Mapping Magnetic Memory in [Co/Pd]IrMn Thin Films Under Exchange Bias Conditions

C. Alex Safsten

A senior thesis submitted to the faculty of
Brigham Young University
in partial fulfillment of the requirements for the degree of

Bachelor of Science

Karine Chesnel, Advisor

Department of Physics and Astronomy

Brigham Young University

March 2015

Copyright © 2015 C. Alex Safsten

All Rights Reserved

ABSTRACT

Mapping Magnetic Memory in [Co/Pd]IrMn Thin Films Under Exchange Bias Conditions

C. Alex Safsten
Department of Physics and Astronomy, BYU
Bachelor of Science

Magnetic domain memory (MDM) is the tendency of a magnetic material to recall its domain pattern after a disturbance resulting from an application of an external magnetic field. In general, magnetic materials do not exhibit this property, but some specialized thin films composed of layered ferromagnetic and antiferromagnetic materials do display magnetic memory. We measure the amount of memory of such films using the x-ray resonant magnetic scattering (XRMS) technique, and use a speckle-correlation technique to quantify the degree of magnetic memory in the sample. In particular, we examine how the magnetic memory in these films persists when we introduce an exchange bias between the ferromagnetic and the antiferromagnetic layers. We find that when bias is applied to the sample, the magnetic memory persists. However, the amount of memory and its behavior throughout the magnetization loop varies with the magnitude of the bias applied. We further find that the amount of memory and its behavior throughout the magnetization loop does not, however, vary much with multiple magnetization loop passes.

Keywords: Magnetic Memory, Exchange Bias, Magnetic Thin Films, XRMS, Cross-Correlation

ACKNOWLEDGMENTS

I would like to thank my wonderful wife, Emily, for her support and encouragement throughout this process. I would also like to thank my advisor, Dr. Karine Chesnel for teaching me all about magnetic thin films and how to study them. Finally, I appreciate the BYU Department of Physics and Astronomy for funding my research.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Ferromagnetic materials often exhibit uniformly magnetized regions known as domains [1]. The *domain pattern* of a magnetic material is the spatial arrangement of the domains within the sample. The domain pattern of a magnetic sample naturally depends on the external magnetic field. The magnetization of the domains within the sample tend to align with the direction of an external magnetic field.

Normally, magnetic moments of domains can be oriented in any direction in space. If we construct a material which is sufficiently thin in one dimension, the magnetic moments will be oriented primarily in the direction of the thin dimension. We say that the magnetization is either up or down. A magnetic material constructed in this manner is known as a *perpendicular magnetic thin film* . The domain patterns of ferromagnetic materials are very complex. There are three spatial degrees of freedom, and the magnetization of the domains can be oriented in any direction. The domain pattern of perpendicular magnetic thin films, though still complex, is somewhat reduced in complexity because: (1) there are only two spatial degrees of freedom, and (2) there is only

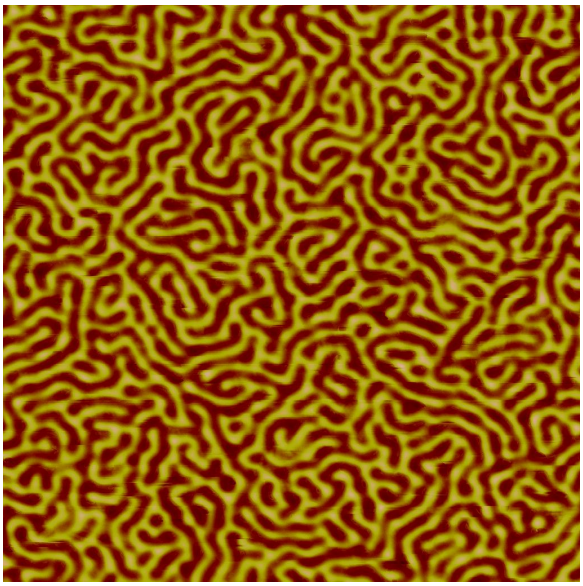


Figure 1.1 The domain pattern the [Co/Pd]IrMn thin film as measured using magnetic force microscopy. The dark regions are domains with spins oriented up while the lighter regions have spins directed down. The physical width of this image is about 20 microns, so we can see that the domain size is about 200 nm

one possible direction for the orientation of the domains. Figure 1.1 shows the domain pattern of a perpendicular magnetic thin film. Figure 1.1 shows a domain pattern in a "maze" state, as the configuration of up-oriented and down-oriented states resembles a maze. Sometimes, the domain pattern manifests itself as bubbles, with a large section of the sample up-oriented, with several small down-oriented regions.

Suppose a magnetic thin film exhibits a certain domain pattern in the absence of an external field. If the domain pattern is changed by an external field, we are usually very unlikely to recover the original domain pattern at some later time. Since there is no preference for the original structure, there is no reason to return to it. In some specialized thin films, however, there is a preference for certain domain patterns. When a domain pattern in such a thin film is disturbed by an external field, there is a tendency for the domain pattern to return to a configuration similar to the original after the field is removed. In other words, the thin film "remembers" its domain pattern. Hence,

we refer to this phenomenon as *magnetic domain memory* (MDM) .

Normally, perpendicular magnetic thin films are isotropically magnetized, which is to say that that there is no preferred direction for the magnetization of the domains. We may, however, introduce a preferred direction by cooling the sample in the presence of a magnetic field, and maintaining this low temperature for the duration of the experiment. This is known as a *field cooling* (FC), as opposed to cooling without a field known as *zero field cooling* (ZFC).

In this work, we will study thin films discovered to exhibit a high degree of MDM. We will produce examine MDM maps, or arrays which completely characterize the amount of MDM exhibited by our samples. Specifically, we will study MDM in thin films experimentally under a variety of cooling field strengths in order to better understand the magnetic interactions which are responsible for MDM within the samples.

1.2 Background

1.2.1 Ferromagnetism, Antiferromagnetism, and Couplings

The domain pattern of a magnetic material is one which minimizes the magnetic energy. The least energy configuration for magnetic domains is, in part, determined by the type of magnetic order of the material. Magnetic ordering is the preferred orientation of neighboring spins, depending on the sign of the exchange interaction associated with the material [2].

If a material has a negative exchange interaction, neighboring spins store less magnetic energy if they are aligned. This magnetic ordering is ferromagnetic ordering. Groups of spins near each other decrease their total energy by mutually aligning their spins and thereby form a domain [1]. A sample is said to be *saturated* if all of its ferromagnetic domains are aligned in the same direction.

If a material has a positive exchange interaction, neighboring spins store less magnetic energy if their spins are anti-aligned. This magnetic ordering is known as antiferromagnetic ordering.

Antiferromagnetic films tend to develop a "checkerboard" pattern at the atomic level, where as many spins are anti-aligned as possible. Antiferromagnetic materials have an associated *Néal temperature* below which the antiparallel ordering occurs.

Figure 1.2 The spins in a ferromagnetic material tend to be aligned with their neighbors. Spins in antiferromagnetic materials are anti-aligned with their neighbors. Above the blocking temperature, spins in the antiferromagnetic layer near the boundary with the ferromagnatic layer tend to align in the direction of the ferromagnetic spins. When cooled below the blocking temperature, the antiferromagnetic layer retains this configuration regardless of any changes in the ferromagnetic layer. Spins so affected are known as uncompensated spins, and this interaction is known as exchange coupling.

Figure 1.2 also gives us some insight on an interesting phenomenon occurring in antiferromagnetic thin films. Observe that each "up" spin in the antiferromagnetic layer is balanced by a "down" spin on every side except the spins near the boundary with the ferromagnetic layer [3]. Spins in the antiferromagnetic layer near the boundary with the ferromagnetic layer are affected by ferromagnetic domain, and at high temperature (e.g., room temperature), these boundary spins

tend to flip in the direction of the neighboring ferromagnetic spins. This effect is known *exchange coupling* [4]. This coupling has an associated *blocking temperature*. Below the blocking temperature, the spins in the antiferromagnetic layers are frozen in their configuration, regardless of the configuration of the neighboring ferromagnetic spins. As ferromagnetic domains conform to the lowest energy state, they couple to uncompensated spins. Thus, the pattern of uncompensated spins is determined by the initial ferromagnetic domain pattern and serves as a template for the recovery of this initial ferromagnetic domain pattern. This is the mechanism which encourages MDM in the ferromagnetic layers [5].

1.2.2 Sample

We use a sample which takes advantage of the exchange coupling between ferromagnetic and antiferromagnetic layers. Specifically, the ferromagnetic layers in our sample consist of alternating cobalt and palladium [3]. The antiferromagnetic layers consist of an alloy of iridium and manganese [6] . We have three samples which vary slightly in the thickness of some of the layers. Formally, the samples' layering scheme is given by,

Sample 0: [(Co 4Å/Pd 7Å)$_{12}$/ Co 4Å/ IrMn 43Å]$_3$/ (Co 4Å/Pd 7Å)$_{12}$/Co 4Å

Sample 1: [(Co 3.8Å/Pd 7Å)$_{12}$/ Co 3.8Å/ IrMn 43Å]$_3$/ (Co 3.8Å/Pd 7Å)$_{12}$/Co 3.8Å

Sample 2: [(Co 3.9Å/Pd 7Å)$_{12}$/ Co 3.9Å/ IrMn 50Å]$_3$/ (Co 3.9Å/Pd 7Å)$_{12}$/Co 3.9Å

with a 5Å capping layer of tantalum on each sample. Here, the / indicates a new layer and a subscript indicates the number of times the preceding structure is repeated in the layering. The samples are produced by sputtering, and each has a thickness of about 70 nm. From the Figure 1.1, we measure the average domain size is about 200 nm, these samples qualify as thin films. Figure 1.3 shows a detailed schematic of the sample construction. We will abbreviate the cumbersome notation for this type of magnetic thin film simply as [Co/Pd]IrMn thin film. Each of the samples were assembled via sputtering by Eric Fullerton and his research team at the University of

California, Davis.

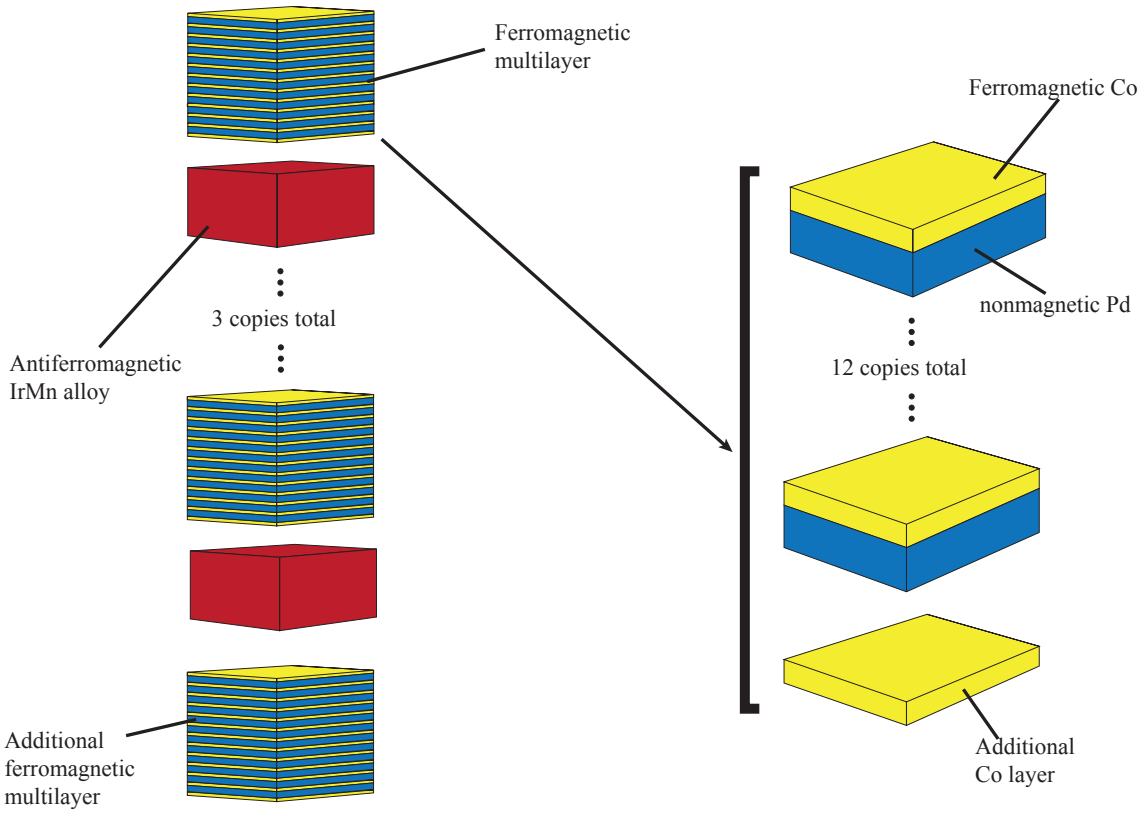


Figure 1.3 The complex multilayered structure of the [Co/Pd]IrMn sample is optimized for magnetic memory.

1.2.3 Magnetization Loops and Exchange Bias

Magnetic materials famously exhibit a property known as hysteresis. Hysteresis is the dependence of the output of a system on the history of the system as well as the current input. Hysteresis in ferromagnetic materials manifests itself in the magnetization loop. A magnetization loop, also known as a hysteresis loop, results when we measure the magnetization of a ferromagnetic sample

with respect to a changing external magnetic field. We find that the magnetization of the sample depends not only on the current external magnetic field, but also on the history of the external field. A magnetization loop may be obtained experimentally by taking the following steps:

1. Begin with a high magnetic field perpendicular to the sample so as to saturate the sample.
2. Reduce the magnetic field gradually, measuring the magnetization at regular intervals.
3. Continue reducing the field until the sample is saturated in the opposite direction.
4. Reverse the process, increasing the field until the sample is once again saturated in the original direction.

The result is a magnetization loop such as in Figure 1.4. The hysteresis is evident in the separation between the ascending and descending branches of the magnetization loop. There are three points of interest in figure. The coercive point, H_c is the field which must be applied in order for the sample to be returned to a neutral state. The saturation point, H_s, is the field required to saturate the sample. The nucleation point, H_n, is the field at which the sample first begins to depart from saturation.

My thesis focuses on the effect of exchange bias on magnetic memory. Exchange bias is the introduction of anisotropy to the sample [7]. Generally, we cool the sample to below the blocking temperature to lock the spin configuration in the antiferromagnetic layers. If we cool the sample in the absence of a magnetic field, the sample is isotropic; there is no preferred direction. We may, however, introduce an external field perpendicular to the film during the cooling process so that the antiferromagnetic layers are biased in the direction of the external field [8] [9]. The external field introduced while cooling is called a *cooling field*. Introducing bias to the sample has the effect of shifting the magnetization loop in Figure 1.4 to the left or right.

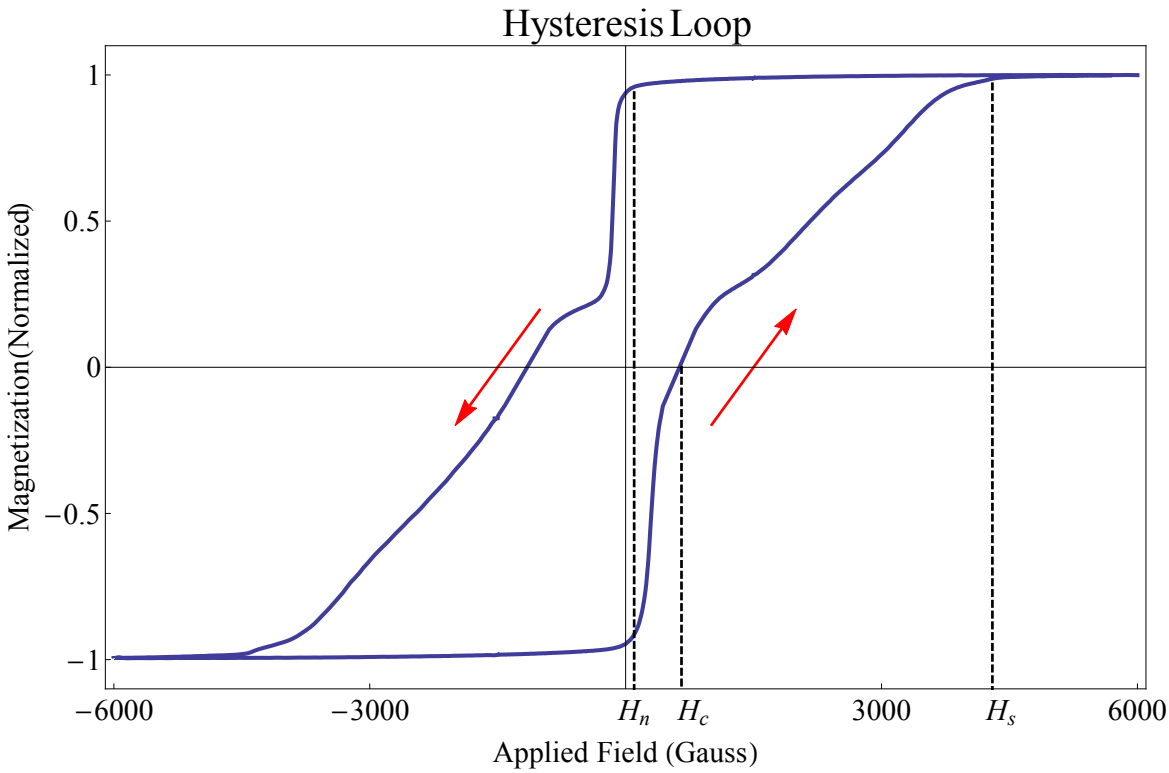


Figure 1.4 The magnetization of the [Co/Pd]IrMn sample 1 as a function of the applied external field. We traverse the loop in the counter clockwise direction. The separation between the ascending and descending branches indicates the hysteresis characteristic of ferromagnetic materials. This data was obtained at 300 K. Note the locations of the coercive, saturation, and nucleation points: H_c, H_s, and H_n.

1.2.4 Mapping Magnetic Memory

In order to determine the amount of MDM exhibited by the sample, we must be able to obtain some information about the domain pattern in situ. We can get some information about the domain pattern using scattering of synchrotron radiation (see Section 2.1 for details) [10]. The result is an image recorded by a CCD detector which contains details related to the domain pattern of the sample. We take images as the sample follows the magnetization loop described in Subsection 1.2.3. We traverse the loop several times and take dozens of images during each loop. We can determine the degree of MDM by comparing images taken at various points along the loop.

After obtaining a scattering image, the next step in determining the amount of magnetic memory is comparing the MDM between two sets of images. We compare two sets of images by producing a map, or an array, which displays the degree of magnetic memory between every pair of images with one in each set. The map shows a general trend in the MDM displayed by the sample over each cycle of the magnetization loop. This allows us to determine over which sections of the magnetization loop the domains in the sample best reproduce their structure. Furthermore, we can tell how the sample "remembers" its domain pattern over several cycles of the magnetization loop.

We will review the specifics of measuring the domain pattern and preparing the retrieved data for presentation in maps in Chapter 2.

1.3 Previous Work and Goals

1.3.1 Previous Work

Studies in recent years have focused on measuring magnetic memory and domain patterns of multilayered magnetic thin films [11]. A conclusion of these studies is that [Co/Pd]IrMn thin films do demonstrate a high degree of magnetic memory induced by the coupling between ferromagnetic and antiferromagnetic layers. These studies have demonstrated MDM in [Co/Pd]IrMn thin films under zero field cooling (ZFC) conditions. We will extend these studies by experimenting on [Co/Pd]IrMn thin films after a cooling field has been applied.

1.3.2 Goals

The goal of our study is to characterize the effects of nonzero cooling field on [Co/Pd]IrMn thin films. We will examine the effects of cooling fields of various strengths.

Chapter 2

Methods

2.1 Experiment

The first step in determining the degree of magnetic memory a sample exhibits is to measure the domain pattern of the sample experimentally.

2.1.1 Principles of the XRMS technique

Xray resonant magnetic scattering (XRMS) is a technique used to measure the domain pattern of magnetic thin films [12]. When xrays pass through a material, they interact weakly with the spins of electrons in the material. This effect is magnified when the energy of the xrays is tuned to an absorption edge of the material (this is what is meant by "resonant"). In our case, we tune the energy of xrays to the L_3 edge of cobalt, 778 eV.

The scattering pattern of xrays through the magnetic material reveals information about the domain pattern of the material. In a process similar to Fraunhofer diffraction, the scattering pattern is related to the Fourier transform of the domain pattern [13]. If the incoming xrays are not spatially coherent, the scattering pattern is blurred. We can still measure general properties like average

domain size, but we lose information about details of the domain pattern. Thus, we use synchrotron radiation which is relatively spatially coherent in comparison to more accessible xray sources. To further enhance the spatial coherence of the xrays, we drill a small aperture in the silicon substrate on which the sample is mounted. The xrays are *still* not perfectly coherent, but the quality is sufficient to distinguish domain patterns with subtle differences.

2.1.2 Experimental Setup

Measuring magnetic memory at a synchrotron facility requires a sophisticated experimental setup. Xrays at 778 eV have an attenuation length of about 1230 μm in air at atmospheric pressure, so we must perform the experiment in high vacuum (micro-Torr levels). Furthermore, we need to measure how the sample responds to an external magnetic field, so we use a vacuum chamber equipped with an octopolar electromagnet capable of producing a field up to 4500 Gauss in any direction.

The sample must be cooled below the blocking temperature. The blocking temperature varies between samples, but each has a blocking temperature higher than 200 K. Even below the blocking temperature, MDM is better observed at temperatures as low as possible, so the sample is affixed to a cryostat and cooled to about 25 K using liquid helium. The cryostat and sample are both mounted on an electronic 3-axis translation stage which we can control digitally to align the sample with the xray beam.

The xray scattering pattern is read and recorded by a CCD camera. Unfortunately, the direct beam is too bright for the detector, and would damage the camera, so we use a small blocker to obscure the direct beam. Because the beam tends to drift slightly, the blocker is attached to a manipulator so we can adjust its position in vacuum. Finally, upstream from the sample, we have installed two shutters: one to regulate the radiation intensity, and the other to expose the sample for a fixed amount of time. Figure 2.1 gives a schematic of the experimental setup.

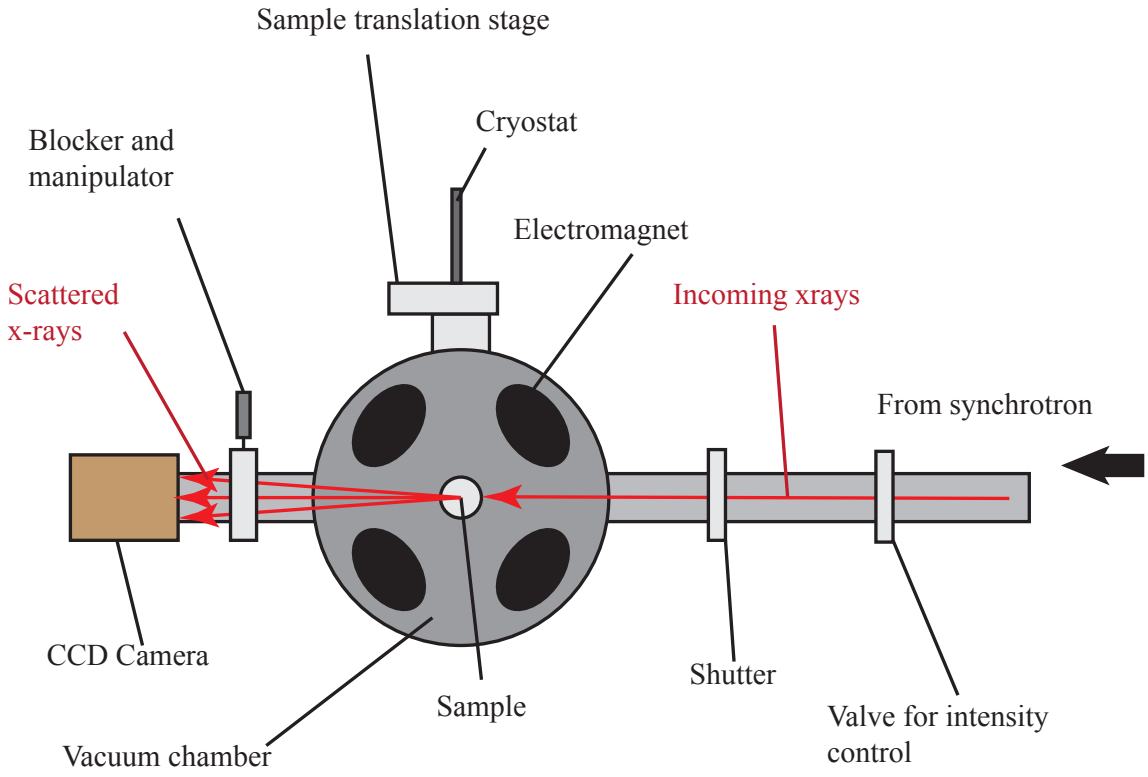


Figure 2.1 A diagram of the experimental setup for measuring magnetic memory at end-station ID4-c at APS.

2.1.3 Experimental Procedure

After assembling the experiment as described above, the first step in performing an XRMS experiment is to evacuate the chamber to micro-torr levels. Next, we decide on the magnitude of cooling field to use (we may also choose not apply any cooling field). We apply the cooling field in the direction perpendicular to the sample, and cool the sample with liquid helium to about 25K. Then we remove the cooling field while maintaining the temperature of the sample. The scattering pattern is brightest when there is no external field. Therefore, we adjust the intensity control valve and

exposure time for the CCD camera so that we collect as much data as possible without saturating or damaging the detector. Finally, we apply the following steps several times:

1. Decide a direction perpendicular to the sample plane to be positive. Apply a saturation field of -4500 Gauss. Record the scattering pattern with the CCD detector.

2. Increase the field in increments of 50 to 200 Gauss (depending on the desired level of detail), and record the scattering pattern at each step.

3. When the field reaches +4500 Gauss, begin decreasing the field at increments of 50 to 200 Gauss, recording the scattering pattern at each step.

We repeat the experiment as described several times for variable cooling fields. Each data set measured between cooling cycles is referred to as a *series*. The parts of series of data taken during the process of increasing applied field are referred to as *ascending branches*, while the parts of a series taken with decreasing field are *descending branches*. Each series has one ascending branch for each repetition of step (2) and one descending branch for each repetition of step (3). Typically, each series has four ascending branches and four descending branches, but some have as few as one of each or as many as seven of each. Four branches is enough to demonstrate magnetic memory over several cycles, but not so many as to waste resources we could use to test the sample under different field conditions. We will measure magnetic memory by comparing images from different ascending or descending branches in the same series. An example of the scattering images obtained through this process is given in Figure 2.2.

2.1.4 Challenges in Experiment

Magnetic memory experiments pose some unique challenges. One such challenge is that the very hysteresis which allows for the phenomenon of magnetic memory can ruin an experiment if we

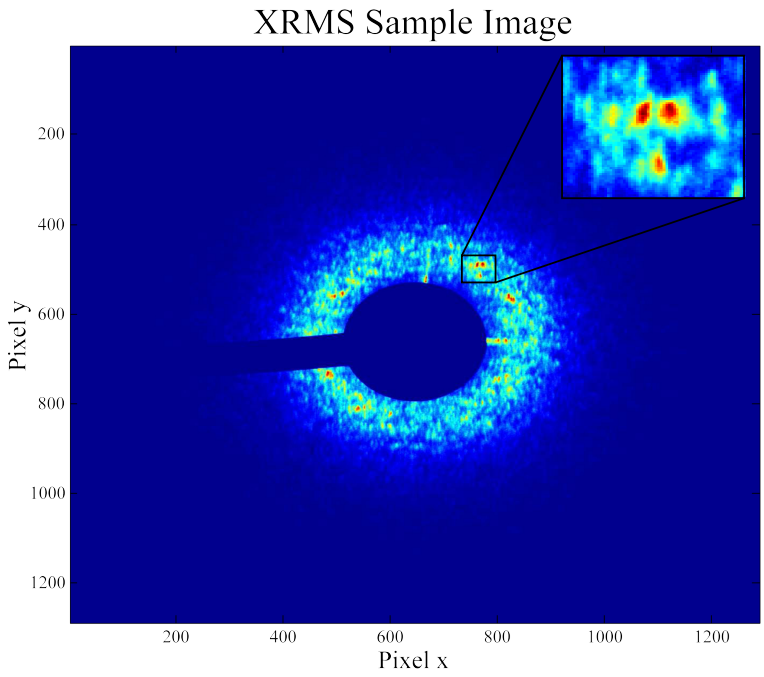


Figure 2.2 This is an example of a scattering pattern obtained through the XRMS technique. Inset is an example of speckles within the image.

are not careful. We use digitally controlled electromagnets to produce an external field for the sample. The computer controlling these magnets accepts a value for a magnetic field goal, and then employs an algorithm which adjusts the current delivered to each of the eight magnets until the field is sufficiently close to the desired value.

The electromagnets' algorithm does not always adjust the field monotonically. If the adjustment is not monotonic, the sample will follow a *minor hysteresis loop*, where the field is increased, and then decreased (or vice versa) without achieving saturation at both extremes. If the field is increased too much, and then decreased to the right value, the domain pattern will be different than if the field is only increased to the particular value. This effect is outlined in Figure 2.3.

Generally, the electromagnets' algorithm does not overshoot very much, so the effect of minor loops is small. Occasionally, though, the algorithm generates a saturation field, and the sample remains saturated for the rest of the images in that branch. We mitigate this problem by changing

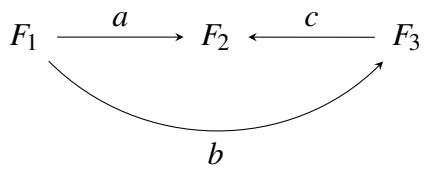


Figure 2.3 Suppose we test the sample at magnetic fields $F_1 < F_2 < F_3$. Two possible paths from F_1 to F_2 are a and $b * c$, where $*$ denotes the concatenation of paths. But because of the effects of hysteresis, these two paths result in different states for the sample's domain pattern when measured at F_2. In particular, if F_3 is a saturation field while F_2 is not, the sample may be saturated if we follow $b * c$ and unsaturated if we follow a. Thus, the process of changing the magnetic field is not commutative.

the field in finer steps in regions where we know the algorithm is faulty.

2.2 Computational Analysis

Through experiment, we obtain the data related to the domain pattern of the sample. To determine how much magnetic memory the sample presents, we must analyze this data using computational techniques.

2.2.1 Purpose

There are two major goals in the computational analysis of experimental data. First, we want to isolate the details of the scattering pattern which correspond to details in the domain pattern. Second, we want to use this information to determine how similar two domain patterns are and organize this information into a chart showing how magnetic memory persists in the sample at different external field strengths.

The features in the scattering pattern corresponding to details of the domain pattern are spots known as *speckles*. The collection of all speckles is the *speckle pattern*. Speckles result from the scattering of coherent x-rays scattering through the sample. Unfortunately, the speckles are

difficult to discern because of the scattering of incoherent x-rays. The incoherent scattering is known as the *envelope*. While the speckle pattern reveals details about the domain pattern, the envelope gives only general information. We will separate the speckle pattern from the envelope, and use speckle patterns to compare images.

The ultimate product of the computational process is the map. A map is an array M generated by two branches of images $\{A_1,...,A_m\}$ and $\{B_1,...,B_n\}$. Each element M_{ij} of the array is a number corresponding to the similarity of image A_i to image B_j. If the images are speckle patterns, then their similarity is related to the amount of magnetic memory exhibited in the sample between those two images [14].

2.2.2 Techniques and Algorithms

While performing our analysis, we will find a few common techniques useful. In particular, we will make use of the convolution of functions. Regarding f and g as L^1 functions $f,g:\mathbb{R}^2\to\mathbb{R}$, the convolution is defined as

$$(f\otimes g)(x)\equiv\int_{\mathbb{R}^2} f(x-\xi)g(\xi)d\xi.$$

More particularly, we employ a cyclic convolution defined for periodic functions.

Definition 2.2.1. Let $f,g\in L^1(\mathbb{R}^2)$ be doubly periodic, both having periods a and b. The *cyclic convolution* of f and g is

$$(f\otimes_{\mathbb{T}} g)(x)=\int_{\mathbb{T}} f(x-\xi)g(\xi)d\xi$$

Where $\mathbb{T}=[0,a)\times[0,b)$.

We will find the following useful in constructing periodic functions.

Definition 2.2.2. Given a function $f:\mathbb{R}^2\to\mathbb{R}$ which vanishes outside of $[0,a)\times[0,b)$ for $a,b>0$,

the *periodic summation* of f is a function $\tilde{f}:\mathbb{R}^2\to\mathbb{R}$ defined by

$$\tilde{f}(x,y)=\sum_{n=-\infty}^{\infty}\sum_{m=-\infty}^{\infty}f(x-na,y-mb).$$

Proposition 2.2.3. *Let f be a function $f:\mathbb{R}^2\to R$ which vanishes outside of $[0,a)\times[0,b)$ for $a,b>0$. Then the periodic summation of f, $\tilde{f}(x,y)$, is periodic in both x and y.*

A proof of this proposition is given in the appendix.

We will convolve images, which we can regard as a matrix of pixel values. Of course, we do not have a convolution defined for matrices, so we will convert each image into a function. Let A be an m-by-n matrix corresponding to an image. If $A_{ij}=z$, define a function

$$f_A(x,y)=\begin{cases}A_{\lceil x\rceil\lceil y\rceil} & \text{if } 0\le x<n, 0\le y<m\\ 0 & \text{otherwise.}\end{cases}$$

Since A has mn values, this is a simple function, and therefore $f_A\in L^1(\mathbb{R}^2)$.

Definition 2.2.4. Let A and B be matrices corresponding to images with dimension n-by-m. Then the *image convolution* of A and B is

$$C=A\otimes B$$

where C is the matrix,

$$C_{ij}=(\tilde{f}_A\otimes_{\mathbb{T}}\tilde{f}_B)(i-1/2,j-1/2)$$

for $1\le i\le n$, $1\le j\le m$. Note that $A\otimes B=B\otimes A$.

We will also take advantage of the convolution theorem, which states

$$\mathscr{F}(f\otimes g)=\mathscr{F}(f)\mathscr{F}(g),$$

where $\mathscr{F}$ denotes a Fourier transform. The convolution theorem also applies to image convolutions.

We will have many opportunities to apply image convolution. One opportunity is in Gaussian smoothing . A Gaussian matrix is an n-by-n matrix G given by

$$G(n,\sigma)_{ij} = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(i-n/2)^2+(j-n/2)^2}{2\sigma^2}},$$

where σ is a real number corresponding to the width of the Gaussian peak. The Gaussian smoothing of an image A is $A \otimes G$. Physically, the process of Gaussian smoothing results in an image similar to A, but blurred slightly. An example of the Gaussian smoothing to Figure 2.2 is given in Figure 2.4.

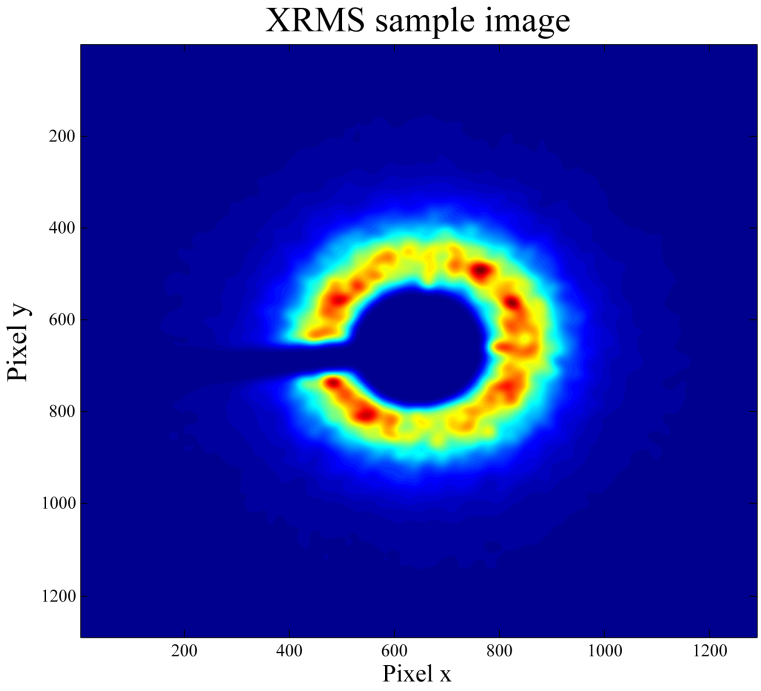


Figure 2.4 Gaussian smoothing applied to Figure 2.2 with the Gaussian matrix $G(50,10)$.

Another application of image convolution is the process of cross correlation. Cross correlation is a method of comparing two images. Cross correlation is a mapping f which maps two images to a real number (usually denoted ρ). It is defined for n-by-m images by

$$\rho = f(A,B) = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m}(A \otimes B)_{ij}}{\sqrt{\sum_{i=1}^{n}\sum_{j=1}^{m}(A \otimes A)_{ij}\sum_{i=1}^{n}\sum_{j=1}^{m}(B \otimes B)_{ij}}}. \tag{2.1}$$

It is not hard to apply Hölder's inequality to show that $0 \leq f(A,B) \leq 1$. If $f(A,B) = 1$, we may conclude that $A = B$. If $f(A,B) \approx 1$, we know that A is very similar to B. We will find this method invaluable in calculating magnetic memory.

2.2.3 Computational Procedure

We now describe the procedure for extracting the speckle pattern and determining the magnetic memory demonstrated between speckle patterns. In order to expedite the computational analysis, we have developed a robust collection of MATLAB functions to perform various tasks. Here we present the steps in the computational analysis.

1. First, we must enter all information from experiment into appropriate files so other functions can access it. This information includes the image numbers recorded by the camera, the applied magnetic field for each image, and the cooling field applied for each series.

2. Each scattering pattern is occluded by the blocker. This results in a scattering image with artificially low values in the shadow of the blocker. To prevent this data from corrupting the results, we must find the location of the blocker within each image. Fortunately, the position of the blocker is nearly constant for each series. (We sometimes have to move the blocker when the beam position shifts slightly due to optical elements changing temperature).

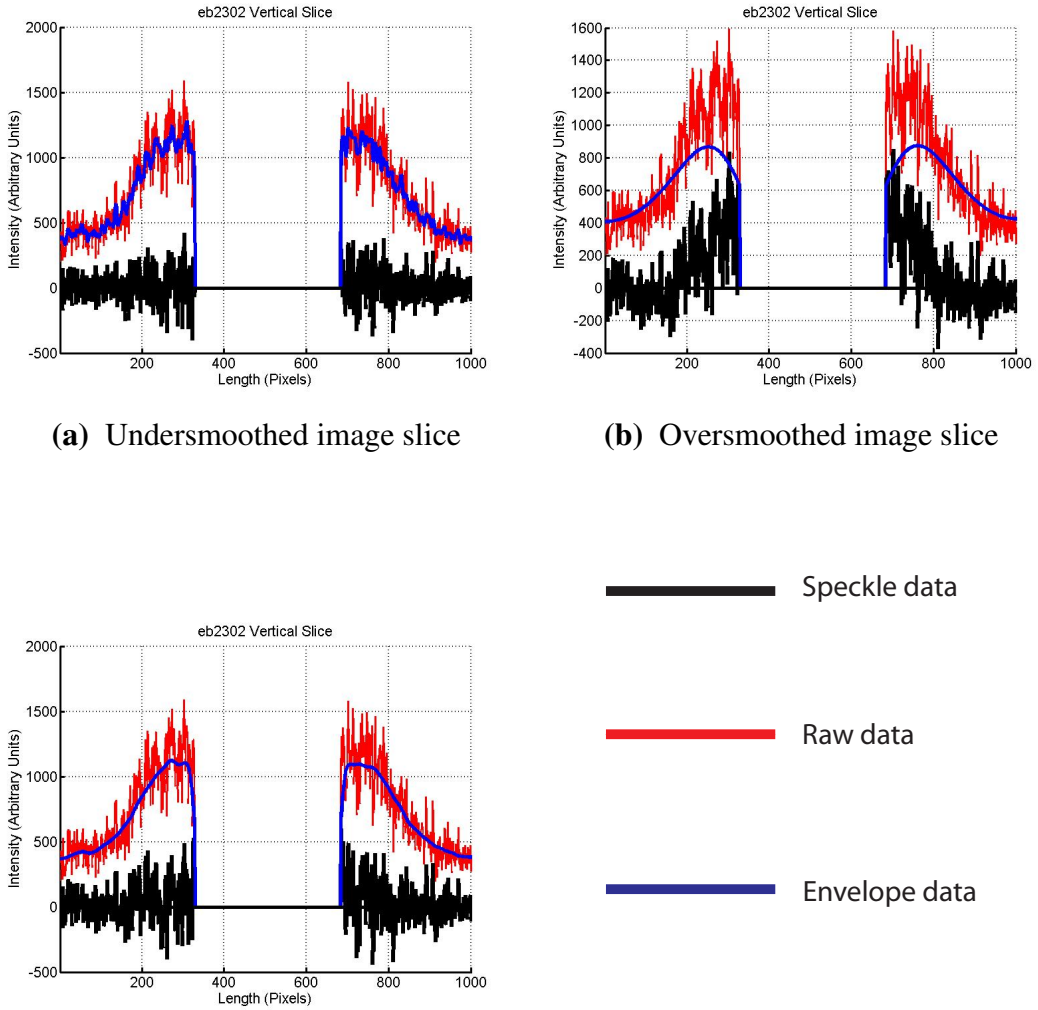
3. We will apply Gaussian smoothing repeatedly to each image to find the smooth underlying envelope of the scattering pattern, which corresponds to the scattering from incoherent radiation. We will subtract the envelope from the rest of the image to extract the speckle data. If we do not smooth enough, we will lose some of the speckle data. If we smooth too much, we will include too much of the envelope in the extracted data. In this step, we pick a few sample images out of a series to smooth at various levels, and determine the best one. Figure 2.5 show the results of oversmoothing, undersmoothing, and the proper level of smoothing.

Once we have determined the proper level of smoothing for each series, we smooth each member of the series to this level.

4. The longest step in the computation is the cross correlation. The cross correlation is performed using the convolution theorem, and takes about one second for each pair of images. We cross correlate every pair of images within a series, so if there are 200 images in a series, the cross correlation can take over ten hours per series when computed in series. The result of the cross correlation between two images of dimension n-by-m is another image of dimension $2n$-by-$2m$. There is a sharp peak centered close to (n,m), but the rest of the image has pixel values close to zero. To save hard drive space, we only save an 81-by-81 subimage of the correlation centered at the peak.

5. Next, we compute
$$\sum_{i=1}^{n}\sum_{j=1}^{m}(A\otimes B)_{ij}$$
for each pair of images A and B in the series. There is a slight complication: there are sometimes small secondary peaks near the primary peak due to neighboring speckles correlating with each other. To avoid summing these secondary peaks, we identify an ellipse which contains only the primary peak, and only sum these values.

6. The final step is to assemble the maps. A *cross-correlation map* is an interpolation of the $f(A_i,B_j)$ from Eq. (2.1) over images $A=\{A_1,...,A_\alpha\}$ and $B=\{B_1,...,B_\beta\}$ where A and B are both ascending or descending branches of images. We assemble the maps by recalling the values of sums from the previous step, and arranging these values in an array. Then we interpolate over the resulting array. Rather than image number, we put the associated applied external field on the axes. This shows us which regions on the magnetization loop have the highest degree of correlation. We call the difference between the branch number for the two branches the separation quality of the map. For example, a map formed by cross correlation



(a) Undersmoothed image slice

(b) Oversmoothed image slice

(c) Properly smoothed image slice

(d) Figures legend

Figure 2.5 These are slices, or columns of the image matrix showing the raw data, the smoothed envelope, and their difference, the speckle. It is important to identify an optimal level of smoothing. Too much smoothing results in a portion of the envelope included in the speckle pattern. Too little smoothing results in omitted speckle detail.

of ascending branch 1 and ascending branch 3 has separation 2. We average all of the maps with the same separation to observe the overall effect.

2.2.4 Challenges in Computational Analysis

The biggest challenge in computation is adjusting for the presence of the blocker. As described above, we must identify the location of the blocker to account for it. Even so, when we smooth the image, the pixels near the blocker are affected by the lower pixel values of the blocker. We do not yet have a good way to avoid this, so we must dispose of data in the neighborhood of the blocker. Thus, when entering the location of the blocker, we intentionally create a blocker about twenty pixels larger than necessary in both directions. When we construct an array with is zero on the enlarged blocker, and one elsewhere, we may multiply this array by the original image and thereby dispose of the original blocker shadow and the surrounding pixels.

We would, of course, prefer not to dispose of data. It is possible that we may interpolate over the blocker region so that the pixels in the neighborhood of the blocker shadow are affected by the low values during the smoothing. This is an area of further study.

Chapter 3

Results and Conclusions

3.1 Experimental Results

The XRMS experiment yields thousands of scattering images. We index the images by number, and organize them into several series. Each cooling cycle has an associated series consisting of (usually) four ascending branches and four descending branches, corresponding to a total of about two hundred images. Between each series, we adjust various parameters of the experiment such as cooling field strength, or final temperature. We may also change which sample we test. Table 3.2 shows the parameters for each series. Some series have a wide temperature range because the liquid helium was depleted before completing the series. Fortunately, enough helium remained to keep the temperature remained far below the blocking temperature in every case.

The data in this analysis are measurements from one of four samples. Recall from Section 1.2.2 that we are working with a set of distinct multilayer types denoted samples 0, 1, and 2. Each sample is mounted on a silicon wafer with a small aperture to increase coherence. Samples 0a and 1a, and 2a have an aperture of size 20μm$\times$ 30μm, while sample 1b has an aperture of size 100μm$\times$ 100μm. The aperture size is the only difference between sample 1a and sample 1b. This

Sample name	Sample number	Aperture size ($\mu m \times \mu m$)
0a	0	20×30
1a	1	20×30
1b	1	100×100
2a	2	20×30

Table 3.1 A description of each sample used in the XRMS experiment

information is summarized in Table 3.1. The size of the aperture is important. A smaller aperture size results in a more contrasted scattering pattern, as shown in Figure 3.1. The larger aperture size produces smaller and less contrasted speckle spots, so we use the smaller aperture size to measure magnetic memory. Thus, we only used sample 1b for a few images, not for large series.

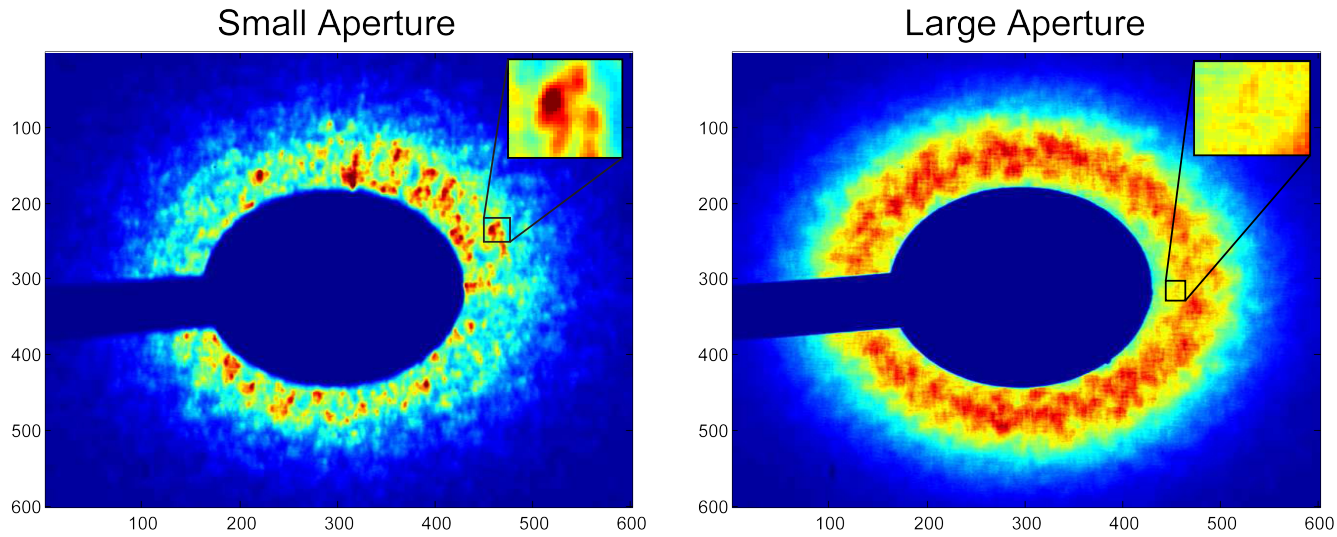
An experiment at the Advanced Light Source (ALS) at Berkeley tested sample 0a, while an experiment at the Advanced Light Source (APS) at Argonne tested samples 1a, 1b, and 2a. The results from the ALS data and APS data are different, so we separate them accordingly. The ALS data comprise series 900-3400, while the APS data comprise series 5000-7200.

3.2 Maps

Here we present the results of the computational analysis. The results consist of maps generated as described in Section 2.2.3. Table 3.3 indexes the series with their corresponding figures. We divide the results in two groups: one obtained in an experiment at ALS and the other at APS. The maps presented here are all given a title starting with the series number, then an “A” or “D,” meaning ascending or descending, followed by a number indicating the number of branch separations. This means that the image is an average of all the maps where the two correlated branches are separated by a fixed number of loops. Four example, the 1200 series has four ascending branches, so the map

Series #	Experiment Location	Sample	Temperature (K)	Cooling Field (Oe)	Image Range
900	ALS	0a	21	3200	953-965
1200	ALS	0a	19	3200	1238-1370
1600	ALS	0a	27	3200	1631-1704
1900	ALS	0a	18	2560	1975-2140
2100	ALS	0a	21	2240	2144-2270
2200	ALS	0a	22	1920	2290-2394
2400	ALS	0a	28	1280	2407-2511
3400	ALS	0a	20	640	3452-3631
3600	ALS	0a	22	0	3635-3759
5000	APS	2b	303	0	5010-5128
5200	APS	1a	22	0	5280-5495
5500	APS	2b	22-70	0	5496-5624
5700	APS	1a	23	4500	5714-5922
5800	APS	1a	23	2500	5923-5974
5900	APS	1a	22	2500	5977-6190
6100	APS	1a	24-130	2000	6192-6393
6400	APS	1a	23	1500	6395-6693
6700	APS	1a	21	1000	6703-6913
7000	APS	1a	22	500	6916-7121
7200	APS	1a	22-160	0	7124-7330

Table 3.2 The parameters used in each series. We organize the experimental data into series. Each cooling cycle has an associated series.



(a) Scattering through a small aperture **(b)** Scattering through a large aperture

Figure 3.1 Scattering through a smaller aperture results in a more distinct speckle pattern. Inset are selected speckles from each image. Note that the speckles from the smaller aperture image are both larder and more distinct than those of the larger aperture image.

with a title "1200-A2" is an average of correlations between images on the first and third branch and images on the second and fourth branch.

3.2.1 ALS Data

Each series from the ALS data has a distinct shape featured in its maps. In particular, we note the following:

- All zero-separation maps are exactly symmetric because the correlation operation is commutative. See Figure 3.2.
- The 1200 series has particularly low correlation levels. See Figure 3.3.
- The ascending branches of the 2100 series demonstrate a "plateau" feature. See Figure 3.4.
- The descending branches of the 2100 series demonstrate a "bird" shape. See Figure 3.5.

Series #	Figure(s)
900	3.7
1200	3.8-3.9
1600	3.10-3.11
1900	3.12-3.13
2100	3.14-3.15
2200	3.16-3.17
2400	3.18-3.19
3400	3.20-3.21
5200	3.22-3.23

Table 3.3 The figures corresponding to each series.

- The maps in the 2400 series feature an "X" shape. See Figure 3.6.

Every map has the same color scale; a red color indicates a higher correlation, while a blue color indicates a lower correlation. We observe a general trend showing that series with lower cooling field tend to show more memory (on average) than those with a higher cooling field. We also see some other trends. Most of the maps have strong correlation on the diagonal, but not all. Many also have strong correlation on a plateau in the center, and others have strong correlation on the anti-diagonal.

Notice that the zero separation, or same-loop correlation maps always have a strongly-correlated diagonal. This is because the diagonal consists of autocorrelation. The near-diagonal correlations are high because we do not expect the domain pattern of the sample to change much over a small interval within a magnetization loop.

Figures 3.7-3.21 give most of the correlation maps obtained from ALS data.

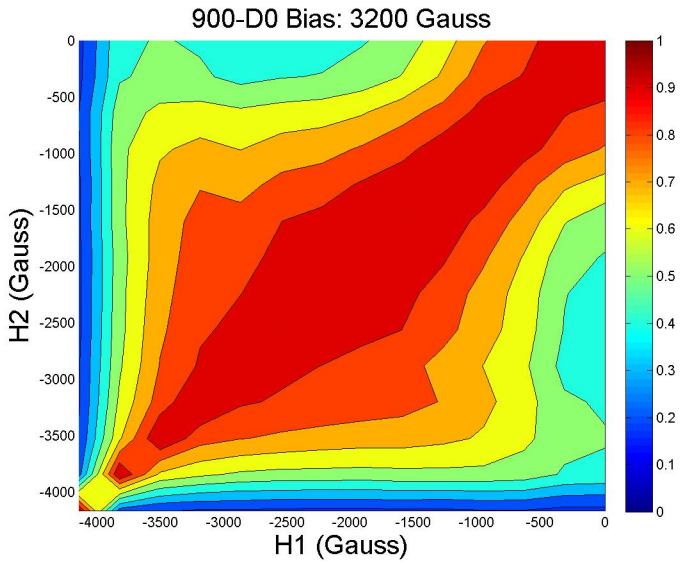


Figure 3.2 Autocorrelation maps are exactly symmetric since the correlation operation is commutative. In the plot title, the letter D stands for descending, and the zero next to the D means that this is a composite average of all maps which show a correlation of two branches with a zero loop separation (i.e., all autocorrelations).

3.2.2 APS Data

These series have a consistent plateau of high correlation in the low-field corner of the map, while the remainder of the map increases in correlation with decreased cooling field. This behavior is not expected based on the results of ALS and other experiments. Further experiment is required to determine whether this is authentic, or a property of samples 1a and 2a. Figures **??** and **??** shows a sample of results from APS data.

3.3 Quantitative Analysis of Maps

3.3.1 Slices

While correlation maps give us a good qualitative understanding of MDM properties of thin films, a quantitative analysis requires a different sort of plot. Such a plot is a slice of the correlation maps.

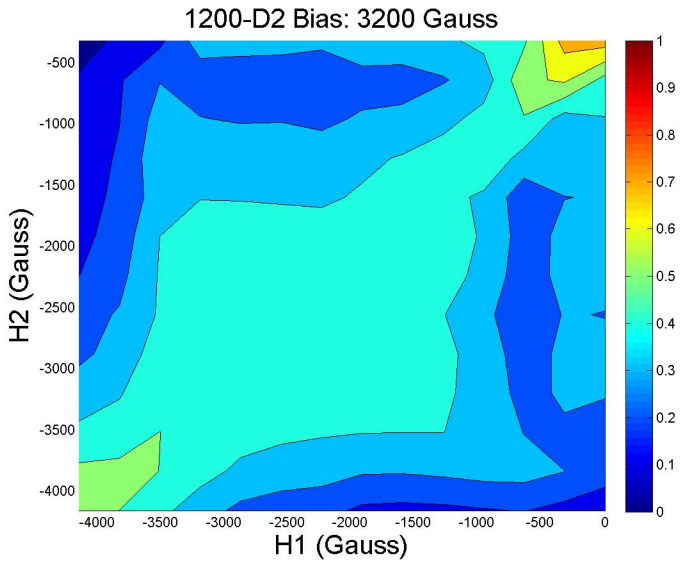


Figure 3.3 The 1200 series has, ing general, a relatively low degree of correlation. In the plot title, the letter D stands for descending, and the two next to the D means that this is a composite average of all maps which show a correlation of two branches with a two loop separation

A slice is simply a row, column, or diagonal of the correlation map matrix plotted against field value. We can superimpose many slices on the same plot to observe different effects. For instance, Figures 3.24 and 3.25 column and diagonal slices (respectively) for each ascending branch separation. We usually refer to a column slice as a vertical slice, and we will always present a vertical slice corresponding to the middle column. Notice that there is not very much decrease in memory over many branch separations. This is a confirmation of a previous paper, which demonstrated that memory persists over many branch separation [15].

Of particular interest to us is the collection slices corresponding to the same branch separation across series with different cooling fields. Such a plot of slices is given in Figures 3.26 and 3.27. Besides the rather anomalous results from the series with a 640 Oe cooling field (this is the 3400 series), we notice the general trend that increased cooling field leads to decreased overall correlation. We shall proceed with a statistical analysis to confirm this rigorously.

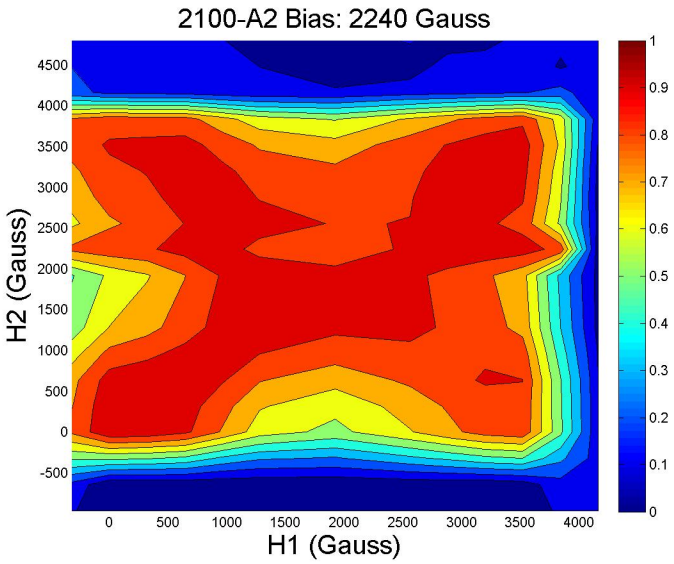


Figure 3.4 Maps from the ascending branches of the 2100 series feature a "plateau" shape. In the plot title, the letter A stands for ascending, and the two next to the A means that this is a composite average of all maps which show a correlation of two branches with a two loop separation

3.3.2 Statistical Analysis

We study two statistics for every map with positive separation: maximum correlation and mean correlation. So as not to confuse these with other maxima and means, we will call these variables map maximum and map mean. We study these variables with respect to two factors: cooling field and branch separation. While we omit a complete presentation of this data, summary statistics are given in Table 3.4. Figure 3.28 shows the effect of cooling field on the means of map maximum and map mean.

To test the significance of cooling field and branch separation on both variables, we will use the statistical tools ANOVA (analysis of variance) and Tukey HSD (honest significant difference). ANOVA tables given in Tables 3.5 and 3.6.

The most important results from the ANOVA tables are the rows for which $\mathscr{P}(>F)$ has a value less than 0.05. This means that the corresponding factor has a statistically significant effect at the

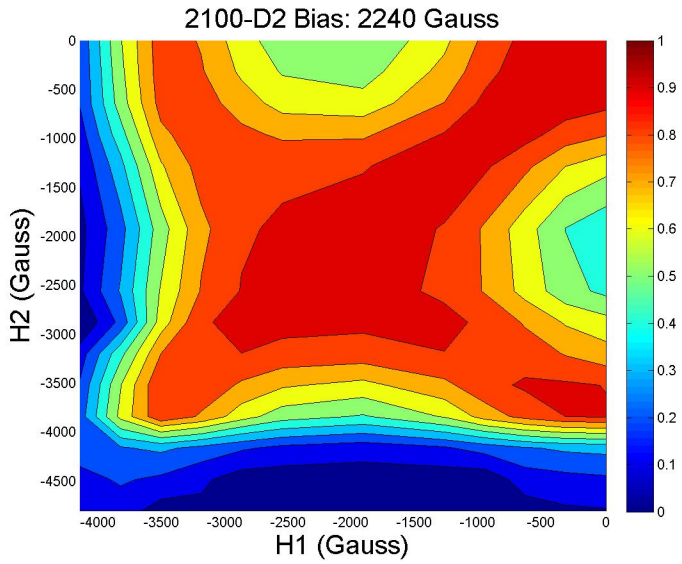


Figure 3.5 Maps from the descending branches of the 2100 series feature a "bird" shape. In the plot title, D stands for descending, and the two next to the D means that this is a composite average of all maps which show a correlation of two branches with a two loop separation

	Map Maximum		Map Mean	
Cooling Field (Oe)	Mean	Std. Dev.	Mean	Std. Dev.
1280	0.989	0.009	0.740	0.058
1920	0.893	0.031	0.583	0.029
2240	0.976	0.012	0.661	0.078
2560	0.899	0.020	0.549	0.076
3200	0.815	0.120	0.389	0.056
Total	0.9207	0.0722	0.5937	0.1257

Table 3.4 Means and standard deviations for the experimental data.

95% confidence level. To summarize the preliminary results of this analysis:

- Cooling field and branch separation both have a significant effect on the map maximum

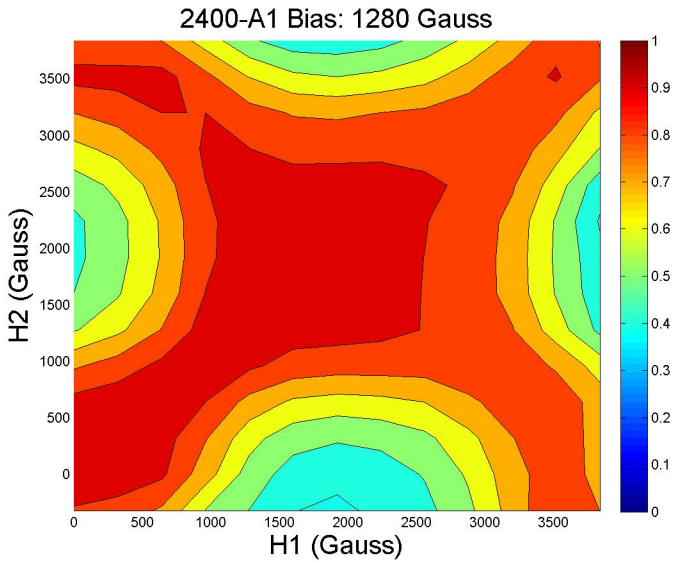
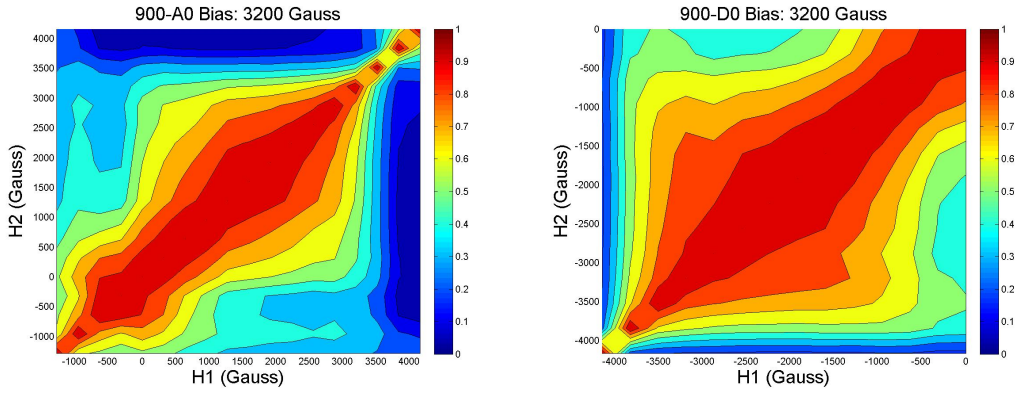


Figure 3.6 Maps from the 2400 series feature an "X" shape. In the plot title, the letter A stands for ascending, and the one next to the A means that this is a composite average of all maps which show a correlation of two branches with a one loop separation

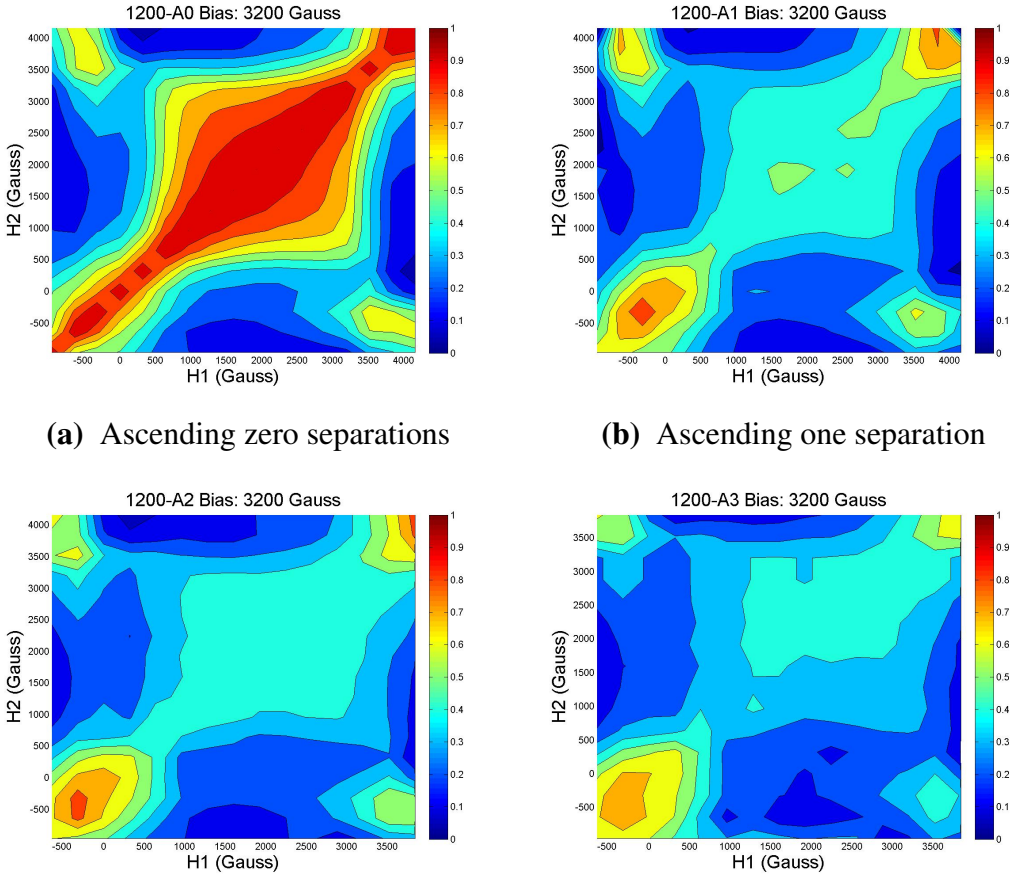


(a) Ascending zero separation **(b)** Descending zero separation

Figure 3.7 Ascending and descending averaged maps for the 900 series.

correlation, as does the interaction between them.

- Cooling field and branch separation both have a significant effect on the map mean correlation.



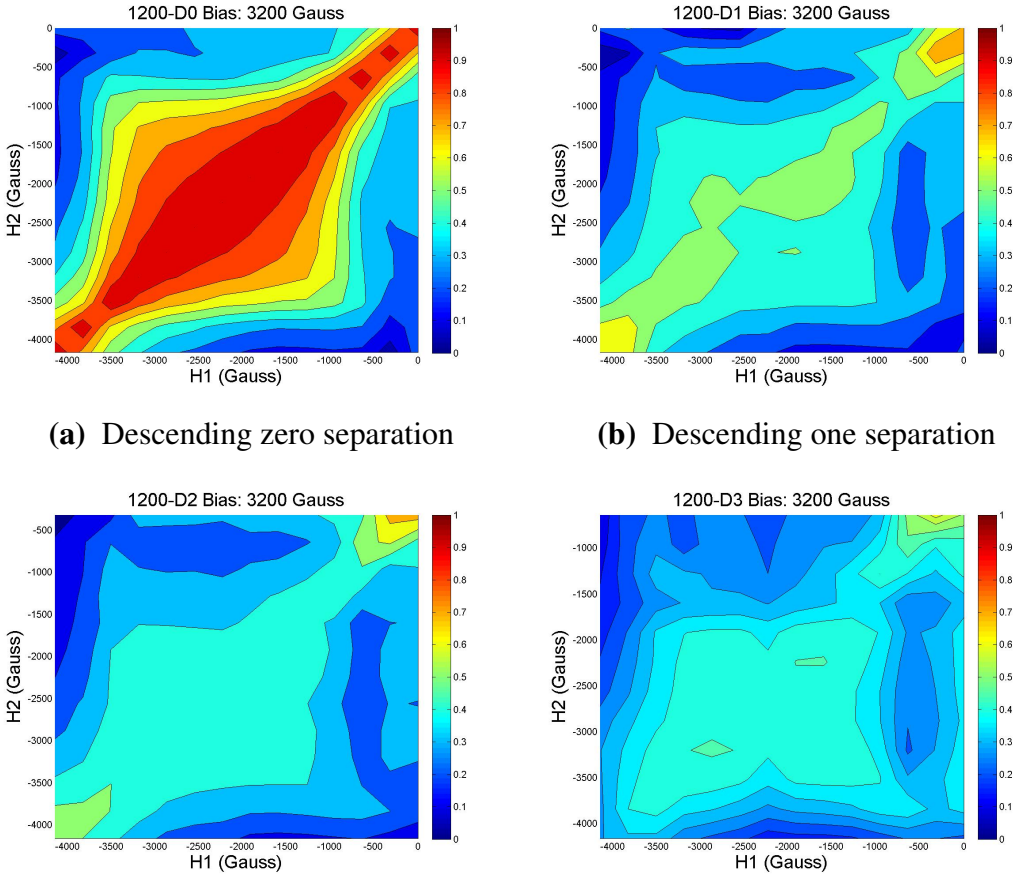
(a) Ascending zero separations

(b) Ascending one separation

(c) Ascending two separations

(d) Ascending three separations

Figure 3.8 Ascending averaged maps for the 1200 series.



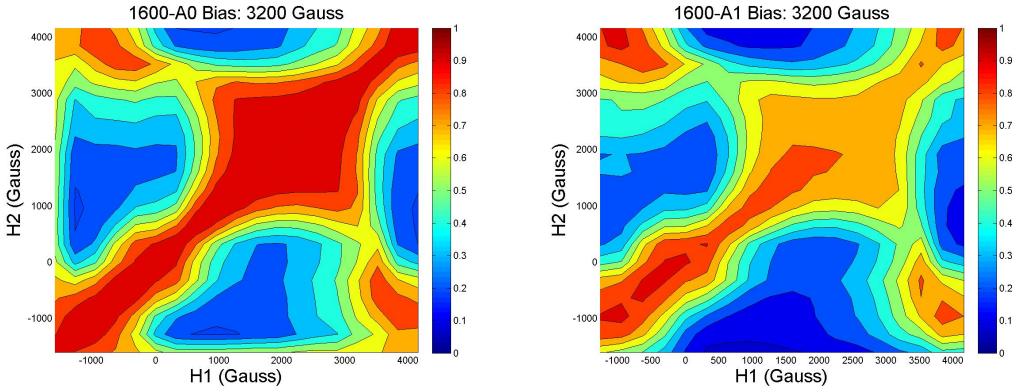
(a) Descending zero separation

(b) Descending one separation

(c) Descending two separations

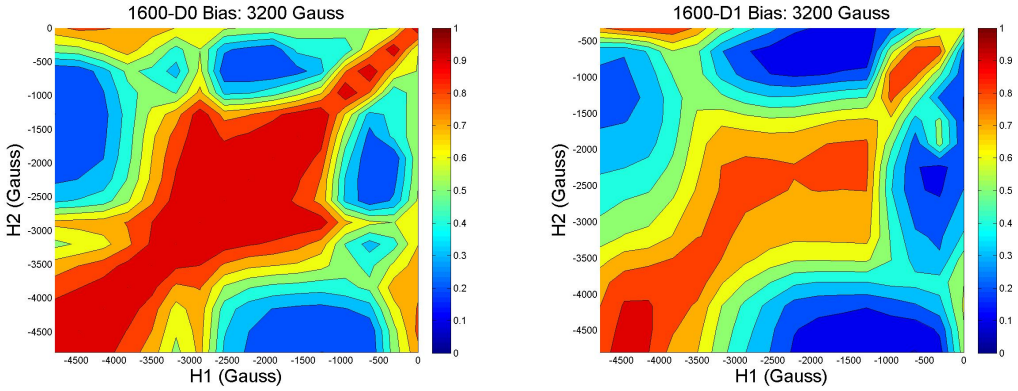
(d) Descending three separations

Figure 3.9 Descending averaged maps for the 1200 series.



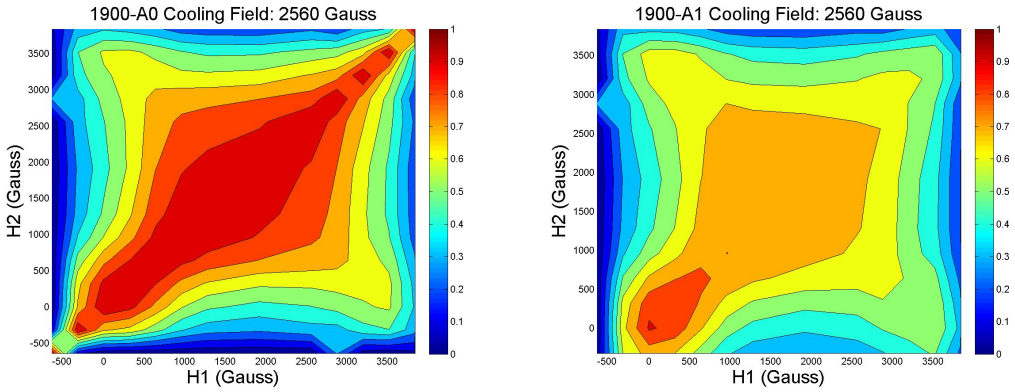
(a) Ascending zero separations **(b)** Ascending one separation

Figure 3.10 Ascending averaged maps for the 1600 series.

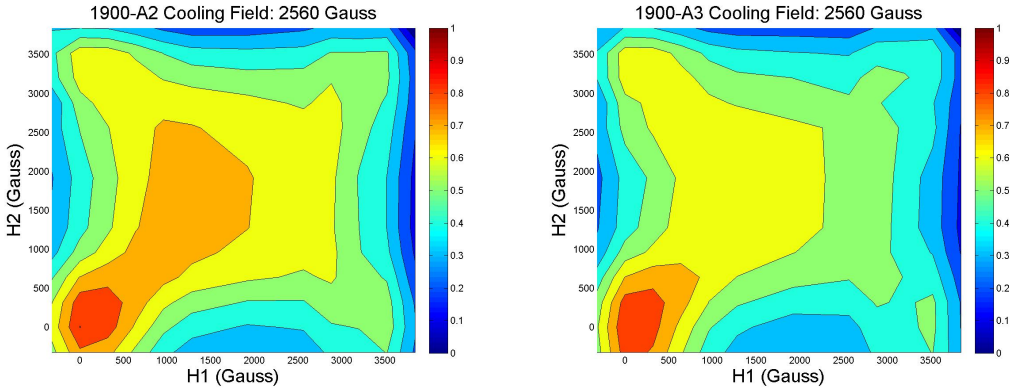


(a) Descending two separations **(b)** Descending three separations

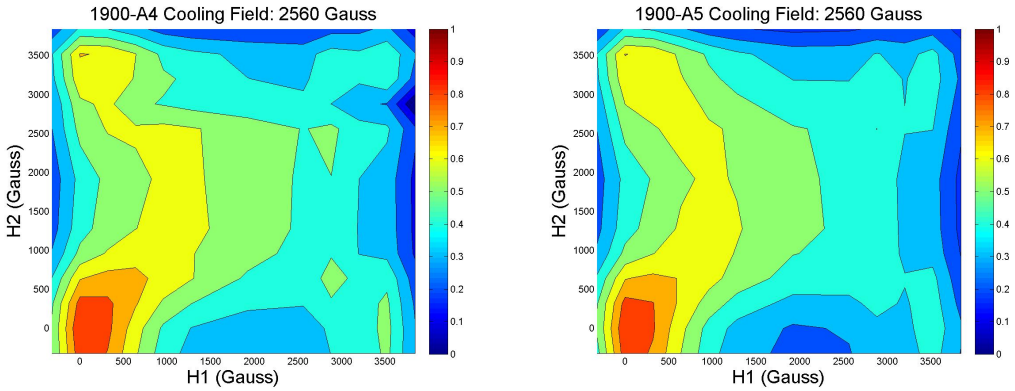
Figure 3.11 Descending averaged maps for the 1600 series.



(a) Ascending zero separations **(b)** Ascending one separation

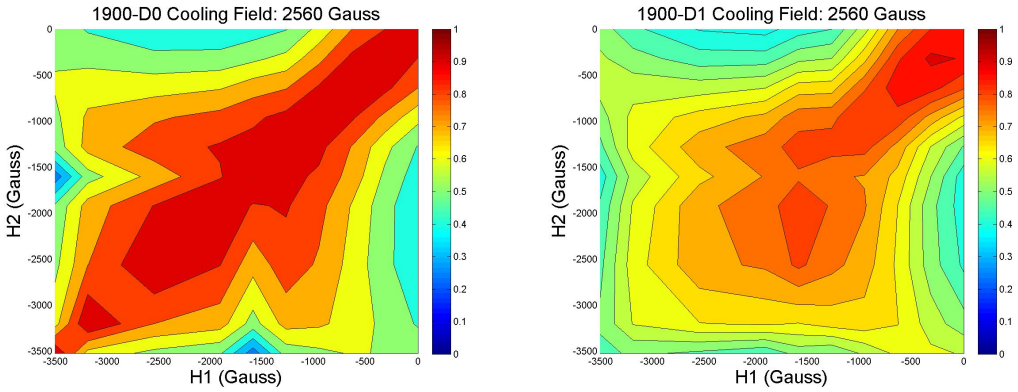


(c) Ascending two separations **(d)** Ascending three separations

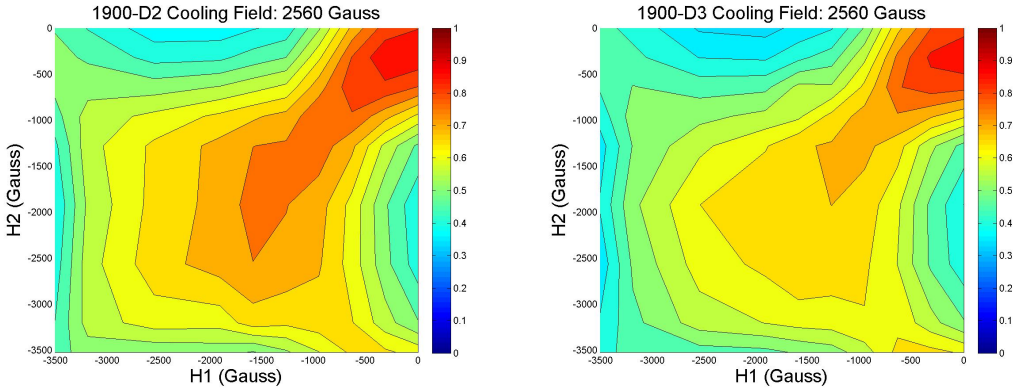


(e) Ascending four separations **(f)** Ascending five separation

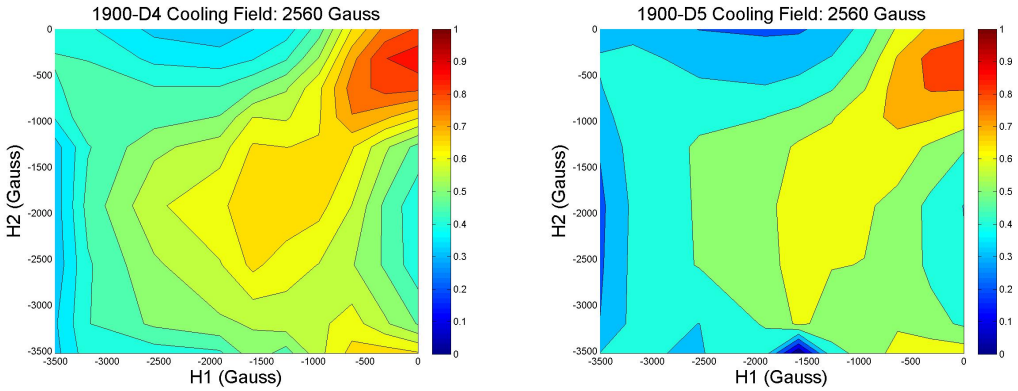
Figure 3.12 Ascending averaged maps for the 1900 series.



(a) Descending zero separations **(b)** Descending one separation

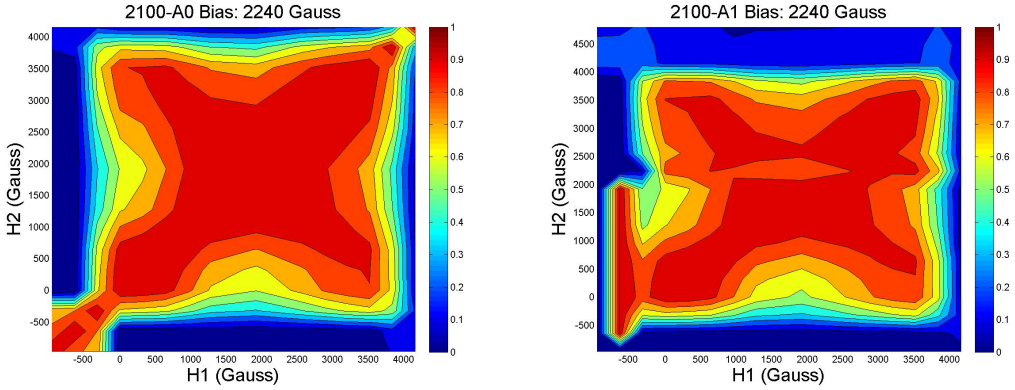


(c) Descending two separations **(d)** Descending three separations

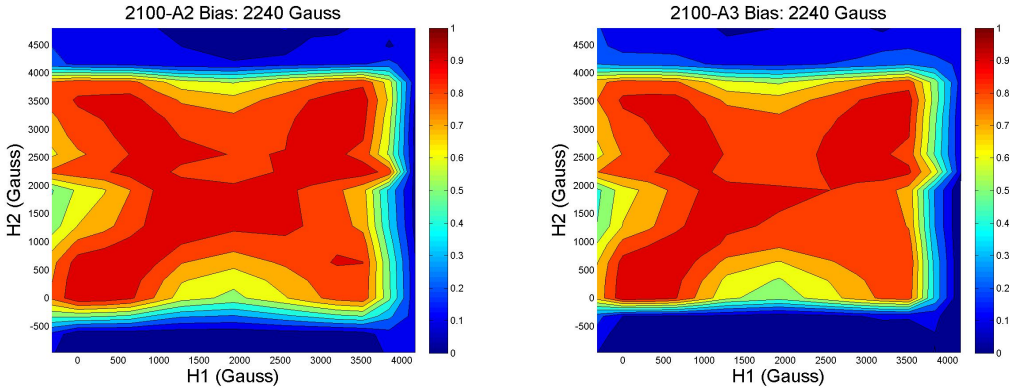


(e) Descending four separations **(f)** Descending five separation

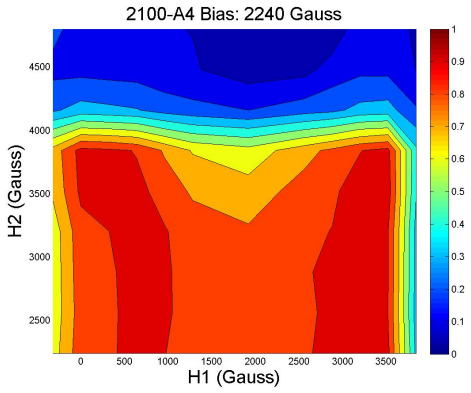
Figure 3.13 Descending averaged maps for the 1900 series.



(a) Ascending zero separations **(b)** Ascending one separation

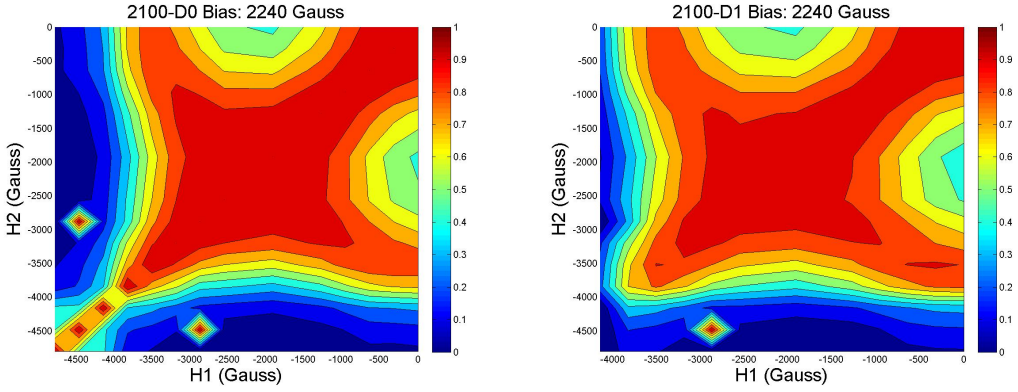


(c) Ascending two separations **(d)** Ascending three separations

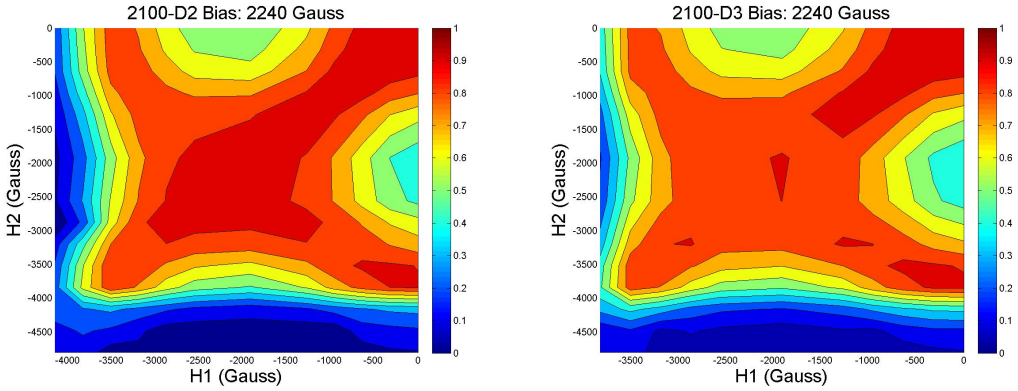


(e) Ascending four separations

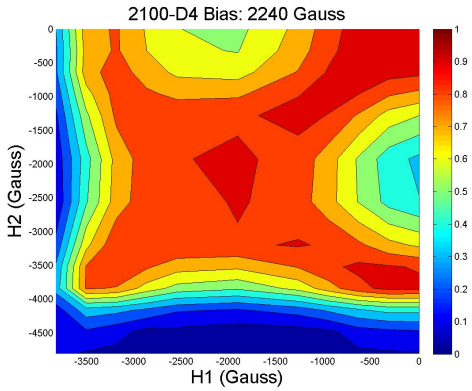
Figure 3.14 Ascending averaged maps for the 2100 series.



(a) Descending zero separations **(b)** Descending one separation

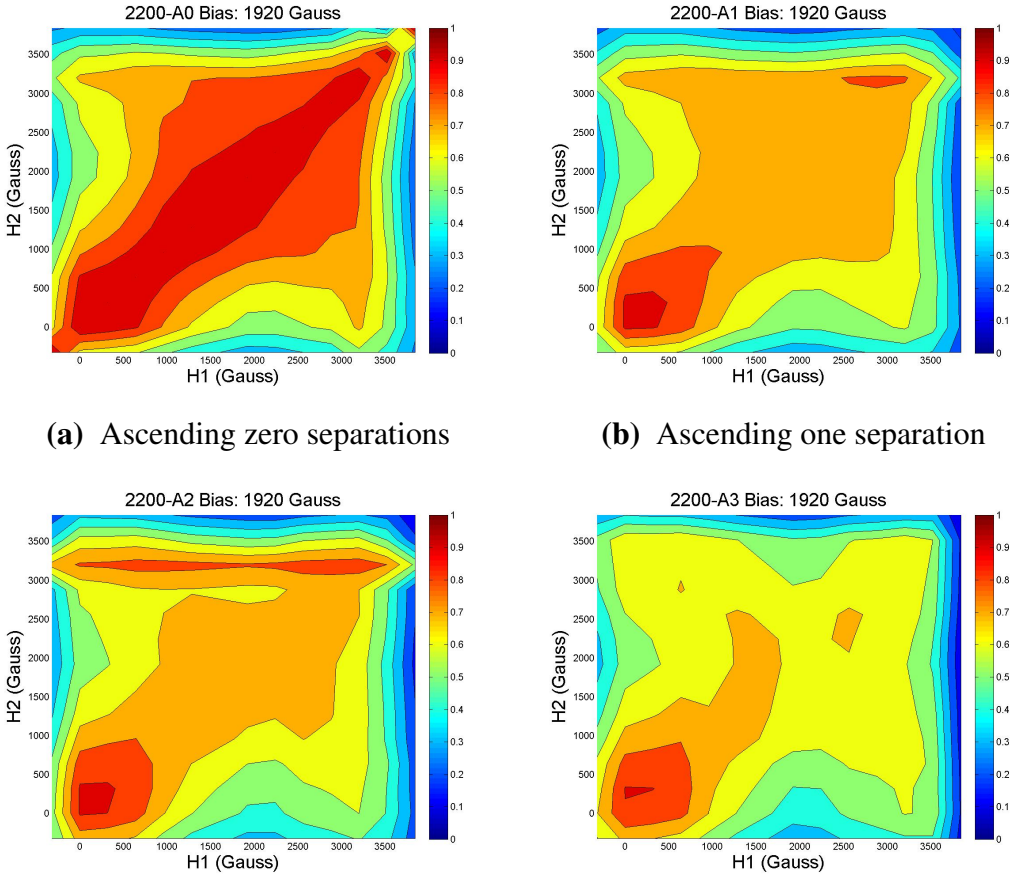


(c) Descending two separations **(d)** Descending three separations



(e) Descending four separations

Figure 3.15 Descending averaged maps for the 2100 series.



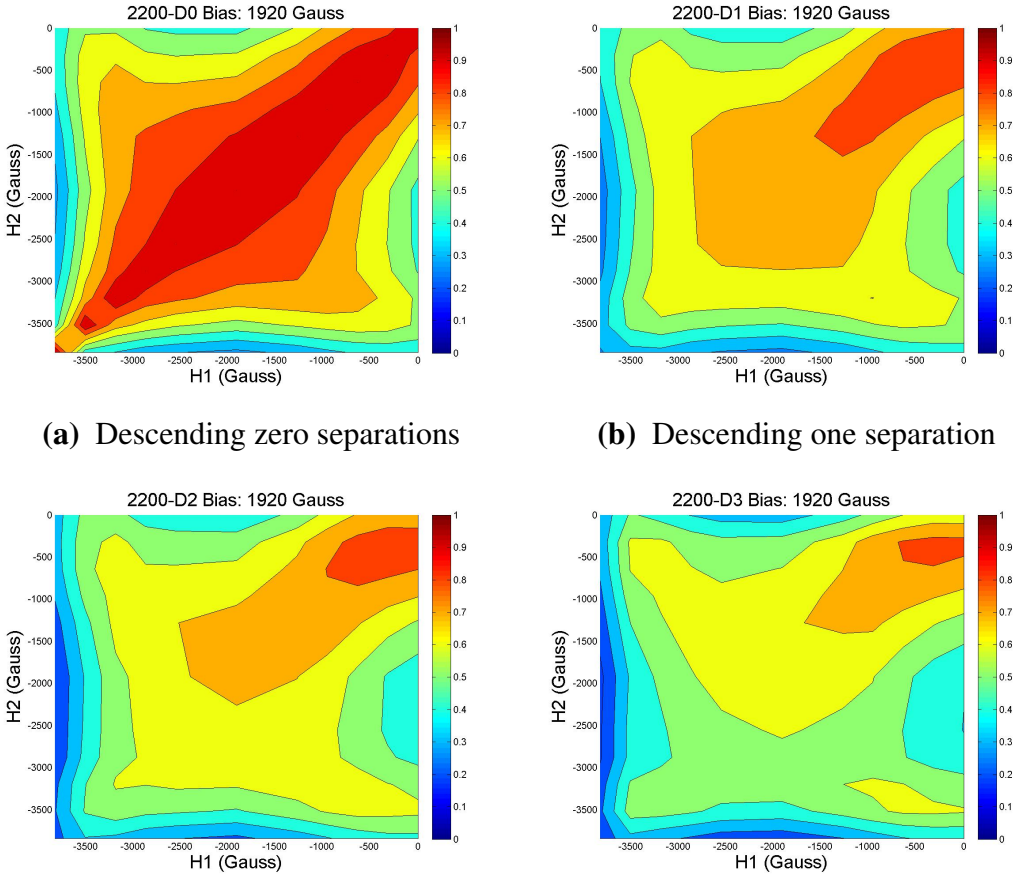
(a) Ascending zero separations

(b) Ascending one separation

(c) Ascending two separations

(d) Ascending three separations

Figure 3.16 Ascending averaged maps for the 2200 series.



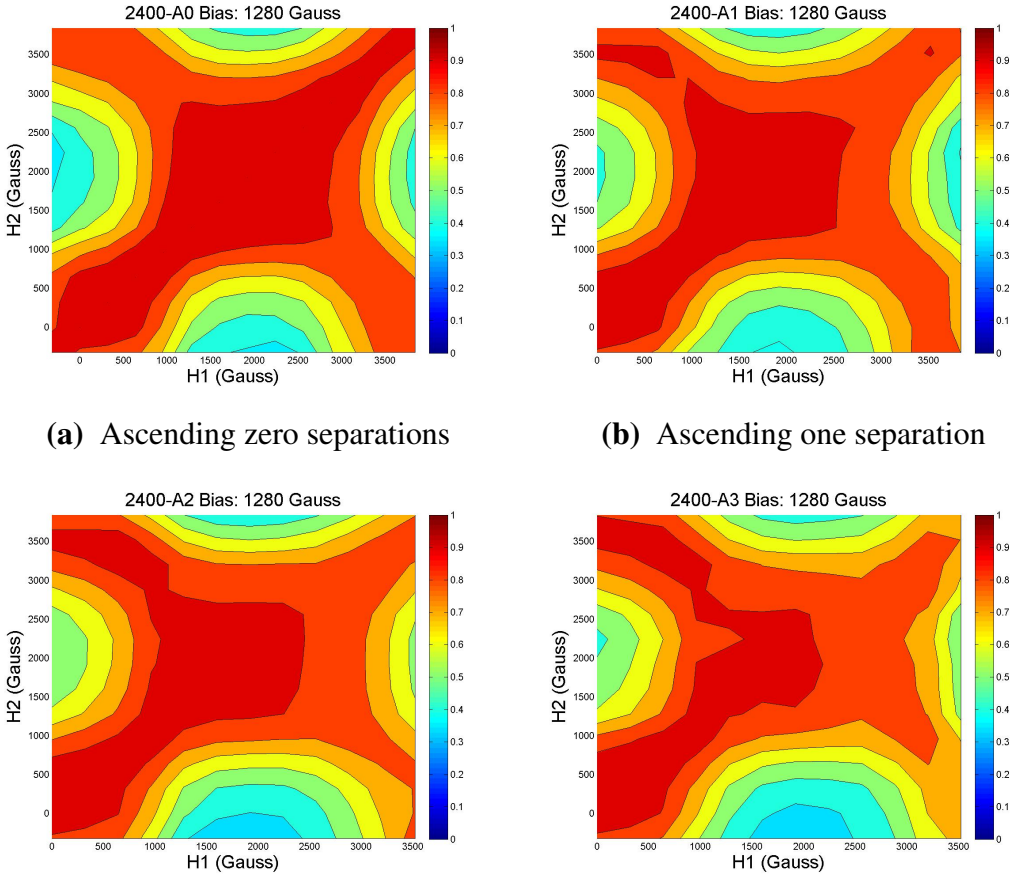
(a) Descending zero separations

(b) Descending one separation

(c) Descending two separations

(d) Descending three separations

Figure 3.17 Descending averaged maps for the 2200 series.



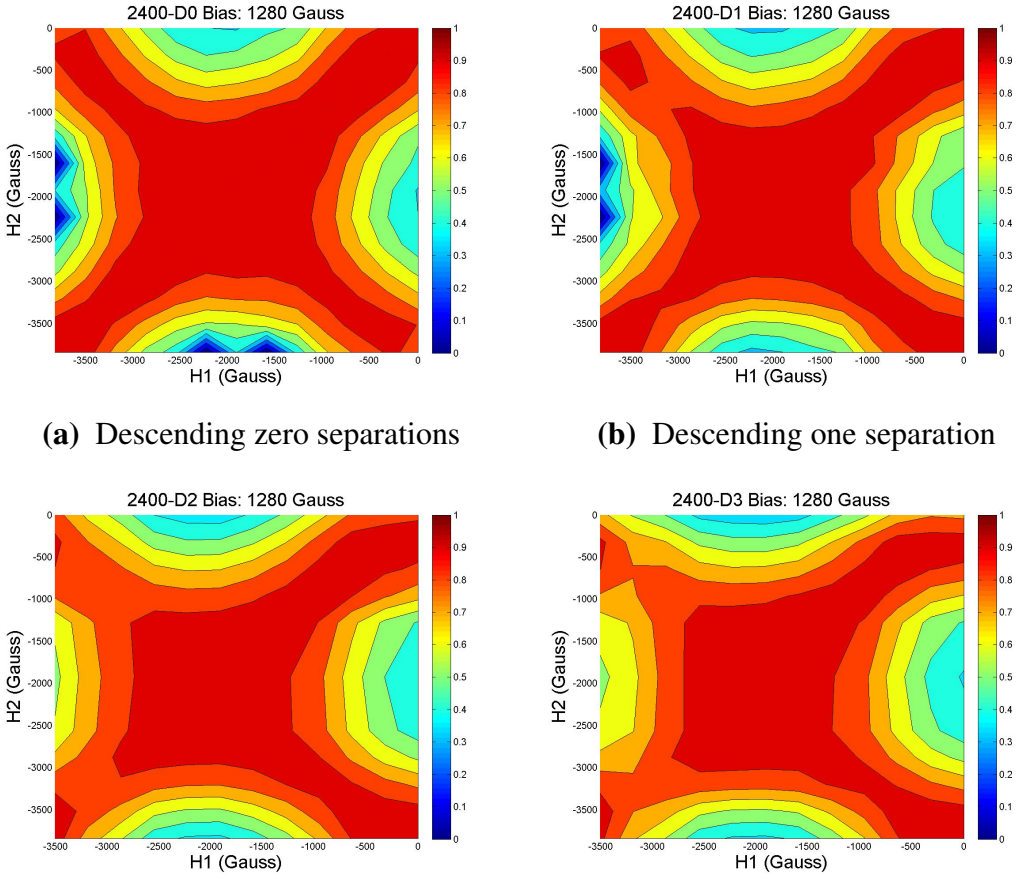
(a) Ascending zero separations

(b) Ascending one separation

(c) Ascending two separations

(d) Ascending three separations

Figure 3.18 Ascending averaged maps for the 2400 series.



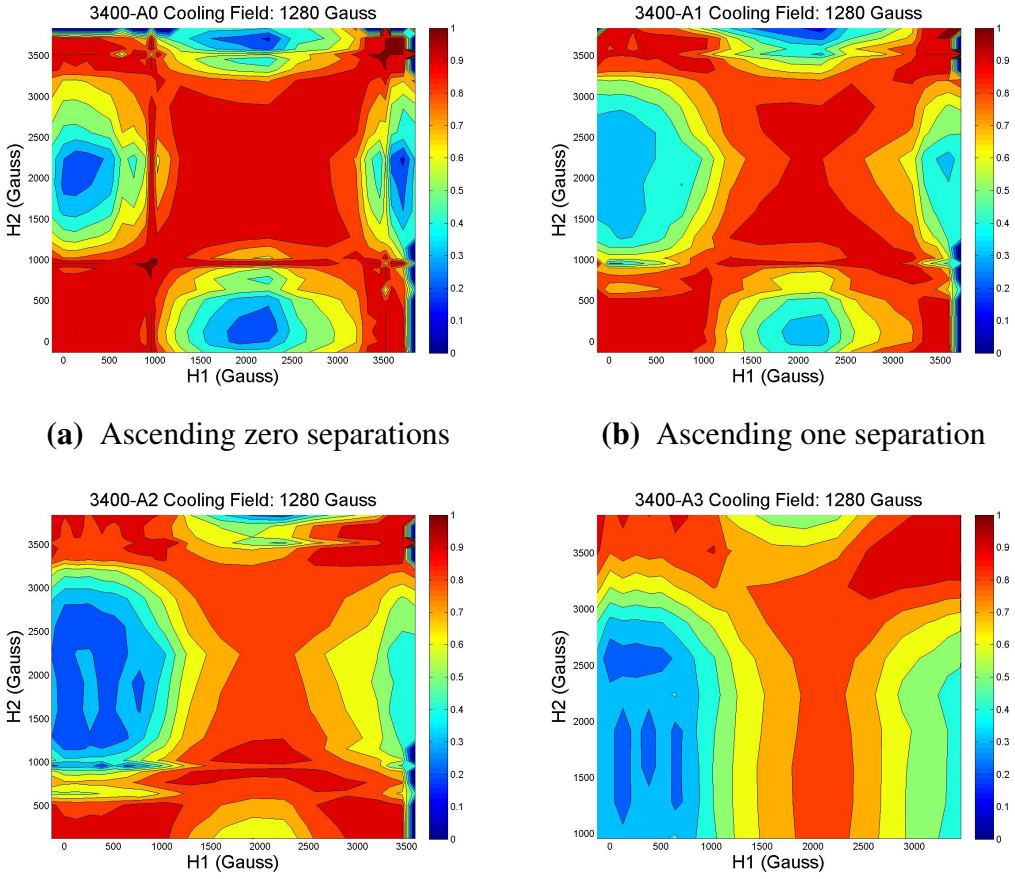
(a) Descending zero separations

(b) Descending one separation

(c) Descending two separations

(d) Descending three separations

Figure 3.19 Descending averaged maps for the 2400 series.



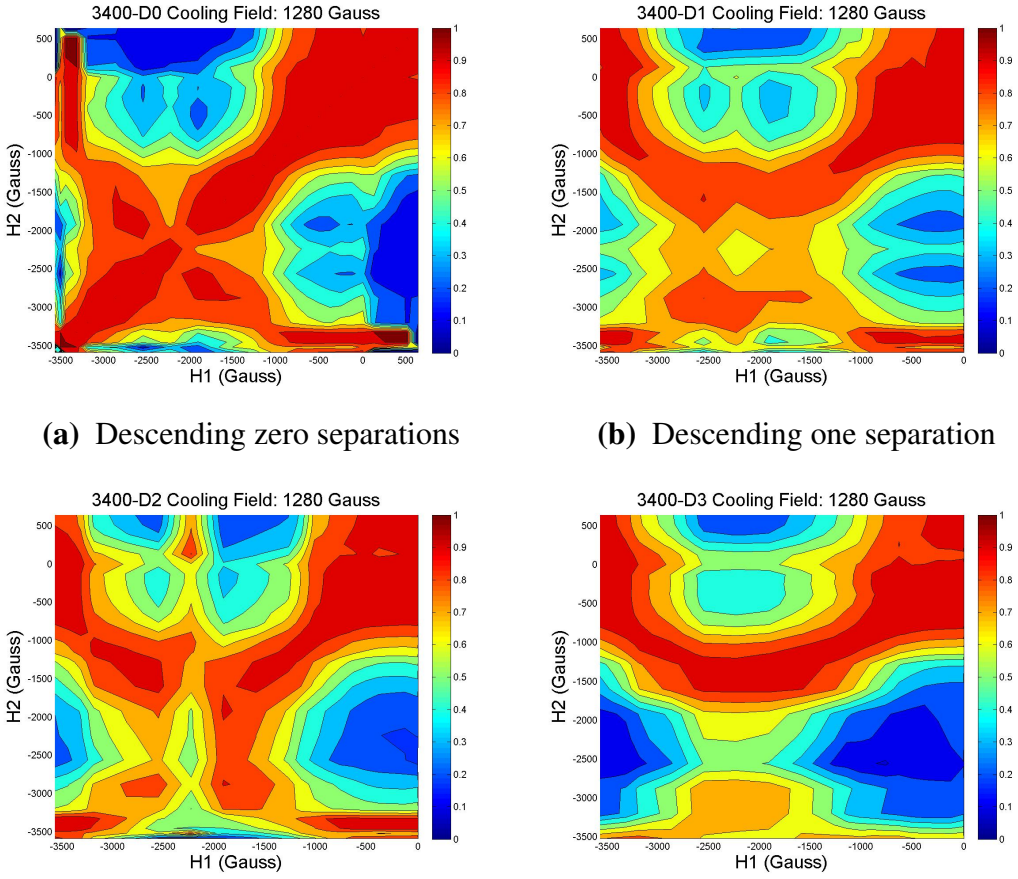
(a) Ascending zero separations

(b) Ascending one separation

(c) Ascending two separations

(d) Ascending three separations

Figure 3.20 Ascending averaged maps for the 3400 series.



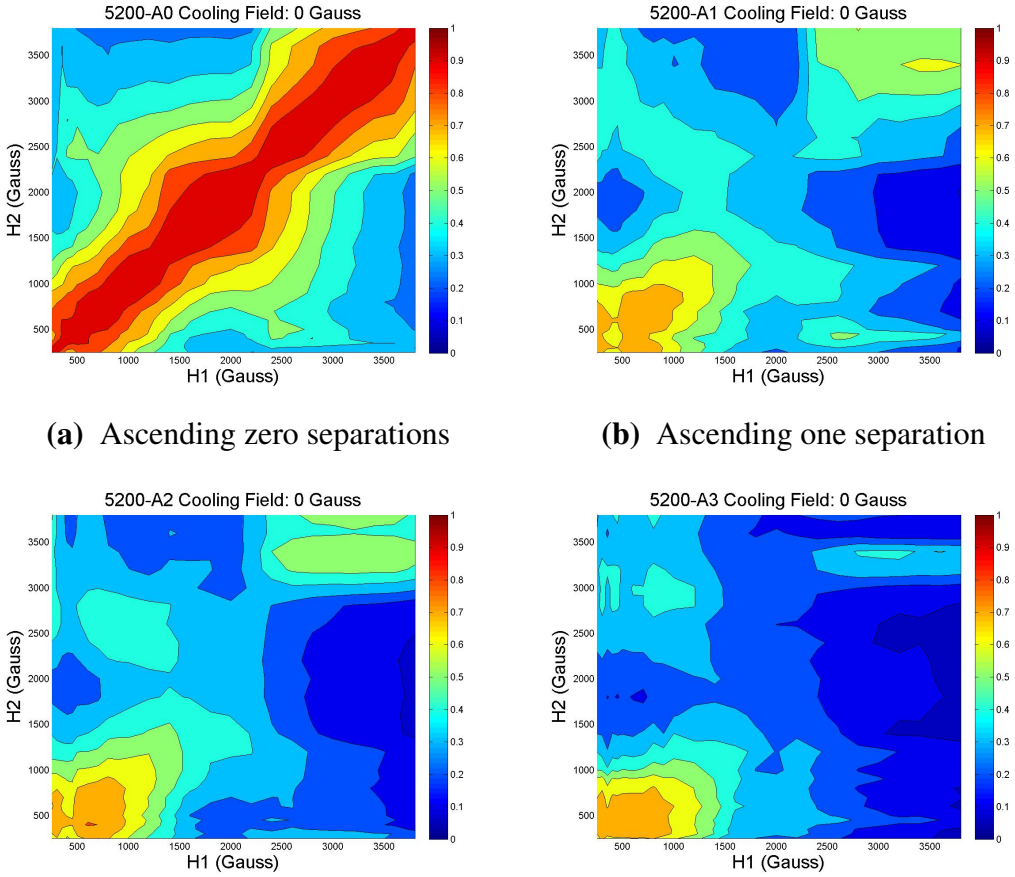
(a) Descending zero separations

(b) Descending one separation

(c) Descending two separations

(d) Descending three separations

Figure 3.21 Descending averaged maps for the 3400 series.



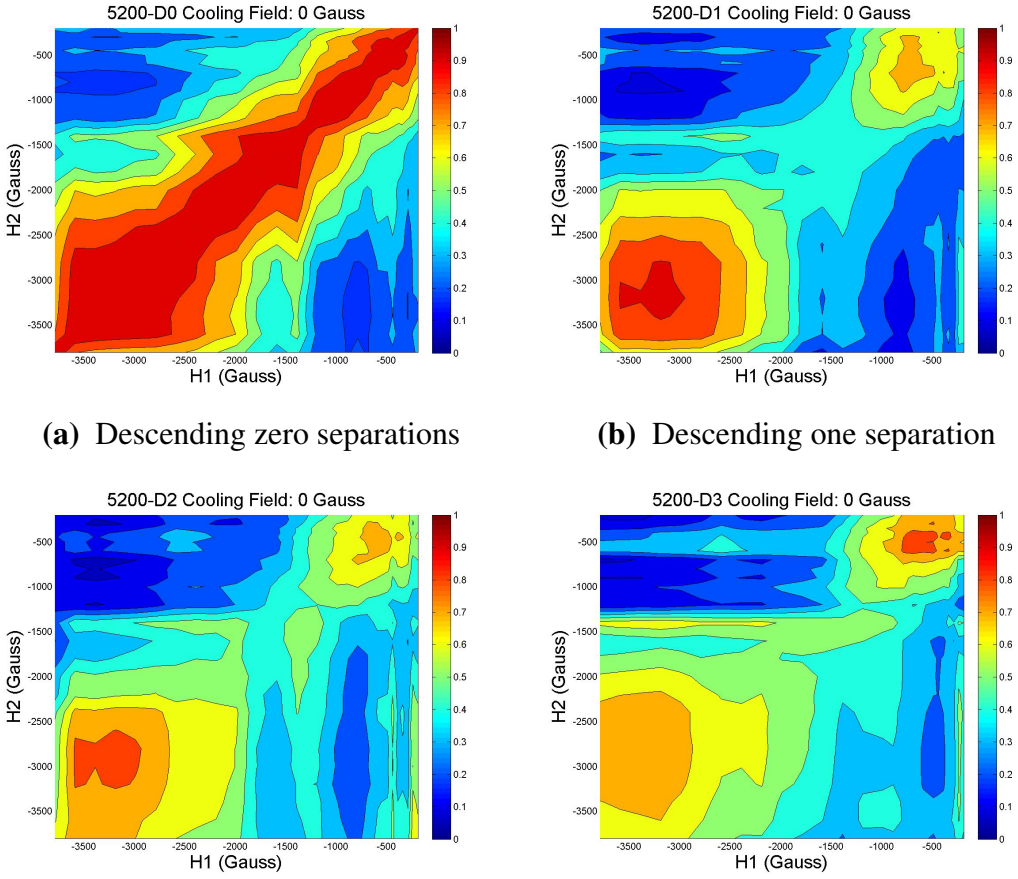
(a) Ascending zero separations

(b) Ascending one separation

(c) Ascending two separations

(d) Ascending three separations

Figure 3.22 Ascending averaged maps for the 5200 series.



(a) Descending zero separations

(b) Descending one separation

(c) Descending two separations

(d) Descending three separations

Figure 3.23 Descending averaged maps for the 5200 series.

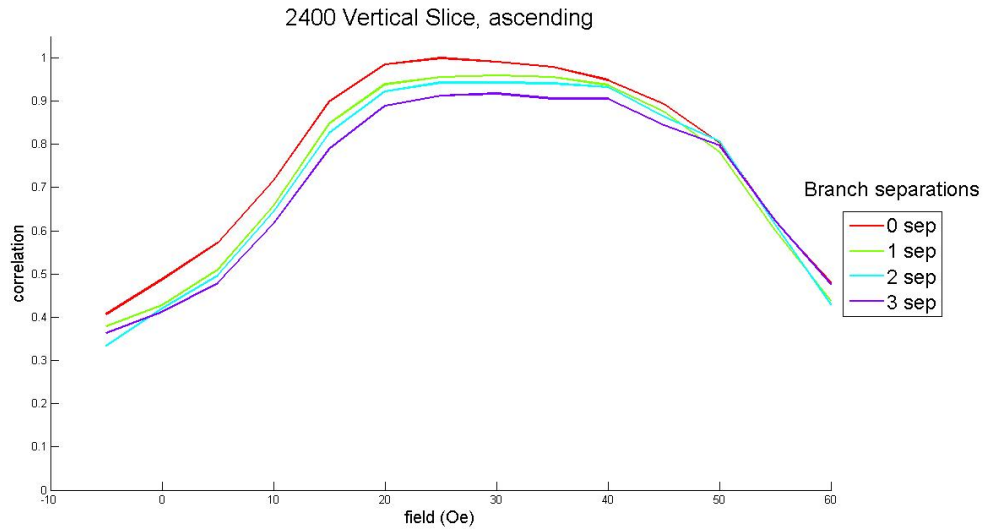


Figure 3.24 Vertical slices for each ascending branch separation of the 2400 series at the middle pixel. Cooling field here is 1280 Oe.

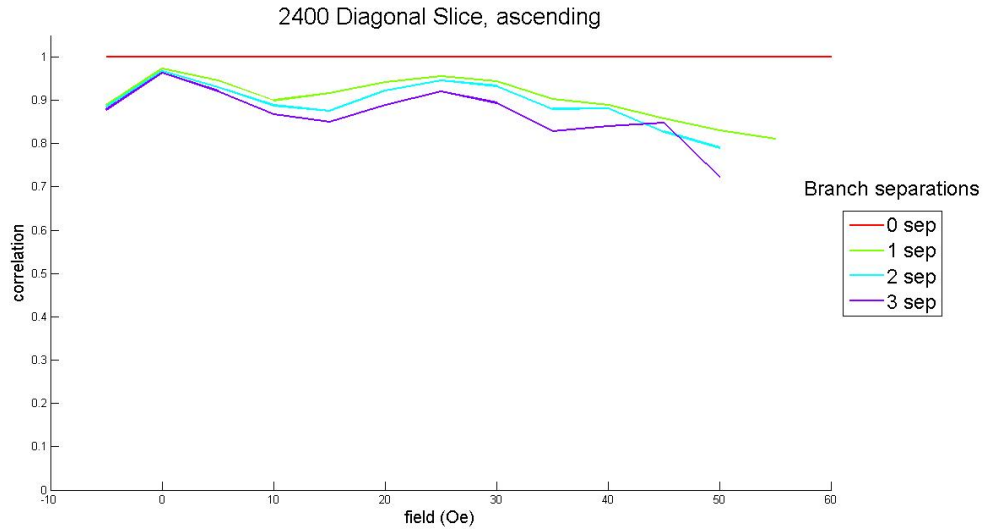


Figure 3.25 Diagonal slices for each ascending branch separation of the 2400 series. Note that the zero separation slice is a constant. This is because the diagonal of a zero separation map consists of autocorrelations. Cooling field here is 1280 Oe.

So we know that cooling field *has* a significant effect, but what is it? To determine what the effect of significant factors are, we can construct Tukey HSD 95% overall confidence intervals.

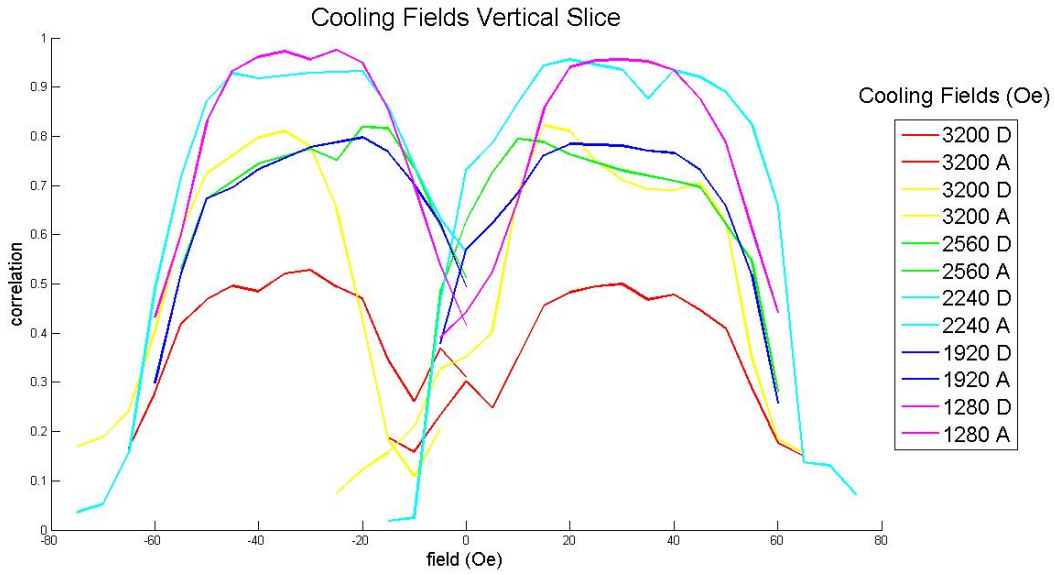


Figure 3.26 Each of the vertical slices for both ascending and descending one separation maps. We have omitted the data from the 3400 series, which was cooled under 640 Oe, because it has some strange behavior, which may not be physical. We observe that, in general, series with a higher cooling field have lower correlation. The vertical slice is at the center pixel.

Map max ρ	D.F.	Sum Sq.	Mean Sq.	F-value	$\mathscr{P}(>F)$
Cooling Field	4	0.3591	0.0898	51.6464	<0.0001
Separation	5	0.0233	0.0047	2.6767	0.0263
Interaction	9	0.0344	0.0038	2.1956	0.0291
Residuals	93	0.1617	0.0017		

Table 3.5 ANOVA table for the map maxima. Note that every effect is significant at the 95% confidence level.

There are hundreds of such intervals for each effect and every interaction over both response variables. For the sake of brevity, we present only the confidence intervals for the effect of cooling field on map maximum and map mean correlation. These confidence intervals are given in Table 3.8.

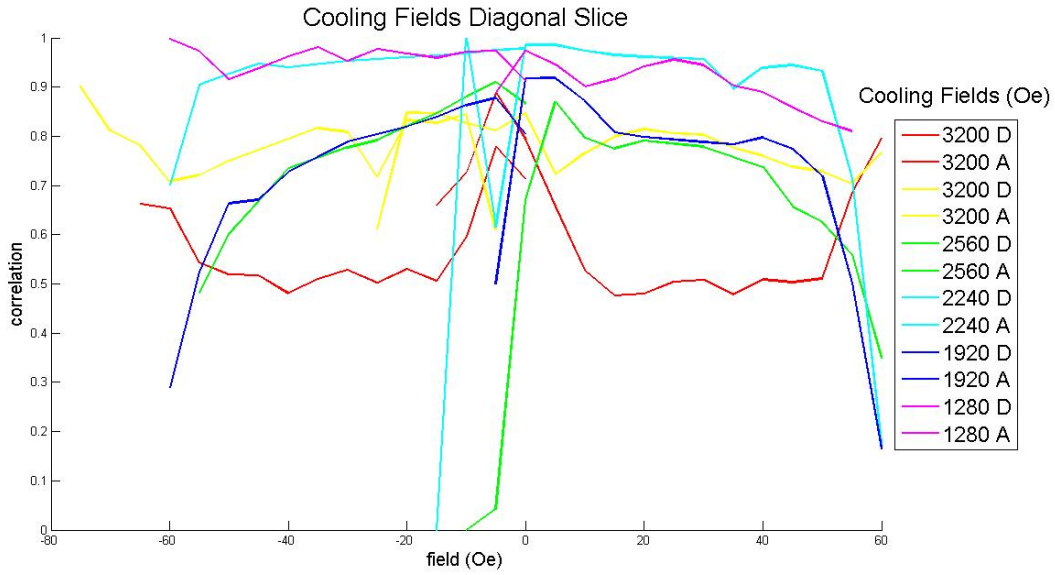


Figure 3.27 Each of the diagonal slices for both ascending and descending one separation maps. We have omitted the data from the 3400 series, which was cooled under 640 Oe, because it has some strange behavior, which may not be physical. We observe that, in general, series with a higher cooling field have lower correlation.

Map mean ρ	D.F.	Sum Sq.	Mean Sq.	F-value	$\mathscr{P}(>F)$
Cooling Field	4	1.2733	0.3183	86.6464	<0.0001
Separation	5	0.1352	0.0270	7.3581	<0.0001
Interaction	9	0.0037	0.0004	0.1130	0.9993
Residuals	93	0.3417	0.0037		

Table 3.6 ANOVA table for the map means. Note that both main effects are significant at the 95% confidence level.

Each row in this table shows the difference between the average values for map mean correlation for two cooling field values, along with a lower and upper bound. We are 95% confident that the true value of the difference between the average map mean correlation for these two cooling field levels lies between these bounds.

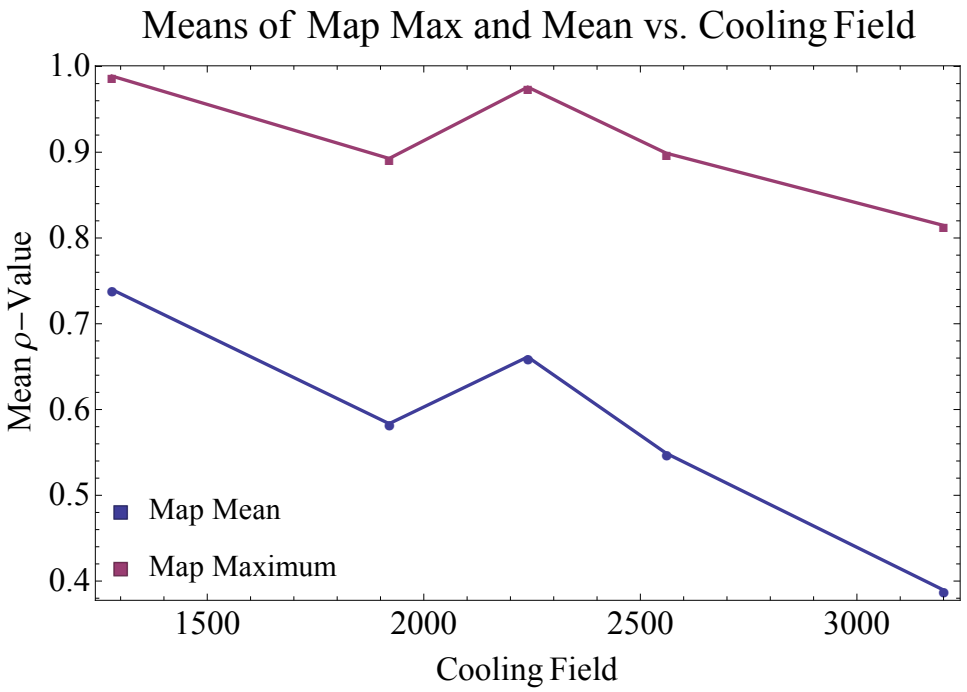


Figure 3.28 The effect of cooling field on the means of map maxima and map means for each cooling field value.

For each case where the confidence interval does not contain zero, we conclude that cooling field does have an effect. Note that in almost every case, the difference between a larger cooling filed map mean and lower cooling field map mean is negative. This means that higher cooling fields result in lower map mean correlations. For example, all else being equal, if ρ_1 is the average map mean of the 3200 Oe maps, and ρ_2 is the average map mean of the 1280 maps, then we are 95% confident that $29\% < \rho_2 - \rho_1 < 40\%$.

To summarize the results of this statistical study, cooling field has an effect on the map maximum and map mean correlation values. A higher cooling field almost always results in lower correlations.

Difference	$\Delta\hat{\rho}$ estimate	Lower bound for $\Delta\hat{\rho}$	Upper bound for $\Delta\hat{\rho}$	p-value
$\hat{\rho}_{1920} - \hat{\rho}_{1280}$	-0.096	-0.137	-0.055	<0.0001
$\hat{\rho}_{2240} - \hat{\rho}_{1280}$	-0.013	-0.048	0.022	0.847
$\hat{\rho}_{2560} - \hat{\rho}_{1280}$	-0.090	-0.120	-0.060	<0.0001
$\hat{\rho}_{3200} - \hat{\rho}_{1280}$	-0.174	-0.213	-0.135	<0.0001
$\hat{\rho}_{2240} - \hat{\rho}_{1920}$	0.083	0.041	0.126	<0.0001
$\hat{\rho}_{2560} - \hat{\rho}_{1920}$	0.006	-0.032	0.044	0.991
$\hat{\rho}_{3200} - \hat{\rho}_{1920}$	-0.078	-0.124	-0.032	<0.0001
$\hat{\rho}_{2560} - \hat{\rho}_{2240}$	-0.077	-0.109	-0.046	<0.0001
$\hat{\rho}_{3200} - \hat{\rho}_{2240}$	-0.161	-0.202	-0.121	<0.0001
$\hat{\rho}_{3200} - \hat{\rho}_{2560}$	-0.084	-0.120	-0.048	<0.0001

Table 3.7 Tukey HSD 95% confidence intervals for the effect of cooling field on the map maximum correlation value. We are 95% confident that the true difference between the average map maximum correlation values of each pair of cooling fields given here lies between the given bounds. Here, $\hat{\rho}_{cf}$ denotes the average map maximum for maps with cooling field cf.

3.4 Conclusion

We began with a goal of determining the effects of a cooling field on magnetic memory. After carrying out experiments to measure magnetic domain memory in the presence of a cooling field, we have determined that magnetic domain memory persists even in high cooling field, but to a diminished degree. The observed decrease in magnetic domain memory when a biasing field is applied may be explained as following: when the film is cooled in the absence of field (zero field cooling), there are about the same number of up and down domains in the imprinted pattern in the antiferromagnetic layer, therefore leaving a large degree of freedom for the domains in the ferromagnetic layer to nucleate at random locations and grow in a way to eventually match the

Difference	$\Delta\overline{\rho}$ estimate	Lower bound for $\Delta\overline{\rho}$	Upper bound for $\Delta\overline{\rho}$	p-value
$\overline{\rho}_{1920} - \overline{\rho}_{1280}$	-0.1564	-0.2160	-0.0967	<0.0001
$\overline{\rho}_{2240} - \overline{\rho}_{1280}$	-0.0785	-0.1296	-0.0275	0.0004
$\overline{\rho}_{2560} - \overline{\rho}_{1280}$	-0.1910	-0.2341	-0.1478	<0.0001
$\overline{\rho}_{3200} - \overline{\rho}_{1280}$	-0.3503	-0.4070	-0.2936	<0.0001
$\overline{\rho}_{2240} - \overline{\rho}_{1920}$	0.0778	0.0162	0.1394	0.0060
$\overline{\rho}_{2560} - \overline{\rho}_{1920}$	-0.0346	-0.0898	0.0206	0.4114
$\overline{\rho}_{3200} - \overline{\rho}_{1920}$	-0.1940	-0.2603	-0.1276	<0.0001
$\overline{\rho}_{2560} - \overline{\rho}_{2240}$	-0.1125	-0.1583	-0.0666	<0.0001
$\overline{\rho}_{3200} - \overline{\rho}_{2240}$	-0.2718	-0.3305	-0.2130	<0.0001
$\overline{\rho}_{3200} - \overline{\rho}_{2560}$	-0.1593	-0.2114	-0.1073	<0.0001

Table 3.8 Tukey HSD 95% confidence intervals for the effect of cooling field on the map mean correlation value. We are 95% confident that the true difference between the average map mean correlation values of each pair of cooling fields given here lies between the given bounds. Here, $\overline{\rho}_{cf}$ denotes the average map mean for maps with cooling field cf.

underlying antiferromagnetic pattern, but when the film is cooled in the presence of magnetic field (this is, the cooling field), there is an unbalance between the up and down domains in the imprinted antiferromagnetic pattern, reducing the ability for the domains in the ferromagnetic pattern to match this underlying pattern. As a result, the overall magnetic memory is somewhat reduced when the sample is field cooled. These hypotheses are confirmed by quantitative and statistical analysis.

As always, we will seek to confirm these conclusions through further experiment. The data obtained at APS is meant to confirm this. Indeed there is less magnetic memory as cooling field increases in the APS data, but other features of the APS data are inconsistent with several previous

experiments. We do not yet know the cause of this discrepancy. Perhaps there was an error in the experimental setup, or perhaps samples 1 and 2 are different in some way from sample 0. Whatever the case, we will seek to better understand these results through further study in future experiments.

We are also always in search of methods to improve the computational process for analyzing magnetic memory data. The biggest challenge to the computational process is the identification and removal of the blocker. As described in Section 2.2.4, we will attempt to build a robust interpolation function designed to "guess" what data would have been in the region of the blocker. This will allow us to perform all of the analysis without regard for the pixels in the neighborhood of the blocker. Other improvements to the computational process include complete automation of the process. Next to the blocker finding and removal, the computational step requiring the most human involvement is the identification of the proper level of smoothing. We will seek to automate this and other computational steps.

Moving forward, there are several questions about magnetic domain memory and our samples which require further study to answer:

- Which of the three samples exhibits the most magnetic domain memory?
- Why do samples 1 and 2 behave differently than sample 0?
- What do magnetic memory maps look like for minor loops?
- How does magnetic memory persist at higher temperature?

We will answer these questions and more through further experiment. Though there is still work to do, the characterization of magnetic memory under cooling field conditions is an important step towards an understanding of magnetic domain memory.

Appendix A

Proofs

A proof of Proposition 2.2.3.

Proposition A.0.1. *Let f be a function $f : \mathbb{R}^2 \to R$ which vanishes outside of $[0,a) \times [0,b)$ for $a,b > 0$. Then the periodic summation of f, $\tilde{f}(x,y)$, is periodic in both x and y.*

Proof. Fix $(x,y) \in \mathbb{R}^2$. Let N be the largest integer such that $Na < x$, and let M be the largest

integer such that $Mb < y$. Then

$$\begin{aligned}
\tilde{f}(x,y) &= \sum_{n=-\infty}^{\infty}\sum_{m=-\infty}^{\infty} f(x-na, y-mb) \\
&= \sum_{n=-\infty}^{\infty} (\ldots + f(x-na, y-(M-1)b) + f(x-na, y-Mb) + f(x-na, y-(M+1)b) + \ldots) \\
&= \sum_{n=-\infty}^{\infty} (\ldots + 0 + f(x-na, y-Mb) + 0 + \ldots) \\
&= \sum_{n=-\infty}^{\infty} f(x-na, y-Mb) \\
&= \ldots + f(x-(N-1)a, y-Mb) + f(x-Na, y-Mb) + f(x-(N+1)a, y-Mb) + \ldots \\
&= \ldots + 0 + f(x-Na, y-Mb) + 0 + \ldots \\
&= f(x-Na, y-Mb).
\end{aligned}$$

If we translate by a in the x direction, we see

$$\begin{aligned}
\tilde{f}(x+a,y) &= \sum_{n=-\infty}^{\infty}\sum_{m=-\infty}^{\infty} f(x+a-na, y-mb) \\
&= \sum_{n=-\infty}^{\infty} (\ldots + f(x+a-na, y-(M-1)b) + f(x+a-na, y-Mb) \\
&\qquad + f(x+a-na, y-(M+1)b) + \ldots) \\
&= \sum_{n=-\infty}^{\infty} (\ldots + 0 + f(x+a-na, y-Mb) + 0 + \ldots) \\
&= \sum_{n=-\infty}^{\infty} f(x+a-na, y-Mb) \\
&= \ldots + f(x+a-Na, y-Mb) + f(x+a-(N+1)a, y-Mb) + f(x+a-(N+2)a, y-Mb) + \ldots \\
&= \ldots + f(x-(N-1)a, y-Mb) + f(x-Na, y-Mb) + f(x-(N+1)a, y-Mb) + \ldots \\
&= \ldots + 0 + f(x-Na, y-Mb) + 0 + \ldots \\
&= \tilde{f}(x,y).
\end{aligned}$$

If we translate by b in the y direction, we see

$$\begin{aligned}
\tilde{f}(x,y+b) &= \sum_{n=-\infty}^{\infty}\sum_{m=-\infty}^{\infty} f(x-na,y+b-mb) \\
&= \sum_{n=-\infty}^{\infty} (\ldots + f(x-na,y+b-Mb) + f(x-na,y+b-(M+1)b) \\
&\qquad + f(x-na,y+b-(M+2)b) + \ldots) \\
&= \sum_{n=-\infty}^{\infty} (\ldots + f(x-na,y+(M-1)b) + f(x-na,y-Mb) + f(x-na,y-(M+1)b) + \ldots) \\
&= \sum_{n=-\infty}^{\infty} (\ldots + 0 + f(x-na,y-Mb) + 0 + \ldots) \\
&= \sum_{n=-\infty}^{\infty} f(x+a-na,y-Mb) \\
&= \ldots + f(x-(N-1)a,y-Mb) + f(x-Na,y-Mb) + f(x-(N+1)a,y-Mb) + \ldots \\
&= \ldots + 0 + f(x-Na,y-Mb) + 0 + \ldots \\
&= \tilde{f}(x,y).
\end{aligned}$$

A symmetric argument is sufficient to show that $\tilde{f}(x-a,y) = \tilde{f}(x,y)$ and $\tilde{f}(x,y-b) = \tilde{f}(x,y)$. Thus f is doubly periodic. □

Appendix B

Code

Here, I present original MATLAB code I wrote to expedite the analysis process. First, this script assists in locating the blocker.

```
clear;
close all;
clc;


series=5200;
imagenum=5322;
imloc=['\\physics\Shares\Research\Magnetic Speckle\Alex\Speckle...
\data\eb' num2str(imagenum) '.fit'];
im=fitsread(imloc);
s=size(im);
multiplier=50000;
m=multiplier;
```

```
%ellipse values

    %center coordinates
x = 644;
y = 660;

    %axes
a = 100;    %verticaly
b = 100;    %horizontaly

phi = 0;%rotation angle

%tetragon values

x1 = 1;%first corner // top left
y1 = 678;

x2 = 525; %second corner //top right
y2 = 640;

x3 = 536; %third corner //bottom right
y3 = 715;

x4 = 1;%fourth corner // bottom left
y4 = 757;
```

```
e=getEllipse(x,y,a,b,phi,s,0)*m;
t=getTetragon(x1,y1,x2,y2,x3,y3,x4,y4,s,0)*m;
imb=im+e+t;%image with blocker added

%image without blocker
fig1=figure('Name','Image without blocker','NumberTitle','off');
imagesc(im)
caxis([min(min(im)),max(max(im))]);

%image with blocker
fig2=figure('Name','Image with blocker','NumberTitle','off');
imagesc(imb)
caxis([min(min(im)),max(max(im))]);
```

This script allows the user to view an image smoothed to several degrees in order to determine the optimal amount of smoothing.

```
clear;
close all;
clc;

tols=[.01,.05];%Tolerances to test
X=zeros(length(tols),1289,1289);%This is where we will store the data
%index counter
i=1;
```

```
s=5200;%series
im=5323;%image #

%Get the raw data from the data folder
raw=fitsread(['\\physics\Shares\Research\Magnetic Speckle\Alex\Speckle...
\data\eb',num2str(im),'.fit']);

for t=tols
    %This runs the smoothing if it has not already been done
    PreProcessor(s,t,'single',im);
    %Formatting the tolerance string
    ts=num2str(t,'%7.6f');
    %Subtract the speckle from the raw data to get the envelope
    Speckel=getfield(load(['\\physics\Shares\Research\Magnetic Speckle\Alex...
    \Speckle\GeneratedDataStore\',num2str(s),'series\speckle ',ts,'\eb',num2str(im)
    ,'.mat']),['eb',num2str(im)]);
    X(i,:,:)=raw-getfield(load(['\\physics\Shares\Research\Magnetic Speckle...
    \Alex\Speckle\GeneratedDataStore\',num2str(s),'series\speckle ',ts,'\eb',num2str(im)
    ,'.mat']),['eb',num2str(im)]);
    figure('Name',['dtol: ' num2str(t)],'NumberTitle','off');
    l=1:1289;
    %plot the raw data and envelope
    plot(l,reshape(X(i,:,650),size(l)),l,raw(:,650))
    i=i+1;
end
```

This script allows the user to choose a degree of smoothing, and automatically smooth an entire series to that degree.

```
function GroupPreProcess(series,tol)
[~,~,ex]=getImagePreProcessTools(series);
aname=['\\physics\Shares\Research\Magnetic Speckle\Alex\Speckle...
\XRMS program\dataListings\imageListings\ascend' num2str(series) '.mat'];
dname=['\\physics\Shares\Research\Magnetic Speckle\Alex\Speckle...
\XRMS program\dataListings\imageListings\descend' num2str(series) '.mat'];
a=getfield(load(aname),'a');
d=getfield(load(dname),'d');

s=size(a);
for i=1:s(1)
    for j=1:s(2)
        im=a(i,j);
        if (sum(find(ex==im))==0 && im~=0)
            try
                PreProcessor(series,tol,'single',im)
            catch
                err=lasterror;
                disp(err);
                disp(err.message);
                disp(err.stack);
                disp(err.identifier)
            end
```

```
        end
    end
end


s=size(d);
for i=1:s(1)
    for j=1:s(2)
        im=d(i,j);
        if (sum(find(ex==im))==0 && im~=0)
            try
                PreProcessor(series,tol,'single',im)
            catch
                err=lasterror;
                disp(err);
                disp(err.message);
                disp(err.stack);
                disp(err.identifier)
            end
        end
    end
end
```

This script generates an averaged map. It also eliminates rows and columns consisting of zeros, which sometimes occur when an image is skipped.

```
function [map,xfields,yfields]=AvgMapMaker(series,dTol,btype,sep)
```

```
intFile=['\\physics\Shares\Research\Magnetic Speckle\Alex\Speckle...
\GeneratedDataStore\' num2str(series) 'series\integrations...
\integrations '  num2str(dTol,'%7.6f') '.mat'];
rhos=getfield(load(intFile),'integrations');

aname=['\\physics\Shares\Research\Magnetic Speckle\Alex\Speckle...
\XRMS program\dataListings\imageListings\ascend' num2str(series) '.mat'];
dname=['\\physics\Shares\Research\Magnetic Speckle\Alex\Speckle...
\XRMS program\dataListings\imageListings\descend' num2str(series) '.mat'];
faname=['\\physics\Shares\Research\Magnetic Speckle\Alex\Speckle...
\XRMS program\dataListings\fieldListings\fieldsA' num2str(series) '.mat'];
fdname=['\\physics\Shares\Research\Magnetic Speckle\Alex\Speckle...
\XRMS program\dataListings\fieldListings\fieldsD' num2str(series) '.mat'];

a=getfield(load(aname),'a');
d=getfield(load(dname),'d');
fieldsA=getfield(load(faname),'fieldsA');
fieldsD=getfield(load(fdname),'fieldsD');

errorCheck=1;
if strcmpi('a',btype(1))
    s=size(a);
    images=a;
    errorCheck=0;
    fields=fieldsA(:,1);
```

```
end
if strcmpi('d',btype(1))
    s=size(d);
    images=d;
    errorCheck=0;
    fields=fieldsD(:,1);
end

if errorCheck==1 || sep>=s(2)
    disp 'branch type not recognized'
else
    N=s(2)-sep;
    maps=zeros(s(1),s(1),N);

    for k=1:N
        im1=images(:,k);
        im2=images(:,k+sep);
        for i=1:s(1)
            for j=1:s(1)
                if im1(i)==0 || im2(j)==0
                    maps(i,j,k)=0;
                    continue
                end

                fs=['eb' num2str(im1(i)) 'eb' num2str(im2(j))];
```

```
ns1=['eb' num2str(im1(i)) 'eb' num2str(im1(i))];
ns2=['eb' num2str(im2(j)) 'eb' num2str(im2(j))];

try
    f=getfield(rhos,fs);
    n1=getfield(rhos,ns1);
    n2=getfield(rhos,ns2);
    score=f/sqrt(n1*n2);
    if score>2
        score=0;
    end
    if score>1
        score=1;
    end

    maps(i,j,k)=score;
catch
    maps(i,j,k)=0;
    disp 'error'
    err = lasterror;
    disp(err);
    disp(err.message);
    disp(err.stack);
    disp(err.identifier)
end
```

```
        end
    end
end
%---------------------------
assignin('base','maps',maps)
map=zeros(s(1),s(1));
for i=1:s(1)
    for j=1:s(1)
        layers=maps(i,j,:);
        if nnz(layers)==0
            cell=0;
        else
            cell=sum(layers)/nnz(layers);
        end
        map(i,j)=cell;
    end
end


   %--------------------------------------------
[xfields, yfields]=meshgrid(fields,fields);


row0=0;
col0=0;
if series==2100 && sep==0
    zeroToleranceRatio=.7;
```

```
else
    zeroToleranceRatio=2;
end
ztr=zeroToleranceRatio;
for i=1:s(1)
    if sum(map(i,:)==0)>=ztr*sum(map(i,:)>0)
        row0=row0+1;
    end
    if sum(map(i,:)==1)>=ztr*sum(map(i,:)~=1)
        row0=row0+1;
    end
end
for i=1:s(1)
    if sum(map(:,i)==0)>=ztr*sum(map(:,i)>0)
        col0=col0+1;
    end
    if sum(map(:,i)==1)>=ztr*sum(map(:,i)~=1)
        col0=col0+1;
    end
end
newmap=zeros(s(1)-row0,s(1));
new_xfields=zeros(s(1)-row0,s(1));
new_yfields=zeros(s(1)-row0,s(1));

r=1;
```

```
c=1;
for i=1:s(1)
    if sum(map(i,:)==0)<ztr*sum(map(i,:)>0) &&...
            sum(map(i,:)==1)<ztr*sum(map(i,:)~=1)
        newmap(r,:)=map(i,:);
        new_xfields(r,:)=xfields(i,:);
        new_yfields(r,:)=yfields(i,:);
        r=r+1;
    end
end

map=newmap;
xfields=new_xfields;
yfields=new_yfields;

newmap=zeros(s(1)-row0,s(1)-col0);
new_xfields=zeros(s(1)-row0,s(1)-col0);
new_yfields=zeros(s(1)-row0,s(1)-col0);

for i=1:s(1)
    if sum(map(:,i)==0)<ztr*sum(map(:,i)>0) &&...
            sum(map(:,i)==1)<ztr*sum(map(:,i)~=1)
        newmap(:,c)=map(:,i);
        new_xfields(:,c)=xfields(:,i);
        new_yfields(:,c)=yfields(:,i);
```

```
            c=c+1;
        end
    end
    map=interpolateOverZeros(newmap);
    xfields=new_xfields;
    yfields=new_yfields;

    %figure;
    %contourf(xfields,yfields,map,12)
    %caxis([0,1])
    %imagesc(map)

    assignin('base','map',map)
    assignin('base','xfields',xfields)
    assignin('base','yfields',yfields)
end
end
function map=interpolateOverZeros(map)
s=size(map);
for i=2:s(1)-1
    for j=2:s(2)-1
        if map(i,j)==0
            map(i,j)=1;
            boarders=zeros(8,1);
            a=1;
```

```
            for k=-1:1
                for l=-1:1
                    if k~=0 || l~=0
                        boarders(a,1)=map(i+k,j+l);
                        a=a+1;
                    end
                end
            end
            if prod(boarders)~=0
               map(i,j)=sum(boarders)/8;
            end
        end
    end
end
end
```

Bibliography

[1] C. Kittel, "Physical Theory of Ferromagnetic Domains," Rev. Mod. Phys. **21,** 541–583 (1949).

[2] R. M. White, *Quantum Theory of Magnetism: Magnetic Properties and Materials*, 3rd rev. ed. ed. (Springer-Verlag, Berlin, 2007).

[3] S. Hashimoto, Y. Ochiai, and K. Aso, "Perpendicular magnetic anisotropy and magnetostriction of sputtered Co/Pd and Co/Pt multilayered films," J. App. Phys. **66,** 4909–4916 (1989).

[4] D. Mauri, E. Kay, D. Scholl, and J. Howard, "Novel Method for determining the anisotropy constant of MnFe in a NiFe/MnFe sandwich," J. Appl. Phys. **62,** 2929–2932 (1987).

[5] K. Chesnel, B. Wilcken, M. Rytting, S. D. Kevan, and E. E. Fullerton, "Field mapping and temperature dependence of magnetic domain memory induced by exchange couplings," New Journal of Physics **15,** 023016 (2013).

[6] J. Eckert, N. Stern, A. Barton, D. Mann, and P. Sparks, "Low temperature properties of spin valves with extremely thin IrMn," J. Appl. Phys. **91,** 8569–8571 (2002).

[7] J. Nogues and I. Schuller, "Exchange bias," J Magn. Magn. Mater. **192,** 203–232 (1999).

[8] W. H. Meiklejohn and C. P. Bean, "New Magnetic Anisotropy," Phys. Rev. **105,** 904–913 (1957).

[9] S. Maat, K. Takano, S. Parkin, and E. Fullerton, "Perpendicular exchange bias of Co/Pt multilayers," Phys. Rev. Lett. 87 (2001).

[10] M. S. Pierce et al., "Disorder-induced magnetic memory: Experiments and theories," Phys. Rev. B 75 (2007).

[11] K. Chesnel, J. A. Nelson, S. D. Kevan, M. J. Carey, and E. E. Fullerton, "Oscillating spatial dependence of domain memory in ferromagnetic films mapped via x-ray speckle correlation," Physical Review B **83,** 054436 (2011).

[12] J. Kortright, D. Awschalom, J. Stohr, S. Bader, Y. Idzerda, S. Parkin, I. Schuller, and H. Seigmann, "Research frontiers in magnetic materials at soft X-ray synchrotron radiation facilities," J. Magn. Magn. Mater. 207 (1999).

[13] K. Chesnel, E. E. Fullerton, M. J. Carey, J. H. Kortright, and S. D. Kevan, "Magnetic memory in ferromagnetic thin films via exchange coupling," Phys. Rev. B 78 (2008).

[14] K. Chesnel, J. Nelson, B. Wilcken, and S. Kevan, "Mapping spatial and field dependence of magnetic domain memory by soft X-ray speckle metrology," Journal of synchrotron radiation **19,** 293–306 (2012).

[15] J. Nelson, B. Wilcken, and K. Chesnel, "Persistence of Magnetic Domain Memory Through Field Cycling in Exchange Bias Thin Films," Jo. Utah Acad. Sci. **87,** 267–274 (2010).

Index