................................. 19
3.5 General orientations for the muon initial polarization \( P(0) \), the average field \( B \), the local field \( b \), and the crystal direction \( c \) ........................................ 20

3.6 A \( \mu SR \) time histogram showing the muonic lifetime of 2.19\( \mu s \) and a precession signal. Also shown is the "negative time" data used in background calculation ............................................. 24

3.7 Data in the asymmetry representation .................................................. 25

3.8 Schematic of forward \((N_F)\) and backward \((N_B)\) histogram data for a non-magnetic sample under a LF-\( \mu SR \) study. Also shown is the muonic decay curve ............................................. 26

3.9 Theoretical calculations of equation showing how an increasing applied field causes a decrease in the dephasing of the muons ............................................. 27

3.10 A plot of the static Kubo-Toyabe relaxation function. Note the long time recovery to \( \frac{1}{3} \) ............................................... 29

4.1 A sketch of the magnetic field behavior of a type I superconductor. During the linear part of the graph there is no \( B \), so \( H = -4\pi M \). At the critical field \( H_c \) the superconductor can no longer expel the field, so it becomes completely normal ............................................. 33

4.2 A sketch of the magnetic field behavior of a type II superconductor. For fields up to \( H_{c1} \) the behavior is like a type I. For larger fields the material enters into the mixed state where flux can enter in the form of filaments of the quantum \( \Phi_0 \), and a flux line lattice is formed. The material is not completely normal until the value \( H_{c2} \) is reached ............................................. 34

4.3 A sketch of the equilibrium magnetic flux line lattice for anisotropic, unaxial superconductors. The real space basis vectors \( a_1 \) and \( a_2 \) are shown ............................................. 35

4.4 A sketch of the unit cell of YBCO .................................................. 37

4.5 Diagram showing the angle \( \theta \) between the average field \( B \) and the crystal axis \( c \) ............................................... 41

5.1 The area within the FLL used in calculating the surfaces shown below ............................................. 52

5.2 The surface corresponding to the \( z \) component of the fields ........ 53
5.3 The surface corresponding to the magnitude field \( b \) at points within a unit cell of the FLL ............................................. 54
5.4 The surface corresponding to the \( x \) component of the fields ........ 55
5.5 The surface corresponding to the \( y \) component of the fields ....... 56
5.6 The contour plot corresponding to the \( z \) component surface. Notice the minimum near the center as well as the saddle points. The core regions have been excluded because the contours became unresolvable. The numbers on this and the following plots are field contour values in Gauss ................................................................. 58
5.7 The contour plot corresponding to the \( x \) component of the fields ... 59
5.8 The contour plot corresponding to the \( y \) component of the fields ... 60
5.9 A sketch of how the grid is handled for the numerical integration. The value of interest is \( M_0 \), and its associated contour line is shown as the darker dotted line. The lighter, diagonal dotted line is that which separates the grid into the two triangles. A given contour can only intersect two of the three sides of each of the two triangles ................. 63
5.10 The magnetic field distribution for the \( b_z \) surface. The parameters used to generate this and the following distributions are the same as those used to generate the surfaces and contours: \( B = 250 \, G; \lambda = 120 \, nm; \theta = 45^\circ; \Gamma = 25 \) ...... 66
5.11 The magnetic field distribution of the \( b \) surface ...................... 67
5.12 The magnetic field distribution corresponding to the \( b_x \) surface. Note that the area to the left of the origin is equal to the area to the right — confirming the areal average of zero ........................................ 68
5.13 The field distribution corresponding to the \( b_y \) surface. Once again the areas to the left and right of the origin are equal .................................. 70
5.14 Field distributions \( n(b_z) \) for various angles \( \theta \) between the average field \( B = 250 \, G \) and the crystal \( Z \) direction. The other parameters are: \( \lambda = 256.5 \, nm \), and \( \Gamma = 25 \). The vertical line at \( f = 3.3875 \, MHz \) indicates the position of the average field .................. 73
5.15 Field distributions $n(b)$ for various $\theta$. The other parameters are the same as in the last figure. ................................................................. 74

5.16 Field distributions $n(b_x)$ for various $\theta$. The other parameters are the same as in the last two figures. ................................................................. 76

5.17 Field distributions $n(b_y)$ for various $\theta$. The other parameters are the same as in the last three figures. .............................................................. 77

5.18 Field distributions $n(b_z)$ for various $B$. The other parameters are: $\lambda = 256.5$ nm, $\theta = 70^\circ$, and $\Gamma = 25$. Note that the horizontal axis is not real - the curves have been shifted to positions closer together for easier comparison. ................................................................. 79

5.19 Shown above are contour plots of $b_z$ surfaces corresponding to three different average field values. From left to right are: $B = 100, 200, 5000$ G. The other parameters are: $\lambda = 256.5$ nm, $\theta = 70^\circ$, $\Gamma = 25$. Note the movement of the minimum, and its effect on the saddle points, as the average field value increases. ................................................................. 81

5.20 Field distributions $n(b)$ for various $B$. The other parameters are the same as in the last figure; here, too, the horizontal axis is not real. . . 82

5.21 Contour plots for the magnitude of $b$ surfaces for $B = 100, 250,$ and $5000$ G. The behavior of the minimum is similar to the $b_z$ contours, but different for the $B = 100$ G case. This position allows for a more concrete saddle point position and yields a peak on the field distribution curve of the last figure. ................................................................. 83

5.22 Shown are the field distributions for $n(b_x)$ as the average internal field is varied as labeled. The variations here are relatively minor and do not indicate the type of behavior exhibited above for $n(b_z)$ and $n(b)$. . . 84

5.23 Shown are the field distributions for $n(b_y)$ as the average internal field is varied as labeled. The variations here are very minor, and the distributions appear to have reached a limit of change by 1000 G, after which no further variation occurs with increasing field. ............... 85

xii
5.24 Shown are the field distributions for $n(b_x)$ as the effective penetration depth is varied as labeled. The values of the other parameters are: 
$B = 250 \, G$, $\theta = 70^\circ$, and $\Gamma = 25$. The variation in width of the distributions theoretically varies as $1/\lambda^2$. ........................................ 87

5.25 A plot of the FWHM of the $n(b_x)$ distributions as a function of the parameter $\lambda$. The points are the FWHM of the calculated distributions, and the curve is a fit to these points. Anisotropic London theory predicts a $1/\lambda^2$ dependence, the fit produced an exponent of $-1.9996$. ........................................ 89

5.26 Shown are the field distributions for $n(b)$ as the effective penetration depth is varied as labeled. The values of the other parameters are: 
$B = 250 \, G$, $\theta = 70^\circ$, and $\Gamma = 25$. The variation in width of the distributions theoretically varies as $1/\lambda^2$. ........................................ 90

5.27 A plot of the FWHM of the $n(b)$ distributions as a function of the parameter $\lambda$. The points are the FWHM of the calculated distributions, and the curve is a fit to these points. Anisotropic London theory predicts a $1/\lambda^2$ dependence, the fit produced an exponent of $-1.995$. ........................................ 91

5.28 A plot of the $n(b_x)$ distributions as a function of the parameter $\lambda$. ........................................ 92

5.29 A plot of the $n(b_y)$ distributions as a function of the parameter $\lambda$. ........................................ 93

5.30 A plot of the width of the $n(b_x)$ distributions as a function of the parameter $\lambda$. The points are the width of the calculated distributions, and the curve is a fit to these points. The fit produced an exponent of $-1.56$. ........................................ 94

5.31 A plot of the width of the $n(b_y)$ distributions as a function of the parameter $\lambda$. The points are the width of the calculated distributions, and the curve is a fit to these points. The fit produced an exponent of $-1.904$. ........................................ 96

5.32 A theoretical oriented sample field distribution for a material like YBCO: 
$B = 1000 \, G$, $\lambda = 256.5 \, nm$, $\Gamma = 25$. The degree of orientation is roughly 90%, meaning that the crystallite $c$ axes fall to within $\pm 5^\circ$ of the intended orientation direction. ........................................ 98
5.33 A theoretical polycrystalline field distribution for a material like YBCO:

\[ B = 1000 \ G, \ \lambda = 256.5 \ nm, \ \Gamma = 25. \]  
Characteristic features are the gradual rise on the left and the long high frequency tail.  

5.34 Shown are two representations of the same field distribution which are Fourier Transformed from \( \mu \)SR data taken at TRIUMF on a bulk YBCO sample. The temperature of the sample is 20 K. The dotted line is the raw transform, with no tweaking. The solid curve is the result of apodization (multiplication in time space) with a gaussian envelope which has a decay constant of 5 \( \mu s \) – this is essentially convolution.

6.1 The definition of geometry and trigonometry for the discussion in the text. There are detectors along the \( x, y \) and \( z \) directions indicated by the clear squares.

6.2 Top is a theoretical field distribution for the following parameter values:

\[ B = 1000 \ G, \ \lambda = 120 \ nm, \ \theta = 0^\circ, \ \text{and} \ \Gamma = 25. \]  
The bottom is the real Fourier transform of this distribution, which corresponds to the time-space or asymmetry representation.

6.3 This is a plot of the lower data of the last figure after being scaled for \( 10^8 \) total events, run through the Poisson distribution random noise generator, and then having the exponential decay of the muon lifetime removed. A real muon histogram’s statistics follow a Poisson distribution, where the error in a particular histogram bin goes as \( N^{1/2} \), where \( N \) is the counts in the bin.

6.4 Top is the resultant distribution for the total events equalling \( 10^8 \). Here the noise is such that the details of the underlying distribution are difficult to determine. The middle panel is for \( 10^7 \) total events, where the details of the line shape are becoming more clear. The bottom is for \( 10^8 \) events, where most features of the original distribution are reproduced.
6.5 Results of a numerical simulation of muons within the FLL of a material like YBCO. The average magnetic field $B = 100 \, G$ and is in a direction given by $\beta = 0^\circ$, $\phi_B = 0^\circ$. The crystal c axis is at $\alpha = 45^\circ$, $\phi_\alpha = 0^\circ$. The initial polarization is at $\gamma = 80^\circ$, $\phi_\gamma = 0^\circ$. The lines correspond to each detector as labeled in the graph.

6.6 Plots of the three $\mu$SR signals for two different orientations of $P(0)$: $\gamma = 0^\circ$ and $\gamma = 20^\circ$. In the top figure there appears to be a phase shift between the two signals. This phase shift also appears in the center plot for the $y$ signals.

6.7 The variation in the initial part of the curves as the direction of the muonic polarization is varied through the angles labeled in the figure. Note the phase change in the $x$ and $y$ curves as the angle $\gamma$ increases from $0^\circ \rightarrow 4^\circ$.

6.8 Shown from top to bottom are three simulations of $\mu$SR data from three different directions of the initial muon polarization. The top has $P(0)$ along $x$, the middle along $y$, and the bottom along $z$. The other values of the parameters are: $B = 100 \, G$, $\lambda = 256.5 \, nm$, $\theta = 45^\circ$, and $\Gamma = 25$.

6.9 This is the plot resulting from the analysis deriving $\frac{\delta n(b)}{\delta}$ using the simulation data with parameters $B = 100 \, G$, $\lambda = 256.5 \, nm$, $\theta = 45^\circ$, and $\Gamma = 25$.

6.10 The top plot is the moment $n(b)$ resulting from this analysis of the simulation data. The bottom plot is the result for the same moment but with the integration method of last chapter – shown as a check of the method.

6.11 This figure is the same as the last, except now the parameters are: $B = 250 \, G$, $\lambda = 120 \, nm$, $\theta = 45^\circ$, and $\Gamma = 25$. The integration result on the bottom shows more structure, which is well reproduced in the simulation distribution above.
6.12 The top figure is the result for the moment $b^T n(b)$ using the simulation data from this chapter. The bottom plot is the same moment, but arrived at via the integration method developed in last chapter, and is shown to illuminate the validity of the method. .......................... 126

6.13 The top is the simulation result for $b^T n(b)$ for the following parameters: 
$B = 250 \ G$, $\lambda = 120 \ nm$, $\theta = 45^\circ$, and $\Gamma = 25$. The bottom is the corresponding integration result. .......................... 127

6.14 Shown on top is the simulation result for $b_z n(b)$ for the following parameters: 
$B = 100 \ G$, $\lambda = 256.5 \ nm$, $\theta = 45^\circ$, and $\Gamma = 25$. The bottom plot is the corresponding integration method result via the methods of chapter 4. .......................... 129

6.15 Shown on top is the simulation result for $b_x n(b)$ for the following parameters: 
$B = 250 \ G$, $\lambda = 120 \ nm$, $\theta = 45^\circ$, and $\Gamma = 25$. The bottom plot is the corresponding integration method result via the methods of chapter 4. .......................... 130

6.16 Shown on top is the simulation result for $b_y n(b)$ for the following parameters: 
$B = 100 \ G$, $\lambda = 256.5 \ nm$, $\theta = 45^\circ$, and $\Gamma = 25$. The bottom plot is the corresponding integration method result via the methods of chapter 4. .......................... 131

6.17 Shown on top is the simulation result for $b_z n(b)$ for the following parameters: 
$B = 100 \ G$, $\lambda = 256.5 \ nm$, $\theta = 45^\circ$, and $\Gamma = 25$. The bottom plot is the corresponding integration method result via the methods of chapter 4. .......................... 133

6.18 Shown on top is the simulation result for $b_y n(b)$ for the following parameters: 
$B = 250 \ G$, $\lambda = 120 \ nm$, $\theta = 45^\circ$, and $\Gamma = 25$. The bottom plot is the corresponding integration method result via the methods of chapter 4. .......................... 134

D.1 Shown here are the data from TRIUMF run 6730 in the asymmetry representation. The arrow indicates where the “hiccup” occurs between the two apparent frequencies in the data. .......................... 184
D.2 In the top plot is shown the raw data (dots) and the fit to the later time data. In the bottom plot are the raw data (again as dots) and the resulting subtracted data (dotted line). It is hoped that the subtracted spectrum is the signal from the sample only.
Abstract

This thesis is a theoretical, numerical study of the magnetic fields which exist in the anisotropic, high temperature superconductors like \( YBa_2Cu_3O_{7-\delta} \), or YBCO for short, using both the anisotropic London theory and simulations based on existing muon spin rotation techniques. The thesis first describes the muon spin rotation (\( \mu \)SR) techniques, and then gives a brief discussion of superconductivity with regard to the London theory of anisotropic, type II superconductors. Next, numerical results of the application of this theory to YBCO are presented. Three dimensional surface plots of the magnetic field components within the flux line lattice (FLL) are shown, as well as the corresponding contour plots of the fields. Field distributions are calculated from these surfaces, and the graphs are presented. These distributions correspond to the real part of the Fourier transform of the muon histogram, and a comparison between data taken on a polycrystalline sample and the theoretical prediction is made. In addition, variation of the field distributions with parameters such as penetration depth, angle of the average field, and the magnitude of the average field is discussed.

The last part of the thesis is a theoretical study of the behavior of muons which have stopped within a superconductor. The muons are assumed to stop uniformly throughout the FLL area, and the precession of each about its local field is recorded as the projection of its polarization along each of three mutually perpendicular "detectors." The depolarization of these signals as a function of time is an indication of the existence of transverse field components which exist within the FLL due solely to the anisotropy of the material. In order to further investigate these off axis fields, we have developed an extension of the usual \( \mu \)SR techniques, coupled with Fourier analysis, which yields new information. For example, with the proper analysis procedure, one may determine to good precision the direction of the average internal field \( \mathbf{B} \) with respect to the applied field \( \mathbf{H}_a \). Other quantities, which we call moments of the field distribution, may also be determined.
LOW INTERNAL MAGNETIC FIELDS IN ANISOTROPIC SUPERCONDUCTORS
Chapter 1

Introduction

Superconductors have been known about since the turn of the century. Recently there has been a renewed interest with the discovery of the new, high-\( T_c \) materials since 1986[1]. These compounds become superconducting at much warmer temperatures than any previously known. In fact, many of them superconduct at temperatures above the boiling point of liquid nitrogen, making the observation of the transition both accessible and inexpensive. It was obvious immediately that these materials could have a tremendous technological impact, or lead to further materials with even higher transitions. For this reason there has been an intense effort by scientists in both academia and industry to study these materials. The scientific and industrial communities hope to learn what makes these materials work. For, learning how these materials work not only increases mankind's overall knowledge of his world, but could make some person or company quite successful if the information were used and developed correctly.

This thesis is a small part of the scientific community's efforts to better understand the high-\( T_c \) materials. In particular, it is a theoretical and numerical study of anisotropic superconductors using the techniques of muon spin rotation spectroscopy. A phenomenological theory called the London theory is used in its anisotropic form to describe the magnetic fields which can exist within certain kinds of these materials. Once these fields are known, a numerical simulation can be performed by allowing muons to stop uniformly within these fields. The stopped muons' behavior is governed
by the local magnetic environment within the superconductor, and their subsequent
decay into positrons can be modeled numerically, also. It is therefore possible to
predict what detectors would see if muons were to stop within such a magnetic field
distribution. The resulting simulated data is useful in itself, but with further manipu­
lation can be used to obtain a variety of information about the microscopic magnetic
fields which have influenced the muons' behavior.

The thesis is written in the following way. First, a brief introduction to the
fundamentals of muon spin rotation (μSR) is given. This includes where and how
muons are produced, how they are transported from the production target to the
sample, and how they stop in this sample. The third chapter describes how muons
behave in the presence of a magnetic field, as well as the μSR techniques – time
differential, transverse field, longitudinal field, and zero field.

In the forth chapter we switch gears and delve into the world of superconductivity.
Here a brief introduction to the subject is given: penetration depth, coherence length,
and type of superconductor are discussed. The high temperature superconductor
\( \text{YBa}_2\text{Cu}_3\text{O}_{7-\delta} \) (YBCO) is described in some detail because there is much information
about its magnetic behavior available and because it should be properly described
by the London theory. The phenomenological London theory, which is often used
for the calculation of the microscopic magnetic fields in this type of material, is
developed in both the isotropic and anisotropic forms. The anisotropic case is more
fully developed because many of the high-Tc materials (including YBCO) tend to
be anisotropic. In this case, there arises fields which are transverse to the average
magnetic field direction. These fields are the subject of much study later, in chapter 5.
The anisotropic theory is also presented as a prescription for the numerical calculation
of the fields at any point within the superconductor.

Chapter five describes the implementation of this prescription and its results. The
results, which are calculated for parameters close to those of YBCO, are in the form of
magnetic field surfaces, contours, and distributions. A study of the fields (including
the transverse components mentioned above) as functions of various parameters is
presented both for interest and as a check on the proper working of the program.

Chapter six developes a procedure for using the fields calculated in chapter five
to do a numerical simulation of muons stopping in the superconductor. The results of the simulation give information about the fields experienced by the muons after they stop. Further, the simulation results are then used in a novel extension of the usual $\mu$SR techniques to yield even more information about the fields. We can obtain what we call moments of the magnetic field distribution, which can tell us about the transverse fields mentioned above. In addition, we can get a handle on the direction of the average internal magnetic field $B$, which in general is not in the same direction as the applied field $H_a$. 
Chapter 2

Fundamentals of $\mu$SR

This chapter briefly discusses some of the basic concepts and background of $\mu$SR. Topics such as the production of pions, the production of muons from these pions, and the stopping of the muons in a sample are described. The two major types of beam lines are introduced, and the kinematics of the particles in each type are derived. Finally, the effects of stopping on the muon polarization are discussed.

2.1 A Little History

In 1957 the groups consisting of Garwin, Lederman, and Weinrich[2] and Friedman and Telegdi[3], working independently and concurrently verified the theory of Lee and Yang [4, 5] of the failure of conservation of parity and charge conjugation in weak decays. These experiments were carried out by studying the decays of positive and negative muons. It was found, among other things, that the positive muon had a rather large decay asymmetry — i.e. it decays preferentially in the direction of its spin. This led to the suggestion that muons could be used to probe internal magnetic fields in materials. Thus $\mu$SR was born.
2.2 What is $\mu$SR?

$\mu$SR stands for Muon Spin Rotation, Relaxation, or Resonance. The different R's correspond to different uses of muons. Rotation means the precession of the muon's spin about the local magnetic field where it sits. Relaxation refers to the spin-lattice relaxation rate ($T_1$) and the spin-spin relaxation (depolarization) rate ($T_2$) which can be measured with various $\mu$SR techniques. Resonance is analogous to NMR, where a radio frequency (RF) electromagnetic field is applied to the sample, causing transitions between the hyperfine energy levels of the muon. For the proper field values this reduces the decay asymmetry and thus allows the determination of the hyperfine energy levels at the muon's local environment.

The basic $\mu$SR procedure is as follows, described with reference to Fig. 2.1, which was the proposed $\mu$SR beam line for Brookhaven National Laboratory in 1990-1991. Positive muons are produced at or near a target T (Platinum, in this case) in a beam line in one of a variety of ways discussed below. The beam line usually has a bend or two (B1 and B2), as well as a separator or two (not shown), to allow the selection of muons from other particles which may be in the beam line. After selection, the muons are magnetically focused into a small spot at the target, labeled TGT in the figure. (There is much equipment associated with the target apparatus not shown here. It will be discussed later below.) Once stopped in the target, a muon's behavior will be governed by the local magnetic (and possibly thermal) environment within the material. At some later point (0 – 10 $\mu$s) the muon decays into a positron ($e^+$) and two neutrinos ($\nu$). The positron is detected and the muon’s life-time is scored in a histogram. Analysis of the time histograms yields information about the target material, as will be discussed in some detail below.

This work will deal almost exclusively with muon spin rotation, but readers interested in the other techniques, as well as more on rotation, can find information in references [6, 7, 8].
Figure 2.1: A section of the blueprint from the 1990-1991 proposed Brookhaven National Laboratory muon beam line. Shown is the production target T, the bending magnet B1, the channel, the wedge bending magnet B2, and the sample target TGT. Not shown is a separator/spin rotator, which was not a part of this beam line but is a part of most muon beam lines.
2.3 Pion and Muon Production

Most muons which occur naturally are the result of the decay of a pion which has been produced due to the high energy nuclear collision of a cosmic ray with a particle in the earth's upper atmosphere. These muons shower the earth, but their flux density is too low for experimentation. Therefore muons are most often produced by re-creating the cosmic ray collision in a laboratory. The following reactions between proton beams and target nuclear protons result in pions:

\[
p + p \rightarrow p + n + \pi^+ \\
p + n \rightarrow n + n + \pi^+ \\
\rightarrow p + p + \pi^-
\]

Pions can also be produced with high energy electrons. Upon hitting the target, the electrons produce other electrons, positrons, and photons which form a shower of particles. Photons of the right energy (\(\geq 300 \text{ MeV}\)) can interact with nuclei in the target and produce pions. Beams have been produced by this means at both Saclay and NIKHEF, and the possibility of such a muon beam line at CEBAF is discussed in a report by Kossler [9].

The lifetime of the charged pion is approximately 26 ns, at which time it spontaneously decays via:

\[
\pi^+ \rightarrow \mu^+ + \nu_\mu
\]

where \(\nu_\mu\) is the neutrino associated with the muon and enters to conserve lepton number. The pion has zero spin, therefore conservation of spin angular momentum insists that the \(\mu^+ - \nu_\mu\) pair must also be zero spin. It is known [10] that neutrinos are spin \(\frac{1}{2}\), left-handed particles – i.e. their spin is anti-aligned with their momentum, or their helicity is \(-1\). The spin of the muon after pion decay must therefore point opposite to its momentum (in the pion rest frame). See Fig. 2.2. This longitudinal polarization of the muon is crucial to \(\mu\)SR, as will become clear below.

Clearly, the muon's energy, momentum, and polarization direction in the LAB are going to depend on how and where it was created. If the pions with little or no
Figure 2.2: A schematic of positive pion decay into a positive muon and its associated neutrino (in the rest frame of the pion). Momenta $p$ and spin directions $S$ are labeled.

kinetic energy near the surface of the target decay, then the muons which decay into the beam line have relatively low energy and are nearly 100% longitudinally polarized. This type of beam is called a surface beam. The kinematics of the surface muons can be worked out using the relativistic invariance of mass and energy equation, as well as conservation of momentum. The results are:

\[
\begin{align*}
 p_\mu &= \frac{(m_\pi^2 - m_\mu^2)c}{2m_\pi} \\
 E_\mu &= \frac{(m_\pi^2 + m_\mu^2)c^2}{2m_\pi} \\
 T_\mu &= \frac{(m_\pi^2 - m_\mu^2)c^2}{2m_\pi}
\end{align*}
\]

where $p_\mu$, $E_\mu$, and $T_\mu$ are the muon's momentum, total energy, and kinetic energy, respectively, and $m_\pi$ and $m_\mu$ are the pion and muon masses, respectively. Using the following values for the masses\cite{11}:

\[
\begin{align*}
 m_\pi &= 139.57\, MeV/c^2 \\
 m_\mu &= 105.67\, MeV/c^2
\end{align*}
\]
one finds:

\[ p_\mu = 29.79 \text{ MeV/c} \]

\[ E_\mu = 109.78 \text{ MeV} \]

\[ T_\mu = 4.12 \text{ MeV} \]

The low kinetic energy of the surface muons means that their range in materials is relatively short \( \sim 150 \text{ mg/cm}^2 \) in air\[8]. This is both a problem and a blessing. It is a problem in that it is more difficult to deliver the muon to the target through windows and scintillators – these must be relatively thin as seen by the muon. However, the blessing is that, when constructed properly, the surface beam allows many more muons to stop within a less massive target than with a decay beam. Surface beams are therefore the line of choice for those doing muonic studies of liquids and gases, as well as those studying smaller, less massive solid samples.

A decay beam, by contrast, consists of muons which are produced by those pions which exit the target with higher energy. The momentum most often selected is that corresponding to backward muons – those which come out of the decay in a direction opposite to that of the parental pion’s momentum. The kinematics of these muons can also be worked out using the same conservation laws as above, except now \( p_\pi = p_\mu + p_\nu \). The muon’s momentum is:

\[ p_\mu = \frac{(m_\pi^2 - m_\mu^2)(p_\pi^2 + m_\pi^2 c^2)^{1/2} \pm p_\pi(m_\pi^2 + m_\mu^2)}{2m_\pi^2} \]

where (+) is for forward and (−) is for backward muons. Taking a representative value for \( p_\pi \) of 200 \text{ MeV/c}[6], which corresponds to a kinetic energy of 104.7 \text{ MeV}, and putting in the numbers used above yields:

\[ p^f_\mu = 209.37 \text{ MeV/c} \]

\[ T^f_\mu = 128.66 \text{ MeV} \]

\[ p^b_\mu = 105.25 \text{ MeV/c} \]

\[ T^b_\mu = 43.77 \text{ MeV} \]

where superscript \( f \) means forward and superscript \( b \) means backward. Due to the high energy and momentum of the forward muons, the backward muons are preferred.
It should also be noted that the polarization of a backward muon, which in the pion rest frame is opposite to its momentum, is parallel to its momentum in the LAB frame.

### 2.4 Muon Decay

The decay of muons is more complicated because it is a three-body decay:

\[ \mu^+ \rightarrow e^+ + \nu_e + \bar{\nu}_\mu \]

where \( \nu_e \) is the neutrino associated with electrons and \( \bar{\nu}_\mu \) is the anti-neutrino associated with the muon. The energetics for this decay for the case where both neutrinos go off opposite to the positron are found to be:

\[
\begin{align*}
 p_{e^+} &= \frac{(m_\mu^2 - m_e^2)c}{2m_\mu} \\
 E_{e^+} &= \frac{(m_\mu^2 + m_e^2)c^2}{2m_\mu} \\
 T_{e^+} &= \frac{1}{2} m_\mu c^2 - m_e c^2 \left( 1 - \frac{m_e}{2m_\mu} \right) = 52.32 \text{MeV}
\end{align*}
\]

On average the positron energy is closer to 35MeV, due to the distribution in angle between decay neutrinos.

The direction of the emitted positron with respect to the muon’s polarization can be understood in detail by calculating the decay probability for a positron of energy between \( \epsilon \rightarrow \epsilon + d\epsilon \) into an angle between \( \theta \rightarrow \theta + d\theta \). This has been done \([12, 13]\), and the result is:

\[
dW^\pm(\epsilon, \theta) = \frac{G^2 m_\mu^2 (3 - 2\epsilon)(1 + (1 - 2\epsilon)/((3 - 2\epsilon) \cos \theta)\epsilon^2}{192\pi^2} d\epsilon d(\cos \theta) \tag{2.1}
\]

where \((-\) is for \( \mu^- \), \((+\) is for \( \mu^+ \), \( \theta \) is the angle between the muon polarization and the momentum of the outgoing positron, \( \epsilon \) is the reduced energy \( E_\mu/E_{\text{max}} \), and \( G = 1.166 \times 10^{-5} \text{GeV}^{-2} \) is the Fermi coupling constant. The important aspect of this equation for our discussion is the \( \cos \theta \) term. It states that the distribution in angle of the emitted positron is a \( \mp \) function of energy multiplying \( \cos \theta \). Therefore the
Figure 2.3: Muon decay in the muon rest frame. This situation depicts the maximum positron energy case, where the positron and the neutrinos go off in opposite directions. Case \( a \) is not observed in nature, while case \( b \) is.

direction of the emitted positron for positive muons is mainly forward, and for the maximum energy case it is exactly forward. This can be understood qualitatively from looking at the maximum energy situation as depicted in Fig. 2.3. The neutrinos' spin directions cancel by virtue of their being a particle-antiparticle pair. This means that the positron spin direction must be the same as that of the muon. Case \( a \) shows the positron coming off with momentum opposite to the muon spin direction; hence it has negative helicity. Case \( b \) shows the positron coming off with its momentum parallel to that of the muon spin, and hence it has positive helicity. If parity were conserved, then both types of decay would be equally likely. However, case \( a \) is not observed in nature; therefore parity is not conserved in this decay and all of the positrons come off with positive helicity in directions which are mainly forward, as stated above. For this case the angle \( \theta \) is zero and \( \epsilon = 1 \), and the \( \cos \theta \) term in equation 2.1 is 2.
2.5 Thermalization

It was mentioned above that knowing the muon polarization is of crucial importance. First, we have been able to understand the origin of the muon polarization from pion decay, and that its direction parallel to the muon momentum is largely maintained throughout the beamline and up to the target. Second, we know that after stopping, the muon decays into a positron which exits (to first order) in the direction of the muon’s spin at the time of decay. It therefore remains to discuss the effects of thermalization—the stopping of the muon in the target—on the muon polarization. First thermalization will be discussed, and then its effects on the polarization will be described.

One can imagine that a muon impinging on a material would get knocked about ferociously and lose any sense of order it previously had. However, it must be remembered that we are dealing with point particles and that their interactions, which involve electrostatic forces only (spin independent), occur at very small time scales over small distances. The traditional view of thermalization is similar to the following[7, 6, 14]. Initially the muon loses energy via ionizing atoms and scattering off of electrons. This takes place until the muon has roughly $35\, keV$ or less of kinetic energy and takes a time of the order $10^{-10} - 10^{-9}s$. At this point it is believed that muonium forms—an atom consisting of one muon and one electron—and breaks and reforms until the energy of the muon is about $100\, eV$. This step requires roughly $10^{-13}\, s$, and the velocity of the muon is roughly $0.002c$. Lastly, the muonium becomes relatively stable, losing energy by colliding elastically with host atoms and inelastically with phonons down to energies of about $15\, eV$. At this point one of two things happens. If the material is a semi-conductor or an insulator the muon will tend to become a part of the molecular make-up of the system with its electron chemically bound with others[14]—i.e. it remains as muonium. If the material is a metal, then the muon becomes free from the muonium electron and is screened by the free charges. (In both cases one should note that when the positive muon/muonium stops, it stops interstitially in the material.)

The polarization of the muon is affected during all of this, but the effects are
negligible. It was mentioned above that during the initial phases of stopping there is scattering from electrons. After calculating cross sections for scattering of longitudinally polarized muons from unpolarized electrons, Ford and Mullin [15] found that the depolarization of the muons is proportional to the fractional energy loss due to the scattering:

\[ \text{Depol} \propto U = \frac{m_e \beta^2 \sin^2 \alpha}{2} \]

where \( U \) is the fractional energy loss, \( \beta = v/c \), and \( \alpha \) is the center of momentum scattering angle. The key term is \( \beta^2 m_e/m_\mu \), which is very small and gets smaller as the muon slows. Therefore electron scattering has small depolarization effects on the muon.

The middle stages of thermalization are the muonium stages, where the muon is cycling through periods of being bound and free and its speed is similar to the orbital speed of valence electrons. The amount of depolarization occurring here depends on how long the muon is in a state of muonium, which in turn depends heavily on the density of the material. That is, denser materials like metals cause more collisions, and hence more cycles of bound/free for the muon, in a shorter time. Therefore the amount of time the muon is bound in muonium, which primarily causes depolarization, is less in dense materials and the net loss of depolarization is less. The depolarization in liquids and gases is significantly higher [7] due to their lower density.

The last and lowest energy stage also causes little depolarization in metals. Once again the time period is extremely short, and the conduction electrons present quickly cause any bound muons to become "free," i.e. shielded, leaving the muon with a polarization vector pointing in essentially the same direction as it started at higher energy.
Chapter 3

The $\mu$SR Technique

This chapter describes the $\mu$SR techniques which are employed in most experiments today. Transverse, longitudinal, and zero field geometries are discussed and the kinds of experimental results often obtained are shown. A brief introduction to precession is given and applied to the time evolution of the muon's polarization.

3.1 Time Differential $\mu$SR

Time differential $\mu$SR (TD-$\mu$SR) is the experimental procedure most often employed by experimenters today. This procedure is described in reference to Fig 3.1. A typical TD-$\mu$SR experimental arrangement consists of: Helmholtz coils, plastic scintillators, a target, and the associated vacuum and cryogenic equipment. The elements are labeled in the figure, but the vacuum and cryogenic equipment are left out for clarity.

In the figure, muons enter one at a time from the left and traverse the scope scintillator S. The muon then encounters the target m and either stops (thermalizes) or continues out the back of the target to the veto scintillator V. The scope and veto are connected to a coincidence box which will reject any events corresponding to a muon not stopping. If the muon does stop, the scope pulse becomes a start signal for a time-to-digital converter (TDC). The TDC stop signal is obtained when one of the large plastic scintillators (B,F) around the sample detects the decay positron from muon decay. Typically the TDC signal is stored in a buffer memory and then
Figure 3.1: A typical experimental setup for \( \mu \)SR experiments. Muons enter at left and encounter the scope detector \( S \), the target \( m \), and possibly the veto counter \( V \). \( B \) and \( F \) are backward and forward positron detectors, and the coils produce an external magnetic field.
Figure 3.2: A schematic of a typical $\mu$SR detection system configured to logically reject bad events as discussed in the text. Taken from Ref. [6].

periodically transferred to a computer in which histograms are later generated and used for analysis.

A typical schematic of a $\mu$SR detection and electronics system is shown in Fig. 3.2. The electronics is configured to take care of certain contingencies which may arise. For example, the TDC will accept a stop signal up until a certain time has elapsed (a time window usually about $5\tau_{\mu} \sim 10\mu s$). If there is no positron after this time the whole event is thrown out and things start over.

Other possibilities include two muons stopping before the window has expired. There are two cases for this. In one, the second muon arrives before the first decays – an obvious problem in distinguishing which positron comes from which muon. In the other, the second muon comes after the first decays, but still within the time window. For this case the second muon may decay before the window expires, or not. This would artificially enhance the short time part of the histogram. Therefore any time
window in which two muons enter causes the event to be discarded.

In addition, accidental events may occur. For instance, a cosmic ray (muon) may be interpreted as a decay positron. This will be stored as a real event as part of the histogram. However, this is a random event and may happen at any time. It is therefore a part of a uniform, random background upon which sits the actual histogram. It is like a DC offset for an AC signal, and can, in principle, be taken care of during analysis.

Finally, it must be realized that the electronic rejection of events causes the overall count rate to suffer. The rate of second muons entering the apparatus \( I_{\text{second}} \) during the time window \( \Delta T \) is given by\(^7\):

\[
I_{\text{second}} = \Delta T I_{\mu}^2
\]

where \( I_{\mu} \) is the incident muon stop rate. If \( \Delta T = 10 \mu s \) and we wish to keep the percentage of two muon rejections \( I_{\text{second}}/I_{\mu} \) to no more than 5%, then the overall count rate upper limit is \( 5 \times 10^3 \text{ s}^{-1} \). This type of limitation is simply an experimental fact of life, and must be lived with in time differential \( \mu \)SR.

### 3.2 Rotation

As mentioned above, this work concerns muon spin \textit{rotation}, and hence the following discussion proceeds from that perspective. In particular, it is the ability of the muon to rotate, or, more correctly, \textit{precess}, which makes it a useful tool in the study of materials. A discussion of the origin of this rotation follows, accompanied thereafter by brief discussions of various \( \mu \)SR techniques.

Of the many characteristics of muons, those important for understanding rotation are \textit{spin} and \textit{magnetic moment}. Quantitatively, the magnetic moment of the muon \( \mu_{\mu} = 1.001165932 \frac{eA}{2m_{\mu}c} \) [11], and its spin \( S_{\mu} = \frac{1}{2} \). Qualitatively, the knowledge of 1/2 spin allows the use of a vector picture to describe the muon's behavior. Any magnetic moment, be it a particle or a bar magnet, will experience a torque when placed in a magnetic field. The torque will act to orient the magnetic moment vector so that it
Figure 3.3: The torque produced on a magnetic moment when experiencing a magnetic field $B$. is parallel with the field – Fig. 3.3. Mathematically, the situation is described by:

$$\tau = \mu \times B$$

If the moment is spinning, then the torque acts so as to cause the magnetic moment vector to precess, or rotate, about the direction of the magnetic field. This is depicted in Fig. 3.4. We know from classical mechanics that the torque equals the time rate-of-change of the angular momentum, which in this case is spin angular momentum: $\tau = dS/dt$. It is also true that the magnetic moment is proportional to the spin via:

$$\mu = \gamma S$$

where $\gamma$ is the gyromagnetic ratio of the spin under consideration. Putting this together yields:

$$\frac{dS}{dt} = \gamma S \times B$$

This equation defines the direction of precession of the moment about the field, as seen in Fig. 3.4. From geometry we see that $|dS| = S \sin \theta d\phi$, and using $|a \times b| = ab \sin \alpha$
Figure 3.4: The torque on a spinning magnetic moment causes it to precess about the direction of the magnetic field.

we get:

\[ S \sin \theta \frac{d\phi}{dt} = \gamma S B \sin \theta \]

which leaves:

\[ \frac{d\phi}{dt} = \gamma B = \omega \]

This is the frequency of precession of the moment about the field. The linear relation between field and precession frequency is of vital importance and forms the basis for the use of muons as probes of magnetic field distributions.

3.3 Time Evolution of the Muon Polarization

In order to describe the use of muons as probes of magnetic field distributions we need to develop the time evolution of the polarization vector. This is done by finding its components as functions of time along three mutually perpendicular directions. The results of this development will aid in the description of the various $\mu$SR experimental techniques in the following sections.
Figure 3.5: General orientations for the muon initial polarization $P(0)$, the average field $B$, the local field $b$, and the crystal direction $c$.

We first consider a muon stopped in some material with its polarization pointing in a general direction — Fig.3.5. There exist various definable directions. One is the average field $B$ direction, which we also define as the $z$ direction. The second is a crystalline direction of the material, termed $c$. $c$ is defined to lie within the $x - z$ plane with polar angle $\theta$, thereby determining the coordinate system axes. There exist a local field $b$ at the muon site which in general is not parallel to the average field $B$. The direction of $b$ is given by the polar angle $\delta$ and azimuthal angle $\phi_b$. Lastly, the polarization is initially at a polar angle $\alpha$ and azimuthal angle $\phi_p$.

To find the time evolution of the muon's polarization, we break the polarization vector into two components, one parallel to $b$ and one perpendicular to $b$. The one parallel to $b$ is independent of time and is found by the dot product $P(0) \cdot \hat{b}$. The perpendicular component is obtained from $P(0) \times \hat{b}$ and $P(0) - P(0) \cdot \hat{b}$, and rotates at an angular frequency $\omega = \gamma b$. These two components can then be projected onto any direction. A compact representation for the polarization as a function of time is
via the following equation:

$$P(t) = \left( \frac{P(0) \cdot b}{b} \right) \frac{b}{b} + \left[ P(0) - \left( \frac{P(0) \cdot b}{b} \right) \frac{b}{b} \right] \cos \omega t + \left( \frac{P(0) \times b}{b} \right) \sin \omega t \quad (3.2)$$

To find a component of the polarization one simply dots this expression into a direction of interest.

### 3.4 The Transverse Field (TF) \(\mu\)SR Technique

The most popular TD-\(\mu\)SR technique is the transverse field technique. The experimental arrangement in this case is one where the muon's initial polarization direction is perpendicular to an externally applied magnetic field. This situation is depicted in Fig 3.1, with the muon's polarization in the plane perpendicular to the field applied by the coils.

As discussed above, the muon will precess in this plane (neglecting local effects) until it decays. The probability of the muon still being around at a time \(t\) is \(n(t) = e^{-t/\tau_w}\). From equation 2.1 we see that the muonic decay as a function of angle is of the form \(1 + a \cos \theta\). If now we let the muon precess at a frequency \(\omega\) we can rewrite this equation as \(1 + A_s P(t)\), where \(A_s\) is an initial asymmetry and \(P(t)\) is from equation 3.2. The precession will cause modulations in the decay curve, yielding the expression:

$$n_d(t) = n_e e^{-t/\tau_w} [1 + A_s P(t) \cdot \hat{d}] \quad (3.3)$$

where the term \(P(t) \cdot \hat{d}\) is now the projection upon a detector direction \(\hat{d}\). In experiments the muons stop at different sites within the target, and in general will sample many local fields \(b\). The term \(P(t) \cdot \hat{d}\), while strictly correct for one muon, will for an experiment correspond to an average over the sample volume, or, equivalently, the field distribution. That is, the net or average polarization vector \(< P(t) >\) will precess about some average field \(< B >\) at a frequency \(< \omega > = \gamma_m < B >\) and have a time dependent amplitude which is found by:

$$g(t) = \int n(B) P(t; B) dB \quad (3.4)$$
where \( n(B) \) is the magnetic field distribution in three dimensions throughout the volume of the sample. The histogram equation for many muons now looks like the following (with the experimental background term added):

\[
N(t) = N_0 e^{-t/\tau}\left[1 + g(t)\right] + B
\]

Defining \( g(t) \equiv A_0 G(t) \cos(<\omega > t + \phi) \) gives the alternative expression:

\[
N(t) = N_0 e^{-t/\tau}\left[1 + A_0 G(t) \cos(<\omega > t + \phi)\right] + B
\]

where \( \phi \) is an initial phase angle which can take into account both polarization misalignment and electronic dead time, and \( B \) is the background term. Note here that \( G(t) \) for a TF experiment is the envelope of the asymmetry function, the so-called relaxation function.

In many materials where there are static fields, the assumption of a gaussian field distribution for the \( z \) component at the muon is valid, and for a large enough transverse field the effects of the local fields can be ignored. This leaves a one-dimensional integral:

\[
g_x(t) = \int n(B_z)P_x(t; B_z) dB_z
\]

where now:

\[
n(B_z) = \frac{\gamma\mu}{\sqrt{2\pi}\Delta_z} \exp\left(-\frac{\gamma^2 B_z^2}{2\Delta_z^2}\right)
\]

where \( \Delta_z \) is the width or second moment of the gaussian distribution, and from equation 3.2: \( P_x(t; B_z) = \cos(\gamma\mu B_z t) = \text{Re} \exp(i\gamma\mu B_z t) \). This results in a \( G_x(t) \) which looks like:

\[
G_x(t) = \exp\left(-\frac{\Delta_z^2 t^2}{2}\right)
\]

This development demonstrates the important concept that the observed function \( g_x(t) \) is related to the magnetic field distribution \( n(B_z) \). Since \( B \) and \( \omega \) are essentially the same thing, we see that it is the Fourier Transform which relates the two quantities. As an example it will now be shown that \( n(B_z) = FT(g_x(t)) \):

\[
n(B_z) = \int_{-\infty}^{\infty} e^{-i\omega t} g_x(t) dt = \int_{-\infty}^{\infty} dt \int_{-\infty}^{\infty} e^{-i\omega t} n(B_z') P_x(t; B_z') dB_z'
\]
plugging in from equation 3.2 for $P_z(t; B'_z)$ we get:

$$n(B_z) = \int_{-\infty}^{\infty} dB'_z \int_{-\infty}^{\infty} n(B'_z) e^{-i(\omega_z - \omega'_z)t} dt$$  \hspace{1cm} (3.11)

Recognizing that:

$$\delta(\omega_z - \omega'_z) \equiv \int_{-\infty}^{\infty} e^{-i(\omega_z - \omega'_z)t} dt$$  \hspace{1cm} (3.12)

leaves:

$$n(B_z) = \int_{-\infty}^{\infty} n(B'_z) \delta(\omega_z - \omega'_z) dB'_z$$  \hspace{1cm} (3.13)

which ends the proof.

It is therefore possible with this $\mu$SR technique to almost directly measure the field distribution in metals. This has been the fundamental contribution of $\mu$SR. All other quantities extracted from $\mu$SR data have been found via indirect methods and/or with assumptions about the field distribution. (Later in this work we will develop a technique which will allow the experimenter to measure the field distribution without using a gaussian, or any other, assumption about the field distribution.) Of course, even in the above proof a gaussian field distribution was assumed. This has shown to be a quite valid assumption in metals where there are static dipolar fields at the muon (i.e. the TF relaxation function has a gaussian envelope). This assumption breaks down when the muon hops or, equivalently, when there is field motion. For motion there are other theories and analysis techniques. The interested reader can peruse references [7, 16].

Questions have also been raised as to how well the Fourier Transform of $g(t)$ can reproduce the actual field distribution. What does one get when the gaussian assumptions are not made? How do phase angles from possible misalignments of detectors and/or the initial polarization of the muon affect the transforms? These questions have been addressed, and in an effort not to re-invent the wheel the reader is referred to the PhD thesis of Riseman [17].

An actual TF histogram is shown below in Fig. 3.6, where a gaussian-like envelope of the oscillations is evident. Note the early time part of the graph. This is the electronics dead time - the time it takes the logic to decide whether or not it has a
Figure 3.6: A μSR time histogram showing the muonic lifetime of 2.19 μs and a precession signal. Also shown is the "negative time" data used in background calculation.
muon. This part of the spectrum is rarely displayed, but is often used in determining the random background for each detector.

Knowledge of the background (before fitting) is mainly used in an alternative representation of the data in which paired histogram data (i.e. F and B) are combined to allow the direct extraction of $g(t)$:

$$g(t) = \frac{(N_1 - B_1) - \alpha(N_2 - B_2)}{(N_1 - B_1) + \alpha(N_2 - B_2)}$$

where $N_1$ and $N_2$ correspond to the raw data from each histogram, $B_1$ and $B_2$ are the backgrounds, respectively, and $\alpha$ is an experimentally determined parameter meant to correct for physical differences between the detectors (i.e. efficiencies and geometries). Data represented via this scheme are shown in Fig. 3.7.
3.5 Longitudinal Field (LF) $\mu$SR

In LF-$\mu$SR the muon's initial polarization is parallel to the external field, which is usually much smaller than in a TF-$\mu$SR experiment. The positron detectors are generally placed along an axis parallel to the initial muon polarization direction.

The histograms can be represented by:

$$N(t) = N_0 e^{-t/\tau} [1 + A_e G_z(t) \cos \phi] + B$$

where $\phi$ is usually 0° or 180°, depending on which detector one is viewing. The shape of $G_z(t)$ will depend on the type of sample under study. If the sample is magnetic, then $G_z(t)$ will generally have some wiggles as a result of muons precessing about fields which are slightly off of the applied field. For non-magnetic samples there will be no precessing, and the histograms will look roughly like Fig. 3.8.

The static magnetic field distribution is now a shifted gaussian[16]:

$$n(B_z) = \frac{\gamma_\mu}{\sqrt{2\pi}\Delta_z} \exp \left( -\frac{\gamma_\mu^2 (B_z - B_0)^2}{2\Delta_z^2} \right)$$

Figure 3.8: Schematic of forward ($N_F$) and backward ($N_B$) histogram data for a non-magnetic sample under a LF-$\mu$SR study. Also shown is the muonic decay curve.
which yields the following for a relaxation function[6]:

\[
g_s(t) = 1 - \frac{2\Delta^2}{\omega_A^3} \left[ 1 - e^{-\frac{\Delta^2}{2}} \cos \omega_A t \right] + \frac{2\Delta^2}{\omega_A^3} \int_0^t e^{-\frac{\Delta^2}{2}} \sin \omega_A t' dt' \tag{3.17}
\]

where \(\omega_A = \gamma_B B_A\) is the applied field.

The shape of this function depends on the strength of the external field. For small applied fields (\(\omega_A\)) a random distribution of fields causes each muon to precess at different frequencies with differently directed precession cones. Therefore although all muons start in phase, they quickly become dephased and the value of \(G_s(t)\) decreases. If \(\omega_A\) is quite a bit larger than the values of the local fields, then the frequencies and precession cones are all about the same and a much smaller amount of dephasing occurs. This field dependence is shown in Fig. 3.9.

Once again there are effects due to the motion of either the muons or the fields at the muon sites. The interested reader can read references [7, 16].
3.6 Zero Field (ZF) $\mu$SR

ZF-$\mu$SR is an obvious special case of LF-$\mu$SR. The experimental arrangement is exactly the same as before, only now no field is applied. The muons are only subject to the fields intrinsic to the sample.

The longitudinal component of the polarization is obtained from equation 3.2:

$$P_z(t) = \frac{b^2}{b^2} + \frac{b^2}{b^2} + \frac{b^2}{b^2} \cos \omega t$$

which when averaged over all possible $b$ directions (for isotropic field distribution functions in all directions) yields:

$$< P_z(t) > = \frac{1}{3} + \frac{2}{3} < \cos \omega t >$$

If again a gaussian field distribution is assumed then we obtain for the relaxation function:

$$G_s(t) = \frac{1}{3} + \frac{2}{3} (1 - \Delta^2 t^2) e^{-r^2/\Delta^2}$$

which is the famous Kubo-Toyabe formula [18].

An interesting aspect of this equation is that at long times it approaches a value of $\frac{1}{3}$, which is a keynote or signature of static gaussian field distributions and shown in Fig. 3.10. It should be noted that this $\frac{1}{3}$ recovery has come into question for some materials under certain conditions where longer time oscillations have been calculated [19, 20].

The usual sources can be consulted if the reader is interested in the motional aspects of this technique.
Figure 3.10: A plot of the static Kubo-Toyabe relaxation function. Note the long time recovery to $\frac{1}{3}$. 

\[ G(t) \]
Chapter 4

Superconductivity

Some of the basic ideas of superconductivity are discussed in this chapter. The penetration depth and coherence length are introduced, as well as the ideas of type I and type II superconductors and their corresponding phase diagrams. The high-$T_c$ superconductor YBCO is shown as a particular example of a type II material. The isotropic London theory is then introduced as a starting point for calculating the microscopic fields within the mixed state of the type II materials. This theory is then extended to anisotropic materials, where a prescription is given for calculating the microscopic field components at points within the equilibrium, anisotropic Flux Line Lattice (FLL). Lastly, the assumptions and limitations of the London theory are addressed.

4.1 Introduction

The first superconductor was discovered in 1911 by a Dutch physicist, Kammerlingh Onnes [21] (actually his graduate student), just three years after he became the first person to liquify helium. The effect Onnes found was that of zero resistance to current flow, which various elements displayed, each at their own characteristic critical temperature $T_c$. In all, 26 elements (and 11 others under special conditions)[22] as well as a whole host of inorganic and organic compounds are superconductors.

Later, in 1933, another effect of superconductivity was discovered by Meissner and
Ochsenfeld [23] - the Meissner effect. A superconductor cooled in a magnetic field will expel the field as it is cooled below its critical temperature. This leaves the material with zero field inside except for in a small layer near the surface where superconducting currents flow. That is, the superconductor becomes a perfect diamagnet, and the applied field lines must bend around the sample. This state exists until either the sample warms above $T_c$ or the field increases above a critical value.

A third characteristic of superconductors is the existence, at least in some form, of an energy gap $\Delta$. An energy gap for superconductors manifests itself as $\frac{1}{3}$ of the energy needed to break a pair of particles which have condensed into a superconducting state. The existence of a gap means that only energies above twice the gap value are sufficient to break the pair, and that there should be an absence of low energy excitations within a pure material. Most elemental superconductors exhibit behavior consistent with an energy gap which is isotropic, or $s$-wave, which means that the surface in momentum space is spherical, and there exists a gap in all directions between the valence and conduction bands. Recently, much research using various techniques (including $\mu$SR) has been done in an attempt to probe the energy gap of new superconducting materials, such as $YBa_2Cu_3O_7-\delta$ and $La_{1.85}Sr_{0.15}CuO_4$ and its related compounds. (The first compound, termed YBCO, will be the subject of much discussion in this work). It was initially generally held that they exhibited $s$-wave properties [24, 25], but then a $d$-wave picture arose [26]. This theory allows for nodes in the gap in certain directions - i.e. directions where pairs can be broken with no energy. At present there is no clear, unambiguous body of experimental evidence supporting either model. The following references, on both sides of the argument, are cited for the interested reader [27, 28, 29, 30]. A good theoretical development of the origin of some pairing states is given in [31] and the references cited therein.

A monumental achievement on the theoretical side came in 1957, when Bardeen, Schrieffer, and Cooper published their Nobel Prize-winning work on the microscopic theory of superconductivity [32]. This was the first successful, non-phenomenological theory of superconductivity. It describes how an attraction between electrons can arise within a background of free electrons, allowing them to pair up into Cooper pairs, and form a part of a ground state wave function in which all electrons may be
bound. This theory is quite robust, and is still used as a "yard stick" by which other theories and experimental results are measured. Due to its robustness, as well as its complexity, the details of the BCS theory are beyond the scope of this work. The interested reader can consult the original work[32], but will probably have better luck with other people's interpretations[33, 34].

In 1986, researchers at IBM's Zürich laboratory discovered a material which became superconducting at a $T_c$ of $\sim 30-35 \, K[1]$. This discovery shook the scientific community because up until then the highest $T_c$ had been only $23.3 \, K$. Soon after, researchers at the Universities of Alabama and Houston developed the material $YBa_2Cu_3O_7$ (called YBCO) which exhibited the even higher critical temperature of $\sim 92 \, K[35]$, which is above the $77 \, K$ boiling point of liquid nitrogen and therefore can be made to go superconducting more cheaply than those materials with cooler $T_c$'s. Soon after, even more materials were discovered with even higher critical temperatures and even more interesting characteristics.

These materials have caused a lot of excitement in both the academic and industrial research communities throughout the world. It is obvious that there is great scientific and technological potential in the superconductors, and that they deserved a great deal of careful study in order to be understood. A better understanding will not only increase our overall knowledge of our world, but will allow for the best possible applications of the materials that we now have, and will help us to develop even more materials with more useful properties. It is for these reasons that works such as this one are undertaken.

### 4.2 Fundamental Parameters and Superconductor Type

There are two fundamental parameters associated with all superconductors: $\lambda$ and $\xi$. Lambda is the *penetration depth*, the length in the exponential sense in which the magnetic fields die off within a superconductor. $\xi$ is the *coherence length*. It is a fundamental length scale which originated with Pippard[36] and concerns the distance
Figure 4.1: A sketch of the magnetic field behavior of a type I superconductor. During the linear part of the graph there is no \( B \), so \( H = -4\pi M \). At the critical field \( H_c \) the superconductor can no longer expel the field, so it becomes completely normal.

over which local fields within a material have an appreciable effect on the current at a nearby point. The coherence length is often mentioned as a characteristic pair wave packet minimum length, and it is also the order of the size of the flux core in type II superconductors (discussed below). The type of a superconductor is determined by the parameter \( \kappa \), the Ginzburg-Landau parameter, defined as \( \kappa \equiv \lambda / \xi \). Type I superconductors have \( \kappa < 1/\sqrt{2} \), while type II superconductors have \( \kappa > 1/\sqrt{2} \).

Type I superconductors are the classic superconductors — Pb, Sn, Al, etc. Their behavior can be described via Fig. 4.1. For low magnetic fields at a temperature below \( T_c \), \( B = 0 \), and the curve is linear with a slope of 1. Then at a critical field \( H_c \) there is a sharp transition to the normal state. Therefore the magnetic field is either completely expelled or completely allowed within the type I material.

For Type II materials the behavior is as depicted in Fig. 4.2. Here the field is completely expelled up to \( H_{c1} \), but then is allowed to penetrate more and more until an upper critical field \( H_{c2} \) is reached, where the field penetrates completely.
Figure 4.2: A sketch of the magnetic field behavior of a type II superconductor. For fields up to $H_{c1}$ the behavior is like a type I. For larger fields the material enters into the mixed state where flux can enter in the form of filaments of the quantum $\Phi_0$, and a flux line lattice is formed. The material is not completely normal until the value $H_{c2}$ is reached.

region between $H_{c1}$ and $H_{c2}$ is termed the *mixed state* and will be the subject of much discussion in this work.

The mixed state is characterized by magnetic flux entering the samples in units of the flux quantum $\Phi_0 = \hbar c/2e = 2.07 \times 10^{-7} \text{Gcm}^2$. The resulting equilibrium geometry formed is usually that of a triangular lattice, with a flux quantum at each corner of the triangle. See Fig. 4.3. It has been shown that the triangular flux line lattice (FLL) geometry is most often the proper one for minimizing the free energy of the system at higher and intermediate fields [37]. The question of very low fields will be discussed later.

The density of the FLL is a function of the magnetic induction, going as $n = \frac{\Phi}{\Phi_0} \text{cm}^{-2}$. When the density gets very high the flux cores begin to overlap. This costs energy, so the sample becomes normal conducting.
4.3 The $YBa_2Cu_3O_{7-\delta}$ Superconductor

An example of a type II superconductor is the relatively new material $YBa_2Cu_3O_{7-\delta}$, termed YBCO for short. This superconductor was discovered in 1987 by T.C. Chu et al.\[38\] and has hence been possibly the most studied of the high-$T_c$ materials. The term high-$T_c$ refers to the then incredibly high critical temperature that this material exhibited of $T_c = 92$ K, which is about an order of magnitude higher than most type I superconductors. For reference a sketch of the unit cell crystal structure is shown in Fig. 4.4. The dimensions of the cell are: $a = 3.8198$ Å, $b = 3.8849$ Å, and $c = 11.6762$ Å[39]; hence the crystal structure has a pronounced anisotropy between the $c$ axis and the $a$-$b$ directions, which are almost equal. The chains are the $Cu-O-Cu$ rungs labeled in the figure and correspond to the $B$ direction, and the planes – the $CuO_2$ planes labeled in the figure – are of considerable importance because it is believed that superconductivity arises here. The superconducting currents flow most easily parallel to, and with the most difficulty perpendicular to, the planes. Therefore there is not only a crystalline anisotropy to these materials, but a superconducting one as
well. This superconducting anisotropy will be addressed below when a theoretical
description of anisotropic superconductors is developed. The superconducting charge
carriers, in this case holes, are believed to originate in the planes. The crystalline
structure and electrical behavior of the planes show almost no anisotropy within the
plane. It is for these reasons that the material is considered uniaxial.

YBCO is the material chosen for study in this thesis because its properties and
behavior seem to allow a theoretical description within the anisotropic theory just
mentioned. Its properties are well known because of the wealth of information and
knowledge obtained from intense study for many years. These properties will be
discussed later in section 4.9. First, however, we need to develop more on supercon­
ductivity.

4.4 Isotropic London Theory

It was mentioned above that the BCS theory was the first successful rigorous theory
of superconductivity. By contrast, the London theory of isotropic superconductors
is a phenomenological theory which is much easier to understand, and it provides a
simple method for the calculation of the magnetic fields of the FLL.

The theory starts by considering the free energy associated with an isolated vortex
within a superconductor in the mixed state. The vortex has a core of radius $\xi$, and
the associated fields extend out isotropically over a length of order $\lambda$. The term
“isolated vortex” implies that the magnetic situation is near $H_{c1}$, where few vortices
have penetrated the sample.

The isolated vortex can be described by an order parameter (sometimes called a
vortex wavefunction) which has the following form for a vortex: $\psi = \psi_{\infty} f(r) e^{i\theta}$, where
$|\psi|^2 = n_s$ (the superconducting charge carrier density, described further below), $f(r)$
is some function of the distance $r$ from the center of the core, and $f(r) \to 1$ as
$r \to \infty$\[33\]. The angle $\theta$ is the phase of the order parameter, which changes by $2\pi$
with one circuit of the core. It has been shown\[33\] that for superconductors with large
$\kappa$ (the high $\kappa$appa limit) the function $f$ varies quite rapidly within the core region,
rises to a value of 1 at a distance on the order of $\xi$, and remains constant wherever
Figure 4.4: A sketch of the unit cell of YBCO.
there are no vortices. The order parameter therefore follows this behavior, too. This is significant in that if the wavefunction is constant, then the superconducting charge carrier density is also constant over these regions. If this density is not a constant, then a theory which allows for its variation must be used, such as the Ginzburg-Landau theory\[40\]. If the superconducting charge carrier density is constant, then one can use the London theory\[41\], which is discussed next below.

One begins the London theory by writing down the free energy density of the superconductor. There will be a term for the energy density of the magnetic field, given by $b^2/8\pi$. There will also be a term for the kinetic energy density of the superconducting particles. This term is $\frac{1}{2}Mv^2n_s$, where $M$ is the effective mass of the superconducting particle, $v$ is its drift velocity, and $n_s$ is the superconducting charge carrier density ($cm^{-3}$). It is important to note again that the charge carrier density is assumed constant, except for right at the vortex core, where it is zero – if this is not true then the London theory will not hold.

We now use the equation for current density:

$$J = n_s ev$$  \hspace{1cm} (4.1)

where $e$ is the charge per particle. This yields for the kinetic energy density:

$$E_k = \frac{1}{2} \frac{M}{n_s e^2} J^2$$  \hspace{1cm} (4.2)

Next recall the Maxwell equation:

$$\nabla \times b = \frac{4\pi}{c} J$$  \hspace{1cm} (4.3)

where $c$ is the speed of light (Gaussian units). Doing some algebra yields:

$$E_k = \frac{1}{8\pi} \frac{Mc^2}{4\pi n_s e^2} |\nabla \times b|^2$$  \hspace{1cm} (4.4)

The free energy density of the system is now:

$$\mathcal{F} = \int V \left( b^2 + \frac{Mc^2}{4\pi n_s e^2} |\nabla \times b|^2 \right)$$  \hspace{1cm} (4.5)
where the integral is over a volume $V$. The term $\frac{M^2}{4\pi H_{c2}} \Phi_0^2$ is recognized as the square of the London penetration depth, $\lambda_L^2[41]$. The free energy density now takes on the familiar form:

$$\mathcal{F} = \frac{1}{8\pi} \int_V \left( b^2 + \lambda_L^2 |\nabla \times b|^2 \right) dV$$  \hspace{1cm} (4.6)

In order to find field equations from the free energy density we must perform a variational calculation, minimizing the free energy with respect to variations in the magnetic field $b$. This can be done by first writing the free energy as a function of $b + \delta b$, and then subtracting off the free energy as a function of $b$. The result is:

$$\delta \mathcal{F} = \frac{1}{4\pi} \int_V \left( b \cdot \delta b + \lambda_L^2 (\nabla \times b) \cdot (\nabla \times \delta b) \right) dV$$  \hspace{1cm} (4.7)

Integrating the second term by parts yields:

$$\frac{1}{4\pi} \int_V \left( \lambda_L^2 (\nabla \times b) \cdot (\nabla \times \delta b) \right) dV = \frac{\lambda_L^2}{4\pi} (\nabla \times b) \cdot \delta b + \frac{\lambda_L^2}{4\pi} \int_V (\nabla \times (\nabla \times b)) \cdot \delta b \, dV$$  \hspace{1cm} (4.8)

The first term on the right hand side is recognized as the current $J$ dotted into $\delta b$. This term therefore integrates to zero far from the core where the currents go to zero. However, within the core there is a problem because the superconducting charge density is no longer constant. It actually goes to zero at the center of the core, causing a singularity. This is taken care of by introducing a delta function "source term" in the equation which multiplies the flux quantum $\Phi_0$. One can now write the equation as:

$$\delta \mathcal{F} = \frac{1}{4\pi} \int_V \left[ b + \lambda_L^2 (\nabla \times (\nabla \times b)) - \Phi_0 \delta(r - r_\nu) \right] \cdot \delta b \, dV$$  \hspace{1cm} (4.9)

The minimum energy condition is satisfied when the integrand is zero; hence, the equation for the fields is:

$$b + \lambda_L^2 \nabla \times (\nabla \times b) = \Phi_0 \delta(r - r_\nu)$$  \hspace{1cm} (4.10)

This equation, in principle, allows the calculation of the magnetic field $b(x, y)$ in the mixed state of an isotropic superconductor. A method of doing so will be described in the next section, where the London theory is extended to the anisotropic, uniaxial superconductors.
4.5 Anisotropic, Uniaxial London Theory

The London theory for anisotropic, uniaxial superconductors starts out with the free energy, also. The development follows exactly the isotropic case until equation 4.6. At this point the uniaxiality needs to be included, and this is done using the charge carrier masses both within the superconducting planes and perpendicular to these planes. The mass within the plane is \( M_X = M_Y = M_1 \); the mass perpendicular to the plane is \( M_Z = M_3 \), which is more massive and reflects the difficulty of current flow in this direction. The mass is now a tensor with \( M_1 \)'s and an \( M_3 \) on the diagonal as represented in the crystal frame. It is generally more convenient to normalize the masses with respect to an average mass defined by \( M_{\text{av}} = (M_1^2 M_3)^{1/3} \), which leaves:

\[
\begin{align*}
  m_1 &= \frac{M_1}{M_{\text{av}}} \\
  m_3 &= \frac{M_3}{M_{\text{av}}}
\end{align*}
\]

(4.11) (4.12)

If we now go to a frame where the geometry is defined as depicted in Fig. 4.5, the mass tensor needs to be transformed. The angle \( \theta \) is the angle between the crystal c axis and the direction of the average field \( B \), the \( x \) axis is defined to lie in the plane of \( c \) and \( B \), and the \( y \) axis is perpendicular to this plane. The mass tensor can be represented in this frame by transforming the crystal mass tensor via a rotation about the \( Y \) (or \( y \)) axis, leaving:

\[
\begin{align*}
  m_{xx} &= m_1 \cos^2 \theta + m_3 \sin^2 \theta & m_{xy} &= m_{yx} = 0 \\
  m_{zz} &= m_1 \sin^2 \theta + m_3 \cos^2 \theta & m_{yy} &= m_1 \\
  m_{xz} &= (m_1 - m_3) \sin \theta \cos \theta
\end{align*}
\]

(4.13)

The uniaxiality is folded into the London theory via these masses through the penetration depth \( \lambda_L^2[42, 43] \), where:

\[
\lambda_L^2 = \frac{M c^2}{4 \pi n_s e^2} \rightarrow \lambda^2 m_{ij} = \frac{M_{\text{av}} c^2}{4 \pi n_s e^2} m_{ij}
\]

(4.14)

where \( \lambda \) is now an effective penetration depth, and the subscripts on \( m_{ij} \) are \( x, y, \) or \( z \).
CHAPTER 4. SUPERCONDUCTIVITY

Figure 4.5: Diagram showing the angle $\theta$ between the average field $B$ and the crystal axis $c$.

It is often convenient when speaking in these terms to introduce an anisotropy parameter $\Gamma \equiv \frac{m_3}{m_1}$, which for YBCO is 25 [44] and for the Bi-2212 compound is $\sim 3025$ [45]. It should also be noted that knowledge of $\Gamma$ is sufficient for finding the masses $m_3$ and $m_1$: $m_3 = \Gamma^{2/3}$ and $m_1 = \Gamma^{-1/3}$. With these changes the free energy density becomes:

$$F = \frac{1}{8\pi} \int_V \left[ b^2 + \lambda^2 m_{ij} (\nabla \times b)_i (\nabla \times b)_j \right] dV$$  \hspace{1cm} (4.15)

The field equations are obtained via the same type of minimization derivation used above. The resulting field equations are:

$$b_k = \lambda^2 m_{ij} \epsilon_{ist} \epsilon_{jkl} \frac{\partial^2 b_s}{\partial x_t \partial x_l} + \Phi_0 \delta_{sk} \delta(r - r_\nu)$$  \hspace{1cm} (4.16)

where the equation is written in component form using the summation convention, and $\epsilon_{abc}$ is the fully anti-symmetric tensor. This equation, like its isotropic partner equation 4.10 above, allows in principle the calculation of the fields within the superconductor.
CHAPTER 4. SUPERCONDUCTIVITY

4.6 Calculation of Fields in Anisotropic Superconductors

The method described here for calculating the magnetic field at any point within a FLL in an anisotropic, uniaxial superconductor, follows closely the method described by Thiemann, et al. [46]. Similar work has been done by Balatski\"i and others[47, 48], and more recently this development has been used as a part of the thesis of Riseman[17].

We first expand equation 4.16 and, recognizing the following general vector identity $\nabla^2 A = \nabla(\nabla \cdot A) - \nabla \times (\nabla \times A)$, write the field components as:

$$b_x - \lambda^2 (m_{zz} \nabla^2_{xy} b_x - m_{zz} \frac{\partial^2 b_x}{\partial y^2}) = 0$$

$$b_y - \lambda^2 (m_{zz} \nabla^2_{xy} b_y + m_{zz} \frac{\partial^2 b_x}{\partial x \partial y}) = 0$$

$$b_z - \lambda^2 (m_1 \frac{\partial^2 b_z}{\partial x^2} + m_{zz} \frac{\partial^2 b_z}{\partial y^2} - m_{zz} \nabla^2_{xy} b_z) = \Phi_0 \sum_{\nu} \delta(r - r_{\nu})$$

where $\nabla^2_{xy} = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ is the two dimensional Laplacian in the $xy$ plane, $r$ is some vector in the FLL, and $r_{\nu}$ are vectors to the vortices.

These equations are strictly correct for an isolated vortex, but can be used with good confidence for situations where there are many vortices which are well spaced and do not overlap. We will assume for the moment that we have vortices which are well separated in the equilibrium arrangement shown in Fig. 4.3. With this assumption, we now invoke the well known periodicity of the FLL, which enables us to perform a Fourier transformation. What we will do is transform the equations into Fourier space (reciprocal lattice space), perform a summation in this space, and transform back into real space for the final result.

The Fourier relationship is written as:

$$b(r) = \sum_{G} b(G)e^{iG \cdot r}$$

$$b(G) = \frac{B}{\Phi_0} \int b(r)e^{-iG \cdot r} \, dr$$

(4.18)
where \( \mathbf{G} \) are reciprocal lattice vectors and \( B \) is the magnitude of the average field over the real space FLL unit cell. The integral is over the unit cell in real space.

Plugging the upper of equations 4.18 into equations 4.17 and solving gives the components of the field as a function of reciprocal lattice vectors:

\[
\begin{align*}
 b_x(\mathbf{G}) &= B \lambda^2 m_{zz} G_y^2 / d \\
 b_y(\mathbf{G}) &= -B \lambda^2 m_{zz} G_x G_y / d \\
 b_z(\mathbf{G}) &= \frac{5}{2} \left( 1 + \lambda^2 m_{zz} G^2 \right) / d 
\end{align*}
\]

(4.19)

where the denominator \( d \) is given by:

\[
d = \left( 1 + \lambda^2 m_{11} G_x^2 + \lambda^2 m_{zz} G_y^2 \right) \left( 1 + \lambda^2 m_{zz} G^2 \right) - \lambda^4 m_{zz}^2 G^2
\]

(4.20)

These equations — for a uniaxial, anisotropic superconductor — will lead to the real space microscopic magnetic field at some general point within the FLL. When the real space field is found (described below), it will in general have components which are transverse to \( z \), i.e. \( x \) and \( y \) components. These transverse components are a result of the anisotropy of the material being introduced via the mass tensor, and do not arise within the isotropic London theory. These transverse components will be the subject of much discussion in chapter 6.

It now remains only to state the real and reciprocal lattice vectors. From various studies it has been found \([42, 49]\) that the basis vectors of the real space, equilibrium FLL, for directions of \( B \) with respect to the material axis of \( 0^\circ \leq \theta \leq 70^\circ \), are given by:

\[
\begin{align*}
 a_1 &= \left( \frac{2 \Phi_{\alpha}}{B} \right)^{1/2} \left( \frac{m_{zz}}{3m_3} \right)^{1/4} \hat{x} \\
 a_2 &= \frac{a_1}{2} \left[ \hat{x} + \left( \frac{3m_3}{m_{zz}} \right)^{1/2} \hat{y} \right]
\end{align*}
\]

(4.21)

The angle between the vectors (\( \eta \)) is given by\([46]\) \( \tan \eta = (3m_3/m_{zz})^{1/2} \). These vectors map out a FLL which consists of isosceles triangles, and for \( \theta = 0 \) the triangles are equilateral (isotropic case).
The reciprocal lattice vectors corresponding to these real space vectors are:

\[ G_x = \pi \left( \frac{2B}{\Phi_0} \right)^{1/2} \left( \frac{3m_3}{m_{zz}} \right)^{1/4} n \]  
\[ G_y = \pi \left( \frac{2B}{\Phi_0} \right)^{1/2} \left( \frac{m_{zz}}{3m_3} \right)^{1/4} (2m - n) \]  

where

\[ m, n = 0, \pm 1, \pm 2, \ldots \]

These vectors map out a reciprocal FLL which consists of isosceles triangles, much like its real space counterpart.

The prescription for the calculation of the local magnetic field \( b(x,y) \) within the FLL is now clear:

1. Choose an angle \( \theta \), an anisotropy \( \Gamma \), and an effective penetration depth \( \lambda \) for a material of interest.
2. Calculate the mass tensor \( m_{ij} \) for these values.
3. Calculate the components in the upper of equations 4.19.
4. Sum equation 4.18 over a sufficient number of reciprocal lattice vectors for each component of the field.
5. Repeat steps 1-4 above for many points within a FLL unit cell.

More needs to be said about the phrase "sufficient number" in the fourth item above. First, the sum over equation 4.18 may not converge for all points in the FLL due the \( G \) dependence in each of the terms. The question becomes one of how many reciprocal lattice vectors are needed to obtain reasonable results. A procedure often used is to cut off the sum at an appropriate number of reciprocal lattice vectors, \( G_{\text{max}} \sim 1/\xi \). Reciprocal lattice vectors of length greater than this fold in Fourier components which are the result of the logarithmic singularity at the core[33]. For a two-dimensional reciprocal lattice there will be two different cutoffs, one for each direction of the lattice, and in general these will be different.
A similar method, and the one used in this work, is to perform an empirical investigation as to the degree of convergence for various numbers of reciprocal lattice vectors. Once a number is found which, when exceeded, causes little change in the field values, this number is used as the cutoff. This method was used for my calculations, and I decided on a value of ±50 for the integers \( n \) and \( m \) in equations 4.22 above. The field values at nearly all points within the FLL showed little variation for numbers greater than this, and for lesser numbers it was obvious that not enough were used.

### 4.7 Limitations of the London Theory

The London theory, like all theories, has limitations. The origin of the limitation is that for the London theory to be valid, the superconducting charge carrier density must be essentially constant throughout the superconductor. As discussed above, within the isolated vortex approximation the vortex wave function increases quickly until it reaches a value of \( \sim 1 \) at the core edge. Outside the core, the wave function, and hence the superconducting charge carrier density, varies little[33].

The development assumes one is near the lower critical field \( H_{c1} \), and is strictly only valid there. However, for intermediate field values, \( H_{c1} \ll H \ll H_{c2} \), the London theory may also be applied with good confidence for materials which are in the high-\( \kappa \) limit, because here again the vortex wave function is still largely a constant, and the cores may be taken care of via the delta functions used above.

For high-\( \kappa \) materials, it is only when in the high field limit, \( H \sim H_{c2} \), that the London theory begins to break down. In this situation the flux cores are very close together, and there is a significant variation in the vortex wave function. This is a result of the charge density varying, and hence the basic assumption upon which the London theory is based is no longer valid. In such situations one must abandon the London theory in favor of a theory which is valid near the upper critical field, like the famous Ginzburg-Landau theory[40, 34].

An obvious fault of the London theory is that the field at the flux core is logarithmically divergent at the point which is assumed to be the core — see, e.g. deGennes [34]. Therefore the effect of finite core size may have to be taken into account. It was
mentioned above that for equilibrium FLL's the reciprocal lattice sum can be cut off at \( G_{\text{max}} \sim \frac{1}{\xi} \), which causes the field within the core to reach a maximum, finite level. This method has relatively minor effects on the overall field values within the FLL and only significantly affects the fields very near (or within) the cores. In addition, there are times where the core size is such that this type of cutoff is very necessary. This can be due to actual large \( \xi \) values, or possibly due to longitudinal fluctuations along the vortex length enhancing the vortex core contribution to the fields within the FLL.

The isotropic and anisotropic theories developed above have additional assumptions. The use of the Fourier ideas assumes a regular, periodic array of vortices in a triangular lattice which has formed within a single crystal. This is the equilibrium condition as solved by Campbell et al. The high-\( T_c \) materials, while certainly being in the high-\( T \) limit discussed above, have historically been full of pinning centers which trap flux and cause significant distortion within the FLL. The distortion often manifests itself in the bending of the flux tubes, causing additional fields which are transverse to the previously defined vortex direction, and generally causing disorder in the FLL.

Single crystals pose problems in that they are often twinned, meaning that while there is a well-defined crystal c axis, there is no defined a or b direction due to their being skewed at right angles to each other along various planes parallel to the c axis direction. Early studies on YBCO crystals using Bitter pattern techniques showed that the twin boundaries caused severe pinning of flux tubes, increasing the flux density along them by nearly a factor of 2. However, this technique only samples the vortex behavior at the surface of the sample, and we are more interested in the bulk behavior. The bulk is readily probed with neutron scattering, and studies in various twinned single crystals have been performed. These have shown that the FLL becomes slightly distorted if twin planes are present, but that the structure is essentially a triangular geometry. One must be aware that these results have been obtained with relatively high applied fields, on the order of 8 - 10 kG in most cases. This is because the FLL spacing goes like \( d \sim \sqrt{\Phi_0 / B} \), and slow neutrons have wavelengths on the order of 10 Å. For the neutrons to "see" the FLL the field...
must therefore be in this range. Neutrons are therefore an excellent probe of the FLL within the bulk at these fields, but are incapable of seeing anything at lower fields.

However, recently it has been shown\[58, 59\] that largely untwinned single crystals can be grown. The FLL within these kinds of samples is expected to be nearly distortion free\[55\] and has been shown to be so in\[53\].

### 4.8 Effects of Anisotropy on the FLL

The first effect of the crystalline uniaxial anisotropy on the FLL is the change with angle $\theta$ from an equilateral to an isosceles geometry. This effect is the most obvious and comes directly out of the analysis given in section 4.6. The second effect is the existence of the transverse magnetic fields, which for the isotropic case do not exist. In addition, there are other, more subtle, effects of anisotropy which can cause changes within the FLL. These are briefly described below.

At sufficiently weak fields ($H < 100 G$) the FLL may undergo a phase transformation from a triangular to what has been termed a chain state. The effect was predicted theoretically\[60, 61, 62, 63, 47, 64, 65\] by realizing that the vortices, which normally repel each other, become attracted to each other for large angles $\theta$ in these anisotropic materials. The effect has recently been shown to exist in Bitter patterns\[66, 58\], and is caused by the interaction potential between the vortices in the high-$T_c$ materials becoming attractive in the $z$ direction. That is, at low enough inductions and high enough angles, the field component $b_z$ becomes negative over a certain range of distances from the core. Since the interaction potential between vortices is proportional to this component of the field\[34, 65\], the interaction becomes attractive. (For a good theoretical discussion, see ref. \[65\].)

Once formed, the spacing between fluxons in the chain direction is independent of field for a certain range of fields\[55\]. Therefore the FLL geometry only changes in one direction as $H$ is varied. In principle the London theory is still valid in this situation, however a different Fourier analysis is needed to properly describe the situation\[17\].

In addition, some theoretical studies at very low inductions have recently pointed to other equilibrium lattices which are not triangular\[67\]. The studies show that the
triangular lattice at low inductions, and with the average field at an angle to the crystal c axis, can become unstable and yield to other structures. These studies were pursued in an attempt to explain the phenomenon of flux melting, where it is believed that the FLL may become disordered due to thermally activated forces within the material, as well as the recent sighting of chains mixed within a normal vortex lattice [66]. The theoretical development of this interesting case has been done[68], and this particular instability within the FLL is found to be a result of the high anisotropy of the material and the geometry of the sample. Instead of the free energy density minimization done above, one must now minimize the Gibbs free energy subject to boundary conditions on $B$ and $H$. This leads to a mixed state within the mixed state, with some flux at an angle to c and some lying parallel to c. This phenomenon is very interesting, but it lies beyond the scope of this work.

Anisotropy in the high $T_c$ materials seems to be linked to the crystal structure. The more anisotropic the crystal structure, the more anisotropic the magnetic and electrical properties. In materials like BSCCO, which has a large ratio of $c/a$, the superconducting planes can be many angstroms apart, so that the currents and fields associated with one plane may have little interaction with the others. It is for this reason that the highly anisotropic materials have recently been described with a theory originally developed to describe materials consisting of superconducting layers separated by non-superconducting material. The Lawrence-Doniach theory[69] states that the superconducting properties of a material may be described as existing within the layers, which are coupled weakly to the other layers via Josephson tunneling. (If the variation of the superconducting properties along c is slow compared to the c-axis coherence length $\xi_c$, then an effective mass tensor may be employed in the description. However, the effective mass idea is not essential to the theory.) The vortices are usually pictured as “pancakes” within the superconducting plane and weakly coupled to the others in neighboring planes. The description has now gone from the 3-dimensional idea of tubes to a 2-dimensional idea of pancakes. This theory has been used recently with fair success in describing some of the more anisotropic materials[53, 70, 71, 72]. It is believed that the anisotropic London theory breaks down in these materials at low temperatures due to the c-axis penetration depth.
becoming shorter than the spacing between CuO₂ planes.

4.9 Favorable Properties of YBCO

It would appear, with all of the possible effects of anisotropy coupled with the limitations of the London theory, that the theory as developed has no application. This is not the case. The compound YBCO seems perfectly suited to a description by the uniaxial, anisotropic London theory. The four properties which makes it suitable are now briefly described.

The first property YBCO displays is a high-$\kappa$ value. It is generally believed that the $\kappa$ value for YBCO is about 100\cite{73}. This certainly places YBCO in the high-$\kappa$ limit discussed above and therefore makes possible the application of the London theory.

YBCO is anisotropic, so the anisotropic London theory is applicable. However, it is believed to have a $\Gamma = 25$, so the amount of anisotropy is not so great that a layered description is needed to describe the behavior\cite{70,72}. Hence the London theory as described above should still be applicable.

In YBCO there exists a low-field regime where there is a well developed FLL without too much distortion\cite{55,74}. There is the possibility of forming a chain state\cite{68}, but for fields which are not too low and angles $\theta$ which are not too large the vortex state should dominate\cite{58}.

Lastly, there has lately been a plethora of good quality single crystals made of YBCO. These have displayed low twinning and pinning\cite{58,75}. Crystals of this caliber are essential if a description based on London is to be used.

These four properties of YBCO should allow a low-field study in good single crystals where a well defined, triangular FLL exists. The London theory should be directly applicable to a YBCO single crystal in this state, and the development in the remainder of this thesis assumes such a condition can be obtained.
Chapter 5

Magnetic Field Surfaces and Distributions

The anisotropic London theory, described in the previous chapter, allows for the numerical calculation of the microscopic magnetic fields within the flux line lattice of the mixed state of type II superconductors. The results of the numerical implementation of this theory are now presented by way of three dimensional and contour plots of the various magnetic field components. The calculations are done over a unit area of the flux line lattice, which encompasses an effective area of two of the icosceles triangles making up the lattice, and the plots reflect this area. Next, the distribution of fields contained within these plots is calculated. This representation is interesting in itself in that certain predictions of the theory can be checked. These predictions are studied by allowing the parameters of the program to vary and watching the distributions change. In addition, the results for some distributions can be compared to experimental results obtained on real samples by way of Fourier transforming $\mu$SR data in the asymmetry representation. A comparison of theory to data, in this way, is made at the end of the chapter.
5.1 Magnetic Field Surfaces

The prescription described in the last chapter for the calculation of magnetic fields via the anisotropic London theory has been followed and magnetic fields at points within an equilibrium FLL have been calculated numerically. The program (see appendix A) needs values for the following parameters: \( B \) - the average magnetic field; \( \lambda \) - the effective penetration depth; \( \theta \) - the angle between the average field and the crystalline \( c \) direction; and \( \Gamma \) - the anisotropy parameter from which the effective masses \( m_1 \) and \( m_3 \) are found. Using only these four parameters the program calculates the real and reciprocal lattice basis vectors, calculates the effective mass tensor, sums over the appropriate number of reciprocal lattice vectors, and transforms back into real space to give the magnetic field components at the point.

The points at which the field is calculated are determined within the program, and lie in a region of the FLL as depicted in Fig. 5.1. This area was chosen as the minimum in which the vector nature of the fields is not repeatable. That is, even though the magnitudes of the fields may be reproduced within this area, the vector nature is not. This can be more easily understood by looking at Fig. 4 of reference [46], where the streamlines of the transverse field are shown. The reason for this choice of area will become clear later when the ideas of the off-axis fields are introduced. The actual points within the area where the fields are calculated are on a grid 51x51, for a total of 2601 points. The grid lengths are found by first calculating the real space basis vectors of equation 4.21 and dividing \( a_1 \) and the \( y \) component of \( a_2 \) by 50.

The values of the parameters corresponding to \( Y \) \( B \) \( a_2 \) \( C \) \( u_3 \) \( O \) \( \phi \) are: \( \Gamma = 25, \lambda = 256.5 \text{ nm} \) [44]. The value for \( \lambda \) is found from using \( \Gamma^{1/2} = \lambda_c/\lambda_{ab}, \lambda = (\lambda_c\lambda_{ab}^3)^{1/3} \), and a value for \( \lambda_{ab} = 1.5 \times 10^{-5} \text{ cm} \), which is in the range found in much of the literature [17, 76, 55, 77, 78, 39]. The symbols \( \lambda_c \) and \( \lambda_{ab} \) are the penetration depths for fields oriented parallel and perpendicular to the basal planes, respectively (or, currents flowing parallel to the \( c \) and a-b directions, respectively). It should be noted that for many of the calculations a slightly different \( \lambda \) value has been used \( \lambda = 120 \text{ nm} \). This was used initially due to a mistake in the calculation, but proved to yeild quite
interesting results. Therefore the results were kept and used as illustrative examples.

Results of calculations using a similar λ parameters will be shown. The reason for using a different λ parameter is that the results for the distributions (calculated later) show a more interesting structure than for the more "exact" YBCO parameter. The value used here is λ = 120 nm, which is about one half of the value quoted above for YBCO. Using this value, B = 250 G, θ = 45°, and Γ = 25, the following set of figures was created. Figures 5.2 and 5.3 are the surfaces corresponding to the z component and the magnitude of the field b within the FLL area. Figures 5.4 and 5.5 are the surfaces corresponding to the x and y components of the fields within this area of the FLL.

Figs. 5.2 and 5.3 are almost identical, and have obvious tall, column-like features which are the flux tubes. These tubes extend higher than shown in the figures, but they have been graphically cut off in order to show the field behavior near the central region with more clarity. The fields fall off smoothly from the cores and become relatively flat in the mid-region. The minimum field exists near the center of the plot. In addition, there are three saddle points within the surfaces. Two are equivalent and exist on the diagonals between the cores, and one exists along the x
Figure 5.2: The surface corresponding to the $z$ component of the fields.
Figure 5.3: The surface corresponding to the magnitude field \( b \) at points within a unit cell of the FLL.
Figure 5.4: The surface corresponding to the $x$ component of the fields.
Figure 5.5: The surface corresponding to the $y$ component of the fields.
direction between the cores. These saddle points will become very important later when the field distributions are discussed.

Fig. 5.4 is the surface of the $x$ component of the fields. Its shape is almost opposite to the shapes just described for the $b_t$ and $b$ surfaces. Now the surface descends steeply at the cores, becoming negative. The central region again is mainly flat, and the values of the field $x$ component are positive here.

Fig. 5.5 is the $y$ component surface of the FLL. This surface exhibits interesting behavior near the cores, where the fields have an inflection as one passes through the core in the $x$ direction. There is an obvious asymmetry with respect to a plane half way along the $x$ axis, where one side is exactly inverted with respect to the other.

An interesting point concerning both of these last two surfaces is that the average of each over the FLL unit area must be zero. This is almost immediately obvious when one looks at the $b_y$ surface, but is somewhat less apparent when looking at the $b_x$ surface. The reason is rather simple. The average field lies along the $z$ direction, and therefore the dominant field is the longitudinal, or $z$ component. For the average field to be along $z$, each of the $x$ and $y$ fields, when averaged over the FLL area, must be zero, or else the average field would lie in some other direction. This assertion has been used as a check on the reliability of the surfaces produced with the program. Both the $x$ and $y$ surfaces have been shown to average to zero over the FLL area to within numerical accuracy.

Shown next are the contour plots corresponding to these surfaces. Fig. 5.6 is the contour for the $b_z$ surface. Once again the core regions are graphically cut off so that the region away from the cores may be viewed more clearly. The minimum is clearly visible near the center of the plot. The saddle points are more evident on this plot; one can clearly see by the numbers labeling the contours that there is a saddle point on the line midway between any two cores. Note also that this plot, as well as the two which follow, is drawn with the proper aspect ratio so that the isosceles nature of the vortex lattice can be seen.

The next figure is the contour plot of the $b_x$ surface – Fig. 5.7. The great variation in the values of $b_x$ near the cores is evident in the tight packing of the contour lines there. The flatness away from the cores is also evident, but here some subtle field
Figure 5.6: The contour plot corresponding to the $z$ component surface. Notice the minimum near the center as well as the saddle points. The core regions have been excluded because the contours became unresolvable. The numbers on this and the following plots are field contour values in Gauss.
CHAPTER 5. MAGNETIC FIELD SURFACES AND DISTRIBUTIONS

Figure 5.7: The contour plot corresponding to the $x$ component of the fields.

variation – islands – which was not so obvious on the surface plot, is more easily seen.

The next contour plot is Fig. 5.8, the contour plot of the $b_y$ surface. The symmetry mentioned above is apparent at first glance here, also. From the numbers one can easily see that the two sides are simply inverted reflections of each other. It is also very easily understood from this plot how the fields average out to zero over the area.

It is also possible to calculate the surface of the transverse field $b_\perp$ and show its contour. This has been done by the author, but is not particularly illuminating. More informative is the plot of the lines of the transverse field, often called the streamlines of the transverse field because the direction of the transverse field at points is given
Figure 5.8: The contour plot corresponding to the $y$ component of the fields.
and it appears as though the lines are flowing. The calculations for these plots has been done by various authors, and two particularly good examples showing interesting streamline plots are references [79, 46]

5.2 Method of Calculating Distributions from Surface Grids

This section describes the method of integration employed to allow the distribution of magnetic fields to be calculated. (As a reminder, a distribution is simply a probability of a particular item being found within a certain set of data.) This section is rather tedious and is included for completeness. Uninterested readers may skip this section with no loss of understanding or context.

Calculations of the sort discussed above often leave one with a lot of data and therefore a real need to organize and display it in a meaningful way. Plotting is of course the first option, both as a surface and as contours. This can shed much light on the correctness of the calculations, assuming that the user has some idea of the final result. It can also help in giving some insight as to what the behavior of the particular physical system at hand is doing – the averaging to zero of the $b_x$ and $b_y$ surfaces, for example. It is for these reasons that plotting is of particular importance.

In addition, it is often interesting to determine the distribution of values contained within the grid; or, in mathematical terms, to calculate the probability distribution of values of the quantity contained within the matrix. Our grids contain magnetic field values which (theoretically) are the actual values of the fields within the FLL. It was discussed in Chapter 2 how the muonic precession signal obtained in a $\mu$SR experiment can be Fourier Transformed to yield the distribution of fields that the muons sense. It would therefore be useful to devise a technique to obtain the theoretical field distributions from the grids for comparison to experiment.

The technique described below does just that. It is not unique, and has been used by others to obtain similar kinds of results for things like densities of states[80]. The
method does a numerical integration of the following integral:

\[ n(M) = \int_A \delta (M(x, y) - M_0) \, dA \]  

(5.1)

where \( A \) connotes the area of the region of interest (in our case the FLL area), \( M(x, y) \) is the value of the quantity contained within the matrix at position \((x, y)\), \( M_0 \) is the particular value of the quantity at some point of the integration, and \( \delta \) is the Dirac delta function. For our surfaces the \( M \) values will be things like \( b, b_x, \) etc. Physically, what this equation does is integrate over contours, finding the fractional amount of area the lines take up within the FLL area, and weight that with the inverse of the steepness of the area on which this particular contour is found. This means that longer contours will contribute more, but if they exist on very steep terrain they are weighted back down.

This can be seen more easily if one uses the properties of delta functions to express the integral in a more physically meaningful form. For a one dimensional function the following relationship is true:

\[ \delta(f(x)) = \sum_i \frac{1}{|f'(x_i)|} \delta(x - x_i) \]  

(5.2)

where the function \( f(x) \) has simple zeroes at \( x_i \) in the sum. The extension to a two dimensional function is obvious:

\[ \delta(f(x, y)) = \sum_i \frac{1}{|\nabla f(x_i, y_i)|} \delta(x - x_i) \delta(y - y_i) \]  

(5.3)

Replacing the delta function over \( M \) in equation 5.1 with the form shown in equation 5.3 yields:

\[ n(M) = \int_A \sum_i \frac{1}{|\nabla M(x_i, y_i)|} \delta(x - x_i) \delta(y - y_i) \, dxdy \]  

(5.4)

where of course \( dA = dxdy \). Evaluation of the integral now leaves a sum over the \( i \) places in the area where \( x \) and \( y \) meet the delta condition:

\[ n(M) = \sum_i \frac{1}{|\nabla M(x_i, y_i)|} \]  

(5.5)

The technique works as follows, with reference to Fig. 5.9. The grid is made up of uniformly spaced points in two dimensions. The step sizes in each direction are
Figure 5.9: A sketch of how the grid is handled for the numerical integration. The value of interest is $M_0$, and its associated contour line is shown as the darker dotted line. The lighter, diagonal dotted line is that which separates the grid into the two triangles. A given contour can only intersect two of the three sides of each of the two triangles.

In general different, but that is irrelevant for the present discussion. The only thing necessary is an N x N grid of values which are relatively smoothly varying in both directions. This can be checked by plotting and is certainly true in the case of our surfaces.

One first chooses a value for $M_0$ and scans through the grid until a specific criterion is met - see Fig. 5.9. The criterion for this check is that the value $M_0$ lie between the minimum and maximum values of the three grid points making up a triangle which is formed by slicing the grid cell along a diagonal - i.e. $M_{\text{min}}(x, y) < M_0 < M_{\text{max}}(x', y')$. The reason for the triangle is that it contains the minimum amount of grid points needed to completely determine the field within its area. If the condition is satisfied, one next finds the two points where the $M_0$ value actually intersects the sides of the triangle. Beware: there is some implied knowledge here. First, it is assumed that
the "contour" corresponding to $M_0$ only intersects the grid sides at two points. In general this may not be true, but the implication here is that the grid steps are small compared to variations in the values $M(x, y)$. If this condition is not met the method will not work. Hand-in-hand with this is the implication that we can find the actual intersection points along the sides of the grid given only values at the corners. At this point we require again that the step sizes are small and assume that a linear extrapolation between points will suffice.

Given these conditions and that the value $M_0$ is within the present triangle, we find the intersection points using the slopes of the assumed linear sides. At this point we can apply equation 5.5. The gradient over a triangle is constant, so we must now sum over the length of the contour within the triangle. Therefore, once the intersections are found, we calculate the length of the line connecting the two points $- l$. Here again is the assumption that the "contour" is slow enough in its variation to allow a linear approximation. In addition to the length we also find the gradient as:

$$\nabla M(x, y) = \frac{\partial M(x, y)}{\partial x} + \frac{\partial M(x, y)}{\partial y} \rightarrow \frac{\Delta M(x, y)}{\Delta x} + \frac{\Delta M(x, y)}{\Delta y}$$  \hspace{1cm} (5.6)

where the \(\rightarrow\) indicates the partials going over in the linear approximation to differences. The actual contribution to the total probability for a given triangle is:

$$n_\Delta(M) = \frac{l}{\frac{\Delta M(x, y)}{\Delta x} + \frac{\Delta M(x, y)}{\Delta y}}$$  \hspace{1cm} (5.7)

The contribution to the distribution for this value of $M_0$ for this particular triangle is the ratio $l/\nabla M(x, y)$. This gets added to all other contributions from all other triangles for the given $M_0$ value. This is then the probability of finding a given $M_0$ within the surface. The next value of $M_0$ is then used and the process is repeated, building up the entire distribution.

### 5.3 Magnetic Field Distributions

The method described in the previous section for the calculation of distributions has been implemented and applied to various magnetic field surfaces. The distribution is
the probability of finding a particular value of a quantity within the set of data under investigation. In this case, the sets of data contain magnetic field values, and so the distributions are magnetic field distributions. The results of this implementation to surfaces like those shown previously in Figs. 5.2 - 5.5 are shown below in Figs. 5.10 - 5.13, respectively.

Figs. 5.10 and 5.11 represent the field distributions for the $b_z$ and $b$ surfaces, respectively. These distributions look very similar, as do their field surfaces in Figs. 5.2 and 5.3. These distributions are characterized by the number of distinctive steps and discontinuities in the curves. For example, both have zero probability for fields below a certain minimum value, which suddenly appears with a very finite probability. This feature is due to the minimum near the center of the surface, which lies in a region which is very flat and therefore has a considerable contribution from the $1/\nabla M(x, y)$ part of the sum (equation 5.7).

Another feature which is characteristic of these surfaces is the peak just above the minimum - which in this case is two peaks. The peaks here are due to the saddle points contained within the field surfaces. These were noted above when both the surfaces and contour plots were discussed. The saddle point along the $x$ axis of Fig. 5.6 is the cause of the left peak, while the saddle points along the diagonals connecting the cores are equivalent and cause the right peak. This can be seen by checking the field values which are printed on the contours. Once again the flatness of the surface around these regions is the cause of the peakiness.

In general there is one further feature common in these particular distributions, and that is a step similar to the minimum at the maximum field value. This step is greatly reduced in magnitude from the minimum, but is likewise caused by the similar effect of there being no field values after it. The reduced magnitude is due to the higher fields existing near the core regions on the surfaces and thereby having shorter lengths and larger gradients contributing to equation 5.7. On the plots in Figs. 5.10 and 5.11 this upper step is not visible. This is because the vertical scale of the plots is large compared to the amplitude of the upper cutoff. (Technically speaking, since the London theory has no intrinsic cutoff, the high field tail will continue on forever. Numerically, however, the reciprocal sum is cutoff and the core value is finite,
Figure 5.10: The magnetic field distribution for the $b_z$ surface. The parameters used to generate this and the following distributions are the same as those used to generate the surfaces and contours: $B = 250$ G; $\lambda = 120$ nm; $\theta = 45^\circ$; $\Gamma = 25$. 
Figure 5.11: The magnetic field distribution of the $b$ surface.
Figure 5.12: The magnetic field distribution corresponding to the \( b_z \) surface. Note that the area to the left of the origin is equal to the area to the right – confirming the areal average of zero.
producing a step at some value $b_{\text{max}}$).

The field distribution for the $b_x$ surface is shown in Fig. 5.12. It has a highly asymmetric shape and also an asymmetry with respect to the origin. In contrast to the distributions just described, this has a sharp, very finite maximum field discontinuity. This is due to the very flat region in Fig. 5.4, where the maximum field situation is almost exactly the same as for the minimum above. The minimum here behaves as the maximum above, with a long tail seeming to be asymptotic to the $x$ axis at low fields. Here, as with the maximum above, there exists a very small discontinuity or step corresponding to the absolute minimum in the distribution. It is not visible here for the same reason that the maximum step is not visible above. The third feature which catches the eye for this plot is the sharp peak just below zero frequency. This is due to the saddle point most easily seen in Fig. 5.7 at the center of the $x$ axis, and also existing equivalently in both the upper left and right corners of the plot.

The $b_y$ field distribution is shown in Fig. 5.13. In contrast to the other distributions shown it has a very symmetric shape. The minimum and maximum points are the result of the peaks on the surface in Fig. 5.5. The peaks on the left and right of the origin in the field distribution are again caused by saddle points, this time easily seen as the center of the hourglass-like contours in Fig. 5.8. The cause of the peak at the origin is not so easy to see. It is due to a saddle point which exists at the center of the top of the contour plot. One must imagine the other plots superimposed about this point, with maxima and minima diagonally across from each other such that their common meeting point is a saddle point at the center of the core with zero $y$ component of the field.

A final point about the $b_x$ and $b_y$ distributions is that they show rather clearly the idea of zero net transverse field throughout the FLL unit cell. The $b_y$ distribution obviously has as much area to the left of the origin as on the right, confirming that there is no net $y$ component. The $b_x$ distribution has no such symmetry, but after some inspection one can see that in fact there is equal area on both sides of the origin. This prediction of the theory is a good check on the correctness of the numerical calculations, which in this case seem to work well.

It should be noted that the theoretical study of magnetic field distributions in
Figure 5.13: The field distribution corresponding to the $b_y$ surface. Once again the areas to the left and right of the origin are equal.
the mixed state of superconductors did not start here. In fact, it started some years ago and has evolved quite rapidly since the introduction of the high speed computer. Listed here for the interested reader are some references which show the development of numerical calculations of magnetic field distributions as well as some which show very detailed theoretical analyses of things like peaks and steps. The references are: [50, 82, 83, 84, 85]. The list is not complete, but is a good launching point for anyone wishing to know more.

5.4 Field Distribution Dependence on Parameters

The surfaces, contours, and the corresponding distributions shown are representative of the infinite number which can be obtained by varying the parameters of the theory. These particular surfaces were chosen as candidates because the parameters used show a rich variety of structures with interesting behavior. As mentioned before, the only real difference is in the parameter $\lambda$, which was used above as 120 nm and should really be 256.5 nm for YBCO. (The real YBCO values will be used below in the discussion of the off-axis fields.) Since it has already been hinted that varying a parameter affects the results, we will now discuss just how the distributions and surfaces in fact do vary with the parameters.

5.4.1 Angle Dependence

The angular dependence of the field distributions is shown in Figs. 5.14 through 5.15. In all of the figures in this subsection, the parameter values other than $\theta$ are: $B = 250$ G, $\lambda = 256.5$ nm, $\Gamma = 25$. First, the dependence of the longitudinal, or $z$, component of the fields is shown in Fig. 5.14. The distribution becomes more compact as $\theta$ increases to 90°. The minimum field value increases with theta until the left side of the curve becomes a straight, vertical rise to the peak. This means that the minimum and the saddle points have become indistinguishable. The peak also moves toward higher frequencies, and the long high-frequency tail becomes shortened.
This behavior indicates that the overall range of $b_z$ within the FLL unit cell decreases rather markedly with angle.

In addition to this behavior is the appearance of the second peak for $0^\circ < \theta < 90^\circ$. At $\theta = 0^\circ$ the FLL is isotropic, and hence the saddle points are equivalent in field value. As theta increases, the fields slowly distort (due to the anisotropic masses) and the FLL changes over from equilateral to isosceles, causing differing values for the two saddle points. The FLL remains isosceles all the way to $90^\circ$, but here the second peak disappears. In theory the two saddle points are still distinct, but their values have become so close at this angle that one needs to look extremely closely to resolve them. It is worth noting that multiple peaks of any nature have yet to be found experimentally, let alone for the difficult $90^\circ$ case.

A final point on this graph is to note that the variations are all about the average field value, which for a field of 250 $G$ is a frequency of $f = 3.3875$ MHz for positive muons (recall $\omega = \gamma \mu B$). It is also important to note that the peaks are below the average field value for all angles, as can be seen in the figure where the average field is labeled as a vertical dotted line.

Very similar behavior is shown in Fig. 5.15, where field distributions of the magnitude $b$ are plotted for various angles $\theta$. The same minimum and tail behavior is seen as in Fig. 5.14, as well as the overall decrease in the width of the distribution. The actual values of the minima and peaks are of course slightly different from those in the previous figure, but the angular dependence is the same. However, the distributions at $0^\circ$ and $90^\circ$ are the same as in the previous figure, once again because of the lack of transverse components at angles where the average field is aligned with a crystal axis. In this figure the frequency corresponding to the average field $B$ is again labeled as a vertical dotted line. The behavior of the various distributions with respect to this line mirrors that in the last figure.

Somewhat similar behavior is shown in Fig. 5.16, where the $b_x$ distributions are shown for various angles $\theta$. The distribution shrinks dramatically in width as theta is increased, and (due to areal conservation) becomes much more pronounced vertically. The saddle point peak on the left is almost gone at low angles, but moves left and increases as $\theta$ gets large. Notice in the graph that the angles $0^\circ$ and $90^\circ$ are not
Figure 5.14: Field distributions \( n(b_z) \) for various angles \( \theta \) between the average field \( B = 250 \) \( G \) and the crystal \( Z \) direction. The other parameters are: \( \lambda = 256.5 \) \( nm \), and \( \Gamma = 25 \). The vertical line at \( f = 3.3875 \) \( MHz \) indicates the position of the average field.
Figure 5.15: Field distributions $n(b)$ for various $\theta$. The other parameters are the same as in the last figure.
shown; this is because there are no transverse components at angles where the field $B$ is along a crystal direction.

The distributions of $b_y$ are shown in Fig. 5.17. The same angular behavior is observed here as before, with the distribution becoming much more compressed as $\theta$ increases. The gradual, symmetric, wing-like decreases to zero at lower angles become sheer drops at higher angles. The three peaks, which are very widely spaced at 10°, become almost indistinguishable at an angle of 70 degrees.

In sum, the distributions are all similarly affected by variations in the direction of the average field $B$. All show significant narrowing as theta increases to higher angles, which is likely a reflection of the increasing "distortion" of the FLL away from equilateral triangles at $\theta = 0^\circ$ to ever more isosceles triangles as $\theta \to 90^\circ$. (The reader should note the use of the term "distortion" in this context means the natural moving of the FLL from equilateral to isosceles, and not distortion by any means such as pinning or thermal fluctuations). The distributions for $n(b_z)$ and $n(b)$ are very similar (even at $B = 250$ G), and show for these parameters separate peaks for the two different saddle points within the field surfaces. These peaks arise once theta is non-zero and persist until theta nears 90°, when they disappear again to within the limits of numerical accuracy.

5.4.2 Field Dependence

The dependence of the various distributions on the magnitude of the average field $B$ is discussed here. In all plots the parameters other than $B$ are held fixed at the following values: $\lambda = 256.5$ nm, $\theta = 70^\circ$, and $\Gamma = 25$. The values for $B$ are 100, 250, 500, 1000, and 5000 G.

The $n(b_z)$ plots are in Fig. 5.18 and again show the characteristic shape, but are seen to be displaced with respect to one another. The displacement as shown is not real – the horizontal axis is not the correct axis and has been left unlabeled in an attempt to avoid confusion. The actual positions of the lineshapes on this axis are quite far apart. When they are plotted in these positions they cannot be easily compared. For this reason the distributions above 100 G have been shifted down
Figure 5.16: Field distributions $n(b_x)$ for various $\theta$. The other parameters are the same as in the last two figures.
Figure 5.17: Field distributions $n(b_y)$ for various $\theta$. The other parameters are the same as in the last three figures.
It is seen at first glance that the magnitude of the average field has some effects on the shapes of the distributions. However, the widths and heights are all very similar, especially when possible numerical effects are considered. It is therefore almost entirely correct to state that variations of this sort, especially at intermediate fields, to first order only shift the distribution along the frequency axis.

As mentioned above, there are some differences between the curves. All of the curves, except for the 100 and 5000 \( G \), have some type of double peak structure. These two peaks are caused by the existence of two saddle points within the magnetic field surface (see above). For angles which are intermediate, like \( \theta = 70^\circ \) for the figure, one expects the two saddle points to give rise to the peaks. Therefore the lack of a double peak for these two curves is curious, and deserves closer inspection.

For a deeper understanding we go back to the surfaces and inspect the contour plots for three of the curves of Fig. 5.18. These are shown in Fig. 5.19, for the cases where \( B = 100, 250, \) and 5000 \( G \). The other parameters for these curves are: \( \lambda = 256.5 \) nm, \( \theta = 70^\circ \), and \( \Gamma = 25 \). Note that the angle here is different than that used earlier for the surface/contour plots; this angle was chosen for two reasons: 1. the geometry of the FLL is different than earlier; 2. this angle is closer to that which yields the maximum saddle point separation – see Fig. 5.14. In Fig. 5.19, it is immediately apparent that the field surfaces undergo changes, especially with respect to the position of the minimum. In the center graph we see a plot like above, where it is easy to see the minima and the saddle points. It is also easy to tell that the two different saddle points (bottom center and along either of the diagonals connecting the lower cores and the upper one) are at distinct field values, separate from each other by a few gauss. When one looks at the left plot one sees that for a lower field of 100 \( G \) the minimum has moved down to a position almost colinear with the cores. Its movement has caused the saddle point which resides there at 250 \( G \) to be suppressed, because there is now very little area near it which is flat. In the right plot is the contour for \( B = 5000 \) \( G \). Here again the minimum has moved, but this time it has shifted up toward the top core. If one looks closely one sees that the saddle points are now almost equivalent in field value. In fact, the field surface now has characteristics...
Figure 5.18: Field distributions $n(b_z)$ for various $B$. The other parameters are: $\lambda = 256.5\; nm$, $\theta = 70^\circ$, and $\Gamma = 25$. Note that the horizontal axis is not real – the curves have been shifted to positions closer together for easier comparison.
which resemble quite closely that of a $\theta = 0^\circ$ equilateral FLL surface. There is still a slight difference in the two saddle points, but the difference at $5000 \, G$ is already too small to see without integrating over a range which stresses the limits of the numerics.

The behavior of the $n(b)$ curves is almost exactly the same as just described for the $n(b_z)$ curves – see Fig. 5.20. Once again the horizontal axis is not real, and the curves have been shifted to the left as described above. The shape and size of the curves remains almost the same for all $B$. But, one can also see differences between the curves. Here, like above, there are some curves with multiple peaks. However, unlike above, the multiple peak graphs exist at all fields. Shown in Fig. 5.21 are the contour plots which correspond to those in Fig. 5.19 for $b_z$. The $100 \, G$ plot clearly shows that the minimum is not shifted down to the $X$ axis, thereby allowing the saddle point to exist with reasonable area. The $5000 \, G$ plot is similar to that in Fig. 5.19, with the distribution looking isotropic. In this case one can see a small peak on the $n(b)$ curve corresponding to this slightly different field-valued saddle point.

We next look at the behavior of the field distributions $n(b_x)$ as the average field magnitude is varied. The curves are shown in Fig. 5.22, where it is seen that the distributions maintain a very consistent shape over a very wide range of average fields. The characteristic asymmetric peaks spread out as the average field is increased. The maximum field value seems to be approaching some upper limit, and the left peak shifts down but also seems to approach a limit. Therefore the values of the fields at the max and left peak saddle do vary with magnitude of the average field. However there appears to be no behavior at all similar to that of the $n(b_z)$ and $n(b)$ distributions, since these saddle points all remain intact throughout the field variation.

Lastly, in Fig. 5.23 are the field distributions for $n(b_y)$. Here there is a very slight variation between the $100 \, G$ curve and the others, which are almost right on top of each other. These distributions therefore seem to converge to a limiting form much sooner than the previous ones, and the overall variation with average field magnitude is almost negligible.

In sum, the variation with average magnetic field of the various component distributions is rather slight, especially for the $b_x$ and $b_y$ distributions. The distributions
Figure 5.19: Shown above are contour plots of $b_z$ surfaces corresponding to three different average field values. From left to right are: $B = 100, 200, 5000 \, G$. The other parameters are: $\lambda = 256.5 \, nm$, $\theta = 70^\circ$, $\Gamma = 25$. Note the movement of the minimum, and its effect on the saddle points, as the average field value increases.
Figure 5.20: Field distributions $n(b)$ for various $B$. The other parameters are the same as in the last figure; here, too, the horizontal axis is not real.
Figure 5.21: Contour plots for the magnitude of $b$ surfaces for $B = 100$, 250, and 5000 $G$. The behavior of the minimum is similar to the $b_z$ contours, but different for the $B = 100$ $G$ case. This position allows for a more concrete saddle point position and yields a peak on the field distribution curve of the last figure.
Figure 5.22: Shown are the field distributions for $n(b_z)$ as the average internal field is varied as labeled. The variations here are relatively minor and do not indicate the type of behavior exhibited above for $n(b_z)$ and $n(b)$.
Figure 5.23: Shown are the field distributions for $n(b_y)$ as the average internal field is varied as labeled. The variations here are very minor, and the distributions appear to have reached a limit of change by $1000 \, G$, after which no further variation occurs with increasing field.
for \( b_z \) and \( b \) generally retain their shape, but closer inspection reveals interesting behavior of the field surfaces. Specifically, this behavior involves the movement of the minimum on both plots, and on the \( b_z \) surface it moves enough to cancel out the effect of one saddle point at \( \theta = 0^\circ \) and \( \theta = 90^\circ \).

### 5.4.3 Effective Penetration Depth Dependence

We next look at the dependence of the field distributions on the value of the effective penetration depth, \( \lambda \). Following the order established above, we will first inspect the distributions of \( n(b_z) \). It is interesting first to establish the theoretical dependence of \( n(b_z) \) on \( \lambda \), and then to check that the theory is self consistent. The quantity of interest is \( < (b_z - B)^2 >^{1/2} \), which can be derived using equations 4.19 and 4.20. If we assume intermediate field values, we can use the approximation \( \lambda^2 G^2 \gg 1[46] \), and the result is:

\[
< (b_z - B)^2 >^{1/2} = \frac{m_{zz}}{\lambda^2 m_1} \sum_{\mathbf{p}=0}^{\infty} \frac{1}{m_{zz} G_z^2 + m_3 G_y^2}
\]  

(5.8)

The important relation of course is that the width of the \( n(b_z) \) distribution varies as \( 1/\lambda^2 \), which we will check below.

The plot of the \( \lambda \) dependences for \( n(b_z) \) is in Fig. 5.24, where one can immediately see a very strong variation. These curves have \( \lambda \) values of 100, 256.5, 500, and 1000 nm. The other parameters, in this as well as the other graphs in this section, are: \( B = 250 \) G, \( \theta = 70^\circ \), and \( \Gamma = 25 \). The values appear at least qualitatively to have the behavior predicted by the theory.

As a check on the theory, the full width at half max (FWHM) is plotted as a function of \( \lambda \) in Fig. 5.25. Shown as points are the widths, and the curve is a fit of \( y = a_0 \cdot \lambda^{a_2} \) to the points. The value of the fit parameter \( a_2 \) is \(-1.9996 \), indicating that the calculation of the lineshapes is correct (at least as far as the theory holds).

The phrase full width at half max needs a little explanation. Theoretically, the peaks on the curves arise from a singularity and are therefore infinite. Numerically they are finite. The method used in this and the following section for determining the FWHM was as follows. First, the peak was found. Second, an average of many points
Figure 5.24: Shown are the field distributions for $n(b_z)$ as the effective penetration depth is varied as labeled. The values of the other parameters are: $B = 250 \, G$, $\theta = 70^\circ$, and $\Gamma = 25$. The variation in width of the distributions theoretically varies as $1/\lambda^2$. 
on either side of the peak (the higher of the two, if two) was found, being careful not
to include points too far from the peak. Third, this average value was then halved,
and the width at this level was found. This method worked well for the \( n(b_x) \) and
\( n(b) \) curves, as will be shown below.

The theoretical dependence of the distributions \( n(b) \) on the effective penetration
depth is the same as for \( n(b_x) \). Using the same approximations as above, it can be
shown that the FWHM of these distributions also vary as \( 1/\lambda^2 \). The plot of the field
distributions for various \( \lambda \) is in Fig. 5.26, and the plot of FWHM for these curves
versus \( \lambda \) is in Fig. 5.27. The solid curve in the latter figure is another fit to the data,
with the exponent this time having a value of \( -1.995 \).

The \( \lambda \) dependence of the distributions \( n(b_x) \) and \( n(b_y) \) are shown in Figs. 5.28
and 5.29. Once again a very strong dependence on \( \lambda \) is evident. Here, we look for the
width of the distribution as a variation with respect to zero Gauss \(- \langle (b_x - 0)^2 \rangle^{1/2} \)
and \(- \langle (b_y - 0)^2 \rangle^{1/2} \). Using similar methods to those used above, we can show that
these relationships are given by:

\[
\langle (b_x - 0)^2 \rangle^{1/2} = \sum_{G \neq 0} \frac{B m_{zz} G_y^2}{\lambda^2 G^2 m_1 (m_{zz} G_x^2 + m_3 G_y^2)}
\]

\[
\langle (b_y - 0)^2 \rangle^{1/2} = \sum_{G \neq 0} \frac{B m_{zz} G_x G_y}{\lambda^2 G^2 m_1 (m_{zz} G_x^2 + m_3 G_y^2)}
\]

Hence both should also display a \( 1/\lambda^2 \) behavior. A plot of the widths of the \( n(b_x) \)
lineshapes versus \( \lambda \) is shown in Fig. 5.30. The points show the correct qualitative
behavior, but the fit produced an exponent of \( -1.56 \) when fit to only the right four
points. When fit with all points the exponent was \( -1.35 \), which is worse. The widths
were determined by the spread between the left peak and the farthest right non-zero
point of the distribution. The calculation of the widths was somewhat complicated
by the fact that the lineshape form changed at \( \lambda = 50 \, \text{nm} \). The steep fall on the right
side became a more gradual decline, and the two peaks moved closer together. This
made defining a consistent width difficult, and hence a fit was done over the final four
points to help avoid this problem. The value for the four points is still not great, and
this is due to the fact that the 50 nm point is crucial for a more complete mapping
of the behavior.
Figure 5.25: A plot of the FWHM of the $n(b_x)$ distributions as a function of the parameter $\lambda$. The points are the FWHM of the calculated distributions, and the curve is a fit to these points. Anisotropic London theory predicts a $1/\lambda^2$ dependence, the fit produced an exponent of $-1.9996$. 
Figure 5.26: Shown are the field distributions for \( n(b) \) as the effective penetration depth is varied as labeled. The values of the other parameters are: \( B = 250 \) G, \( \theta = 70^\circ \), and \( \Gamma = 25 \). The variation in width of the distributions theoretically varies as \( 1/\lambda^2 \).
Figure 5.27: A plot of the FWHM of the $n(b)$ distributions as a function of the parameter $\lambda$. The points are the FWHM of the calculated distributions, and the curve is a fit to these points. Anisotropic London theory predicts a $1/\lambda^2$ dependence, the fit produced an exponent of $-1.995$. 
Figure 5.28: A plot of the $n(b_x)$ distributions as a function of the parameter $\lambda$. 
Figure 5.29: A plot of the $n(b_y)$ distributions as a function of the parameter $\lambda$. 
Figure 5.30: A plot of the width of the $n(b_x)$ distributions as a function of the parameter $\lambda$. The points are the width of the calculated distributions, and the curve is a fit to these points. The fit produced an exponent of $-1.56$. 
The widths of the $n(b_y)$ distributions were more well-behaved, as shown in Fig. 5.31. The exponent was fit to a value of $-1.904$, which is fairly close to the expected $-2$. The widths here were measured between the parts of the curves in Fig. 5.29 where the distribution drops vertically to zero probability. The lineshapes for $n(b_y)$, unlike those of $n(b_z)$, maintain a consistent form for all $\lambda$ and therefore allow for a more consistent width measurement. This is evidenced by the much better behavior of the points and the better fit parameter for the exponent.

To summarize, the anisotropic London theory predicts a $1/\lambda^2$ dependence for the width of the various distributions discussed. Examination of the calculated distributions yields almost exactly this behavior for both the $n(b)$ and $n(b_z)$ curves, very close correspondence by the $n(b_y)$ curves, and a somewhat less than perfect match for the $n(b_x)$ curves. This last is mostly due to difficulties in analysis, and with a more consistent method would probably yield better results.

### 5.5 Oriented and Polycrystalline Distributions

The distributions presented so far correspond to those of single-crystal samples — there is a known crystalline direction $c$ at an angle $\theta$ to the average field $B$ direction. Therefore the entire discussion up to now is applicable to single-crystals. However, the techniques can be extended to cover samples which are not single crystals, such as oriented and polycrystalline superconducting YBCO materials.

Oriented samples are generally fabricated:

1. in a suspension which is placed in high magnetic fields ($\sim 8$ T)[86, 87]. The crystallites of superconducting material in the suspension experience a torque which tends to align their $c$ axis with the field, which leaves the resulting bulk sample with a high degree of orientational order [88].

2. by melt texturing a thick film[89].

The materials produced in this way generally have $\sim 90\%$ orientation, meaning that the crystallites making up the material have their $c$ axes aligned to within $\pm 5^\circ$ of that of the applied field during fabrication.
Figure 5.31: A plot of the width of the $n(b_y)$ distributions as a function of the parameter $\lambda$. The points are the width of the calculated distributions, and the curve is a fit to these points. The fit produced an exponent of $-1.904$. 
Polycrystalline samples are generally produced from precursor powders which are mixed in the proper proportions and pressed into pellets. These pellets are then reacted in a furnace with oxygen, bringing the oxygen level up to that required for superconductivity[90, 91, 92]. In these samples there is no unique crystalline direction – the crystallites are isotropically oriented in space.

The extension of the field distribution calculations to these types of materials is simple. One calculates many single crystal distributions \( n_{sc}(b, \theta) \) for various angles and then sums and averages appropriately. This is expressed mathematically as:

\[
    n(b_z) = \int_{\theta_{\min}}^{\theta_{\max}} n_{sc}(b_z, \theta) \sin \theta d\theta
\]

(5.11)

where \( n(b_z) \) is the oriented or polycrystalline distribution corresponding to the choice of angles \( \theta_{\min} \) and \( \theta_{\max} \).

For example, one can find the oriented sample field distribution of a 90% oriented sample by setting \( \theta_{\min} = 0^\circ \) and \( \theta_{\max} = 5^\circ \). A plot of this calculation is shown in Fig. 5.32. The parameters for the calculation are: \( B = 1000 \, G \), \( \lambda = 256.5 \, nm \), and \( \Gamma = 25 \). This distribution looks similar to the typical single crystal distributions, with the sharp discontinuity at the minimum field, the sharp peak, and then the long, high field tail. The peak is not quite as sharp as before, and the width is slightly enhanced also. These effects are similar to what is seen experimentally in single crystal lineshapes and attributed to disorder in the FLL[17, 29, 76]. It would therefore be difficult to distinguish these subtle differences from experimental reality.

An example of a polycrystalline distribution is in Fig. 5.33. The parameters here are the same as those in Fig. 5.32. This distribution is much different than the single crystal lineshapes, although it retains the sharp peak and long high field tail. The big difference in this plot is the gradual increase on the left-hand side of the distribution leading up to the peak. This is a consequence of the \( \sin \theta \) averaging over lower \( \theta \) single-crystal distributions. The single crystal peaks, which start out as tall as the remaining peak in the figure, are reduced by \( \sin \theta \) and hence give the smooth rise.

Experimental polycrystalline distributions on YBCO have a shape similar to the theoretical one. The differences lie in the smoothing of the overall line shape (presumably due to disorder in the FLL) and also in the notable lack of the long high
Figure 5.32: A theoretical oriented sample field distribution for a material like YBCO: $B = 1000 \, G$, $\lambda = 256.5 \, nm$, $\Gamma = 25$. The degree of orientation is roughly 90%, meaning that the crystallite c axes fall to within $\pm 5^\circ$ of the intended orientation direction.
Figure 5.33: A theoretical polycrystalline field distribution for a material like YBCO: $B = 1000 \, G$, $\lambda = 256.5 \, nm$, $\Gamma = 25$. Characteristic features are the gradual rise on the left and the long high frequency tail.
frequency tail \cite{83, 82}. An example of an experimental polycrystalline lineshape is in Fig. 5.34, which is of a bulk sample of YBCO at $T = 20K$. The data were taken in 1990 at TRIUMF by Dr. Carey Stronach and Christof Niedermayer on a polycrystalline YBCO sample fabricated by the author. It can be seen here that the curve is not as sharp as the theory, and that the high-field tail is almost non-existent. In order to make the theory appear more like the data, various schemes have been developed to alter the shape of the theoretical curves. The most popular is convolution with another curve, usually a gaussian, which causes the theoretical curve to broaden and smooth out, like the data \cite{17, 76}. The convolution curve is also shown in Fig. 5.34, but is shown on a reduced scale for clarity.

Other techniques have been developed to alter the appearance of the theoretical polycrystalline lineshapes. All attempt to decrease the theory's high frequency tail and increase the lower frequency side. A few references are included here for the interested reader \cite{93, 94}. These references suggest that the London theory is correct and applicable, and that the discrepancies between the theoretical and experimental results are largely due to things like disorder, bad samples, and pinning.
Figure 5.34: Shown are two representations of the same field distribution which are Fourier Transformed from $\mu$SR data taken at TRIUMF on a bulk YBCO sample. The temperature of the sample is 20 K. The dotted line is the raw transform, with no tweaking. The solid curve is the result of apodization (multiplication in time space) with a gaussian envelope which has a decay constant of 5 $\mu$s – this is essentially convolution.
Chapter 6

Off-Axis Fields in Anisotropic Superconductors

In this chapter the microscopic magnetic field components perpendicular to the average field \( B \) direction are investigated. A numerical simulation of muon behavior within the field distributions of the last chapter is presented. The resulting muon time histograms are theoretical predictions of the experimental \( \mu \text{SR} \) data. Coupling existing \( \mu \text{SR} \) techniques and Fourier transform analysis, the data are treated to reveal what we call moments of the microscopic field distributions. These moments tell us interesting things about the nature of the fields within the materials, including an estimate of how far the average field is off in direction from the applied field.

6.1 Simulating Muon Behavior within the FLL

In chapter 4 the anisotropic London theory was developed. It describes a method for calculating the magnetic fields within the FLL of an anisotropic superconductor. In chapter 5 this method was implemented on a computer, allowing the magnetic fields of the FLL to be numerically determined. Using these fields, and our knowledge of how muons precess in magnetic fields (chapter 3), allowed a simulation program to be implemented where muons stop uniformly within a triangular FLL. The simulation results in data which should correspond to high statistics \( \mu \text{SR} \) data. The manipulation
One first must understand the geometry used for the simulation. The geometry and trigonometric symbols used are shown in Fig. 6.1. Particular values of the angles will be referenced later when simulation results are discussed.

First, the directions $x$, $y$, and $z$ are those which line up with three mutually perpendicular detectors - the LAB axes. All further directions are defined with respect to these three. The average field $\mathbf{B}$ direction is determined by the angles $\beta$ and $\phi_{\beta}$. The $c$ axis direction is determined by the angles $\alpha$ and $\phi_{\alpha}$. Once $\mathbf{B}$ and $\mathbf{c}$ are specified, the angle $\theta$ between them is determined by the dot product: $\cos \theta = \mathbf{B} \cdot \mathbf{c} / |\mathbf{B}|$. Knowing $\theta$, the magnitude of $\mathbf{B}$, and the material (i.e. $\lambda$ and $\Gamma$) allows the calculation of the fields within the FLL by the methods developed in chapter 4. Recall that the crystal (field) axes directions $X$, $Y$, and $Z$ are defined by
the relations given in chapter 4 as $B \times (B \times c)/B^2$, $B \times c/B$, and $B$, respectively, and are not shown in the figure for the sake of clarity.

With the fields known, the simulation can commence. An outline of the program which carries out the simulation follows:

1. Pick a direction for the initial polarization $P(0)$ of the muon by choosing the angles $\gamma$ and $\phi_\gamma$.
2. Pick a field value $b_i$ to search for in the FLL.
3. If $b_i$ is found at a point within the FLL, calculate the projection of the muon's polarization along each of the three directions $x$, $y$, and $z$ for a length of time $T$ as it precesses about the vector local $b$.
4. Using the integration method of chapter 4, weight each muon's contribution according to the amount of area that the field occupied within the cell.
5. Keep a running field/areal average for each of the three mutually perpendicular directions where one has imagined the detectors to be.

The program is contained in appendix B for the interested reader to peruse.

The details of point 3 above deserve some discussion. Recall from chapter 4 that the FLL unit cell is divided into rectangular sub-cells. The field values of the FLL are calculated at the corners of the sub-cells. Once a field $b_i$ is found to lie within a sub-cell, its direction must be determined. We only know the field values at the corners of a sub-cell, so to find the direction of the field $b_i$ at its intersection point of the sub-cell, we average the components at the two closest known points. These averaged components are then used for the direction of the local field. The muon polarization is then broken into components along this average field direction and perpendicular to it. The perpendicular component is further broken into two components, where the $x'$ direction is defined to lie in the plane defined by $P(0)$ and $b_{av}$. The $x'$ and $y'$ components then rotate about the $b_{av}$ direction in discrete time steps for some
amount of time $T$ via the equation:

$$
\begin{pmatrix}
x'(t) \\
y'(t) \\
z'(t)
\end{pmatrix} =
\begin{pmatrix}
\cos \omega t & \sin \omega t & 0 \\
-\sin \omega t & \cos \omega t & 0 \\
0 & 0 & 1
\end{pmatrix}
\begin{pmatrix}
x'(0) \\
y'(0) \\
z'(0)
\end{pmatrix}
$$ (6.1)

The components $x'$, $y'$, and $z'$ are then transformed into the LAB axes by rotations about the appropriate axes. The rotation matrix for this is:

$$
R =
\begin{pmatrix}
\cos \delta \cos \phi_b & -\sin \delta \cos \phi_b & \sin \phi_b \\
\cos \delta \sin \phi_b & \cos \phi_b & \sin \delta \sin \phi_b \\
-\sin \delta & 0 & \cos \delta
\end{pmatrix}
$$ (6.2)

where $\delta$ and $\phi_b$ are the polar and azimuthal angles determining the direction of $b_{av}$ with respect to the LAB axes.

6.2 Assumptions of the Simulation Program

This model assumes certain things. For instance, it assumes that all muons arrive with their polarization in the same direction. This is largely true, but it should be remembered that real muon beams may be only 90% polarized. This program, like the previous one, assumes that the FLL consists of a well defined, equilibrium set of straight cores which are not distorted in any way. The possible problems with this assumption were discussed above in section 4.7. Further, this program assumes a uniform stopping distribution of muons throughout the FLL. That is, that the muons are as likely to stop in one spot in the FLL as in any other, and that there will be no feature within the FLL or material which will cause any denser stopping of muons than any other. In order to satisfy this assumption it will be necessary to have very good quality samples which are relatively pin-free and uniform. This assumption is generally quite good with respect to the FLL, because the length scale of the FLL is of the order of $\lambda$, while the muon is believed to stop near the oxygen in the plane or chain of the unit cell (Fig. 4.4). The length scale of the FLL is therefore something like 150 nm, while the muons are stopping within the unit cells (with a bond length
of approximately 0.1 nm\(^2\) which are at most of order 1 nm. Therefore many muons should stop within the the FLL and sample it uniformly.

The model also assumes no type of interaction of the muon with anything other than the magnetic field, and that the muons do not diffuse throughout the material. Since, as mentioned above, the muons are believed to stop near an oxygen, then there should be no dipole-dipole interaction since oxygen nuclei are spin zero in their most abundant form. Due to the \(1/r^3\) nature of the dipole-dipole interaction, no other nuclei will have an influence even close to that of the FLL fields. In addition, it has been shown [90] that the muons do not diffuse in the high-\(T_c\) materials, so motion effects should not play a role in the results.

A further assumption is that the detectors are of infinitesimal size and all of equal efficiency. In reality the detectors cover some solid angle in space and therefore the projection is over this area some distance away from the muon. The efficiency of each detector will in general be different, and there may even be differences within a single detector depending on where the positrons are incident. The actual signal in a finite sized detector will therefore be an average of the type of signal calculated here over the solid angle of the detector and also over the efficiency as a function of position on the detector. This type of averaging will cause the signal to become smeared somewhat, but for higher statistics the problem should be less noticable.

Lastly, it must be understood that the simulation calculates the projection of the muon polarization simultaneously along all three directions. Therefore each muon contributes to each detector’s signal for the entire time \(T\) that the simulation runs. This is certainly much different than a real monte carlo simulation, where actual decays with the decay asymmetry folded in would take place at times weighted by the muon lifetime. However, real data will approach our simulation for high statistics.

In order to see the kind of experimental statistics needed to approach our simulations, we did a numerical test using a theoretical lineshape calculated via the methods of the last chapter. We first calculated the lineshape from the \(z\) surface, and then Fourier transformed into time space to simulate the asymmetry histogram – see Fig. 6.2; top panel is the theoretical distribution, bottom is the FT of the
Figure 6.2: Top is a theoretical field distribution for the following parameter values: $B = 1000 \, G$, $\lambda = 120 \, nm$, $\theta = 0^\circ$, and $\Gamma = 25$. The bottom is the real Fourier transform of this distribution, which corresponds to the time-space or asymmetry representation.
CHAPTER 6. OFF-AXIS FIELDS IN ANISOTROPIC SUPERCONDUCTORS

Figure 6.3: This is a plot of the lower data of the last figure after being scaled for $10^6$ total events, run through the Poisson distribution random noise generator, and then having the exponential decay of the muon lifetime removed. A real muon histogram’s statistics follow a Poisson distribution, where the error in a particular histogram bin goes as $N^{1/2}$, where $N$ is the counts in the bin.

distribution. These data were then transformed into raw histogram-like data by including the muonic decay and a user-specified total count parameter. This total count parameter could be varied to simulate the statistics of the histogram – higher total counts meaning better statistics. The now raw histogram was then made noisy by running it through a random number generator [97] which produced a spectrum with a Poisson distribution of noise in it. This noisy spectrum was then changed back to an asymmetry type of spectrum – see Fig. 6.3 – and then Fourier transformed into frequency space to see the effects on the lineshape.

The results of three such trials are shown in Fig. 6.4. The amount of statistics is varied by an order of magnitude as one goes from top to bottom from $10^6$ to $10^5$. The effect of better statistics is obvious, as the original lineshape of Fig. 6.2 is virtually recovered in the bottom plot. We therefore conclude that at least $10^7$ total events are necessary for an experiment to reasonably reproduce these results.
Figure 6.4: Top is the resultant distribution for the total events equalling $10^6$. Here the noise is such that the details of the underlying distribution are difficult to determine. The middle panel is for $10^7$ total events, where the details of the line shape are becoming more clear. The bottom is for $10^8$ events, where most features of the original distribution are reproduced.
CHAPTER 6. OFF-AXIS FIELDS IN ANISOTROPIC SUPERCONDUCTORS

6.3 Results of the Simulation

6.3.1 Results for a General Case

An example of the results of a simulation are shown in Fig. 6.5. This case has the average field at a position given by $\beta = 0^\circ$, $\phi_\beta = 0^\circ$, the c axis at $\alpha = 45^\circ$, $\phi_\alpha = 0^\circ$, and the initial polarization at $\gamma = 80^\circ$, $\phi_\gamma = 0^\circ$. One can see the three separate signals in the figure as different line types. Each shows a relatively large initial amplitude and oscillations which decay at a certain rate. The oscillation frequency closely corresponds to that given by $\omega = \gamma B$, but the signal is actually not simply composed of one frequency, as one might have guessed. The decay of the signal with time is interesting because it is evidence of the existence of the off-axis fields. That is, the muons are all arriving in phase due to their polarization, but because each sees a slightly different local field (in both magnitude and direction) each precesses at its own rate in its own direction. Therefore the muons all slowly become out of phase with each other and the net signal seen at the detector decreases. The proof of the existence of the off-axis fields is this dephasing in the signal. If there were no off-axis fields, then all muons would see the same local field and they would stay together as they precessed. Therefore this dephasing, if all other sources of field inhomogeneity can be ruled out, is definite proof of the existence of off-axis fields.

6.3.2 Coupling to/from Local Fields

An interesting detail of these simulations is found by observing the behavior of the various signals as the direction of the initial polarization of the muons is varied. For example, given a field value of $B = 100 \text{ G}$ in the direction $\beta = 0^\circ$, $\phi_\beta = 0^\circ$, a c axis direction of $\alpha = 45^\circ$, $\phi_\alpha = 0^\circ$, and a YBCO-like material, Fig. 6.6 shows what appears to be a phase shift in the signals for two directions of initial polarization. The top panel shows the $x$ signals, the center shows the $y$, and the bottom shows the $z$ signals. There appears to be a phase shift in both the $x$ and $y$ signals between the $\mathbf{P}(0) \parallel z \ (\gamma = 0^\circ)$ and $\gamma = 20^\circ$. The increase in the initial $x$ amplitude is explainable in that as $\gamma$ is increased, the projection onto $x$ must therefore increase. But this does
Figure 6.5: Results of a numerical simulation of muons within the FLL of a material like YBCO. The average magnetic field $B = 100 \, G$ and is in a direction given by $\beta = 0^\circ, \phi_\beta = 0^\circ$. The crystal $c$ axis is at $\alpha = 45^\circ, \phi_\alpha = 0^\circ$. The initial polarization is at $\gamma = 80^\circ, \phi_\gamma = 0^\circ$. The lines correspond to each detector as labeled in the graph.
not explain the phase change in the \( y \) signal.

To investigate further, we calculate the signals as the initial polarization is varied from \( \gamma = 0^\circ \) to \( 4^\circ \). These are shown in Fig. 6.7. We again see what appears to be phase shifts in the \( x \) and \( y \) signals as the angle \( \gamma \) is varied. When \( P(0) \) is farther from \( B \), the oscillations begin in the proper direction for this geometry via \( dP/dt = P \times B \). One can see this by noting that the initial amplitude along the \( y \) direction goes negative, corresponding to precession about the \( z \) axis in a clockwise sense as viewed along \( +z \) back toward the origin. For lower angles of \( P(0) \) there is a definite change in the muon behavior, for now the initial amplitude along \( y \) is no longer negative, but positive. It seems that the muon's behavior shifts from one where the precession is governed by the direction of the average field \( B \) at larger angles (\( \gamma \geq 3^\circ \)) to one which is controlled by the local fields \( b \) near \( \gamma = 0^\circ \) - the muons will decouple from the local fields at large \( \gamma \). It would therefore be interesting to look experimentally for this type of behavior, as it would be yet another indication of the existence of off-axis fields.

6.4 M onents of the Field Distributions

6.4.1 Starting Point

It was mentioned above that a technique was developed which enables the determination of various moments of magnetic field distributions. This technique will now be discussed in some detail.

The starting point for this discussion is the examination of three special cases of experimental geometries. In the first case, the average field is along the LAB \( z \) direction and the initial polarization is along the \( x \) direction. For this case, the polarization projections along each of the three LAB axes (corresponding to equation 3.2) are:

\[
P_{xy}(t) = \frac{b_x b_z}{b^2} (1 - \cos \omega t) - \frac{b_y}{b} \sin \omega t \tag{6.3}
\]

\[
P_{xz}(t) = \frac{b_x b_z}{b^2} (1 - \cos \omega t) + \frac{b_y}{b} \sin \omega t \tag{6.4}
\]
Figure 6.6: Plots of the three $\mu$SR signals for two different orientations of $P(0)$: $\gamma = 0^\circ$ and $\gamma = 20^\circ$. In the top figure there appears to be a phase shift between the two signals. This phase shift also appears in the center plot for the $y$ signals.
Figure 6.7: The variation in the initial part of the curves as the direction of the muonic polarization is varied through the angles labeled in the figure. Note the phase change in the $x$ and $y$ curves as the angle $\gamma$ increases from $0^\circ \rightarrow 4^\circ$. 
\[ P_{xx}(t) = \frac{b_x^2}{b^2} + \frac{b_z^2}{b^2} \cos \omega t \] (6.5)

where the first of the double subscripts indicates the direction of the initial polarization, and the second is the LAB direction upon which the polarization is projected.

The next case is the same, except that now the initial polarization is along the LAB \( y \) direction:

\[ P_y(t) = \cos \omega t - \sin \omega t \] (6.6)

\[ P_{yx}(t) = \frac{b_y b_x}{b^2} (1 - \cos \omega t) + \frac{b_z}{b} \sin \omega t \] (6.7)

\[ P_{xy}(t) = \frac{b_x^2}{b^2} + \frac{b_z^2}{b^2} \cos \omega t \] (6.8)

The final case has the initial polarization along \( z \):

\[ P_{zx}(t) = \frac{b_x b_z}{b^2} (1 - \cos \omega t) - \frac{b_x}{b} \sin \omega t \] (6.9)

\[ P_{yz}(t) = \frac{b_y b_z}{b^2} (1 - \cos \omega t) + \frac{b_z}{b} \sin \omega t \] (6.10)

\[ P_{zy}(t) = \frac{b_x^2}{b^2} + \frac{b_z^2}{b^2} \cos \omega t \] (6.11)

The simulations for each of these three situations are in Fig. 6.8, presented from top to bottom as \( P(0) \) along \( x, y, \) and \( z \), respectively. The situation in the bottom case is analogous to a LF \( \mu \)SR experiment, and the simulation correspondingly shows a set of signals consistent with this geometry. The other two cases correspond to transverse field (TF) \( \mu \)SR geometries, as indicated by the large precession amplitudes at small times.

It is interesting to note the long-time (asymptotic) values of each of the three signals from the bottom plot. The \( z \) signal starts at 1.0 and oscillates with a decaying amplitude. The asymptotic value of this signal is about 0.92, which is a reflection of the long-time value of the term \( b_z^2/b^2 \) in equation 6.11, averaged over the FLL unit cell area. The \( x \) signal of this plot has a long time value which is non-zero, and in fact has a value which is near 0.066. This value is the long-time value of the correlation function \( < b_x b_z/b^2 > \) from equation 6.9, where the \( < \ldots > \) now indicate the average over the FLL area. The \( y \) signal also oscillates and decays with time, but...
its long-time value is zero. This means that in equation 6.10 the correlation function
\[ \langle b_x b_x / b^2 \rangle = 0 \text{ as } t \to \infty. \]

The average of the functions over the area of the FLL can be expressed mathematically as, \textit{e.g.}:

\[
\frac{b_x b_x}{b^2} \rightarrow \frac{b_x b_x}{b^2} = \int_A n(b) \frac{b_x b_x}{b^2} dA
\]  \hspace{1cm} (6.12)

where \( b_x, b_z, \) and \( b \) are functions of the position within the FLL, \( dA = dx \, dy \), and
the arrow in the equation indicates that the term from equation 6.9 is taken over to
an average over the FLL area. The lines on the graphs in Fig. 6.8 therefore really
represent these averaged functions as functions of time, and should be thought of as
such.

Another interesting interpretation of this long-time behavior is to visualize a net,
ensemble average polarization vector and its projection onto the three directions.
This vector begins, \textit{e.g.}, aligned along \( z \), and precesses on a cone which causes small
oscillations in the \( x \) and \( y \) directions. As time progresses this cone becomes tilted off
of the \( z \) axis and up toward the \( x \) axis by a small angle. The net polarization can
then also be thought to have shifted its orientation from along \( z \) to along this newly
defined direction. This effect has been seen independently (and published earlier) by
Riseman [17].

6.4.2 Finding Moments

The asymptotic behavior of the simulations noted above gives some information about
particular moments of the field distributions at long times. However, this has arisen
in an almost accidental fashion. It would therefore be useful to develop a method to
actively extract further information \textit{(i.e.} moments\textit{)} from the simulations.

Our method of arriving at these moments involves manipulating the simulation
data of the various plots in Fig. 6.8 using elementary mathematics, algebra, and
Fourier transform techniques. For example, if we wish to find the moment \( \langle b_z / b \rangle \),
we can do it by subtracting the \( z \) simulation curve of the center plot of Fig. 6.8 from
the \( y \) simulation curve of the bottom plot. We then divide each element by 2, and
Figure 6.8: Shown from top to bottom are three simulations of $\mu$SR data from three different directions of the initial muon polarization. The top has $P(0)$ along $x$, the middle along $y$, and the bottom along $z$. The other values of the parameters are: $B = 100 \, G$, $\lambda = 256.5 \, nm$, $\theta = 45^\circ$, and $\Gamma = 25$. 
take the Fourier sine transform over time. This results in a function of frequency, \( < b_x(\omega)/b > \), which when integrated over frequencies yields a number for \( < b_x/b > \).

How do we know this will give \( < b_x/b > \)? The answer requires inspection of equations 6.3 through 6.11. The equation corresponding to the \( y \) curve in the bottom plot is 6.10, while that corresponding to the \( z \) curve in the center plot is 6.6. Both of these equations contain the term \( b_x/b \), and the curves for both are averaged over the same FLL area. It is therefore possible to do the subtraction of equation 6.6 from equation 6.10:

\[
< \frac{b_x}{b^2} (1 - \cos \omega t) > + < \frac{b_x}{b} \sin \omega t > - \left[ < \frac{b_x}{b^2} (1 - \cos \omega t) > - < \frac{b_x}{b} \sin \omega t > \right] = 2 < \frac{b_x}{b} \sin \omega t > \quad (6.13)
\]

Dividing by 2 and taking the sine Fourier transform gives:

\[
\int_0^T < \frac{b_x}{b} \sin \omega t \sin \omega' t \, dt \to \int_0^T \int_A n(b) \frac{b_x}{b} \sin \omega t \sin \omega' t \, dt \, dA \quad (6.14)
\]

Now switch the order of integrations:

\[
\int_A n(b) \, dA \int_0^T \sin \omega t \sin \omega' t \, dt \quad (6.15)
\]

The integral over time is approximately a delta function in frequency:

\[
\int_0^T \sin \omega t \sin \omega' t = \frac{\sin(\omega - \omega')t}{2(\omega - \omega')} - \frac{\sin(\omega + \omega')t}{2(\omega + \omega')} \approx \delta(\omega - \omega') \quad (6.16)
\]

Substituting this in equation 6.14 gives:

\[
\int_A n(b) \frac{b_x}{b} \delta(\omega - \omega') \, dA = < \frac{b_x}{b} > n(b) \quad (6.17)
\]

which is the moment \( b_x/b \, n(b) \) of the field distribution. For the data shown in Fig. 6.8, the result using this procedure is in Fig. 6.9.

This plot represents the distribution \( n(b) \) of the magnitude of the local fields weighted by the term \( b_x/b \) for each \( b \) within the distribution \( n(b) \). It is not the distribution of \( b_x/b \) itself, which would most likely look much different. This idea of a weighted \( n(b) \) distribution is central to understanding the concept of the moments, and is necessary for an understanding of what follows.
Figure 6.9: This is the plot resulting from the analysis deriving \( b_x n(b) \) using the simulation data with parameters \( B = 100 \, G \), \( \lambda = 256.5 \, nm \), \( \theta = 45^\circ \), and \( \Gamma = 25 \).
CHAPTER 6. OFF-AXIS FIELDS IN ANISOTROPIC SUPERCONDUCTORS

6.5 Interesting Moments

6.5.1 n(b)

Back in chapter 4 a theoretical procedure was discussed in which the distribution of the magnitude of the local field \( b \) could be obtained. This is of interest because it sheds light on the microscopic properties of the equilibrium FLL. It would therefore be useful if, in addition to this purely theoretical calculation, there were an experimental means to see this distribution. Using the techniques described above, such an experimental procedure is now described.

One first needs to acquire high statistics data according to the geometries of Fig. 6.8. This means that the data will correspond to the averaged versions of equations 6.3 through 6.11. Using these equations as a guide, we recognize that the averaged forms have terms which look like \( < n(b) \frac{b_2^2 + b_2^2}{b_2^2} \cos \omega t > \), which appear in equations 6.5, 6.8, and 6.11. We first subtract off the long-time non-zero values, since these will be due to the non-sinusoidal terms in the equations. Next, we add these terms together:

\[
< n(b) \frac{b_2^2 + b_2^2}{b_2^2} \cos \omega t > + < n(b) \frac{b_2^2 + b_2^2}{b_2^2} \cos \omega t > + < n(b) \frac{b_2^2 + b_2^2}{b_2^2} \cos \omega t >
\]  

(6.18)

All terms are averaged over the same FLL, so the \(< ... >\) are all the same. Also, \( n(b) \) is exactly the same in each term. Recognizing this, we may write:

\[
< n(b) \left( \frac{b_2^2 + b_2^2 + b_2^2 + b_2^2 + b_2^2 + b_2^2}{b_2^2} \right) \cos \omega t >
\]  

(6.19)

The term within the parentheses becomes \( 2b_2^2/b_2^2 \), leaving:

\[
< n(b) 2 \cos \omega t >
\]  

(6.20)

Dividing by 2 and Fourier cosine transforming leaves \( n(b) \), which is what we were after.

This analysis has been carried out using the simulation data of Fig. 6.8, and the result is shown in Fig. 6.10. Also shown is the result of the purely theoretical calculation of chapter 4 for comparison sake. The simulation result is not as clean and
smooth as the integrated curve, but that is to be expected due to the lesser overall resolution afforded by the Fourier transform, as well as the "ringing" oscillations which occur as a result of the finite time range of the integration (recall the approximation to a gaussian above in equation 6.16).

The important thing to keep in mind here is that this is a method which allows experimental extraction of the microscopic distribution of the magnitude of the local field \( b \). What is most commonly done in \( \mu \)SR experiments is a Fourier transform of TF-\( \mu \)SR data at high applied fields. To look at what this actually means, we take, e.g., equations 6.3 and 6.5, which detect the transverse signals in the simulation. For large applied fields, the components \( b_x \) and \( b_y \) are quite small compared to the longitudinal component \( b_z \), which is nearly equal to \( b \). Neglecting the off-axis field components, these equations become:

\[
P_x(t) \approx -\frac{b_x}{b} \sin \omega t
\]
\[
P_z(t) \approx \frac{b_z}{b} \cos \omega t
\]

Taking the sine and cosine transforms, respectively, of the averaged forms of these equations (and using the same delta function approximation as above) yields:

\[
< n(b) \frac{b_x}{b} \delta(\omega - \omega') > \tag{6.23}
\]
\[
< n(b) \frac{b_z}{b} \delta(\omega - \omega') > \tag{6.24}
\]

Now, in the limit that \( b_z \to b \), each of these becomes the field distribution \( n(b) \). This limiting case is closely approximated when there is a high applied field, but, as mentioned earlier in chapter 4, this approximation does not hold for low applied fields. Our method, while also using the delta function approximation, is independent of the high-field approximation, and in principle will work for any value of applied field.

The example above for \( n(b) \) is descriptive, but the line shape is rather smooth. To show that the method of moment finding is in fact a good method, capable of reproducing more complicated forms, simulation data have been prepared corresponding to a field \( B = 250 \, G \), an effective penetration depth \( \lambda = 120 \, nm \), anisotropy parameter
Figure 6.10: The top plot is the moment $n(b)$ resulting from this analysis of the simulation data. The bottom plot is the result for the same moment but with the integration method of last chapter – shown as a check of the method.
Γ = 25, and angle θ = 45°. Although these values do not correspond to any real material, the line shapes have more structure and therefore are a better test of the method. The results of the simulation and the integration are shown in Fig. 6.11, top and bottom, respectively.

This distribution has more structure, as indicated in the bottom plot of Fig. 6.11. There is a sharp rise at the minimum frequency, followed by a peak, and then a dip, then another peak, and then a long high-frequency tail which is asymptotic to zero. The simulation does a good job of reproducing this structure, getting not only both peaks correctly but also showing the long high-frequency tail. We will show more of this type of simulation later for different moments.

6.5.2 \( b_1^2 n(b) \)

Another moment of interest is \( b_1^2 n(b) \), which is the distribution of the magnitude of the field \( b \) weighted by the square of the transverse field at each \( b \) value within the FLL. This distribution can be obtained in the following way. The transverse field components are of course designated \( b_x \) and \( b_y \), and therefore the square of the transverse component is simply:

\[
b_1^2 = b_x^2 + b_y^2
\]

(6.25)

Upon inspection of equations 6.3 - 6.11, one sees this form contained in equation 6.11 as the right hand term. Subtracting off the time independent term from this equation leaves:

\[
< \frac{b_x^2 + b_y^2}{b^2} \cos \omega t >
\]

(6.26)

If we now take the Fourier cosine transform, and while doing so multiply each term by the field (frequency) squared, we have the following:

\[
\int_0^T b^2 < \frac{b_x^2 + b_y^2}{b^2} \cos \omega t > \cos \omega' t dt = \int_0^T \cos \omega t \cos \omega' t dt = < \frac{b_x^2 + b_y^2}{b^2} > n(b)
\]

(6.28)

The result of doing this calculation on the data of Fig. 6.8 is shown in Fig. 6.12 at the top. As a check on its validity, the integration method was again employed, but
Figure 6.11: This figure is the same as the last, except now the parameters are: $B = 250 \, G$, $\lambda = 120 \, nm$, $\theta = 45^\circ$, and $\Gamma = 25$. The integration result on the bottom shows more structure, which is well reproduced in the simulation distribution above.
now modified to calculate the quantity \( (b_x^2 + b_y^2) n(b) \) at each sub-cell of the FLL area, and weight appropriately as described above. The result from this method is shown in Fig. 6.12 in the bottom panel.

Once again, the integration method yields a much smoother result with finer detail. There is the rise at the minimum, a sharp peak, and then a high frequency tail which is rather pronounced. The plot in the upper half is the simulation data, which reasonably well matches the integration result in that it has all of the same features of the shape. It does have some rather pronounced ringing, which is a result of the Fourier transformation approximation to a delta function. If one were to smooth this ringing out then the shape would be an almost exact match to the integration data.

It is worth emphasizing again that this is not the distribution of \( b_x^2 \) itself, but is the distribution \( n(b) \) weighted at each \( b \) of its calculation by the corresponding value of \( b_x^2 \). The theoretical analysis given above for the simulations lends itself to this type of analysis and does not permit any measure of \( b_x^2 \) directly. This is also true of the other moments discussed here.

As before, another run at the alternate parameters has been done to show the reproducibility of shapes with more structure. The simulation and integration distributions are shown in Fig. 6.13, with the simulation on top. This shape has a sharp rise at the minimum, followed by a rise to a peak, and then a gentle slope toward zero. The simulation reproduces these features well, but has a little difficulty with ringing on the slope. This is most likely an artifact of the Fourier transform approximation mentioned above, in which the function is not a true delta but is of the form \( \sin(a - b)/(a - b) \). Overall, however, the simulation reproduces the line shape well.

6.5.3 \( b_x n(b) \)

To find the moment \( b_x n(b) \), one subtracts equation 6.6 from equation 6.10, which leaves:

\[
< 2 \frac{b_x}{b} \sin \omega t > \tag{6.29}
\]

These data are then divided by two and Fourier sine transformed with each \( b \) as a
Figure 6.12: The top figure is the result for the moment $b_1^2 n(b)$ using the simulation data from this chapter. The bottom plot is the same moment, but arrived at via the integration method developed in last chapter, and is shown to illuminate the validity of the method.
Figure 6.13: The top is the simulation result for $b_{m}^2 n(b)$ for the following parameters: $B = 250$ G, $\lambda = 120$ nm, $\theta = 45^\circ$, and $\Gamma = 25$. The bottom is the corresponding integration result.
CHAPTER 6. OFF-AXIS FIELDS IN ANISOTROPIC SUPERCONDUCTORS

coefficient in the transform. This leaves the moment $b_x n(b)$, which is the distribution of the $x$ component of the microscopic fields within the FLL.

The calculation for this moment is shown in Fig. 6.14, with the simulation result on top and the integration result on the bottom. For this case the parameters are: $B = 100 \, G$, $\lambda = 256.5 \, nm$, $\theta = 45^\circ$, and $\Gamma = 25$. The simulation does a very good job in this case of reproducing the structure of the distribution. It is interesting to note for the theoretical moment that the area under the curve is zero, because the average over the FLL must be zero as stated in the theory. The simulation also does a good job in showing this feature.

The same distribution is again calculated for the alternate parameters of $B = 250 \, G$, $\lambda = 120 \, nm$, $\theta = 45^\circ$, and $\Gamma = 25$. The plots are shown in Fig. 6.15, with the simulation again on top. There is a little more structure here, which is well represented in the simulation. The theoretical area is also zero, as it was above.

6.5.4 $b_y n(b)$

Another moment of interest is $b_y n(b)$, which is the distribution $n(b)$ weighted by the appropriate factor $b_y$ at each $b$. This moment is found by subtracting equation 6.9 from 6.4, dividing this by 2, and then sine Fourier transforming while multiplying by the appropriate $b_y$. The result for this is shown in Fig. 6.16, again accompanied by the integration result.

These two curves are quite similar. The integration result is largely zero, being broken up only by a very narrow up-down spike near 1 $MHz$. This makes sense if one recalls the surface of $b_y$, where there was almost perfect asymmetry within the FLL unit cell. At every point within the FLL cell there existed a $b_y$ value whose opposite in sign could be found symmetrically across the FLL. It is therefore understandable to get a zero at all (most) points because the $b_y$ factor which multiplies in the integral should sum to zero. This moment should integrate to zero via this line of reasoning, and it does.

The simulation result has the same general shape. It starts out looking like the integration result and for most of the curve it is also zero. The up-down spike is
Figure 6.14: Shown on top is the simulation result for $b_x n(b)$ for the following parameters: $B = 100 \, G$, $\lambda = 256.5 \, nm$, $\theta = 45^\circ$, and $\Gamma = 25$. The bottom plot is the corresponding integration method result via the methods of chapter 4.
CHAPTER 6. OFF-AXIS FIELDS IN ANISOTROPIC SUPERCONDUCTORS

Figure 6.15: Shown on top is the simulation result for $b_x n(b)$ for the following parameters: $B = 250 \, G$, $\lambda = 120 \, nm$, $\theta = 45^\circ$, and $\Gamma = 25$. The bottom plot is the corresponding integration method result via the methods of chapter 4.
Figure 6.16: Shown on top is the simulation result for $b_y n(b)$ for the following parameters: $B = 100 \, G$, $\lambda = 256.5 \, nm$, $\theta = 45^\circ$, and $\Gamma = 25$. The bottom plot is the corresponding integration method result via the methods of chapter 4.
CHAPTER 6. OFF-AXIS FIELDS IN ANISOTROPIC SUPERCONDUCTORS

not as pronounced as in the integration result, but it is still there somewhat. As for the integration result, this also should integrate to zero. It is obvious that it does to within the limits of numerical accuracy.

6.5.5 $b_z \, n(b)$

As a final example, the moment $b_z \, n(b)$ is calculated. This moment is found by subtracting equation 6.3 from equation 6.3, dividing by 2, and doing the sine transform while multiplying by each field within the transform – exactly as described above. This then results in the distribution $n(b)$ weighted by the microscopic $z$ component of the fields $b_z$. The simulation and integration results for the $B = 100 \, G$ case are shown in Fig. 6.17. These line shapes are similar to those for $n(b)$, where there is a large peak followed by a long, high-frequency tail.

The comparison data are shown in Fig. 6.18 for the alternate field and parameter values. These look similar to the above $n(b)$ data for the alternate simulation data – Fig. 6.11. Here, as there, is more structure than in the main simulation case, with two peaks followed by a long tail. This time, as in most cases we have seen, the simulation does a very good job of producing the same result as the integration method.

6.5.6 Other Moments

It is possible to find other moments, such as $b_x \, b_z$, $b_x \, b_y$, $b_x^2 + b_y^2$, etc. These can be obtained via the methods discussed above, but will not be discussed because of their limited usefulness.

6.6 The Direction of $B$

Using the techniques of above it is possible to determine the direction of the average internal field $B = \langle b \rangle$ to reasonable accuracy. One first finds the moments $b_x \, n(b)$, $b_y \, n(b)$, and $b_z \, n(b)$ as described above. These moments are now distributions in frequency of the field distribution $n(b)$ weighted by the components $b_x$, $b_y$, and $b_z$. 
Figure 6.17: Shown on top is the simulation result for $b_z \cdot n(b)$ for the following parameters: $B = 100 \text{ G}$, $\lambda = 256.5 \text{ nm}$, $\theta = 45^\circ$, and $\Gamma = 25$. The bottom plot is the corresponding integration method result via the methods of chapter 4.
Figure 6.18: Shown on top is the simulation result for \( b_2 n(b) \) for the following parameters: \( B = 250 \) G, \( \lambda = 120 \) nm, \( \theta = 45^\circ \), and \( \Gamma = 25 \). The bottom plot is the corresponding integration method result via the methods of chapter 4.
as discussed above. If one now takes each of these moments and integrates over frequencies (fields), one has (e.g.):

\[
\int_{b_{\text{min}}}^{b_{\text{max}}} b_x n(b) \, db = \langle b_x \rangle
\]

(6.30)

which is the average value of the \( x \) component of the fields. What this amounts to is integrating out the curve on the top of Fig. 6.14. The same calculation can be done for both \( b_y \) and \( b_z \), integrating over their weighted distributions in Figs. 6.16 and 6.17, respectively. One is then left with \( \langle b_x \rangle \), \( \langle b_y \rangle \), and \( \langle b_z \rangle \), which are the components of \( B \).

Why is this interesting? It is interesting because in general one knows only the direction of the external applied field \( H_a \). This field is related to \( B \) by the following equation:

\[
H_a = B - 4\pi M
\]

(6.31)

where \( M \) is the magnetization of the material, and the equation is expressed in gaussian units [81]. It is therefore quite possible that if \( M \) is appreciable the applied field \( H_a \) will not be parallel to the average internal field \( B \).

The theory developed in chapters 3 and 4 assumes one has a knowledge of the average field \( B \) in both magnitude and direction. The development in this chapter is built upon this theory and, therefore, this assumption. It is generally assumed in most theories and experiments that the applied and internal average fields are parallel. This method should now allow for an experimental check on the validity of this assumption.

Prior to discussing the results of this analysis, a brief discussion of some important points of the analysis will be presented. First, one must be sure to create a good set of simulation curves for the three, mutually perpendicular directions. The experimental analog is to make sure that all detectors are aligned and calibrated, that the muon initial polarization direction is reasonably well known, and that high statistics data are taken. Second, one must Fourier transform these data over the proper range of frequencies, keeping in mind that the lowest and highest frequencies possible to see are: \( f_{\text{min}} = 1/T \), and \( f_{\text{max}} = 1/2\Delta t \) (the Nyquist limit). Here \( T \) is the overall
CHAPTER 6. OFF-AXIS FIELDS IN ANISOTROPIC SUPERCONDUCTORS

<table>
<thead>
<tr>
<th>Initial Data (G)</th>
<th>Simulation Results (G)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$b_x$</td>
<td>$b_y$</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>170.434</td>
<td>170.434</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 6.1: These are the results of applying the moment method to simple field distributions directed along $z$, $45^\circ$ to each axis, and again along $z$, for $B = 295.2 \, G$. The left side shows the components within the FLL, and the right shows the averaged values resulting from the analysis.

It is apparent from this table that the method works rather well for these simple cases.

Next we apply the method to the reed simulation data, like that of Fig. 6.8. We also apply it to the alternate data set, with $B = 250 \, G$, $\lambda = 256.5 \, nm$, $\theta = 45^\circ$, and $\Gamma = 25$. Also, for one extra comparison, we apply the method to a situation where $B = 1000 \, G$, $\lambda = 256.5 \, nm$, $\theta = 45^\circ$, and $\Gamma = 25$. The results are shown in Table 6.2.

These results are not as good as the simple cases, but that is to be expected due to the much more complicated magnetic field structure within the FLL. Most of the
Table 6.2: These are the results of applying the moment method to the different simulation data. The top results are for the $B = 100 \, G$, $\lambda = 256.5 \, nm$, $\theta = 45^\circ$, and $\Gamma = 25$ case. The middle results are for the alternate simulation data: $B = 250 \, G$, $\lambda = 120 \, nm$, $\theta = 45^\circ$, and $\Gamma = 25$. The last data are for $B = 1000 \, G$, $\lambda = 256.5 \, nm$, $\theta = 45^\circ$, and $\Gamma = 25$.

The results are close to the proper values, and fall within about $\pm 5\%$. Therefore if the actual $B$ is off from the applied field $H_a$ by more than this amount, we should be able to see it (assuming that we can get good $\mu$SR data with high statistics). Based on these results, it does not seem possible to detect any smaller deviation of $B$ from $H_a$ than this amount.

A proposal to the National Science Foundation, based largely on the work in this thesis, has (at the time of this writing) just been funded. It will now be possible for the experimental equivalent of the simulations to be obtained. The same computational machinery used here will then be applied to this data in hopes of obtaining valuable results.
Chapter 7

Conclusions

This thesis set out to study the internal magnetic fields of the high temperature, anisotropic superconductors, particularly the compound \(YBa_2Cu_3O_7\). Most of these new materials have \(T_c\) values (critical temperature – the onset of superconductivity) which are above the boiling point of liquid nitrogen. This fact has made their potential use very attractive due to the relative cheapness of liquid nitrogen compared to other coolants and refrigeration techniques. These newer superconductors, discovered in the late 1980's, have been some of the most intensly studied materials of all time. However, there is still much to be learned about them, especially at lower applied fields where many theoretical and experimental assumptions break down. This work has therefore been an attempt to better understand these materials from a theoretical point of view, and to provide a framework and the tools for an experiment to complement these studies.

To this end, the thesis first described the \(\mu\)SR techniques and how muons behave in the presence of a magnetic field. The data obtained with these techniques, and how it can be used to show the microscopic internal magnetic fields seen by the muons, was discussed in some detail.

Next, the isotropic and anisotropic London theories, which allow theoretical calculation of the microscopic magnetic fields within the mixed state of these superconductors, were discussed at length. Then, using the anisotropic London theory, a computer program was developed which calculates the microscopic magnetic fields at
any point within the equilibrium (triangular) flux line lattice. The program accepted user-specified parameters for values of the average field $B$ magnitude, the effective penetration depth $\lambda$, the angle $\theta$ defining angle between the crystal c axis and the average field $B$ direction, and the anisotropy parameter $\Gamma$. The field components, including those which are transverse (or off-axis) to the average field $B$ direction, were then calculated on a grid which subdivided the unit flux line lattice (FLL) unit cell.

Using these field values, the theoretical distributions of the magnetic fields were calculated. Distributions were calculated for all components of the fields, as well as for the magnitude of the field $b$. A study was then undertaken to determine the dependence of the fields and the distributions on the various parameters of the program, such as the average field $B$, the effective penetration depth $\lambda$, the anisotropy parameter $\Gamma$, and the angle $\theta$ between the crystal c axis and the average field $B$. The $1/\lambda^2$ dependence of the width of the field distributions was shown to hold, as well as the interesting dependence of the shape of the distribution on the magnitude of the average field. These results, and others, are discussed in chapter 5.

The last part of the thesis combined the knowledge gained from the study of muons as well as the results of the theoretical field data. That is, knowing the microscopic magnetic field within the FLL allowed a simulation of the muon behavior within the unit cell. A program was written which simulates the behavior of muons stopping uniformly within the FLL, calculating their polarization projection onto three mutually perpendicular "detectors." This simulation was done in an attempt to probe the microscopic magnetic field components which are transverse (or, off-axis) to the direction of the average magnetic field $B$. These transverse fields arise naturally from the theory when the anisotropy is taken into account, and are not due to pinning or other mechanisms. The simulation results show a depolarization of the muon spins for all three directions – meaning that there is field inhomogeneity within the FLL – and proving (at least at some level) that these "off axis" fields exist. (Incidentally, to the best of this author's knowledge, the simple experimental analog has yet to be done.)
The final endeavor of this work was to use the results of the simulations to extract further information about the magnetic fields within the FLL. A technique was then developed to take the simulation data, manipulate it using both simple math and Fourier analysis, and extract information which we call moments of the field distributions. One useful moment, $n(\delta)$, is the distribution of the magnitude of the microscopic magnetic field $b$ within the FLL. The only approximation in this method is that of the upper limit of a Fourier integral being finite and not infinite, which introduces an approximate delta function as opposed to a real one. This approximation, however, is common to all numerical Fourier transform techniques, and should still yield good, quantitative results. Another useful result of the analysis is that the direction of the average internal magnetic field $B$ may be determined to good accuracy. This is useful in that its direction may in general be different than that of the applied field $H_a$.

The types of superconductors to which this theory naturally applies — single crystals of high quality $YBa_2Cu_3O_7$ — are now being produced with little twinning and defects. It is in these materials where one hopes to best apply the results of the London theory, since they most closely approximate the theoretical assumptions. Some day the experimental analog of this theoretical study will be pursued in hopes of shedding more light onto the still confusing data and theories which surround these materials. This may be stated with confidence because, at the time of this writing, a grant based largely on the work of this thesis has been funded for these experiments.
Appendix A

Field Calculation Program

This appendix contains the program hsumndb.f. It is a FORTRAN program which is an implementation of the ideas developed in chapter 3 concerning the anisotropic London theory. The program uses the prescription developed there and calculates the \( X, Y, \) and \( Z \) components of the microscopic magnetic field on a grid which is \( 51 \times 51 \) units on a side. The user is queried for the strength of the average magnetic field \( B \), the angle \( \theta \) between \( B \) and the crystal \( c \) axis, the value of the effective penetration depth \( \lambda \), and the anisotropy parameter \( \Gamma \). Once these values are input, the program calculates the field components on the grid. Once the field values are known on the grid, a subroutine called \texttt{integ} is called which does the integration of the following delta function:

\[
\delta (b_0 - b(x, y)) \quad (A.1)
\]

over the area of the FLL. This produces the kinds of field distributions shown in chapter 4.

The subroutine \texttt{integ} can be modified to calculate the moments described in chapter 5. One simply does the integration of above, but weights each contribution from each sub-cell of the grid by the corresponding coefficient of \( n(b) \) for the particular calculation at hand. This type of modification is shown below within the \texttt{integ} subroutine in comment lines as, \textit{e.g.}, the variable \texttt{otherc}. 

141
program hsum
   c...The main program -- sets up loops and calculates (x,y) points at 
   c...which the fields hsumx, hsumy, and hsumz will be calculated. The 
   c...points (x,y) lie in a region near the lower left flux quantum of 
   c...an isosceles flux triangle. Rejuvenated 1/14/93 to calculate the 
   c...distribution dn/db for the magnitude of b. Again 2/3/93 to calculate 
   c...bperp2/b2*dndb lineshape. Again 5/28/93 for bx dn/db & by dn/db.
   parameter(IERD=60, PI=3.141592654, LEN=1000, GAMMA=13.56e-3)
   character resp*1, gfil*20, rfil*20, resp1*1
   dimension hsumx(0:IERD,0:IERD), hsumy(0:IERD,0:IERD), 
   1 hsumz(0:IERD,0:IERD), hsumag(0:IERD,0:IERD), field(LEN), 
   2 fldnov(LEN)
   data B/l. 8e3/, phiO/2.07e-7/, am3/8.56/, ami/.342/, alambda/. 256Se-4/, 
   1 theta/0./
   areal=0.
   do i=0, IERD
      do j=0, IERD
         hsumx(i,j)=0.
         hsumy(i,j)=0.
         hsumz(i,j)=0.
      enddo
   enddo
   factor=1.
   c...Get the parameters and the basic g's.
   call gcalc(gx,gy,B,zzm,amxz,amxx,alamda,ami,am3,bix, 
   1 b2y,IERD,phiO,theta)
   c...Calculate angle between basis vectors.
   argy=sqrt(3.*am3/zzm)
   alpha=atan(argy)
   c...See if grid exists; if so read in.
   print*, 'Should the grid be calculated? (y/n)' 
   read(5,'(a1)')resp
   bx=bix/real(IERD)
   dy=b2y/real(IERD)
   if(resp .eq. 'n')then
      2 print*, 'Input file name containing the grid info.' 
      read(5, '(a20)', err=2)gfil
open(9, file=glil, status='old', err=999)
do 5 inc=0,IEHD
do 5 jnc=0,IEHD
read(9,*)hsumx(inc, jnc),hsumy(inc, jnc),hsumz(inc, jnc)
5 continue
close(9)
goto 99
endif
do 10 i=0,IEHD
do 20 j=0,IEHD
x=dx*i
y=dy*j
c...Sum G on this x,y point.
call gsum(x,y,i,j,gx,gy,B,zmm,amxz,amxx,alamda,am1,am3,IEHD,
1 hsumx,hsumy,hsumz)
20 continue
10 continue
99 continue
c...Now calculate the magnitude of b at all points.
do i=0,IEHD
do j=0,IEHD
  hsumag(i,j)=sqrt(hsumx(i,j)**2 + hsumy(i,j)**2 + hsumz(i,j)**2)
enddo
enddo
75 print*, 'Save hsums? (y/n)'
read(6,'(a1)')resp
if(resp .eq. 'y')then
  print*, 'Input name for output file: '
  read(5,'(a20)', err=75) gfil
  open(9, file=gfil, err=998)
  print*, 'Output in file ', gfil
  do i=0,IEHD
do j=0,IEHD
    write(9,*)hsumx(i,j),hsumy(i,j),hsumz(i,j)
  enddo
enddo
APPENDIX A. FIELD CALCULATION PROGRAM

```fortran
close(9)
endif
print*, 'Integ or quit? (i/q)
read(5, '(a1)') resp
if (resp .eq. 'q') goto 1000
call integ(hsumag, hsumz, IEND, dx, dy, fldnow, field, LEN, theta, nflds)
c...Need to normalize the lineshape; convert B into freq.
do 236 k=1, nflds
   field(k) = field(k) * GAMMA
236 areal = areal + fldnow(k)
print*, 'Areal= ', areal
if (areal .eq. 0.) areal = 1.
do 236 k=1, nflds
   fldnow(k) = fldnow(k) / areal
236 c...BEEP so user knows is finished.
   print*, char(7)
c...Save line shape stuff in a file.
72 print*, 'Input file name for field data: '
   read(5, '(a20)', err=72) rfil
   open(9, file=rfil, err=997)
do 65 i=1, nflds
   if ((i .gt. 12).and. (fldnow(i) .eq. 0.)) goto 67
      write(9, *) field(i), fldnow(i)
65 continue
67 close(9)
goto 1000
999 print*, 'Error opening file ', gfil, ' check spelling.'
goto 2
998 print*, 'Error opening file ', gfil, ' check spelling.'
goto 75
997 print*, 'Error opening file ', rfil
1000 end
```

C..........................C

```fortran
subroutine gcalc(gx, gy, B, zzm, amxx, amxz, alamda, ami, am3, bix,
1 b2y, IEND, phi0, theta)
c...This subroutine calculates the components of the vector G and returns them
```
APPENDIX A. FIELD CALCULATION PROGRAM

...to the caller in gx and gy.

parameter(PI=3.1415926)
parameter(GAMMA=25.)
character*1 resp

...Use the accepted values for the parameters. B is in Gauss; Phi0 is in
G*cm*cm; lambda is in cm; theta in radians (duh).

print*,'B , lambda, theta, gamma= ',B,alamda,theta,GAMMA
print*,'Input B (G)' '
read*,B
print*,'Input lambda (cm)' '
read*,alamda
print*,'input theta (deg)'
read*,theta
theta=theta*PI/180.

aml=GAMHA**(-l./3.)
am3=GAMMA**(2./3.)
7 amxx=aml*cos(theta)**2+am3*sin(theta)**2
zzm=aml*sin(theta)**2+am3*cos(theta)**2
amxz=(aml-am3)*sin(theta)*cos(theta)

bstuf=sqrt(2.*B/phiO)
shit=3.*am3/zzm
amstuf=shit**.25

...These are the basis real-space x and y lengths in cm.
blx=sqrt(2.*PHI0/B)*(l./shit)**.25
b2y=(blx/2.)*sqrt(3.*am3/zzm)
gx=PI*bstuf*amstuf
gy=PI*bstuf/amstuf
return
end

subroutine gsun(x,y,i,j,gx,gy,B,zzm,amxx,amxz,alamda, 
1 am1,am3,IEHD,hsumx,hsumy,hsumz)
...This subroutine does the sum over the recip. latt. vectors { G }.
...The fundamental values of these vectors are found in gcalc.
parameter(IGSTOP=50)
dimension hsumx(0:IEHD,0:IEHD),hsumy(0:IEHD,0:IEHD),
hsumz(0:IEHD,0:IEHD)
c character*1 ans
c logical debug
c debug=.false.
c...Check if is for debugging.
c print*, 'Debug? '
c read(5,'(a1)')ans
c if(ans .eq. 'y') debug=.true.
c if(debug) then
c print 62, x,y,i,j,gx,gy,B,zzm,amxz,amxx,alamda,ami,am3,IEHD
c52 format(' In gsum: x,y,i,j= ',2el2.S,2i2,/,' Gx,Gy,B= ',2el2.5,
1 fl2.6,/,' Mzz,Mxz,Mxx= ',3el2.6,/,' Lamda,M1,M3= ',3e12.5,/,c 2 ' IEHD= ',i2,/)c endif
c...Set up loops for sum.
front=B*alamda**2
c if(debug) print*, 'front= ',front
alamdas=alamda**2
f1=am1*alamdas
f2=amxx*alamdas
f3=zzm*alamdas
f4=amxz**2*alamdas
13=zzm*alamdas
do 10 n=-IGSTOP,IGSTOP
do 20 m=-IGSTOP,IGSTOP
gxx=gx*n
gyy=gyy*(2.*m-n)
gyy=gyy**2
gxx=gxx**2
c...Define quant.'s to optimize code.
aglob=f1*gxxs
bglob=f2*gyy
dglob=f4*gyy
13=zzm*alamdas
f4=amxz**2*alamdas
10 continue
do 20 continue
c if(debug) then
write(6,50)aglob,bglob,f3,dglob
APPENDIX A. FIELD CALCULATION PROGRAM

c50  format(' aglob= ',e12.5, ' bglob= ',e12.5, ' cglob= ',e12.5,
c 1 ' dglob= ',e12.5, '/)
c  endif
  g2=gxxx + gyy
  d=(1+ aglob + bglob)*(1+g3*g2) - dglob*g2
  hgx=front*xmx*gyy/d
  hgy=-front*xmx*gxx*gyy/d
  hgz=B/d+front*zzm*g2 /d
  if(debug) then
    write(6,51)g2,d,hgx,hgy,hgz
  c51  format(' g2,d= ',2e12.5, ' hgx,hgy,hgz= ',3e12.5, '/)
  endif
  gdotr=gxx*x + gyy*y
  deltax=hgx*cos(gdotr)
  deltay=hgy*cos(gdotr)
  deltaxshgz*cos(gdotr)
  if(debug)print*, ' deltax,y,z- ',deltax,deltay,deltaz
  hsumx(i, j)=hsumx(i, j)+deltax
  hsumy(i, j)=hsumy(i, j)+deltay
  hsumz(i, j)=hsumz(i, j)+deltaz
  20 continue
  10 continue
  return
end

SUBROUTINE INTEGR(hsum, hsumz, INDEX, DX, DY, FLDNOW, FIELD, LEN, THETA,
i  NFRDS)
c...This routine will run through field values and sum for the line-shape
...at each one. Field values are assumed to vary linearly between successive
...points on the grid, and also across diagonals. Each contribution to the
...sum is calculated by dividing the length of the field line through half
...of the rectangle by the gradient. Updated 2/93 for the calculation of
c...dn/db<>bperp^2/2^2> -- hence the inclusion of the hsumz grid to calc
c...bperp. Again 5/27/93 for bx dn/db & by dn/db.
c...This routine will run through field values and sum for the line-shape

c...This routine will run through field values and sum for the line-shape

dimension hsum(0:INDEX,0:INDEX),FLDNOW(LEN),FIELD(LEN),

APPENDIX A. FIELD CALCULATION PROGRAM

147
1. hsumz(0:INDEX,0:INDEX)
   character ans*1
   dstot=sqrt(dx**2+dy**2)
   print*, 'dx,dy= ',dx,dy
   c..Find m u  and min field values.
   do 4 i=0,INDEX
   do 4 j=0,INDEX
   if((i.eq.0).and.(j.eq.0))then
      fmax=hsum(0,0)
      fmin=hsum(0,0)
   endif
   if(abs(hsum(i,j)) .lt. fmin) fmin=hsum(i,j)
   if(abs(hsum(i,j)) .gt. fmax) fmax=hsum(i,j)
   4 continue
   c...Need to automatically choose the limits for the lineshape calculation.
   c...Fstart is easy, one below its value is sufficient to give a first zero.
   c...To find the end value, typical distributions were analyzed and it was
   c...found that values from fmin up to a certain % of the range of fields held
   c...all of the necessary information. The percentage increases as the angle
   c...increases.
   print*, 'In integ, fmin & fmax= ',fmin,fmax
   itheta=nint(theta/1.571*19.)
   fstart=fmin-1.
   c deltaf=fmax-fmin
   c percnt=(2.*itheta)/100. + .15
   c fend=fmin + deltaf*percnt
   c df=(fend-fstart)/real(LEN)
   print*, 'Min and max field values are ',fmin,fmax
6  print*, 'Input the starting and ending values for the integ: ' 
   read(5,*,err=6)fstart,fend
8  print*, 'Input the number of divisions(< ',LEN,': ' 
   read(5,*,err=8)nflds
   if(nflds .gt. LEN)nflds=LEN
   df=(fend-fstart)/real(nflds)
   print*, 'start,end,df= ',fstart,fend,df
   c...Loop over field values.
   do 10 ifield=1,nflds
field(ifield)=fstart + (ifield-1)*df

c...Loop over the grid.
do 20 j=0,IHDEX-1
do 30 i=0,IHDEX-1
xd=-1.
yd=-1.
xe=-1.
ye=1.
c...Check if the field is in the lower right triangle.
if((field(ifield) .lt. amaxl(hsum(i,j),hsum(i+1,j),
1  hsum(i+1,j+1))).and.(field(ifield) .ge. aminl(hsum(i,j),
2  hsum(i+1,j),hsum(i+1,j+1)))then

c ... Calculate the gradient for the lower right triangle.
dbdx=(hsum(i+1,j)-hsum(i, j))/dx
    dbdy=(hsum(i+1,j+1)-hsum(i+1,j))/dy
    gradb=sqrt(dbdx**2+dbdy**2)
    print*, 'dbdx, dbdy, gradb= ', dbdx, dbdy, gradb
c...Calc these for other contrib.
    dbdxz=(hsumz(i+1, j)-hsumz(i,j))/dx
    dbdyz=(hsumz(i+1, j+1)-hsumz(i+1,j))/dy
    dbdxyz=(hsumz(i+1, j+1)-hsumz(i, j))/dstot
    c..  .Find the intersection points on the loser right triangle,
c...x-axis bottom.
    if((field(ifield) .lt. amaxl(hsum(i,j),hsum(i+1,j))).and.
1 (field(ifield) .ge. aminl(hsum(i,j),hsum(i+1,j)))) then
    xe=(field(ifield)-hsum(i,j))/dbdx
endif
c...y-axis right.
    if((field(ifield) .lt. amaxl(hsum(i+1,j),hsum(i+1,j+1))).and.
1 (field(ifield) .ge. aminl(hsum(i+1,j),hsum(i+1,j+1)))) then
    ye=(field(ifield)-hsum(i+1,j))/dbdy
endif
c...diagonal up-right to low-left.
    if((field(ifield) .lt. amaxl(hsum(i,j),hsum(i+1,j+1))).and.
1 (field(ifield) .ge. aminl(hsum(i,j),hsum(i+1,j+1)))then
dbdds=(hsum(i+1,j+1)-hsum(i, j))/dstot
deltas=(field(ifield)-hsum(i,j))/dbdds
APPENDIX A. FIELD CALCULATION PROGRAM

fract = deltas/dstot
xd = fract*dx
yd = fract*dy
endif

c...Only two of the above three should be satisfied.
    if((xe .gt. 0.).and.(ye .gt. 0.).and.(xd .gt. 0.)) then
        print*, 'Problem at i, j = ', i, j, ' lower right triangle.'
        goto 40
    endif
    if(xe .lt. 0.) then
        dlen = sqrt((dx-xe)**2 + (yd-ye)**2)
        bperp2 = ((hsum(i+1,j)+dbdy*ye)**2-(hsumz(i+1,j)+dbdyz*ye)**2 +
                   (hsum(i,j)+dbdx*xe)**2-(hsumz(i,j)+dbdxz*xe)**2)/2.
        otherc = (hsumz(i+1,j)+dbdyz*ye+hsumz(i,j)+dbdxz*xe)/2.
        print*, 'bp2xe = ', bperp2
    elseif(ye .lt. 0.) then
        dlen = sqrt((xd-xe)**2 + ye**2)
        bperp2 = ((hsum(i,j)+dbdx*xe)**2-(hsumz(i,j)+dbdxz*xe)**2 +
                   (hsum(i+1,j)+dbdy*ye)**2-(hsumz(i+1,j)+dbdyz*ye)**2)/2.
        otherc = (hsumz(i+1,j)+dbdyz*ye+hsumz(i,j)+dbdxz*xe)/2.
        print*, 'hp2ye = ', bperp2
    else
        dlen = sqrt((dx-xe)**2 + ye**2)
        bperp2 = ((hsum(i+1,j)+dbdy*ye)**2-(hsumz(i+1,j)+dbdyz*ye)**2 +
                   (hsum(i,j)+dbdx*xe)**2-(hsumz(i,j)+dbdxz*xe)**2)/2.
        otherc = (hsumz(i+1,j)+dbdyz*ye+hsumz(i,j)+dbdxz*xe)/2.
        print*, 'bp2diag = ', bperp2
    endif

c...Calc the contrib. to the sum.
    cont = dlen/gradb
    cont = dlen/gradb*bperp2
    cont = dlen/gradb*otherc
    fldnow(ifield) = fldnow(ifield) + cont
    print*, 'fldnow, cont = ', fldnow(ifield), cont
APPENDIX A. FIELD CALCULATION PROGRAM

40 xe=-l.
ye=-l.
c...Keep xd and yd in case are needed below.
endif

c...Now for the upper left triangle.
    if ((field(ifield) .lt. amaxl(hsum(i,j),hsum(i,j+1),
        1 hsum(i+1,j+1))).and.(field(ifield) .ge. aminl(hsum(i,j),
        2 hsum(i,j+1),hsum(i+1,j+1))) then
        c...Calc gradient up here.
        dbdx=(hsum(i+1,j+1)-hsum(i,j+1))/dx
        dbdy=(hsum(i,j+1)-hsum(i,j))/dy
        gradb=sqrt(dbdx**2 + dbdy**2)
        c print*, 'dbdx2, dbdy2, gradb2= ',dbdx,dbdy,gradb
        c...Calc these for other cont.
        dbdxz=(hsumz(i+1,j+1)-hsumz(i, j+1))/dx
        dbdyz=(hsumz(i,j+1)-hsumz(i, j))/dy
        dbdxyz=(hsumz(i+1, j+1)-hsumz(i, j))/dstot
        c...Top.
        if ((field(ifield) .lt. amaxl(hsum(i,j+1),hsum(i+1,j+1))).and.
        1 (field(ifield) .ge. aminl(hsum(i,j+1),hsum(i+1,j+1))) then
            xe=(field(ifield)-hsum(i,j+1))/dbdx
        endif
        c...Left side.
        if ((field(ifield) .lt. amaxl(hsum(i,j),hsum(i,j+1))).and.
        1 (field(ifield) .ge. aminl(hsum(i,j),hsum(i,j+1))) then
            ye=(field(ifield)-hsum(i,j))/dbdy
        endif
        c...Diag. done above: if there is a crossing, xd and yd are calculated.
        if((xe .gt. 0.).and.(ye .gt. 0.).and.(xd .gt. 0.))then
            print*, 'Problem at i,j= ',i,j,' upper left triangle.'
            goto 30
        endif
        if(xe .lt. 0.)then
            dlen=sqrt(xd**2 + (yd-ye)**2)
            bperp2=((hsum(i,j)+dbdy*ye)**2-(hsumz(i,j)+dbdyz*ye)**2 +
                1 (hsum(i,j)+dbdxyz*delta)**2-(hsumz(i,j)+dbdxyzz*delta)**2)/2.
            c otherc=(hsumz(i,j)+dbdxyz*ye+hsumz(i,j)+dbdxyzz*delta)/2.
c print*, 'bp2xe2= ',bperp2
elseif(ye .lt. 0.)then
dlen=sqrt((xd-xe)**2 + (dy-ye)**2)
bperp2=((hsum(i,j+1)+dbdx*xe)**2-(hsumz(i,j+1)+dbdxz*xe)**2 +
1 (hsum(i,j)+dbdxs+deltas)**2-(hsumz(i,j)+dbdzs+deltas)**2)/2.
c otherc=(hsumz(i,j+1)+dbdxz*xe + hsumz(i,j)+dbdxz*deltas)/2.
c print*, 'bp2ye2= ',bperp2
else
dlen=sqrt(xe**2 + (dy-ye)**2)
bperp2=((hsum(i,j+1)+dbdx*xe)**2-(hsumz(i,j+1)+dbdxz*xe)**2 +
1 (hsum(i,j)+dbdy*ye)**2-(hsumz(i,j)+dbdyz*ye)**2)/2.
c otherc=(hsumz(i,j+1)+dbdxz*xe + hsumz(i,j)+dbdyz*ye)/2.
c print*, 'bp2diag2= ',bperp2
endif
c...Calc contrib. for this.
c        cont=dlen/gradb
        cont=dlen/gradb*bperp2
c        cont=dlen/gradb*otherc
c        print*, '2, fldnow, cont= ',fldnow(ifield),cont
        fldnow(ifield)=fldnow(ifield)+cont
        endif
30 continue
20 continue
10 continue
return
end
Appendix B

Simulation Calculation Program

The following FORTRAN program calculates the simulation of the muons stopping uniformly within the FLL. It has the capability to calculate the fields, if necessary, by borrowing the subroutines gcalf.f and gsum.f from the last appendix. It will also read in a previously calculated grid, if the user desires. The program accepts as input the polar and azimuthal angles of B, c, and the initial polarization P(0) of the muons. From these it proceeds to calculate the projection of the polarization onto the three LAB directions x, y, and z. The contribution of the present field value at the muon is weighted as in the integration routine integ of the last appendix and discussed in chapter 4. That is, the fractional amount of area that the muon’s field occupies is calculated and multiplies (scales) the polarization data for that muon. In this way the entire grid of field values can be properly taken into account and a uniform muon stopping distribution throughout the FLL is insured.

program goft
   c...This program calculates the grid and then calculates the relaxation
   c...function G(t) from it along x, y, and z directions averaging via
   c...Kossler's suggestion. Also, rotations are included to allow for
   c...a lab frame set of axes that are not coincident with a crystal frame.
   c...In this version the detector axes are defined as x, y, and z. One inputs
   c...the angles (polar and x-y planar) to define the directions of Spin, B, and
   c...the crystal c-axis. The angle theta (B&c) is then calculated and told to
   c...the user and the user is allowed to reset c for another theta before the

153
c...actual calculation begins.

parameter (IEHD=50, PI=3.141592654, LEN=4000, GAMMA=13.55342e-3)
parameter (TRANGE=10.)
character resp*, fil*20, fil2*20
dimension hsumx(0:IEHD, 0:IEBD), hsumy(0:IEHD, 0:IEMD),
1 hsumz(0:IEHD, 0:IEID), time(0:LEN), sigxm(0:LEI),
2 sigym(0:LEI), sigzm(0:LEI), tmat(3, 3), rotb(3, 3), htor(3, 3)
dimension sig1(0:LEN), sigy1(0:LEN), sigz1(0:LEN), sigsp(0:LEN),
1 sig0(0:LEN), sigy0(0:LEN), siga0(0:LEN), sigza0(0:LEN),
2 bsum(0:IEKD, 0:IEHD)
dimension polxt(0:LEN), polyt(0:LEN), polzt(0:LEN)

alamda=1.2e-5
theta=45.
B=250.
contot=0.
314 print 500, B, alamda
print*, 'Input your values: '
read*, B, alamda
print 500, B, alamda

500 format(/, 'Field and lambda are: ', f8.2, e10.3, /)

print*, 'Input the angles alpha and phi-s for the alignment'
print*, 'of spin in degrees: '
read*, alpha, phis
print*, 'The angles are ', alpha, phis
alpha=alpha*PI/180.
phis=phis*PI/180.

print*, 'Input the angles beta and phi-b for the alignment'
print*, 'of B in degrees: '
read*, beta, phib
print*, 'The angles were: ', beta, phib
beta=beta*PI/180.
phib=phib*PI/180.
APPENDIX B. SIMULATION CALCULATION PROGRAM

777  print*, 'Input the angles nu and phi-c for the alignment'
   print*, 'of c in degrees:'
   read*, enu, phic
   print*, 'The angles were:', enu, phic
   enu=enu*PI/180.
   phic=phic*PI/180.

   c...Calculate the angle theta -- NOTE B comps are unit-wise; not need full val.
   Bx=sin(beta)*cos(phib)
   By=sin(beta)*sin(phib)
   Bz=cos(beta)
   cx=sin(enu)*cos(phic)
   cy=sin(enu)*sin(phic)
   cz=cos(enu)
   px=sin(alpha)*cos(phis)
   py=sin(alpha)*sin(phis)
   pz=cos(alpha)
   pz=zchk(pz)
   px=zchk(px)
   py=zchk(py)

   print*, 'Bx,y,z= ', Bx, By, Bz
   print*, 'cx,y,z= ', cx, cy, cz
   print*, 'px,py,pz= ', px, py, pz

   Bdotc=Bx*cx+By*cy+Bz*cz
   theta=acos(Bdotc)
   thetap=theta*180./PI
   print*, 'Theta is ', thetap
   print*, 'Re-align c? (y/n)'
   read(5, '(a1)') resp
   if(resp .eq. 'y') goto 777

   c...Calculate the crystal x, y, and z axis components wrt the lab axes. This
c...is needed below for the transformation matrix. Y first -- yBz is the B
c...frame y's z-component in the lab frame, etc. Careful with the sin(theta)
c...if theta=0 -- see sinthe.
APPENDIX B. SIMULATION CALCULATION PROGRAM

\[ \sin \theta = \sin(\theta) \]
\[ \text{if} (\text{abs}(\sin \theta) \lt 1.0 \cdot 10^{-3}) \sin \theta = 1. \]
\[ y_{Bx} = \text{zchk}((By \cdot cz - cy \cdot Bz)/\sin \theta) \]
\[ y_{By} = \text{zchk}((Bz \cdot cx - cz \cdot Bx)/\sin \theta) \]
\[ y_{Bz} = \text{zchk}((Bx \cdot cy - cx \cdot By)/\sin \theta) \]
\[ \text{print}, 'y_{Bx}, y_{By}, y_{Bz} = ', y_{Bx}, y_{By}, y_{Bz} \]

\[ c...X \text{ next.} \]
\[ x_{Bx} = \text{zchk}(Bz \cdot y_{By} - y_{Bz} \cdot By) \]
\[ x_{By} = \text{zchk}(Bx \cdot y_{Bz} - y_{Bx} \cdot Bz) \]
\[ x_{Bz} = \text{zchk}(By \cdot y_{Bx} - y_{By} \cdot Bx) \]
\[ \text{print}, 'x_{Bx}, y_{By}, z_{Bz} = ', x_{Bx}, y_{By}, z_{Bz} \]

\[ c...Z \text{ next.} \]
\[ z_{Bx} = \text{zchk}(Bx) \]
\[ z_{By} = \text{zchk}(By) \]
\[ z_{Bz} = \text{zchk}(Bz) \]
\[ \text{print}, 'z_{Bx}, z_{By}, z_{Bz} = ', z_{Bx}, z_{By}, z_{Bz} \]

area=0.
\[ g_x = 0. \]
\[ g_y = 0. \]
\[ zzm = 0. \]
\[ amxz = 0. \]
\[ amyx = 0. \]
\[ bix = 0. \]
\[ b_{2y} = 0. \]

\[ \text{print}, 'Read in grid? (y/n)' \]
\[ \text{read}(5,'(a1)')\text{resp} \]
\[ \text{if}(\text{resp} \cdot \text{eq.} 'y') \text{then} \]
\[ \text{print}, 'Input grid file name: ' \]
\[ \text{read}(5,'(a20)')\text{fil} \]
\[ \text{open}(9,\text{file}=\text{fil}) \]
\[ \text{do } m=0,\text{END} \]
\[ \text{do } n=0,\text{END} \]
\[ \text{read}(9,*)\text{hsux}(m,n),\text{hsuy}(m,n),\text{hsuz}(m,n) \]
c...Get the parameters and the basic g's.
call gcalc(gx,gy,B,zzm,amxz,amxx,alamda,ami,am3,bix,
   b2y,IEHD,theta)

print*,'Input output file name: '
read(S,'(a20)')fil2

dx=bix/real(IEHD)
dy=b2y/real(IEHD)
area=dx*dy
print*,'dx,dy= ',dx,dy

if(resp .eq. 'y')goto 444

!...Calculate grid.
do 10 i=0,IEHD
do 20 js0,IEHD
   c...Need to reinitialize each element to zero before calculating.
hsumx(i,j)=0.
hsumy(i,j)=0.
hsumz(i,j)=0.

   x=dx*i
   y=dy*j

   c...Sum G on this x,y point.
call gsum(x,y,i,j,gx,gy,B,zzm,amxz,amxx,alamda,ami,am3,IEHD,
      hsumx,hsumy,hsumz)
celly=celly+dble(dx*dy)

20 continue
APPENDIX B. SIMULATION CALCULATION PROGRAM

10 continue

open(9,file='gridxyz')
do m=0,IEHD
do n=0,IEHD
write(9,*)hsumx(m,n),hsumy(m,n),hsumz(m,n)
enddo
enddo
close(9)

c...Now have all components of the field. Need to calculate the transformation
c...matrix from the B frame to the lab frame using the components of B and c

c...obtained above.
444 tmat(1,1)=xBx
tmat(1,2)=yBx
tmat(1,3)=zBx
tmat(2,1)=xBy
tmat(2,2)=yBy
tmat(2,3)=zBy
tmat(3,1)=xBz
tmat(3,2)=yBz
tmat(3,3)=zBz

c...Calculate the b grid.
do i=0,IEHD
do j=0,IEHD
bsum(i,j)=sqrt(hsumx(i,j)**2+hsumy(i,j)**2+hsumz(i,j)**2)
enddo
enddo

c...Calculate sigma in the time loop.
  pfreq=CAANMA*B
  period=1./pfreq
tdel=period/12.
tstps=nint(TRANSB/tdel)
print*,'ntstps=',ntstps
if(ntstps.gt.LEN)then
APPENDIX B. SIMULATION CALCULATION PROGRAM

ntstps=LEN

print*, 'Too many time steps, had to truncate.'

endif

deltat=.2 Se-2

do i=0, LEN
    time(i)=real(i)*deltat
endo

c...The components of the spin in the LAB frame.
   sx1=px
   syl=py
   sz1=pz

c...Now implement the kossler weighting idea for the Pol funs.

c...This is borrowed from the integration routine.

c subroutine integ(hsum,INDEX,dx,dy,fldnow,field,LEN,itheta)

c...This routine will run through field values and sum for the lineshape at
c...each one. Field values are assumed to vary linearly between successive
c...points on the grid, and also across diagonals. Each contribution to the
c...sum is calculated by dividing the length of the field line through half
c...of the rectangle by the gradient.

c dimension hsum(0:INDEX,0:INDEX),fldnow(LEN),field(LEN)

dstot=sqrt(dx**2+dy**2)

c...Find max and min field values.
   fmax=bsum(0,0)
   fmin=bsum(0,0)

   do 4 i=0, IEND
   do 4 j=0, IEND
     if(bsum(i,j) .lt. fmin) fmin=bsum(i,j)
     if(bsum(i,j) .gt. fmax) fmax=bsum(i,j)
 4 continue
c...Need to automatically choose the limits for the lineshape calculation.
c...Fstart is easy, one below its value is sufficient to give a first zero.
c...To find the end value, typical distributions were analyzed and it was
c...found that values from fmin up to a certain % of the range of fields held
c...all of the necessary information. The percentage increases as the angle
c...increases.

        print*,'In integ, fmin & fmax= ',fmin,fmax
        fstart=fmin-1.
        deltaf=fmax-fmin
        percnt=(2.*itheta)/100. + .15
        fend=fmin + deltaf*percnt
        df=(fend-fstart)/real(LEN)
        print*,'fstart,fend,df= ',fstart,fend,df
        pause 'Waiting ...'
        print*,'Input fstart,fend: '
        read*,fstart,fend
        df=(fend-fstart)/real(LEN)
        print*,'fstart,fend,df= ',fstart,fend,df

        c...Loop over field values.
        do 100 ifield=1,LEN
            do 100 ifield=1,numf

                field=fstart + (ifield-1)*df
                print*,'field is ',field

        c...Loop over the grid.
        do 200 j=0,IEND-1
            do 300 i=0,IEND-1

                xd=-1.
                yd=-1.
                xe=-1.
                ye=-1.

        c...Check if the field is in the lower right triangle.
APPENDIX B. SIMULATION CALCULATION PROGRAM

c...Need to also find the proper components of the field being sought for the
c...wiggles later on.
    if((field .lt. amax1(bsum(i,j),bsum(i+1,j)),
1 bsum(i+1,j+1))).and.(field .ge. amin(bsum(i,j),
2 bsum(i+1,j),bsum(i+1,j+1)))then

c...Calculate the gradient for the lower right triangle.
    dbdx=(bsum(i+1,j)-bsum(i,j))/dx
    dbdy=(bsum(i+1,j+1)-bsum(i+1,j))/dy
    gradb=sqrt(dbdx**2+dbdy**2)
    dbxds=(hsumx(i+1,j)-hsumx(i,j))/dx
    dbxds=(hsumx(i+1,j+1)-hsumx(i+1,j))/dy
    dbxds=(hsumy(i+1,j)-hsumy(i,j))/dx
    dbxds=(hsumy(i+1,j+1)-hsumy(i+1,j))/dy
    dbxds=(hsumz(i+1,j)-hsumz(i,j))/dx
    dbxds=(hsumz(i+1,j+1)-hsumz(i+1,j))/dy

c...Find the intersection points on the lower right triangle.
c...x-axis bottom.
    if((field .lt. amax1(bsum(i,j),bsum(i+1,j))).and.
1 (field .ge. amin(bsum(i,j),bsum(i+1,j)))) then
    xe=(field-bsum(i,j))/dbdx
    bxx=hsumx(i,j)+xe*dbxdx
    byx=hsumy(i,j)+xe*dbydx
    bzx=hsumz(i,j)+xe*dbzdx
    c print*, 'BR'
    endif

c...y-axis right.
    if((field .lt. amax1(bsum(i+1,j),bsum(i+1,j+1))).and.
1 (field .ge. amin(bsum(i+1,j),bsum(i+1,j+1)))) then
    ye=(field-bsum(i+1,j))/dbdy
    bxy=hsumx(i+1,j)+ye*dbxdy
    byy=hsumy(i+1,j)+ye*dbydy
    bzy=hsumz(i+1,j)+ye*dbzdy
APPENDIX B. SIMULATION CALCULATION PROGRAM

```
bzy=hsuzm(i+1,j)+ye*dbzdy

c      print*, 'SR'
   endif

c...diagonal up-right to low-left.
   if ((field .lt. amax1(bsum(i,j),bsum(i+1,j+1))).and.
      (field .ge. amin1(bsum(i,j),bsum(i+1,j+1)))then
      dbdd=(bsum(i+1,j+1)-bsum(i,j))/dstot
      deltas=(field-bsum(i,j))/dbdd
      fract=deltas/dstot
      xd=fRACT*dx
      yd=fRACT*dy

      dbxds=(hsuxm(i+1,j+1)-hsux(i,j))/dstot
      dbysd=(hsumy(i+1,j+1)-hsumy(i,j))/dstot
      dbzds=(hsumz(i+1,j+1)-hsumz(i,j))/dstot
      bxd=hsux(i,j)+deltas*dbxds
      byd=hsumy(i,j)+deltas*dbysd
      bzd=hsumz(i,j)+deltas*dbzds
      print*, 'DG'
   endif

c...Only two of the above three should be satisfied.
   if ((xe .gt. 0.).and.(ye .gt. 0.).and.(xd .gt. 0.)) then
      print*, 'Problem at i,j = ',i,j,' lower right triangle.'
      goto 40
   endif

   if (xe .LT. 0.) then
      dlen=sqrt((dx-xd)**2 + (yd-ye)**2)
   elseif (ye .LT. 0.) then
      dlen=sqrt((xd-xe)**2 + ye**2)
   else
      dlen=sqrt((dx-xe)**2 + ye**2)
   endif

c...Calc the contrib. to the sum.
```
APPENDIX B. SIMULATION CALCULATION PROGRAM  

cont=dlen/gradb  
contot=contot+cont

c...Here do the rotation stuff for a found field.  
c...First get the average field comps for this field -- need for rotations --  
c...only 2 of the 3 should be non-zero.  
c print*, 'bxx, bxy, bxd= ', bxx, bxy, bxd  
c print*, 'byx, byy, byd= ', byx, byy, byd  
c print*, 'bzx, bzy, bzd= ', bzx, bzy, bzd  
  bxavg=(bxx+bxy+bxd)/2.  
  byavg=(byx+byy+byd)/2.  
  bzavg=(bzx+bzy+bzd)/2.  

  bxx=0.  
  bxy=0.  
  byx=0.  
  byy=0.  
  bzx=0.  
  bzy=0.

c...Transform these comps into the LAB frame from the B frame -- for later.  
  hsx=bxavg*tmat(1,1)+byavg*tmat(1,2)+bzavg*tmat(1,3)  
  hsy=bxavg*tmat(2,1)+byavg*tmat(2,2)+bzavg*tmat(2,3)  
  hsv=bxavg*tmat(3,1)+byavg*tmat(3,2)+bzavg*tmat(3,3)  
  hsx=zchk(hsx)  
  hsy=zchk(hsy)  
  hsz=zchk(hsz)  
c print*, 'hsx, hsy, hs= ', hsx, hsy, hsz

  hperp2=hsx**2 + hsy**2  
  h2=hperp2 +hsz**2  
  h=sqrt(h2)  
  hperp=sqrt(hperp2)  

  csdel=zchk(hsz/h)  
  snedl=zchk(sqrt(1.-csdel**2))
c... Check if hperp=0.
    if (hperp.eq.0.) then
        csphib = 1.
        snphib = 0.
    else
        csphib = zchk(hsx/hperp)
        snphib = zchk(hsy/hperp)
    endif

c... Calculate rot matrices going into and out of the b frame.
    rtoh(1,1) = csdel * csphib
    rtoh(1,2) = csdel * snphib
    rtoh(1,3) = -sndel
    rtoh(2,1) = -snphib
    rtoh(2,2) = csphib
    rtoh(2,3) = 0.
    rtoh(3,1) = sndel * csphib
    rtoh(3,2) = sndel * snphib
    rtoh(3,3) = csdel

    htor(1,1) = rtoh(1,1)
    htor(1,2) = rtoh(2,1)
    htor(1,3) = rtoh(3,1)
    htor(2,1) = rtoh(1,2)
    htor(2,2) = rtoh(2,2)
    htor(2,3) = rtoh(3,2)
    htor(3,1) = rtoh(1,3)
    htor(3,2) = rtoh(2,3)
    htor(3,3) = rtoh(3,3)

c... Now rotate spin comps into the b (h) frame.
    sxh = sxl * rtoh(1,1) + syl * rtoh(1,2) + szl * rtoh(1,3)
    syh = sxl * rtoh(2,1) + syl * rtoh(2,2) + szl * rtoh(2,3)
    szh = sxl * rtoh(3,1) + syl * rtoh(3,2) + szl * rtoh(3,3)

c print*, 'sxh, syh, szh = ', sxh, syh, szh
c...Let the spin rotate for all time steps in its frame at proper freq. Is a
c...rotation about z (b-frame).
   do 80 it=0,LEN
      omegat=GAHMA*h*2.*PI*time(it)

      sgxt=szh*cos(omegat)+syh*sin(omegat)
      sgyt=-sxh*sin(omegat)+syh*cos(omegat)
      sgzt=szh

   c...Rotate back into the LAB frame.
      sigxp(it)=sgxt*htor(1,1)+sgyt*htor(1,2)+sgzt*htor(1,3)
      sigyp(it)=sgxt*htor(2,1)+sgyt*htor(2,2)+sgzt*htor(2,3)
      sigzp(it)=sgxt*htor(3,1)+sgyt*htor(3,2)+sgzt*htor(3,3)

   c  write(6,*)time(it),sigxp(it),sigyp(it),sigzp(it)
80 continue

   c...This is the contrib. to the average from this 1/2 triangle. Need to now
   c...add this to the total with the factor cont. above.
   do it=0,LEN
      sigxl(it)=sigxl(it)+cont*sigxp(it)
      sigyl(it)=sigyl(it)+cont*sigyp(it)
      sigzl(it)=sigzl(it)+cont*sigzp(it)
   enddo

40 xe=-1.
   ye=-1.

   c...Keep xd and yd in case are needed below.
   endif

   c...Now for the upper left triangle.
      if((field .lt. anaxl(bsum(i,j),bsum(i,j+1),
           1 bsum(i+1,j+1))).and.(field .ge. amini(bsum(i,j),
           2 bsum(i,j+1),bsum(i+1,j+1))) then

   c...Calc gradient up here.
APPENDIX B. SIMULATION CALCULATION PROGRAM

\[
\begin{align*}
\text{dbdx} &= \frac{\text{bsum}(i+1,j+1) - \text{bsum}(i,j+1)}{dx} \\
\text{dbdy} &= \frac{\text{bsum}(i,j+1) - \text{bsum}(i,j)}{dy} \\
\text{gradb} &= \sqrt{\text{dbdx}^2 + \text{dbdy}^2} \\
\text{dbxdx} &= \frac{\text{hsumx}(i+1,j+1) - \text{hsumx}(i,j+1)}{dx} \\
\text{dbxdy} &= \frac{\text{hsumx}(i,j+1) - \text{hsumx}(i,j)}{dy} \\
\text{dbydx} &= \frac{\text{hsumy}(i+1,j+1) - \text{hsumy}(i,j+1)}{dx} \\
\text{dbydy} &= \frac{\text{hsumy}(i,j+1) - \text{hsumy}(i,j)}{dy} \\
\text{dbzdx} &= \frac{\text{hsumz}(i+1,j+1) - \text{hsumz}(i,j+1)}{dx} \\
\text{dbzdy} &= \frac{\text{hsumz}(i,j+1) - \text{hsumz}(i,j)}{dy} \\
\text{c...Top.} \\
\text{if}((\text{field} \lt \text{amax1}(\text{bsum}(i,j+1),\text{bsum}(i+1,j+1))) \text{and} \quad 1 \quad (\text{field} \ge \text{amin1}(\text{bsum}(i,j+1),\text{bsum}(i+1,j+1))) \text{then} \\
\text{xe} &= (\text{field} - \text{bsum}(i,j+1))/\text{dbdx} \\
\text{bxx} &= \text{hsumx}(i,j+1) + \text{xe} \times \text{dbxdx} \\
\text{byx} &= \text{hsumy}(i,j+1) + \text{xe} \times \text{dbydx} \\
\text{bzx} &= \text{hsumz}(i,j+1) + \text{xe} \times \text{dbzdx} \\
\text{endif} \\
\text{c...Left side.} \\
\text{if}((\text{field} \lt \text{amin1}(\text{bsum}(i,j),\text{bsum}(i,j+1))) \text{and} \quad 1 \quad (\text{field} \ge \text{amax1}(\text{bsum}(i,j),\text{bsum}(i,j+1))) \text{then} \\
\text{ye} &= (\text{field} - \text{bsum}(i,j))/\text{dbdy} \\
\text{bxy} &= \text{hsumx}(i,j) + \text{ye} \times \text{dbxdy} \\
\text{byy} &= \text{hsumy}(i,j) + \text{ye} \times \text{dbydy} \\
\text{bzy} &= \text{hsumz}(i,j) + \text{ye} \times \text{dbzdy} \\
\text{endif} \\
\text{c...Diag. done above: if there is a crossing, xd and yd are calculated.} \\
\text{if}(\text{xe} \gt 0.) \text{and} (\text{ye} \gt 0.) \text{and} (\text{xd} \gt 0.) \text{then} \\
\text{print}, \text{'Problem at i,j= ',i,j,' upper left triangle.'} \\
\text{goto 300}
\end{align*}
\]
APPENDIX B. SIMULATION CALCULATION PROGRAM

endif

if(xe .lt. 0.)then
dlen=sqrt(xd**2 + (yd-ye)**2)
elseif(ye .lt. 0.)then
dlen=sqrt((xd-xe)**2 + (dy-yd)**2)
else
dlen=sqrt(xe**2 + (dy-ye)**2)
endif

c...Calc contrib. for this.
cont=dlen/gradb
contot=contot+cont

c...Now do same as above for these new components of the field.
c...Here do the rotation stuff for a found field.
c...First get the average field comps for this field -- need for rotations --
c...only 2 of the 3 should be non-zero.
bxavg=(bxx+bxy+bxd)/2.
byavg=(byx+byy+byd)/2.
bzavg=(bxz+bzy+bzd)/2.

bxx=0.
bxy=0.
byx=0.
byy=0.
bxz=0.
bzy=0.

hsx=bxavg*tmat(1,1)+byavg*tmat(1,2)+bzavg*tmat(1,3)
hsy=bxavg*tmat(2,1)+byavg*tmat(2,2)+bzavg*tmat(2,3)
hz=bxavg*tmat(3,1)+byavg*tmat(3,2)+bzavg*tmat(3,3)
hx=zchk(hsx)
hsy=zchk(hsy)
hsz=zchk(hsz)
c...Transform these comps into the LAB frame from the B frame -- for later.
print*, '2,hx,hsy,hz= ',hx,hsy,hz

167
APPENDIX B. SIMULATION CALCULATION PROGRAM

\[ h_{perp}^2 = h_{sx}^2 + h_{sy}^2 \]
\[ h^2 = h_{perp}^2 + h_{az}^2 \]
\[ h = \sqrt{h^2} \]
\[ h_{perp} = \sqrt{h_{p}^2 + h_{r}^2} \]
\[ \text{csdel} = \text{zchk}(h_{az}/h) \]
\[ \text{sndel} = \sqrt{1 - \text{csdel}^2} \]

Check if \( h_{perp} = 0 \).

\[ \text{if}(h_{perp} \text{eq.} 0.) \text{then} \]
\[ \text{csphib} = 1. \]
\[ \text{snphib} = 0. \]
\[ \text{else} \]
\[ \text{csphib} = \text{zchk}(h_{sx}/h_{perp}) \]
\[ \text{snphib} = \text{zchk}(h_{sy}/h_{perp}) \]
\[ \text{endif} \]

Calculate rot matrices going into and out of the \( b \) frame.

\[ rtoh(1,1) = \text{csdel} \times \text{csphib} \]
\[ rtoh(1,2) = \text{csdel} \times \text{snphib} \]
\[ rtoh(1,3) = -\text{sndel} \]
\[ rtoh(2,1) = -\text{snphib} \]
\[ rtoh(2,2) = \text{csphib} \]
\[ rtoh(2,3) = 0. \]
\[ rtoh(3,1) = \text{sndel} \times \text{csphib} \]
\[ rtoh(3,2) = \text{sndel} \times \text{snphib} \]
\[ rtoh(3,3) = \text{csdel} \]

\[ htor(1,1) = rtoh(1,1) \]
\[ htor(1,2) = rtoh(2,1) \]
\[ htor(1,3) = rtoh(3,1) \]
\[ htor(2,1) = rtoh(1,2) \]
\[ htor(2,2) = rtoh(2,2) \]
\[ htor(2,3) = rtoh(3,2) \]
\[ htor(3,1) = rtoh(1,3) \]
\[ htor(3,2) = rtoh(2,3) \]
htor(3,3)=rtoh(3,3)

...Now rotate spin comps into the b (h) frame.
  sxh=sxl*rtoh(1,1)+syl*rtoh(1,2)+szl*rtoh(1,3)
  syh=sxl*rtoh(2,1)+syl*rtoh(2,2)+szl*rtoh(2,3)
  szh=sxl*rtoh(3,1)+syl*rtoh(3,2)+szl*rtoh(3,3)

print*, ' 2, sxh, syh, szh = ', sxh, syh, szh

...Rotate the spin for all time steps in its frame at the proper freq.
  do 90 it=0,LEN
    omegat=GAMMA*h*2.*PI*time(it)
    sgxt=sxh*cos(omegat)+syh*sin(omegat)
    sgyt=-sxh*sin(omegat)+syh*cos(omegat)
    sgzt=szh
  c...Rotate back into the LAB frame.
    sigxp(it)=sgxt*htor(l,1)+sgyt*htor(l,2)+sgzt*htor(l,3)
    sigyp(it)=sgxt*htor(2,1)+sgyt*htor(2,2)+sgzt*htor(2,3)
    sigzp(it)=sgxt*htor(3,1)+sgyt*htor(3,2)+sgzt*htor(3,3)
  write(5,*)(time(it), sigxp(it), sigyp(it), sigzp(it)
  90 continue

...This is the contrib. to the average from this 1/2 triangle. Need to now
...add this to the total with the factor cont. above.
  do it=0,LEN
    sigxl(it)=sigxl(it)+cont*sigxp(it)
    sigyl(it)=sigyl(it)+cont*sigyp(it)
    sigzl(it)=sigzl(it)+cont*sigzp(it)
  enddo
  endif

bxd=0.
byd=0.
APPENDIX B. SIMULATION CALCULATION PROGRAM

bzd=0.

300  continue
200  continue
100  continue

c...Scale according to the total areal contribution.
do  i=0,LEN
   sigxl(i)=sigxl(i)/contot
   sigyl(i)=sigyl(i)/contot
   sigzl(i)=sigzl(i)/contot
endo
c...Write results.
open(9,file=fil2)
print*, 'Writing to file ',fil2
do  it=0,LEN
   write(9,*),time(it),sigxl(it),sigyl(it),sigzl(it)
60  continue
close(9)
end

function zchk(val)
c...This function checks for values which are essentially zero and
c...makes them so.
tol=1.e-4
if((val .lt. tol).and.(val .gt. -tol))then
   zchk=0.
elsetzchk=val
endif
return
end
Appendix C

Simulation Analysis Program

The following program (called *wow*) accepts as input the output data from three separate runs of the simulation program — for a total of 9 wiggle lines. These data are assumed to come from the three initial conditions described in chapter 5 — B along z, and the initial polarization of the muons along each of x, y, and z, respectively. The data are then treated as outlined in chapter 5 as far as the subtractions, divisions, and Fourier transformations, to yield the moment one wishes. The Fourier transformations are done in a subroutine called *slosum*, which will perform both cosine and sine discrete Fourier transforms on the data. The data are returned to the main program in two arrays, one containing the cosine transform, and one containing the sine data. One chooses the proper data set depending on the application.

Points of note and care:

1. One should note the factor of 1/2 included in the cosine transformation. This only happens for the first data point of the data which are being transformed, but was found to be of critical importance. If this is not done, then there arises a false, non-zero level in much of the baseline of the resulting transform. This can cause many problems when one attempts to calculate areas of moments, etc.

2. It is of extreme importance to make sure that the simulation data have been properly generated. This means that all fields within the grid should have their
contribution to the wiggles. If this is not done the transform will have much
more ringing than otherwise.

program vvow
  c...This program will accept 3 data sets corresponding to three separate expts
c...studying off-axis fields. The data are assumed to be in 4 cols as: time,
c...x data, y data, z data. These data once read in are then Fourier
  c...Transformed into their corresponding cos and sin parts. These transforms
  c...are then manipulated to extract the relevant information such as dn/db.
  parameter(LEK=2000,GAMMA=13.55e-3)
  dimension dataxx(LEN),dataxy(LEN),dataxz(LEN),datayx(LEN),
  1 datayy(LEN),datayz(LEN),datazx(LEN),datazy(LEN),datazz(LEN),
  2 dataxt(LEN),datayt(LEN),datazt(LEN),dndb(LEN),sumdat(LEN),
  3 difdat(LEN),ftsumc(LEN),ftsums(LEN),ftdifc(LEN),ftdifs(LEN),
  4 sumbzy(LEN),difbx(LEN),ftbzc(LEN),ftbzs(LEN),difbz(LEN),
  5 ftdbzc(Lenk),ftdbzs(LEN),ftxxc(LEN),ftxxs(LEN),fttyc(LEN),
  6 ftyys(Len),ftzzc(LEN),ftzzs(LEN),fdndbc(LEN),fdndbs(LEN),
  7 temp(LEN),sumbxy(LEN),ftbxc(LEN),ftbxs(LEN)
  character fill1*20,fill2*20,fill3*20,outfil*20
  print*, 'Input the three data file names, one per line:'
  read(6,'(a20)')fill,fil2,fil3
  open(9,file=fill)
  open(10,file=fil2)
  open(11,file=fil3)
  do i=l,LEN
    read(9,*),end=10)dataxt(i),dataxx(i),dataxy(i),dataxz(i)
    read(10,*),datayt(i),datayx(i),datayy(i),datayz(i)
    read(11,*),datazt(i),datazx(i),datazy(i),datazz(i)
  enddo
  10 close(9)
  close(10)
  close(11)
  ilast=i-1
  c...Subtract off any DC levels from the data.
  do i=1,ilast
    sumxx=sumxx+dataxx(i)
    sumxy=sumxy+dataxy(i)
    sumxz=sumxz+dataxz(i)
APPENDIX C. SIMULATION ANALYSIS PROGRAM

```fortran
sumyx = sumyx + datayx(i)
sumyy = sumyy + datayy(i)
sumyz = sumyz + datayz(i)
sumxx = sumxx + datazx(i)
sumzy = sumzy + datazy(i)
sumzz = sumzz + datazz(i)

enddo
avgxx = sumxx / real(ilaat)
avgxy = sumxy / real(ilaat)
avgxz = sumxz / real(ilaat)
avgyx = sumyx / real(ilaat)
avgyy = sumyy / real(ilaat)
avgyz = sumyz / real(ilaat)
avgzx = sumzx / real(ilaat)
avgzy = sumzy / real(ilaat)
avgzz = sumzz / real(ilaat)

do i = 1, ilaat
  dataxx(i) = dataxx(i) - avgxx
  dataxy(i) = dataxy(i) - avgxy
  dataxz(i) = dataxz(i) - avgxz
  datayx(i) = datayx(i) - avgyx
  datayy(i) = datayy(i) - avgyy
  datayz(i) = datayz(i) - avgyz
  datazx(i) = datazx(i) - avgzx
  datazy(i) = datazy(i) - avgzy
  datazz(i) = datazz(i) - avgzz
endo

... Now implement the Petzinger idea. Add wigs from x along z to wigs from c...z along x and cos xf; then subtract and do a sin xf.

do i = 1, ilaat
  sumdat(i) = (datazx(i) + dataxz(i))/2.
  difdat(i) = (dataxz(i) - datazx(i))/2.
  sumbzy(i) = (datayz(i) + datayx(i))/2.
```

difbx(i)=(dataxy(i)-datayz(i))/2.
difbz(i)=(datayz(i)-dataxy(i))/2.
sumbx(i)=(dataxy(i)+datayx(i))/2.
c   write(30,*),datazt(i),dataxx(i),datayy(i)
c   write(31,*),dataxt(i),dndb(i)
c   write(32,*),datazt(i),dilbx(i)
c   write(33,*),datazt(i),datayx(i),dataxy(i),dilbz(i)
enddo
c close(30)
c close(31)
c close(32)
c close(33)
c...Setup for the FT's.
   print*, 'Input starting and ending freqs for the FT (MHz):'
   read*, fstart, fend
   print*, 'Input the number of divisions (<', LEI, '):'
   read*, numdiv
deltal=(fend-fstart)/real(numdiv)
   print*, 'fstart, fend, deltal', fstart, fend, deltal

c...dn/db — add the data and divide by 2, then FT.
do i=1,ilast
   dndb(i)=(dataxx(i)+datayy(i)+datazz(i))/2.
   write(66,*),dataxt(i),dndb(i)
enddo
   close(66)
c...This is the FT for n(b) — sin xf with freq mult in slosum.
call slosum(dataxt,dndb,fdndbc,fdn dbs,ilast,fstart,
   1   deltaf,numdiv,LEN,arxxc,arxxs)
c call slosum(dataxt,difbz,ftyyc,ftyy s,ilast,fstart,
c   1   deltaf,numdiv,LEN,aryyc,aryys)
c call slosum(dataxt,datazz,ftzzc,ftzzs,ilast,fstart,
c   1   deltaf,numdiv,LEN,arzzc,arzzs)
c call slosum(dataxt,dibx,ftbxc,ftbxs,ilast,fstart,
c   1   deltaf,numdiv,LEN,arxxc,arxxs)
c...Now find area of dndb and use for all normalizations hereafter.
area=arxxc
   ater=simpn(fdndbc,numdiv,deltaf=1.e6)
APPENDIX C. SIMULATION ANALYSIS PROGRAM

print*, 'area, ater= ', area, ater
do i=1,numdiv
    write(67,'(f12.8,f12.8,f12.8)')
    write(67,*)fstart+deltaf*(i-1),ldndbc(i)/ater
enddo
close(67)
c...Now send the appropriate data to the FT routine.
c  call slosum(dataxt, sumdat, ftsum, ftsums, ilast, fstart,
  1  deltaf, numdiv, len, areayc, areays)
c...These are all sin xf'd with a freq mult for bxn(b), bxn(b), and bzn(b).
c  call slosum(datazt, dildat, ltdils, ltdilc, ilast, fstart,
  1  deltaf, numdiv, len, areayc, areays)
c  print*, 'Area of by curve is ', areays
  call slosum(datazt, dildbx, ltdbxc, ltdbxs, ilast, fstart,
  1  deltaf, numdiv, len, areaxc, areaxs)
c  print*, 'Area of bx curve is ', areaxs
  call slosum(dataxt, dildbz, ltdbz, ltdbzs, ilast, fstart,
  1  deltaf, numdiv, len, areaxc, areaxs)
c  print*, 'Area of bz curve is ', areaxs

c...Now print these out for curiosity sake...
c  open(20, file='sinxls')
c  open(21, file='dndbc.out')
c  open(22, file='ltbx')
c  do i=1,numdiv
c    write(20,'(f12.8,f12.8,f12.8,f12.8,f12.8)')
    write(21,*)fstart+(i-1)*deltaf
    write(22,*)fstart+(i-1)*deltaf, ftyys(i)/area
  enddo
c  close(20)
c  close(21)
c  close(22)
c...Now integrate over the appropriate fit for the average.
  neven=mod(numdiv,2)
  if(neven.eq.0)then
    ndiv=numdiv-1
  else
    ndiv=numdiv
  endif
do i=1,ndiv
  freq=fstart+(i-1)*deltal
  iseven=mod(i,2)
  if((i.eq.1).or.(i.eq.ndiv))then
    fac=1.
  elseif(iseven .eq. 0)then
    fac=4.
  else
    fac=2.
  endif
  avbxz=avbxz+fac*ftsumc(i)/3.
  avby=avby+fac*ltdifs(i)/3.
  avbx=avbx+iac*itbxs(i)/3.
  avbz=avbz+fac*ftbzs(i)/3.
enddo
print*,'avbx, avby, avbz= ', avbx,avby, avbz
print*,'<bx>,<by>,<bz> = ',avbx*daltal/GAMMA/area,
1  avby*deltaf/GAMMA/area,avbz*deltal/GAMMA/area
print*,'<bx>,<by>,<bz> = ',avbx*daltai/GAMMA,
1  avby*deltaf/GAMMA,avbz*deltaf/GAMMA
end

c*****************************************************************************
subroutine slosum(tdata,data,ftdats,ildat,
  istart, deltal, numdiv, ILEI, araac, areas)
  c...This routine sets up for integrated fourier cos & sin transforms:
  c...
  c... amp(t) * cos(2*PI*freq*t) [and sin(2*PI*freq*t)]
  c...
  c...The integration is simply a summation only of the above products.
  c*****************************************************************************
  parameter(PI=3.1415926,LEI=2000)
  dimension freq(LEI),cosans(LEI),sinans(LEI)
  dimension tdata(ildat),data(ildat),ftdats(ILEI),ftdata(ILEI)
  s=0.
  areac=1.
  areas=1.

deltat=tdata(10)-tdata(9)
print*, 'deltat = ', deltat

Calculate the frequencies.
do 45 i=1,numdiv+1
45 freq(i)=fstart+deltaf*(i-1)
print*, 'Thinking.......'

How transform data, looping on frequencies.
do 55 index=1,numdiv

Initialize sum for each new freq.
cosans(index)=0.
sinans(index)=0.
w=2.*PI*freq(index)

Calculate integral for the freq at index, using a simple sum.
do 53 j=1,ildat

Note the use of cos as mentioned above. Also note the division by 2 for
the first element of the cosine transform - a la RLV.
if(j .eq.1) then
cosfac=.5
else
  cosfac=1.
endif

cosans(index)=cosans(index) + 
  1   cosfac*data(j)*cos(w*tdata(j))*deltat

c 1  freq(index)=2*cosfac*data(j)*cos(w*tdata(j))*deltat

sinans(index)=sinans(index) +
c 1   data(j)*sin(w*tdata(j))*deltat

1  freq(index)*data(j)*sin(w*tdata(j))*deltat
53 continue
55 continue

area=simpson(cosans,numdiv,deltaf)
areas=simpson(sinans,numdiv,deltaf)
print*, 'This time, area = ', area, areas
if(abs(area) .lt. 1.) area=1.
if(abs(areas) .lt. 1.) areas=1.

Check for the root of the xfrms.
call fzero(cosans,numdiv,icdxx)
call fzero(sinans,numdiv,isdxx)
APPENDIX C. SIMULATION ANALYSIS PROGRAM

if(icdxx .ne. 0) then
    areac = area
    print*, 'areac=', areac, ', icdxx=', icdxx
else if(isdxx .ne. 0) then
    area = areac
    print*, 'area=', area, ', isdxx=', isdxx
endif

do 91 iow=1, numdiv
    fdatc(iow) = cosans(iow)
    fdatd(iow) = sinans(iow)
91
icdxx = 0
isdxx = 0
9999 end

*************************************************************************
function simsen(data, ndat, dx)
*************************************************************************
dimension data(ndat)
temp = 0.
neven = mod(ndat, 2)
if(neven .eq. 0) then
    nd = ndat - 1
else
    nd = ndat
endif

do 10 i=1, nd
    ieven = mod(i, 2)
    if((i .eq. 1) .or. (i .eq. nd)) then
        fac = 1.
    elseif (ieven .eq. 0) then
        fac = 4.
    else
        fac = 2.
    endif
10
    temp = temp + data(i) * fac * dx / 6.
enddo

return
end
temp = temp + fac * data(i) * dx / 3.
10 continue
simpsn = temp
end

subroutine fzero(ydata, numdiv, indexx)
c...This routine takes data in the array ydata and looks for the
c...prominent zero crossing in it, returning the index in the variable
c...indexx. The prominent zero is defined arbitrarily as that where there
c...exists NUMWIN consecutive values on each side with same sign (pos/neg).
c...Local zeroes should fail this criterion (hopefully).
parameter(NUMWIN=30)
dimension ydata(numdiv)
  indexx = 0
  isin1 = 1
  if(ydata(NUMWIN).lt.0.) isin1 = -1
  do 10 i = NUMWIN + 1, numdiv - NUMWIN
    if(ydata(i).lt.0.) then
      isin2 = -1
    else
      isin2 = 1
    endif
  if(isin1.ne.isin2) then
   c...zero cross.
   c print*, 'Have a zero crossing at i = ', i, ' 181,182=',
   c 1 isin1,isin2
   do j = i - NUMWIN, i - 1
      isnlft = sign(real(isin1), ydata(j))
      if(isnlft.ne.isin1) then ! signs have switched
        isin1 = sign(real(isin1), ydata(i))
      endif
    enddo
  endif
  do j = i + 1, i + NUMWIN
    isnrt = sign(real(isin2), ydata(j))
    if(isnrt.ne.isin2) then ! signs switch
      isin1 = sign(real(isin1), ydata(i))
    endif
  enddo
goto 10
endif
enddo

c...If get to here then is a valid zero cross.
indexx=i
return
endif
10 continue
return
end
Appendix D

TRUIMF Data Analysis

In 1990 Tanya Riseman was taking data at TRIUMF for her PhD thesis. In doing so she built a detector rig much like that imagined in this thesis: three mutually perpendicular (pairs of) detectors which are calibrated and can be considered "the same" in some sense. Much data were taken with this rig, and a lot of the data were recorded at low fields on single crystals of YBCO. The single crystals were of high quality, for the time, and were manufactured at the University of British Columbia, in Vancouver, Canada.

For the gory details of the thesis, one can consult Ref. [17]. I will outline here what was done that is similar to what this thesis suggests, as well as describe what is different from what this thesis suggests. Finally, the ideas behind the analysis program (presented in appendix E) which the author wrote will be discussed, as well as how the TRIUMF data were used as a test run for possible experimental/analysis problems.

D.1 The TRIUMF Experimental Arrangement

The TRIUMF experimental rig consisted mainly of a target surrounded by a box shaped set of 6 detectors. The muon polarization could be changed from antiparallel to the incident beam direction to vertically perpendicular to it. This was accomplished by a separator/rotator which was just upstream from the target area. Therefore the
angle $\gamma$ of Fig. 6.1 can be anywhere in the range $180^\circ \geq \gamma \geq 0^\circ$, assuming that the beam comes in from the left in the figure. The angle $\phi$, is zero, confining the polarization to the vertical plane. The applied magnetic field $H_a$ also lies in this plane, and can be varied in both magnitude and direction by two sets of Helmholtz coils at right angles to each other. Hence the angle $\beta$ of Fig. 6.1 is in the range $0^\circ \leq \beta \leq 90^\circ$, and the angle $\phi_\beta = 0^\circ$.

The target was a single crystal of the superconductor $YBa_2Cu_3O_{6.95}$, but was not one single piece. Actually, the target consisted of a mosaic of smaller crystals arranged on a piece of mylar such that they were aligned together. This presented a cross section to the muon beam of about 3 cm$^2$.

D.2 The TRIUMF Data

The data taken using the above mentioned apparatus were contained in data files on the TRIUMF VAX cluster. Each file contained up to six separate histograms, one for each side of the box detector. The information contained in the file header, coupled with copies from the data log books at the time of the experiment, allowed the geometry of each run to be deduced. It was hoped that some data consistent with the geometry of equations 6.5 to 6.11 would be available. That is, three sets of data where the initial polarization is along each of three mutually perpendicular directions, respectively, for the same applied field direction. After figuring out the geometries used in the TRIUMF data, it was determined that no such combination of data runs was obtained. This was mainly due to the inability of the experimental apparatus to allow for the third direction of the muon initial polarization, as the rotator only rotates the polarization in a plane.

Most of the data taken by Riseman (and others) was in a Transverse Field geometry, where the initial polarization and applied field were kept at right angles to each other. In addition, most runs were of insufficient statistics, based on our calculations in chapter 5, to adequately produce the moment distributions described there. This is most likely due to the considerable time constraints imposed at a major facility like TRIUMF, where beam time is quite valuable. Also, the goal of the work done by
Riseman was not the same as this work — she was extracting penetration depths, coherence lengths, and looking for the low-field phenomenon of the chain state, among other things — so the statistics were not as critical as they will be if this work is ever tackled.

In view of the situation with the TRIUMF data, it was decided to nevertheless attempt to analyze many runs anyway. It was hoped that many mistakes, oversights, and problems could be ironed out early by having some real data which are at least from the type of detectors we wish to use. To this end the author wrote a program, contained in appendix E, which reads in the TRIUMF binary data files, prepares the histograms, and does analysis.

### D.3 Analysis

Particular attention was paid to certain aspects of data analysis in the program. First, the background is determined in one of three ways: one, a semi-log plot of the data is presented to the user — if the long time data do not fall on the straight line, then the background is incorrect and the user is allowed to adjust it for straightening; second, one can include a background term in the fitting function and fit to the background; third, one could simply set the background to some level (not generally a good idea).

The reason for concern over the background was that if the value is not set properly then the asymmetry representation will not be correct. The general effect is to cause the asymmetry data to rise or fall at later times, instead of being nice and straight. This is a bothersome problem which can only be rectified if proper background subtraction in the asymmetry equation 3.14 is done. Some examples of improper background subtraction are shown in Ref. [17], including some extreme cases.

Another area of concern and effort was in the fitting itself. Generally fitting is done to either the individual histograms (with the muon lifetime still there) or to the asymmetry representation of paired histogram data (wiggles only). The fits may contain a simple function or a complicated one, depending on what one expects to find. However, one usually fits over the entire time range, or only out to a few
Figure D.1: Shown here are the data from TRIUMF run 6730 in the asymmetry representation. The arrow indicates where the “hiccup” occurs between the two apparent frequencies in the data.

microseconds for very rapidly behaving materials. In this program, based on what the data seemed to present, we allow for fits to the end of the time range, and also to the whole time range. The user sets the starting time for the “later time” fits, and the least squares routines work out the best fit.

This fitting scheme was devised due to the apparent appearance of two separate signals (frequencies) within the data from TRIUMF. These are presumed to be from two different sources: the actual target under investigation, and the sample holder to which the sample is attached. An example of this behavior is shown in Fig. D.1. One can see a “hiccup” in the wiggles as indicated in the figure.

It was assumed, especially after seeing the results of our simulations, that the early time signal was from the sample, and the long time signal was from the holder.
Therefore the longer time fits were done on the sample holder signal. This fit was then extrapolated back to $t = 0$, and the long time fit was subtracted off of the entire histogram. This was done in the hope of leaving only the signal from the sample, which should correspond to the theory. An example of a later time fit and its subtraction from the data is shown in Fig. D.2.

The third concern was the proper determination of the parameter $\alpha$, seen in equation 3.14. This parameter is supposed to help make up for inequalities in different detectors which are paired about the sample. For example, one detector may be more efficient at detecting the positrons than the other. Or, maybe the discriminators in the coincidence rack are off by some amount, letting in some events to one detector and not into another. Whatever the case, the alpha parameter is a first order attempt to even out any experimental problems. Alpha is generally determined from fits to low TF runs on a sample like $\text{Cu}$ or other well known material. The quantity $N_0$ of equation 3.6 for each fit is then used to determine alpha via $\alpha = N_f^b/N_0^b$, where the superscripted $f$ and $b$ can be any paired detector parameters.

If alpha is determined properly, then the long time offsets which are seen in the simulation data (see Fig. 6.8) should be visible. If alpha is not properly determined, then an offset may be as likely due to an incorrect alpha parameter as a real offset. It is therefore imperative to properly determine the alpha parameter if one wishes to see the long time offsets.

These three issues – proper background subtraction, two frequency fitting and subtraction, and proper alpha determination – have come out of the TRIUMF data analysis as key to doing good experiments corresponding to the theory presented in this work.

D.4 Further Obstacles

Even if all of the procedures above are followed and correct data results, the data may not look like the simulation data. The biggest difference will be that, e.g., the $P_{zz}(t)$ data will not be as far offset from zero as in the theory. Even though $\alpha$ determines the offset, the data may be such that a scaling is needed to bring it up to the proper
Figure D.2: In the top plot is shown the raw data (dots) and the fit to the later time data. In the bottom plot are the raw data (again as dots) and the resulting subtracted data (dotted line). It is hoped that the subtracted spectrum is the signal from the sample only.
level. More correctly stated, a scaling is needed to map the actual level to where it theoretically should be. The scale factor can be determined by running the experiment on something like Cu, which is well known and has a very slow depolarization rate. The data for the three initial polarization directions should be much closer to theory for a uniform field distribution, and the offset for the longitudinal field case should be closer to 1. The copper level is then used as the "1" level, and the subsequent runs on the YBCO can be scaled to this level.

Another problem which can occur is that the initial amplitudes of the wiggles may not be of the proper size. This is closely tied in with the offset being too small, and occurs for much the same reason. If all of the muons were exactly aligned at \( t = 0 \) and all of the detectors were exactly the same and the solid angle effects of the detectors could be ignored, then the amplitude would approach the proper value for high enough statistics. However, this is certainly not the case, so the amplitude of the wiggles suffers. The fix for this is to also run on a material like Cu and use the wiggles from this as the standard by which the YBCO wiggles will be scaled.
Appendix E

TRIUMF Data Analysis Program

The following program, tridat, was written to analyze μSR data which are stored in the TRIUMF data format. This format is different than that used by the BNL μSR group, for which an analysis program already existed.

This program borrows much of the analysis code from the BNL code, written in the late 1980's by W.J. Kossler. Part of the contribution of the author is in the correct reading in of the TRIUMF binary data files, as well as the correct un-flipping of the bytes necessitated by the IBM RISC machine's dyslexia. The subroutine datrd reads in the data, while the function ibmfip does the byte flipping. The non-linear least squares fitting routines have been omitted, but they are the standard ones from Ref. [97].

The routine display displays the data and the fit rather quickly in a separate pop-up window. The routines for this are from the PLPLOT package [98], which works under the X11 windows system. Implementing this was also a contribution of the author.

The program has within it (as a callable option) the routine badbin, also written in the late 1980's by the BNL μSR group. This routine searches through the data and looks for spurious peaks caused by electronic glitches and the like. It then replaces the glitch by a suitably averaged value at the proper bin.

Further contributions of the author are in the areas of background investigation and in fitting style, as described above in appendix D. The background code is in the
subroutine bkg, and the new fitting style is coded in char and tfreq, below.

program tridat
subroutine ibmflp to correct for IBM's goofy byte switching problem. It also
creates the data and time arrays, corrected for packing, and coerces them
into the (1-a2)/(1+a2) asymmetry form. A routine called tfunc holds the
asymmetry fit function, and the normal NUMR routines are called for the
fitting, just as in the BNL part of the program.

parameter(HUMDAT=10240,HPAR=10)
integer*2 ihis1(HUMDAT),ihis2(HUMDAT),ihis3(HUMDAT),
ihis4(HUMDAT),ihis5(HUMDAT),ihis6(HUMDAT)
dimension data1(HUMDAT),data2(HUMDAT),data3(HUMDAT),
data4(HUMDAT),data5(HUMDAT),data6(HUMDAT),dsig1(HUMDAT),
dsig2(HUMDAT),dsig3(HUMDAT),dsig4(HUMDAT),dsig5(HUMDAT),
dsig6(HUMDAT)
dimension asyml(HUMDAT),asym2(HUMDAT),asym3(HUMDAT),at1(HUMDAT),
at2(HUMDAT),at3(HUMDAT),fnc(HUMDAT),sig1(HUMDAT),sig2(HUMDAT),
sig3(HUMDAT),t1(HUMDAT),t2(HUMDAT),t3(HUMDAT),t4(HUMDAT),
t5(HUMDAT),t6(HUMDAT)
dimension parms(HPAR),covar(HPAR,HPAR),alpha(HPAR,HPAR),
dyda(HPAR),lista(HPAR),epar(HPAR)
real*8 y(HUMDAT),t(HUMDAT),f(HUMDAT)
c...These parameters are set depending on the fitting function.
integer*4 nend
logical newdat
external tfunc
external ffunc
data lbl(1),lbl(2),lbl(3),lbl(4),lbl(5),lbl(6)/
  'a(1)','a(2)','a(3)','a(4)','a(5)','a(6)'/
data lbl(7),lbl(8),lbl(9),lbl(10)/'a(7)','a(8)',
  'a(9)','a(10)'/
data lista /1,2,3,4,5,6,7,8,9,10/
newdat=.true.
c...These parameters are set depending on the fitting function.
ma=6
APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

mf=6
nca=10
na=10

...Ask the user what to do.
7    print*, 'Pick your favorite option: '
    print*, 'he = Print options.'
    print*, 'rd = Read data.'
    print*, 'ip = Input parameters and choose spectrum.'
    print*, 'as = Calculate asymmetry function.'
    print*, 'fu = Calculate function.'
    print*, 'ft = Fit data.'
    print*, 'di = Display function and data.'
    print*, 'ba = Badbin data.'
    print*, 'of = Off-Axis stuff.'
    print*, 'pr = Print data to file.'
    print*, 'bk = Check background.'
    print*, 'f2 = Fit 2 frequency data.'
    print*, 'ex = Exit program.'
10   print*, 'Option?'
    read(5, '(a2)') resp
    if (resp .eq. 'rd') then
        goto 1
    elseif (resp .eq. 'ip') then
        goto 2
    elseif (resp .eq. 'fu') then
        goto 3
    elseif (resp .eq. 'ft') then
        goto 4
    elseif (resp .eq. 'di') then
        goto 5
    elseif (resp .eq. 'ex') then
        goto 6
    elseif (resp .eq. 'he') then
        goto 7
    elseif (resp .eq. 'ba') then
        goto 8
    elseif (resp .eq. 'of') then
APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

```
goto 9
elseif(resp .eq. 'as')then
goto 11
elseif(resp .eq. 'pr')then
goto 12
elseif(resp .eq. 'bk')then
goto 13
elseif(resp .eq. 'f2')then
goto 14
else
goto 10
endif

c...Read in the data and the appropriate timing parameters -- assumes up to 4 histograms only.
call datrd(ihisl,it01,itl1,it21,ihis2,it02,it12,it22,
ihis3,it03,it13,it23,ihis4,it04,it14,it24,ihis5,it05,
it15,it25,ihis6,it06,it16,it26,dt,mhist)
c...Create the actual time and histogram data taking into account any packing.
call datcre(ihisi,ihis2,ihis3,ihis4,ihis5,ihis6,datal,data2,
data3,data4,data5,dt,t2,t3,t4,t5,t6,dsig1,dsig2,dsig3,
3 dsig4,dsig5,dsig6,dt01,dt11,dt21,dt02,dt12,dt22,dt03,dt13,
4 dt23,dt04,dt14,dt24,dt05,dt15,dt25,dt06,dt16,dt26,dt,nendi,
nend2,nend3,nend4,nend5,nend6,bkg1,bkg2,bkg3,bkg4,bkg5,bkg6,
npk)
newdat=.true.
goto 10
c...Get parameters for the function.
print*, 'Is this for single histo or asym? (s/a)'
read(5, '(a1)')ftype
if(ftype .eq. 's')then
27 print*, 'Spectrum now is', ians, ' Which spectrum? (1-',mhist,')'
read*, ians
if(ians .gt. mhist)goto 27
elseif(ftype .eq. 'a')then
28 print*, 'Spectrum now is', kans, ' Which spectrum? (1-',mhist/2,')'
read*, kans
```
APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

    if(kans .gt. mhist/2)goto 28
else
    goto 2
endif
print*, 'ma is ', ma, 'Input the number of parameters in function:
read(5,*_,err=2)ma
if(ma .gt. 6)then
    ma=10
    print*, 'ma set to 6, try again'
    goto 2
endif
    call parin(lbl,parms,ma)
goto 10
  c...Calculate the function.
  3 if(ftype .eq. 'a')then
    if(kans .eq. 1)then
        do i=1,kend1
            call tfunc(at1(i),parms,fnc(i),dyda,na)
        enddo
    elseif(kans .eq. 2)then
        do i=1,kend2
            call tfunc(at2(i),parms,fnc(i),dyda,na)
        enddo
    elseif(kans .eq. 3)then
        do i=1,kend3
            call tfunc(at3(i),parms,fnc(i),dyda,na)
        enddo
    else
        print*, 'kans is wrong, = ', kans
    endif
    elseif(ftype .eq. 's')then
        if(ians .eq. 1)then
            do i=1,nend1
                call ffunc(tl(i),parms,fnc(i),dyda,na)
            enddo
        elseif(ians .eq. 2)then
            do i=1,nend2
        endif
    endif

APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

```
call ffunc(t2(i), paras, fnc(i), dyda, na)
enddo
elseif(ians .eq. 3)then
  do i=l,nend3
    call ffunc(t3(i), paras, fnc(i), dyda, na)
  enddo
elseif(ians .eq. 4)then
  do i=l,nend4
    call ffunc(t4(i), paras, fnc(i), dyda, na)
  enddo
elseif(ians .eq. 5)then
  do i=l,nend5
    call ffunc(t5(i), paras, fnc(i), dyda, na)
  enddo
elseif(ians .eq. 6)then
  do i=l,nend6
    call ffunc(t6(i), paras, fnc(i), dyda, na)
  enddo
endif
endif
goto 10
c...Fit the function to the data. The 1 after alanda is called nfp in bnl.dat,
c...it appears to be the starting point for the calc of the function. till
nc...appears where nfirst is in bnl.dat, not sure about it yet (i also??).
4  if(ftype .eq. 'a')then
    if(kans .eq. 1)then
      call curfit(lisa, at1, asym1, sig1, kend1, paras, ma, mf, covar,
        1  alpha, nca, chisq, tfunc, alanda, 1, 1, epax, fnc, kend1, dyda)
    elseif(kans .eq. 2)then
      call curfit(lisa, at2, asym2, sig2, kend2, paras, ma, mf, covar,
        1  alpha, nca, chisq, tfunc, alanda, 1, 1, epax, fnc, kend2, dyda)
    elseif(kans .eq. 3)then
      call curfit(lisa, at3, asym3, sig3, kend3, paras, ma, mf, covar,
        1  alpha, nca, chisq, tfunc, alanda, 1, 1, epax, fnc, kend3, dyda)
    endif
  elseif(ftype .eq. 's')then
    if(ians .eq. 1)then
```
APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

call curfit(lis, t, data, dsig, nend, paras, ma, mf, covar,
1 alpha, nca, chisq, ffunc, alaada, 1, 1, epar, fnc, nend, dyda)
elseif(ians .eq. 2) then
  call curfit(lis, t2, data2, dsig2, nend2, paras, ma, mf, covar,
1 alpha, nca, chisq, ffunc, alaada, 1, 1, epar, fnc, nend2, dyda)
elseif(ians .eq. 3) then
  call curfit(lis, t3, data3, dsig3, nend3, paras, ma, mf, covar,
1 alpha, nca, chisq, ffunc, alaada, 1, 1, epar, fnc, nend3, dyda)
elseif(ians .eq. 4) then
  call curfit(lis, t4, data4, dsig4, nend4, paras, ma, mf, covar,
1 alpha, nca, chisq, ffunc, alaada, 1, 1, epar, fnc, nend4, dyda)
elseif(ians .eq. 5) then
  call curfit(lis, t5, data5, dsig5, nend5, paras, ma, mf, covar,
1 alpha, nca, chisq, ffunc, alaada, 1, 1, epar, fnc, nend5, dyda)
elseif(ians .eq. 6) then
  call curfit(lis, t6, data6, dsig6, nend6, paras, ma, mf, covar,
1 alpha, nca, chisq, ffunc, alaada, 1, 1, epar, fnc, nend6, dyda)
endif
endif

goto 10

c...Display the data and function.
c...Coerce values to their proper precision.
if(ftype .eq. 'a') then
  if(kans .eq. 1) then
    nend = kendl
    do i = 1, kendl
      y(i) = asym1(i)
      t(i) = at1(i)
      f(i) = fnc(i)
    enddo
  elseif(kans .eq. 2) then
    nend = kend2
    do i = 1, kend2
      y(i) = asym2(i)
      t(i) = at2(i)
      f(i) = fnc(i)
    enddo
APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

```c
elseif (kans .eq. 3) then
    nend = kend3
    do i = 1, kend3
        y(i) = asym3(i)
        t(i) = at3(i)
        f(i) = fnc(i)
    enddo
endif
elseif (ftype .eq. 's') then
    if (iane .eq. l) then
        nend = nendl
        do i = 1, nendl
            y(i) = datal(i)
            t(i) = tl(i)
            f(i) = fnc(i)
        enddo
    elseif (ians .eq. 2) then
        nend = nend2
        do i = 1, nend2
            y(i) = data2(i)
            t(i) = t2(i)
            f(i) = fnc(i)
        enddo
    elseif (ians .eq. 3) then
        nend = nend3
        do i = 1, nend3
            y(i) = data3(i)
            t(i) = t3(i)
            f(i) = fnc(i)
        enddo
    elseif (ians .eq. 4) then
        nend = nend4
        do i = 1, nend4
            y(i) = data4(i)
            t(i) = t4(i)
            f(i) = fnc(i)
        enddo
```
APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

```fortran
elseif(ians .eq. 5) then
  nend=nend5
  do i=1,nend5
    y(i)=data5(i)
    t(i)=t5(i)
    f(i)=fnc(i)
  enddo
elseif(ians .eq. 6) then
  nend=nend6
  do i=1,nend6
    y(i)=data6(i)
    t(i)=t6(i)
    f(i)=fnc(i)
  enddo
endif
endif

call display(t,y,f,nend)
goto 10

c...Clean up dirty data with badbin.
8 print*, 'I have the single histo number as ', ians
print*, 'Input the proper number if not correct:'
read*, ians
if(ians .eq. 1) then
  call badbin(data1,1,nend1)
elseif(ians .eq. 2) then
  call badbin(data2,1,nend2)
elseif(ians .eq. 3) then
  call badbin(data3,1,nend3)
elseif(ians .eq. 4) then
  call badbin(data4,1,nend4)
elseif(ians .eq. 5) then
  call badbin(data5,1,nend5)
elseif(ians .eq. 6) then
  call badbin(data6,1,nend6)
endif
goto 10
```
APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

...This is for the call to the off axis stuff????

9 bkgf=parms(6)
   xn0f=parms(1)
   if(ians .eq. 1)then
      call char(ihis1,it01,it11,it21,data1,xn0f,bkg1,t1,nend1)
   elseif(ians .eq. 2)then
      call char(ihis2,it02,it12,it22,data2,xn0f,bkg2,t2,nend2)
   elseif(ians .eq. 3)then
      call char(ihis3,it03,it13,it23,data3,xn0f,bkg3,t3,nend3)
   elseif(ians .eq. 4)then
      call char(ihis4,it04,it14,it24,data4,xn0f,bkg4,t4,nend4)
   elseif(ians .eq. 5)then
      call char(ihis5,it05,it15,it25,data5,xn0f,bkg5,t5,nend5)
   elseif(ians .eq. 6)then
      call char(ihis6,it06,it16,it26,data6,xn0f,bkg6,t6,nend6)
   endif
   goto 10

...Calculate asymmetry function.

11 if(ians .eq. 0)then
   print*, 'You need to input parameters for the asym function.'
   print*, 'type "ip".'
   goto 10
elseiif(ians .eq. 1)then
   call asym(data1,data2,asym1,sig1,at1,it01,it11,it02,it12,
   1 nend1,nend2,bkg1,bkg2,dt,npk,kend1)
elseif(ians .eq. 2)then
   call asym(data3,data4,asym2,sig2,at2,it03,it13,it04,it14,
   1 nend3,nend4,bkg3,bkg4,dt,npk,kend2)
elseif(ians .eq. 3)then
   call asym(data5,data6,asym3,sig3,at3,it05,it15,it06,it16,
   1 nend5,nend6,bkg5,bkg6,dt,npk,kend3)
endif
   goto 10

...Print function to data file.

12 print*, 'Input data file name: '
   read(5,'(a20)')dfil
   open(9,file=dfil)
print*, 'Single or asym (s/a)?'
read(S, '(a1)') type
if(type .eq. 's') then
  print*, 'Input the spectrum number (1-', mhist, '):'
  read*, ispec
  if(ispec .eq. 1) then
    do i=1,nend1
       write(9,*) ti(i), data1(i)
    enddo
  elseif(ispec .eq. 2) then
    do i=1,nend2
       write(9,*) t2(i), data2(i)
    enddo
  elseif(ispec .eq. 3) then
    do i=1,nend2
       write(9,*) t3(i), data3(i)
    enddo
  elseif(ispec .eq. 4) then
    do i=1,nend4
       write(9,*) t4(i), data4(i)
    enddo
  elseif(ispec .eq. 5) then
    do i=1,nend5
       write(9,*) t5(i), data5(i)
    enddo
  elseif(ispec .eq. 6) then
    do i=1,nend5
       write(9,*) t6(i), data6(i)
    enddo
  endif
elseif(type .eq. 'a') then
  print*, 'Input spec number (1-', mhist/2, '):'
  read*, kspec
  if(kspec .eq. 1) then
    do i=1,kend1
       write(9,*) at(i), asymi(i)
    enddo
APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

```fortran
elseif(kspec .eq. 2)then
  do i=l,kend2
    write(9,*),at2(i),asyn2(i)
  enddo
elseif(kspec .eq. 3)then
  do i=l,kend3
    write(9,*),at3(i),asym3(i)
  enddo
endif
endif
d08e(9)
goto 10

c...Check backgrounds.
13 print*, 'Present spectrum is ', lspec
   print*, 'Input the spectrum number (1-*, mhist,):'
   read*, lspec
   if(lspec .gt. mhist)goto 13
   if(lspec .eq. 1)then
     call bkgck(bkg1,ihis1,it01,lspec,newdat,data1,t1,nend1,npk)
   elseif(lspec .eq. 2)then
     call bkgck(bkg2,ihis2,it02,lspec,newdat,data2,t2,nend2,npk)
   elseif(lspec .eq. 3)then
     call bkgck(bkg3,ihis3,it03,lspec,newdat,data3,t3,nend3,npk)
   elseif(lspec .eq. 4)then
     call bkgck(bkg4,ihis4,it04,lspec,newdat,data4,t4,nend4,npk)
   elseif(lspec .eq. 5)then
     call bkgck(bkg5,ihis5,it05,lspec,newdat,data5,t5,nend5,npk)
   elseif(lspec .eq. 6)then
     call bkgck(bkg6,ihis6,it06,lspec,newdat,data6,t6,nend6,npk)
   endif
goto 10

c...Set up for the 2 freq fits. ftype was used above in ip for single or asym.
14 if(ftype .eq. 's')then
   if(ians .eq. 1)then
     call tfreq(ffunc,data1,t1,lista,dsg1,nend1,parms,ma,mf,
            covar,alpha,nca,chipq,alamda,epar,fnc,dyda,ftype)
```

APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

```
else if (iAns .eq. 2) then
    call tfreq(ffunc, data2, t2, lista, dsig2, nend2, parms, ma, mf,
               covar, alpha, nca, chisq, alambda, epar, fnc, dyda, ftype)
else if (iAns .eq. 3) then
    call tfreq(ffunc, data3, t3, lista, dsig3, nend3, parms, ma, mf,
               covar, alpha, nca, chisq, alambda, epar, fnc, dyda, ftype)
else if (iAns .eq. 4) then
    call tfreq(ffunc, data4, t4, lista, dsig4, nend4, parms, ma, mf,
               covar, alpha, nca, chisq, alambda, epar, fnc, dyda, ftype)
else if (iAns .eq. 5) then
    call tfreq(ffunc, data5, t5, lista, dsig5, nend5, parms, ma, mf,
               covar, alpha, nca, chisq, alambda, epar, fnc, dyda, ftype)
else if (iAns .eq. 6) then
    call tfreq(ffunc, data6, t6, lista, dsig6, nend6, parms, ma, mf,
               covar, alpha, nca, chisq, alambda, epar, fnc, dyda, ftype)
endif
else if (ftype .eq. 'a') then
    if (iAns .eq. 1) then
        call tfreq(tfunc, asym1, at1, lista, sig1, kendl, parms, ma, mf,
                   covar, alpha, nca, chisq, alambda, epar, fnc, dyda, ftype)
    elseif (iAns .eq. 2) then
        call tfreq(tfunc, asym2, at2, lista, sig2, kendl, parms, ma, mf,
                   covar, alpha, nca, chisq, alambda, epar, fnc, dyda, ftype)
    elseif (iAns .eq. 3) then
        call tfreq(tfunc, asym3, at3, lista, sig3, kendl, parms, ma, mf,
                   covar, alpha, nca, chisq, alambda, epar, fnc, dyda, ftype)
endif
endif
```

c...Return to main program.
c
```
```
```
This routine reads the binary data files from TRIUMF and sorts data and headers. This program worked on the IBM PC without the calls to ibmflp, but on the RISC it was necessary to implement a version with the bit in order to get the proper readings of the TRIUMF data. What one needs to do is read in all data as integer*1 and flip the bits before printing them out or using them for fitting.

COMPILING/LINKING NOTE -- There was a problem in reading the first histo header in that the time per bin variable was getting switched correctly in ibmflp but the parameter passage was such that the parameter on this end was thought to be int*4 even though it was declared as int*2.

Compiling with the following options worked:

```
xlf -0 -qnoi4 -o <exec> <other routines> readtri.f
```

```fortran
parameter(IBTOT=20480)
character fil*20,fil2*20
character cttl*40,cscbl*72,coment*144,ctitle*12,cihttl*12,
        icttl2*12
integer*4 jtsc(18),jdsc(18),title,ictask,ievttot,nevttot,nevtot2,
        iiloc
integer*2 mdatel(8),mdate2(8),mlston(4),mlocsc(12),
        m2date(12),mlston(8),mlocsl(24),mlogf(5),mspar2(12),mskew(4),ms(2),junk(18),mbuff(256),
        mrun,mhist,msclr,msupd,msup,d,mscl,ms,mbuff,ms2,ms21,mrun1,msclr1,msupd1,m2date(12),mlocsc(12),
        mlscl(2),msupd2(2),mmskew(4),ms(2),mscnsc(2),nsctot(2),msth(4),
        ntot2,nt2,nto,ntt2,ntb1,ntb2,ntb3,ntb4,
        khist
```

RISC fortran supports int*1, otherwise need character.

```fortran
integer*1 ii(iibtot),mskew(4),ms(2),m2date(12),mlston(8),
        mlocsc(24),mlogf(10),mspari(24),mskew2(4),ms(2),junk1(32),
        mbuff1(512),mskew(4),mskew(32),ms21(2),msruni(2),msth(2),
        mlocsc(2),msupd2(2),mmskew(4),ms(2),mscnsc(2),msctot(2),
        nlocsc(2),nlocsc(2),mstot(2),nlocsc(2),mm21(2),ms21(2)
```

equivalence (nevttot,ms(1))
equivalence (nevttot,ms2(1))

print*, 'Input TRIUMF binary file name: '

read(5,'(a20)',err=1) fil

Open the binary file. History seems to show that they use 512 bytes as
the record length, although it may actually be 256 or less.

open(9, file=fil, err=9999, access='direct',
     form='unformatted', recl=512, status='old')

Read in the first record's worth of stuff. The file header is 512 bytes
in length, divided as indicated in the read below.

irec=1
write(6,5)
5  format(' Reading file header.','/)

read(9, rec=irec) mrun1, mhist1, msclrl, msupd1, jtscl, jdscl, mmini, mscli,
mdate, m2date, mlstol, mcmcs1, mlocsi, mrsatl, ictask, mlogfl,
uic, msparl, cttl, cscbl, coment

call ibmflp(mrun1, 1, 1, mrun)
call ibmflp(mhist1, 1, 1, mhist)
call ibmflp(msclrl, 1, 1, msclr)
call ibmflp(msupd1, 1, 1, msupd)
call ibmflp(mmini, 1, 1, mmini)
call ibmflp(msecli, 1, 1, msec)
call ibmflp(mdate, 1, 6, mdatel)
call ibmflp(m2date, 1, 6, mdate2)
call ibmflp(mlstol, 1, 6, mlston)
call ibmflp(mcmcs1, 1, 1, mcmcs)
call ibmflp(mlocsi, 1, 12, mlocsc)
call ibmflp(mrsatl, 1, 1, mrsst)
call ibmflp(ictask, 1, 1, ictask)
call ibmflp(mlogfl, 1, 5, mlogf)
call ibmflp(uic, 1, 1, uic)
call ibmflp(msparl, 1, 1, 12, mspare)
write(6,10)mrun, mhist, mdate1(2), mdate1(3), mdate1(1), mdate1(4), mdate5(2), mdate5(3), mdate5(1), mdate5(4),
mdate2(5), mdate2(8), cttl, cscbl, coment
10  format(/' Run ',i5,' ; # of histograms ',i3,/, date',/,
              1 2(ix, i2, ',', i2, ',', i2, 2x, ',', i2, ',', i2, ',', i2, ',', i2, ',', i40, ',', i40, ',',
              2 a144,/)"
c...Loop over the number of histograms in the file defined by mhist.
do 666 khist=1,mhist
APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

irec=irec+1

print*, 'In khist loop, khist,mhist,irec= ',khist,mhist,irec

...Read in the first histogram header -- 64 bytes in length seems to be
...standard.
write(6,15) khist
15 format(' Reading histogram ',i3,' header.','/)
read(9,rec=irec) ihist1,lengt1,ms11,ms12,duhduh,maskev,nt01,nti1,nt21,ctitle,junk1

...How do the switches.
call ibmflp(ihist1,1,1,ihist)
call ibmflp(lengt1,1,1,length)
call ibmflp(ms11,1,1,ms(1))
call ibmflp(ms12,1,1,ms(2))
call ibmflp(duhduh,1,1,itpb)
call ibmflp(maskev,1,2,maskev)
call ibmflp(nt01,1,1,nt0)
call ibmflp(nti1,1,1,nt1)
call ibmflp(nt21,1,1,nt2)
call ibmflp(junk1,1,16,junk)
write(6,20) ihist,length,nevtot,itpb,maskev,nt0,nt1,nt2,
1 c title
20 format( ' Histogram: ',i3,' length:',i8,' total events:',i8,/,1 ' time/bin: ' , i4 , ' masks: ', z4 , 'H',2x,z4,'H',/,' t0: ' , i4 , ' t1: ' ,
2 i4 , ' t2: ' , i6 ,'/',ix,al2,/)

...Assign the proper histo variables to their parameter values.
if(ihist .eq.1)then
i00=nt0
iti1=nt1
iti2=nt2
itpb1=itpb
elseif(ihist .eq.2)then
i02=nt0
iti2=nt1
iti2=nt2
itpb2=itpb
elseif(ihist .eq.3)then
i03=nt0
APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

it13=nt1
it23=nt2
itpb3=itpb

elseif(ihist .eq.4)then
it04=nt0
it14=nt1
it24=nt2
itpb4=itpb

elseif(ihist .eq.6)then
it06=nt0
it16=nt1
it26=nt2
itpb6=itpb

else
print*, 'Histogram number ',ihist,'exceeds program','
1 bounds -- oops!'
endif

...Due to some FORTRAN or IBM problem the rest of the record could
...not be read in--only ist 128 bytes. Why? Good question.
...Re-read as all integer and take what want.
read(9,rec=irec)mbuff1

...How need to figure out how many more blocks need be read. The first block
...is the file header (512 bytes). The second block (512) is composed of both
...the histogram header (64 bytes) and the beginning of the data (512-64=448).
...It seems that the histo headers/data always occupy a certain number of
...blocks, whose number is given by length/256 (recall that the first has
c...been read in already).
   nblk=length/256
   if(nblk .gt. IBTQT/512)then
      print*, 'Too much data for present arrays, fix program.'
      return
   endif
   do 40 i=1,nblk
   c...How and for all subsequent reads into ihll mult all indices by 2. Record
   c...stuff should be the same. Recall ihloc is now the byte count, not byte
   c...pairs.
      irec=irec+1
      read(9,rec=irec)mbuff1
      do 50 j=1,256+2
         ihloc=224*2+(i-1)*256*2+j
         ihll(ihloc)=mbuff1(j)
      c...Check if at end of the histogram.
      if(ihloc .eq. length*2)then
         goto 55
      endif
      50 continue
   40 continue
   c...Transfer the histogram (defined now by length) to the proper (int*2).
   55 if(ihist .eq.1)then
      call ibmflp(ihll,1,length,ihis1)
   elseif(ihist .eq.2)then
      call ibmflp(ihll,1,length,ihis2)
   elseif(ihist .eq.3)then
      call ibmflp(ihll,1,length,ihis3)
   elseif(ihist .eq.4)then
      call ibmflp(ihll,1,length,ihis4)
   elseif(ihist .eq.5)then
      call ibmflp(ihll,1,length,ihis5)
   elseif(ihist .eq.6)then
      call ibmflp(ihll,1,length,ihis6)
   else
      print*, 'The histogram data is also out of bounds . . .'
   endif
APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

666 continue

C... Check that all data has the same time scale (anal?).

if((itpbl.ne.itpb2).or.(itpb3.ne.itpb4))then
   print*, 'The data time scales are different -- this sucks!'
endif

C... NOTE -- THESE ARE THE TRIUMF TIME MAPPINGS -- THEY MAKE NO SENSE.

elseif(itpbl.eq.3)then
   dt= .0625e-2
elseif(itpbl.eq.4)then
   dt= .125e-2
elseif(itpbl.eq.5)then
   dt= .25e-2
else
   print*, 'Something probably wrong with the read, itpb= ',itpbl
endif
goto 1000

9999 print*, 'Error opening file ',fil
goto 1

1000 return

end

subroutine ibmflp(iarr1,nstart,nflips,iarr2)

C*****************************************************************************
C... This routine was written 8/18/93 by A. Greer to compensate for the IBM
C... internal representation of reversing the bits for integers. It takes an
C... integer*1 array and flips them while equivalencing them to an integer*2
C... variable. Note that if int*1 is not supported then character may be used.
C*****************************************************************************

integer*1 iarr1(1),itemp,iarr1(40960)
integer*2 iarr2(1),itarr2(20480)
equivalence(itarr1,itarr2)

C print*, 'In ibmflp, nstart, nflips= ',nstart, nflips
C print*, 'iarr1 as passed in is ',iarr1
do i=nstart,(nstart-1)+2*nflips
   itarr1(i)=iarr1(i)
endo

C print*, 'Before, itarr1(1),itarr1(2)= ',itarr1(1),itarr1(2)
C... This loop actually does the switching of the bytes using temp.
do i=nstart,(nstart-l)+2*nflips,2
  itemp=itarr1(i)
  itarr1(i)=itarr1(i+1)
  itarr1(i+1)=itemp
enddo
  print*, 'After, itarr1(l), itarr1(2)= ', itarr1(l), itarr1(2)
  c... This loop transfers the data to the parameter for passage to the outside
  c... world — this seems intermittent at the moment....
  do i=nstart/2+1,nstart/2+nflips
    iarr2(i)=itarr2(i)
  enddo
  print*, 'itarr2(1), iarr2(1)= ', itarr2(1), iarr2(1)
  return
end

subroutine datcre(ihis1, ihis2, ihis3, ihis4, ihis5, ihis6, data1,
  1 data2, data3, data4, data5, data6, ti, t2, t3, t4, t5, t6, dsig1, dsig2,
  2 dsig3, dsig4, dsig5, dsig6, it01, it11, it21, it02, it12, it22, it03,
  3 it13, it23, it04, it14, it24, it05, it15, it25, it06, it16, it26, dt,
  4 nendl, nend2, nend3, nend4, nend5, nend6, bkg1, bkg2, bkg3, bkg4, bkg5,
  5 bkg6, npk)
  c******************************************************************************
  c... This routine written 9/93 by A. Greer for the TRIUMF data analysis. It
  c... accepts the data in ihis1 - ihis6 and the corresponding time info in it0x -
  c... it2x, and dt as was read in in datrd. It calculates the proper time and
  c... data arrays and takes into account any packing that the user wishes to use.
  c... These data arrays are then used to calculate the asymmetry functions, which
  c... use the automatically calculated backgrounds and alpha parameters.
  c
  c... Note change on 11/8 to take care of the high statistics overflow problem
  c... for the TRIUMF data. For int*2 data the cutoff is 65535, after which
  c... the numbers go negative. Taken care of below.
  c******************************************************************************
  parameter (CUT=65535.0)
  integer*2 ihis1(l), ihis2(l), ihis3(l), ihis4(l), ihis5(l), ihis6(l)
  dimension data1(l), data2(l), data3(l), data4(l), data5(l), data6(l)
  dimension ti(l), t2(l), t3(l), t4(l), t5(l), t6(l), dsig1(l), dsig2(l),
  dsig3(l), dsig4(l), dsig5(l), dsig6(l), it01(l), it11(l), it21(l), it02(l), it12(l), it22(l), it03(l),
  it13(l), it23(l), it04(l), it14(l), it24(l), it05(l), it15(l), it25(l), it06(l), it16(l), it26(l), dt,
  nendl(l), nend2(l), nend3(l), nend4(l), nend5(l), nend6(l), bkg1(l), bkg2(l), bkg3(l), bkg4(l), bkg5(l),
  bkg6(l), npk(l)
APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

1  dsig3(l),dsig4(l),dsig5(l),dsig6(l)
   integer*4 nend

   c...Query about packing.
   10    print*, 'Input the packing factor: '
       read*, npk

   c...Use the good time and end info for creating the data.
   c...The number to create depends on packing.
   c...Also need to see if have an even # of pts, so do the following.
   itst=it11-(it11/npk)*npk
   nend1=it21/npk
   do i=1, nend1
     data1(i)=0.
     do j=itst, itst+npk-1
       data1(i)=data1(i)+ihis1(j+npk*(i-l)+l+it11)
     enddo
     dsig1(i)=sqrt(1.+abs(data1(i)) )
     t1(i)=(it11-it01+(i-l)*npk)*dt
   enddo
   itst=it12-(it12/npk)*npk
   nend2=it22/npk
   do i=1, nend2
     data2(i)=0.
     do j=itst, itst+npk-1
       data2(i)=data2(i)+ihis2(j+npk*(i-l)+l+it12)
     enddo
     dsig2(i)=sqrt(1.+abs(data2(i)) )
     t2(i)=(it12-it02+(i-l)*npk)*dt
   enddo
   itst=it13-(it13/npk)*npk
   nend3=it23/npk
   do i=1, nend3
     data3(i)=0.
     do j=itst, itst+npk-1
       data3(i)=data3(i)+ihis3(j+npk*(i-l)+l+it13)
     enddo
     dsig3(i)=sqrt(1.+abs(data3(i)) )
     t3(i)=(it13-it03+(i-l)*npk)*dt
   enddo
APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

```plaintext
enddo
itst=it14-(it14/npk)*npk
nend4=it24/npk
do i=1,nend4
  data4(i)=0.
  do j=itst,itst+npk-1
    data4(i)=data4(i)+ihia4(j+npk*(i-1)+1+it14)
  enddo
  dsig4(i)=sqrt(1.+abs(data4(i)))
  t4(i)=(it14-it04+(i-1)*npk)*dt
enddo
itst=it15-(it15/npk)*npk
nend5=it25/npk
do i=1,nend5
  data5(i)=0.
  do j=itst,itst+npk-1
    data5(i)=data5(i)+ihia5(j+npk*(i-1)+1+it15)
  enddo
  dsig5(i)=sqrt(1.+abs(data5(i)))
  t5(i)=(it15-it05+(i-1)*npk)*dt
enddo
itst=it16-(it16/npk)*npk
nend6=it26/npk
do i=1,nend6
  data6(i)=0.
  do j=itst,itst+npk-1
    data6(i)=data6(i)+ihia6(j+npk*(i-1)+1+it16)
  enddo
  dsig6(i)=sqrt(1.+abs(data6(i)))
  t6(i)=(it16-it06+(i-1)*npk)*dt
enddo

C... Now find where the zeroes end at the end of the data set -- can cause fit
C... problems if too many.
  do i=nend1,1,-1
    if(data1(i).ne.0.)goto 111
  enddo
111 nend1=i
```

APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

do i=nend2,l,-l
   if(data2(i).ne.0.)goto 112
endo
112 nend2=i
   do i=nend3,1,-1
      if(data3(i).ne.0.)goto 113
   enddo
113 nend3=i
   do i=nend4,1,-1
      if(data4(i).ne.0.)goto 114
   enddo
114 nend4=i
   do i=nend5,1,-1
      if(data5(i).ne.0.)goto 115
   enddo
115 nend5=i
   do i=nend6,1,-1
      if(data6(i).ne.0.)goto 116
   enddo
116 nend6=i

c... Need to check for the overflow possibility. This should only be a problem
with the first couple of data sets.
do i=1,nend1
   if(data1(i).lt.0.)data1(i)=CUT+CUT+data1(i)
   if(data2(i).lt.0.)data2(i)=CUT+CUT+data2(i)
endo
c... Find the backgrounds for each of these data sets.
do i=20,it01-10
   if(ihis1(i).gt.0)then
      ifirst=i
      goto 1111
   endif
endo
1111 do i=ifirst+10,it01-10
   sum1=sum1+ihis1(i)
   num1=num1+1
endo
do i=20,it02-10
  if(ihis2(i).gt.0)then
    ifirst=i
    goto 222
  endif
enddo
222 do i=ifirst+10,it02-10
  sum2=sum2+ihis2(i)
  num2=num2+1
enddo
do i=20,it03-10
  if(ihis3(i).gt.0)then
    ifirst=i
    goto 333
  endif
enddo
333 do i=ifirst+10,it03-10
  sum3=sum3+ihis3(i)
  num3=num3+1
enddo
do i=20,it04-10
  if(ihis4(i).gt.0)then
    ifirst=i
    goto 444
  endif
enddo
444 do i=ifirst+10,it04-10
  sum4=sum4+ihis4(i)
  num4=num4+1
enddo
do i=20,it05-10
  if(ihis5(i).gt.0)then
    ifirst=i
    goto 555
  endif
enddo
555 do i=ifirst+10,it05-10
APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

\[ \text{sum5} = \text{sum5} + \text{ihis6}(i) \]
\[ \text{num5} = \text{num5} + 1 \]
enddo
do i=20,IT06-10
  if(ihis6(i).gt.0)then
    ifirst=i
    goto 666
  endif
enddo
666 do i=ifirst+10,IT06-10
  sum6=sum6+ihis6(i)
  num6=num6+1
enddo
bkg1=sum1/real(num1)
bkg2=sum2/real(num2)
bkg3=sum3/real(num3)
bkg4=sum4/real(num4)
bkg5=sum5/real(num5)
bkg6=sum6/real(num6)
c...reset these to zero.
  sum1=0
  num1=0
  sum2=0
  num2=0
  sum3=0
  num3=0
  sum4=0
  num4=0
  sum5=0
  num5=0
  sum6=0
  num6=0
return
end

subroutine asym2(dat1,dat2,asy2,sig1,time1,t1,t2,t3,t4,t5,t6,t7,t8,t9,t10,
APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

1  nendl,nend2,bkg1,bkg2,dt,npk,kend)
c==================================================================================================
c... Takes the single histo data from sets 1 and 2 and creates the asymmetry
c... function. Time zero and end time parameters are furnished, as are the
c... two backgrounds.
c==================================================================================================

dimension dat1(l),dat2(l),asym(l),sig(l),time(l)
data alpha/1./
print*, 'Alpha = ',alpha,' Input your value: 'read(5,*),alpha

c... Now coerce data into the form (1-a2)/(1+a2) for fitting. Find min length

c... array and use that length.
222 imin=min(0(nendl,nend2)
jmax=max(0(it11-it01,it12-it02)
kend=imin
do i=1,kend
   time(i)=(jmax+(i-1)*npk)*dt
   f=dat1(i)-bkg1*npk
   b=dat2(i)-bkg2*npk
   xnumr=f-alpha*b
   xdenm=f+alpha*b
   asym(i)=xnumr/xdenm
   sigf2=f+2.*npk*bkg1
   sigb2=b+2.*npk*bkg2
   sig(i)=sqrt((2.*alpha/xdenm**2)**2*(b**2*sigf2+f**2*sigb2))
endo
doto
return
end

subroutine badbin(data,ifst,ilst)
c******************************************************************************
c This routine is adapted from the mid-80's PC-based version. It now is a

c... part of the fitting program and receives the data as a parameter, as

c... opposed to having to read in the data as before. Other than that the code

c... is the same, except that I have changed most math from integer to real.
c... A. Greer (9/93).
c******************************************************************************
APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

dimension data(8192),ksave(10)
character df*16,tl(5)*60,ans*16,tans*l,respl0*l
nsig=4
c write(8,*)(data(i),i=1,25)
print*, 'The start of good data and end of data are in channels:'
print*,ifst,ilst
620 print *, 'Enter first, last channel, nsig for first spectrum'
read *,ifst,ilst,nsig

c...This is my addition (A. Greer). Take the first 10 channels and avg, then

c...check which of the first 10 exceeds the nsig*sdavg barrier and correct

c...those, then do the usual stuff.
666 print*, 'Checking first 10 data points . . . '
do k=1,10
    sum=sum+abs(data(k))
enddo
avg10=sum/10.
sd10=nsig*sqrt(avg10)
do k=1,10
    diff10=abs(data(k)-avg10)
    if(diff10.gt.sd10)then
        print*, 'At bin ',k,' the data is ',data(k)
        print*, 'The avg of 1st 10 is ',avg10
        print*, 'Change its value to avg+sqrt(avg)*nsig?'
        read(5,'(al)')respl0
        if(respl0.eq.'y')data(k)=avg10+sqrt(avg10)*nsig
    endif
enddo

c check first three channels to see if they are ok.
print*, 'Now doing normal checks . . . '
622 ifst=ifst+1
    x=abs(data(ifst))
    sd=nsig*sqrt(x)
    diff=abs(data(ifst)-data(ifst1))
    print*, 'Checking 1st, diff, sd = ',diff,sd
    if(diff.gt.sd) then
        c...Check third value.
APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

\[ x_2 = \text{abs}(\text{data}(\text{ifst}1)) \]
\[ s_2 = \text{nsig} \times \text{sqrt}(x_2) \]
\[ \text{diff}_2 = \text{abs}(\text{data}(\text{ifst}1) - \text{data}(\text{ifst}1+1)) \]

\begin{verbatim}
if(\text{diff}_2 \gt \text{sd}_2) then
    print*, 'There are problems with the data, display to see.'
    print*, 'diff2, sd2 = ', diff2, sd2
else
    c...The first data point is screwed up.
    avgtmp = (\text{data}(\text{ifst}1) + \text{data}(\text{ifst}1+1))/2.
    print*, 'Data in channels 1,2,3 = ', \text{data}(\text{ifst}), \text{data}(\text{ifst}1),
          \text{data}(\text{ifst}1+1)
    print*, 'Changing ', \text{data}(\text{ifst}), ' to ', avgtmp
    \text{data}(\text{ifst}) = \text{avgtmp}
endif
print*, 'Start over? (y/n)'
read(5, '(a1)') \text{tans}
if(\text{tans} .eq. 'y') goto 620
endif
\end{verbatim}

ifst2 = ifst2+2

\begin{verbatim}
c start check for other channels
    do 711 i = ifst2, ilst
        i2 = i-2
        i1 = i-1
        avg = (\text{data}(i1) + \text{data}(i2))/2.
        x = \text{abs}(avg)
        sd = \text{nsig} \times \text{sqrt}(x)
        \text{diff} = \text{abs}(\text{data}(i) - avg)
        c if there is a glitch, see if the following channel has a glitch,
        c if so fix glitch
        c if there is a second glitch, stop execution
        if(\text{diff} \gt \text{sd}) then
            ip1 = i+1
            \text{diff}1 = \text{abs}(\text{data}(ip1) - avg)
            if(\text{diff}1 \gt \text{sd}) then
                print *, 'Two consective channels with glitches'
                print *, 'Previous two channels, data, and avg are ', i2, i1,
                \text{data}(i2), \text{data}(i1), \text{avg}
            endif
        endif
    enddo 711
\end{verbatim}
print *, 'data in channels ', i, ', ipl', ' is ', data(i), data(ipl)
print *, 'Do you wish to go ahead anyway?'
read(5, 'a1') ans
if (ans.eq. 'n') goto 999
endif
avg2 = (data(i1) + data(ipl)) / 2.
print *, 'Bin ', i, ' is found to have a glitch'
print *, 'Do you wish to change the value from ', data(i),
1       ' to ', avg2, '?'
read(5, 'a1') ans
if (ans.eq. 'y') then
    data(i) = avg2
endif
endif
711 continue
999 return
end

subroutine ffunc(t, a, y, dyda, na)
C...The function looks like:
C...
C... y = a1 * [exp(-t/tm) *(1 + a2 * exp((-t*a3)**2/2) *
C... cos(a4*t + a5)) + a6
C...
C...Check that none of the bases are less than zero.
    if(a3 .lt. 0.0)then
        a3 = abs(a3)
    endif
parameter(TM=2.19, DEMAX=709.78271)
dimension dyda(na), a(na)
double precision arg, earg
front = a(1)*exp(-t/TM)
csl = cos(a(4)*t + a(5))
cni = sin(a(4)*t + a(5))
C...
APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

if(a(4) .lt. 0.0) then
a(4)=abs(a(4))
print*, ' a(4) < 0, set=pos,'
endif
arg=-(t*a(3))**2/2
check that the value will not cause an error.
if(arg .ge. DEMAX) arg=709.0
earg=dexp(arg)
derivatives.
dyda(1)=front/a(1)*(1+a(2)*earg*csl)
dyda(2)=front*earg*csl
dyda(3)=-front*a(2)*earg*csl*a(3)*(t**2)
dyda(4)=-front*a(2)*earg*sn1*t
dyda(5)=-front*a(2)*earg*sn1
dyda(6)=1.
function.
y=a(1)*(front/a(1)*(1+a(2)*earg*csl)) + a(6)
return
end

subroutine display(t,y,x,npts)
****************************************************************************
This routine was written 8/93 by A. Greer to replace the GRAFMATIC PC-based
calls for the fitting routines. It plots out the data (and fit function)
on the same graph in different colors. Note that the arrays passed are
double precision -- this is due to Chree Haas building things this way.
The compilation goes something like:

xlf -o <> -L /usr/local/plplotd -I/usr/local/plplot/include
-1 plplot -lx11 -lm

Note that the parameters t,y,x are passed in as double precision and that
npts is int*4.
*****************************************************************************
real*8 t(i),y(i),x(i),ymax,ymin,tmin,tmax,tl,ty,yl,yy
integer*4 just,axis,nx,ny,npts,code,green,yellow,cyan,
salmon,plt
real*8 tplot(16384),yplot(16384),xplot(16384)
c print*, 'In display.'
c write(*,*) (t(i), i=1, 25)
c write(*,*) (x(i), i=1, 25)
c...These need to be declared as variables for the plotting routines which
c...expect int*4 variables.
    just=0
    axis=0
    nx=1
    ny=1
c print*, 'npts= ', npts
code=1
green=3
yellow=2
cyan=11
salmon=14
c...Get upper and lower limits on the data.
tmin=t(1)
tmax=t(1)
ymin=y(1)
ymax=y(1)
do i=2, npts
    if(t(i).gt.tmax)tmax=t(i)
    if(y(i).gt.ymax)ymax=y(i)
    if(y(i).lt.ymin)ymin=y(i)
enddo
write(6,104) tmin, ymin, tmax, ymax
104 format('  The plotting limits are: ',/,' tl= ',f8.2,8x,' yl= ',f8.2,8x,
            ' tu= ',f8.2,8x,' yu= ',f8.2,8x)
print*, 'Input plotting parameters tl, tu, yl, yu: '
read*, tl, tu, yl, yu
if((tl.eq.tu).or.(yl.eq.yu)) then
tl=tmin
tu = tmax + 1
yl = ymin
yu = ymax + 1
else if ((tl .lt. tmin) .and. (tu .gt. tmax)) then
  trange = tmax - tmin
else if (tl .lt. tmin) then
  trange = tu - tmin
else if (tu .gt. tmax) then
  trange = tmax - tl
else
  trange = tu - tl
endif

C ... Need to calculate the actual number of points to plot for the routines
C ... given the user's time range.
dt = t(10) - t(9)
nplt = trange / dt

C ... Need to get the proper starting point for the plot based on the user's
C ... input -- also shift the data for passing.
if (tl .lt. t(1)) then
  n2 = 1
  ni = 1
else
  n2 = tl / dt
  ni = t(1) / dt
endif
if (nplt .gt. npts) nplt = npts
nstart = n2 - ni
do i = 1, nplt
  tplot(i) = t(i + nstart)
  yplot(i) = y(i + nstart)
  xplot(i) = x(i + nstart)
endo

C ... Now call the plotting routine.
call plstart("xwin", nx, ny)
call plenv(tl, tu, yl, yu, just, axis)
call plcol(yellow)
call pllab('Time (#gmsec)', 'N(t)', '')
APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

```fortran
  call plcol(green)
call plpoin(nplt,tplot,yplot,code)
call plcol(cyan)
call ppline(nplt,tplot,xplot)
call plend()
return
end

subroutine parin(lbl,parms,ma)
  c******************************************************************************
c...This routine allows the user to input the parameters for the fitting fun.
c******************************************************************************
  character lbl(10)*7
  dimension parms(10)
  print *,(lbl(i),i=1,ma)
  print 303
  print 302,(parms(i),i=1,ma)
  301 print *,' ? ',
  read(*,*,err=301)(parms(i),i=1,ma)
  print 302,(parms(i),i=1,ma)
  303 format(' Asy(xn) Lamba(MHz)(pa) Power(svv) ',
           ' v(MHz) Phi Bgr etc. ')
  302 format(lx,al2.5,18.4,f10.4,110.4,18.3,18.4,416.2/)
return
end

subroutine prfun(t,y,nmax)
  c******************************************************************************
c...This function prints out the data for viewing on the screen or for plotting
c...to a file.
c******************************************************************************
  dimension t(1),y(1)
  character dat1*20,ans*1
  711 print *,' ? ',
  read(*,*,err=711)nf,nl
  print *,' ? ',nf,nl
  print 701,(y(i),i=nf,nl)
```
APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

701 format(1x,5f10.3)
    print *, 'Print data to disk?'
    read(5, '(a1)') ans
    if(ans .eq. 'y') then

710 print *, 'File name?'
    read(5, '(a20)', err=710) daf1
712 print *, 'Input starting and ending times for data. '
    read(5, *, err=712) strtim, endtim
    open(9, file=daf1, err=799)
    do 720 i = 1, nmax
        if ((t(i) .ge. strtim) .and. (t(i) .le. endtim)) then
            write(9, 730) t(i), y(i)
730 format(2x, f10.4, 3x, f10.4)
        endif
    720 continue
    close(9)
    endif
    goto 790
799 print*, 'Open error for file ', daf1
790 return
end

subroutine curfit(lista, t, y, sig, npts, parms, ma, mf, covar, alpha, nca,
    1  chisq, ffunc, alambda, nfp, nfirst, epar, fnc, nmax, dyda)

c...This routine calls the curve fitting routines of NUMA and does a non-linear
  c...least squares fit.
  c
  c
  c
  c
dimension lista(l), covar(l,1), alpha(1,1), sig(1), parms(1),
  1  t(1), y(1), epar(1), fnc(1), dyda(1)
  external ffunc
  print*, 'In curfit, npts=', npts
  write(6, *) (t(i), i = 1, 26)
  write(6, *) (fnc(i), i = 1, 26)
  809 print *, 'mf=', mf, (the first mf of lista are fit)= ',
    read(*, *, err=809) mf
  print*, 'Input the parameters for lista: '
read(5,*)(lista(i),i=l,ma)

c  print*,,'mf, lista are: ',mf,(lista(i),i=l,ma)
  alambda=-1.
801 call mrqmin(t(nfp),y(nfp),sig(nfp),npts,parms,ma,
  1 lista,mf,covar,alpha,nca,chisq,ffunc,alambda)
  chisqn=chisq/(npts-mf)
  print*,,' chisqn, alambda5 ',chisqn,alambda
811 print*,,' If good enough enter 0 for alambda now '
  read(*,*,err=811) alambda
  if(alambda.ne.0.)goto 801
  call mrqmin(t(nfirst),y(nfirst),sig(nfirst),npts,parms,ma,
  1 lista,mf,covar,alpha,nca,chisq,ffunc,alambda)
  chisqn=chisq/(npts-mf)
  print*,,'chisqn= ',chisqn
  print*,,'
  Parameters are:'
  write(6,*)(parns(i),i=l,na)
  print*,,'Errors are:'
  write(6,*)(sqrt(covar(i,i)),i=l,na)
  do 804 i=l,10
  804 epar(i)=sqrt(covar(i,i))
  do 803 i=l,nnar
    fnc(i)=0.
    if(i.ge.nfp.and.i.1e.nfp+npts)then
      call ffunc(t(i),parms,fnc(i),dyda,10)
    endif
  803 continue
  return
end

subroutine parwrt(parms,epar)
  c**************************************************************************
  c...This routine written by A. Greer around 1991-ish to facilitate the plotting
  c...of data. It is a bit cumbersome since one must recall the record number,
  c...but it can save time.
  c**************************************************************************
  character filnaa*20,resp*1
dimension parms(1), epax(l)
1500 if(numrec .eq. 0) then
    write(6,1501)
1501 format(' Input the name of the file to which you wish',/,
1' to write the parameters.',/)
    read(5,1506,err=1500)filnam
1505 format(a12)
    numrec=1
endif
write(6,1506)numrec
1506 format(' The current record is ',i3,'. If you wish to',/,
1' change it, input y (type y for new file).'/)
read(5,1508)resp
1508 format(a1)
if(resp .eq. ' y') then
1530 print*,' Input the value for the record number, ',
1' a 0 will allow a new file to be started.'
    read(5,*),err=1530)numrec
    if(numrec .eq. 0) goto 1500
endif
1520 write(6,1525)
1525 format(' Input the run # and temperature.',/)
    read(5,*),err=1520)irun,tempat
1559 write(6,1560)
1560 format(' Input the numbers of the two parameters for printing',/,
1' to the file (i.e. 1 = a(1)).',/)
    read(5,*),err=1559)iparl, ipar2
1540 open(9,iostat=ios, err=1599,file=filnam, form='formatted',
1 access=' direct ', recl=60)
    write(9,1550, rec=numrec)irun,tempat, parms(iparl), epars(iparl),
1 parms(ipar2), epars(ipar2)
1550 format(1x,i5, 5(2x,f8.4))
    close(9)
    numrec=numrec+1
    goto 1598
1599 print*, ' Open error in file ', filnam
1598 return
This function is for the TRIUMF data fitter. It is a function to fit the asymmetry, looking like:

\[ tfunc = a_1 \cdot \exp\left(-a_2 t^2/2\right) \cdot \cos(a_3 t + a_5) \]

---

```fortran
subroutine tfunc(t, a, y, dyda, na)
  dimension dyda(na), a(na)
  arg = a(2) * t
  earg = exp(-arg**2/2)
  cs = cos(a(3)*t + a(4))
  sn = sin(a(3)*t + a(4))
  y = a(1) * earg * cs
  dyda(1) = earg * cs
  dyda(2) = -a(2) * t * earg * cs
  dyda(3) = -a(1) * earg * sn
  dyda(4) = -a(1) * earg * sn
  return
end
```

---

This routine is for trying to fit Tanya's data with my analysis techniques.
We do one histo at a time, finding the background from the initial level and initial amplitude from fits and extract the relaxation function from this, via a mult. by \( \exp(t/\tau_{amu}) \). We must be careful not to do the background subtraction wrong because there may be moments of the field distribution causing DC levels in the relaxation function. BEWARE.

```fortran
parameter(\tau_{amu}=2.19, LEM=5000)
dimension ihis(1), data(1), at(1), datnew(LEM)
double precision t(LEM), y(LEM), x(LEM)
integer*4 npts
cchar=1 resp
10 print*, 'The bkg is ', bkgc, ' Input your value.'
```
APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

read*,bkgc
do i=1,kend
   datnew(i)=(data(i)-bkgc)*exp(at(i)/taumu)/xn0f-1.
endo
c...Display for now to see if is ok.
do i=1,kend
   t(i)=dble(at(i))
   y(i)=dble(datnew(i))
   x(i)=0.d0
endo
npts=kend
call display(t,y,x,npts,0)
print*,'Again? (y/n)' read(5,'al') resp
if(resp.eq.'y')then
goto 10
else
   return
endif
end

subroutine bkgck(bkg,ihis, it0,numhis,newdat,data,time,nend,npk)
c***********************************************************************
c...This routine will show the previous bkg level and display the points
c...to check if the level is on the level.
c***********************************************************************
parameter(ISIZE=5000)
dimension ihis(l),iup(l),ilow(l),data(l),time(l)
real*8 y(ISIZE),x(ISIZE),dummy(ISIZE)
integer*4 npts
character resp(2),resp1(1),fill(20)
logical newdat
it00=it0
if(it00.gt.ISIZE)then
   print*, 'ISIZE is too small for it0, setting it0=ISIZE'
it00=ISIZE
endif
APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

if(newdat)then
  do i=1,6
    iup(i)=0
    ilow(i)=0
  enddo
  newdat=.false.
endif

c...Need to find the first real non-zero bkg point.
nzero=20
  do i=20,it00
    if(ihis(i).eq.0)then
      nzero=nzero+1
    else
      goto 10
    endif
  enddo

10 if(iup(numhis) .eq. 0)then
  iup(numhis)=it00-10
  ilow(numhis)=nzero+10
endif

c...Need to allow the user to change the range over which the bkg is calculated
35 print*,'Pick an option:'
print*,'vi = view background bounds'
print*,'re = reset background bounds'
print*,'in = input a background level'
print*,'ln = plot data as semi-log to see linearity'
print*,'pl = print ln data to file'
print*,'eb = exit background stuff'
read(5,'(a2)')resp
if(resp .eq. 'vi')then
  goto 20
elseif(resp .eq. 're')then
  print*,'Present values are',ilow(numhis),' and ',iup(numhis)
  print*,'Input the lower and upper bins for bkg: '
  read*,ilow(numhis),iup(numhis)
  do i=ilow(numhis),iup(numhis)
APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

```plaintext
sum=sum+ihis(i)
endo
bkg=sum/real(iup(numhis)-ilow(numhis))
sum=0.
goto 20
elseif(resp .eq. 'in')then
  print*, 'Bkg is ',bkg,' Input bkg: '
  read*,bkg
goto 35
elseif(resp .eq. 'ln')then
do i=l,nend
  x(i)=dble(time(i))
  arg=data(i)-bkg*npk
  if(arg .le. 0.)then
    print*, 'At i= ',i,' found a neg arg= ',arg
    arg=1.
  endif
  y(i)=dble(alog(arg))
  dummy(i)=0.d0
endo
npts=nend
call display(x,y,dummy,npts)
elseif(resp .eq. 'pl')then
  print*, 'Input file name:'
  read(5,'(a20)')fill
  open(9,file=fill)
do i=l,nend
  write(9,*)(time(i),y(i))
endo
close(9)
elseif(resp .eq. 'eb')then
  goto 45
endif
goto 35
20 do i=l,itOO
  y(i)=dble(ihis(i))
x(i)=dble(i)
```

APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

```fortran
dummy(i)=0.
enddo
dummy(ilow(numhis))=bkg
dummy(iup(numhis))=dummy(ilow(numhis))
nts=it00
print*, 'The present bkg level is:', bkg
call display(x,y,dummy,nts)
goto 35
45 return
end

subroutine tfreq(ffunc,data,t,lista,sig,nend,parms,ma,mf,covar,
1  alpha,nca,chiq,alama,epar,fnc,dyda,ftype)
c******************************************************************************
c...This subroutine takes the data and prepares and performs a two frequency fit.
c...The method of choice now is to fit later time data to a regular muon histo
c...function with gaussian depol. This data is extrapolated back to time=0 and
c...then properly subtracted off of the total data, leaving only the YBCO sig.
c...In theory this should work.
c******************************************************************************
dimension lista(l),covar(l,l),alpha(l,l),sig(l),parms(l),t(l),
1  data(l),epar(l),fnc(l),dyda(l)
dimension dshift(8000),tshift(8000),sigsht(8000),subfnc(8000)
real*8 time(8000),y(8000),f(8000)
integer*4 number
character resp*l,ftype*i
external ffunc
print*, 'Input the time at which you wish to begin the fit:'
read*,tst
deltat=t(10)-t(9)
nts=(tst-t(1))/deltat
numshnt=nend-nts
do i=nts,nend
dshift(i-nts+1)=data(i)
tshift(i-nts+1)=t(i)
sigsht(i-nts+1)=sig(i)
endo
c... Now fit these arrays of data.
10 call curfit(lista,tshift,dshift,sigsh,pnumsht,parms,ma,mf,
  1      covar,alpha,nca,chisq,ffunc,alama,1,1,spar,func,nump,dyda)

c... Display this fit on the orig data.
do i=1,nend
  time(i)=t(i)
y(i)=data(i)
  if(i .ge. ntst) then
    f(i)=func(i-ntst+1)
  else
    f(i)=0.d0
  endif
enddo
number=nend
call display(time,y,f,number)
print*,'Fit again? (y/n)'
read(5,'(a1)')resp
if(resp .eq. 'y')goto 10

c... Extrapolate the data back to t=0 -- create the data from the parameters.
if(ftype .eq. 's') then
  do i=1,nend
    subfnc(i)=parms(1)*exp(-t(i)/2.19)*parms(2)*
    1       exp(-t(i)**2*parms(3)**2/2.)*
    2       cos(parms(4)*t(i)+parms(5))
  enddo
elseif(ftype .eq. 'a') then
  do i=1,nend
    argy=t(i)**2*parms(2)**2/2.
    subfnc(i)=parms(1)*exp(-argy)*
    1       cos(parms(3)*t(i)+parms(4))
  enddo
endif

c... Subtract off the theoretical crap from the YBCD data.
do i=1,nend
  dshift(i)=data(i)-subfnc(i)
  if(ftpye .eq. 's') then
APPENDIX E. TRIUMF DATA ANALYSIS PROGRAM

```plaintext
if(dshift(i) .lt. 0)dshift(i)=0.
endif

time(i)=t(i)
y(i)=dshift(i)
f(i)=0.d0
enddo
c...The data in dshift should be the YBCO data only.

number=nend
call display(time,y,f,number)
print*, 'Should this now become the data for analysis? (y/n)'
read(5,'(a1)')resp
if(resp .eq. 'y')then
do i=1,nend
    data(i)=dshift(i)
endo
print*, 'The data for analysis is now the subtracted one.'
else
    print*, 'The data for analysis remains unchanged.'
endif
return
end
```
Bibliography


[52] D.R. Harshman et al. private communication.


[80] H. Krakauer. private communication.


Vita

Allan John Greer, Jr., was born on January 24, 1965, in Abington, Pennsylvania. He graduated from Central Bucks West High School in June, 1983. He received a BS in Physics (cum laude) from Moravian College in June, 1987. He earned a MS in physics from the College of William and Mary in May, 1989. The Doctor of Philosophy degree in physics should be awarded to him in May, 1994, by the College of William and Mary. As of this writing he remains unemployed, but is hoping to soon find gainful employment.