Improving the Contrast of Neutron Interferometry Phase Measurements Using Online Bayesian Markov Chain Monte Carlo Methods

by

Thomas Alexander

Declaration of Authorship

I, Thomas Alexander, declare that this thesis titled, "Improving the Contrast of Neutron Interferometry Phase Measurements Using Online Bayesian Markov Chain Monte Carlo Methods" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.

- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.

- Where I have consulted the published work of others, this is always clearly attributed.

- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.

- I have acknowledged all main sources of help.

- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

_____

Date:

_____

"Magic's just science that we don't understand yet."

Arthur C. Clarke

MOUNT ALLISON UNIVERSITY

Faculty of Science
Department of Physics

Bachelors of Science with Honours

by Thomas Alexander

Abstract

The Thesis Abstract is written here (and usually kept to just this page). The page is kept centered vertically so can expand into the blank space above the title too...

Acknowledgements

The acknowledgements and the people to thank go here, don't forget to include your project advisor...

Contents

List of Figures

List of Tables

Abbreviations

t transmitted beam amplitude coefficient

r reflected beam amplitude coefficient

Physical Constants

$$\text{Speed of Light} \quad c \quad = \quad 2.997\ 924\ 58 \times 10^8 \ \tfrac{m}{s}$$

Symbols

k	quantum wave number	m^{-1}
v	velocity	ms^{-1}
ω	angular frequency	rads^{-1}

For/Dedicated to/To my...

Chapter 1

Introduction

1.1 Neutron Interferometry

1.1.1 History

Interferometry has long been a powerful tool for experimental physics. Its various forms have been used in the discovery of many historically significant results such as the Michelson-Morley experiment which showed that the speed of light was independent of inertial reference frames and experimental data in support of Bell's Inequality. [4][5]

The key concept of interferometry is the superposition of waveforms upon each other in order to deduce meaningful physical properties from the resultant combination. If one considers two waves of identical frequency than the waves when superimposed will combine constructively when in phase and de-constructively when out of phase. The technique of interferometry can be applied to many different experimental systems, the requirement being that the interferometry medium be described as a wave mathematically. Such systems that have been used in the past include electromagnetic waves, water waves, electrons and neutrons. Although electrons and neutrons classically are described as point particles the development of quantum mechanics allows that all matter is actually described by a waveform and therefore interferometry techniques may be applied to the electron and neutron waveforms. This paper focuses primarily on neutron interferometry.

The first Neutron Interferometer with slow neutrons was constructed by Maier-Leibnitz and Springer in 1962 and was effectively equivalent to a double slit experiment. However, their interferometer was not effective for measuring physical properties of materials. In 1965 the perfect single-crystal interferometer was theorized by Ulrich Bonse

and Michael Hart, however it was not until 1974 that their interferometer was made functional by Helmut Raunch and his student Wolfgang Treimer. Their interferometer used a single perfect crystal in which two horizontal slices were removed from the interior to form a three-blade interferometer.[6] Using the single-crystal design researchers Colella,Overhauser and Werner to perform the famous COW experiment which measured the phase shift due to the gravitational potential difference between two neutron beams separated by a small displacement in height.[1] Further experiments made such contributions to experimental physics such as the measurement of the Aharonov-Bohm effect and the the effect of the Earth's rotation on a quantum system.[6] It was quickly realized that neutron interferometry measurements provide an incredible level of accuracy and isolation in experimental measurements. This is due to the fact that the neutron has essentially zero electric charge and therefore does not feel the Coulomb force. Therefore for the case of slow neutrons there is no need to isolate for stray electric fields.
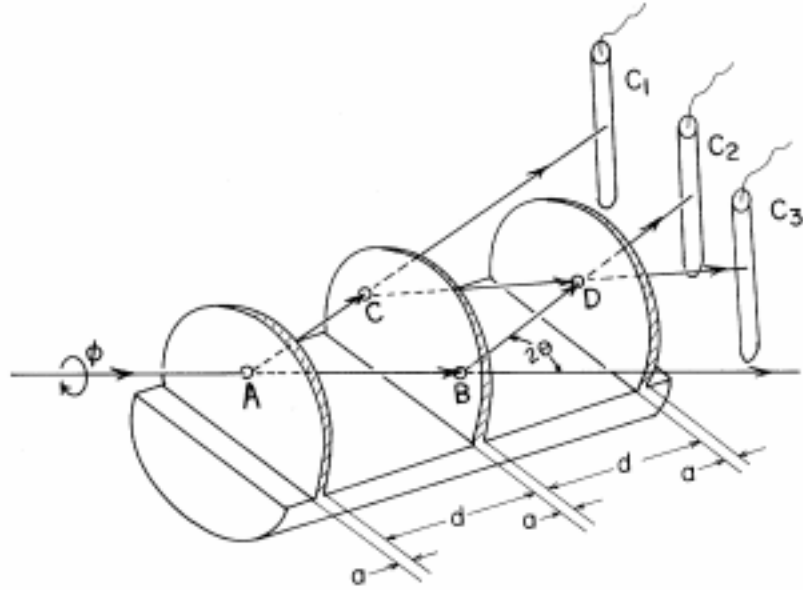
FIGURE 1.1: An illustration of the interferometer used in the COW experiment.[1]

1.1.2 Application to Quantum Information

As the neutron interferometry provides a low-noise experimental system it provides an ideal test-bench for testing certain aspects of quantum information theory. Such an example was the use of a five-blade interferometer which allowed the quantum information encoded in the neutron waveform by using additional blades to exploit the symmetry of mechanical vibrations in the interferometer and decouple these modes.[7]. This is an example of encoding the information into a decoherence-free subspace and is a technique

that may be applicable in future scalable quantum computation systems. Additionally it has been shown that neutron interferometers can be used for the generation of single neutron entangled states. [8] Additionally there is interest in the quantum discord of neutron interferometry systems and there application towards non-classical discord algorithms.[9]. It is unlikely that a scalable quantum computer will be realizable with neutrons due to their low interaction with other quantum systems.

1.1.3 Application to Quantum Fundamentals

Neutron interferometry has played a large role in experimentally gathering information on the fundamental behaviour of quantum systems. Such as the Aharonov-Bohm effect, the effect of gravity,quantum discord and verifying Bell's Inequality. [6][1][9][5]. More recently researchers at the Institute for Quantum Computing are designing an experimental neutron interferometer that is equivalent to a triple-slit experiment in the search for third order interference effects that are theoretically non-existent but if found may be evidence of new quantum theories.[10]

1.1.4 National Institute of Standards and Technology

The majority of the work presented in this thesis applies directly to the neutron interferometry setup at the National Institute of Standards and Technology in Gaithersburg, MD. The neutrons are produced at the NIST Research Reactor and extracted via a dual-crystal parallel-tracking monochromator with energy of $4 - 20meV$. They are fed along wave-guides to the isolated interferometry setup. NIST has three,four and five blade perfect single-crystal interferometry assemblies although we focus on solely the three blade assembly. Neutron detection is provided by ^3He detectors or by high resolution position-sensitive detectors.[11][12]

1.2 Markov Chain Monte Carlo Methods

Often in advanced quantum physics the probability distributions of the systems reside in a high dimensional parameter space. It is often desired to find the correct set of parameters to describe a system. However, it is normally impossible to test every single possible real value of the parameter space as the computational time would be astronomical. Markov Chain Monte Carlo methods attempt to solve this problem by using a bunch of "guesses" that will according to a heuristic attempt to reasonably explore the parameter space for the approximate parameters after many iterations. Markov Chain
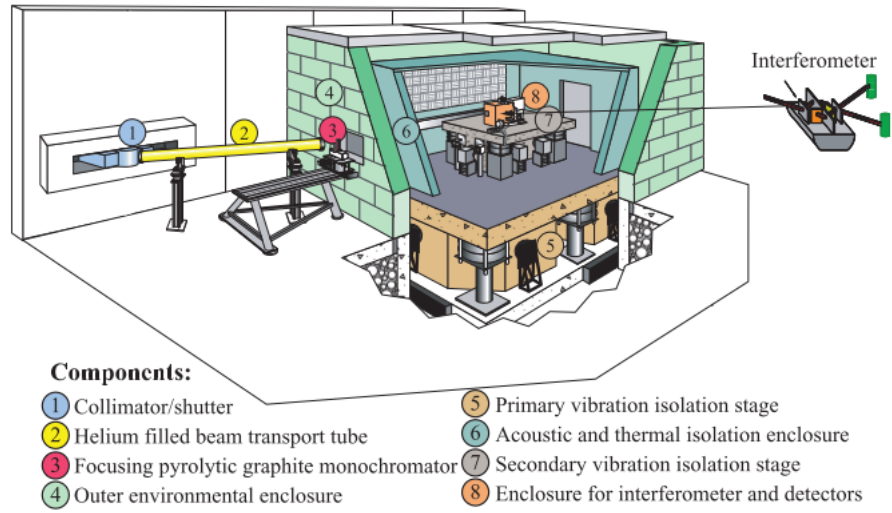
FIGURE 1.2: Neutron interferometry lab at NIST [2]

Monte Carlo Methods have many uses including in Bayesian statistics, physics, biology and linguistics.[13]

The Markov Chain Monte Carlo method can be thought of scattering a large number of particles in a space to be explored. By moving these particles around the space, it is possible to effectively sample the entire space if enough iterations and particles are used. We will use MCMC methods to explore the parameter space of neutron detection probability distributions in order to obtain the optimal set of system parameters.
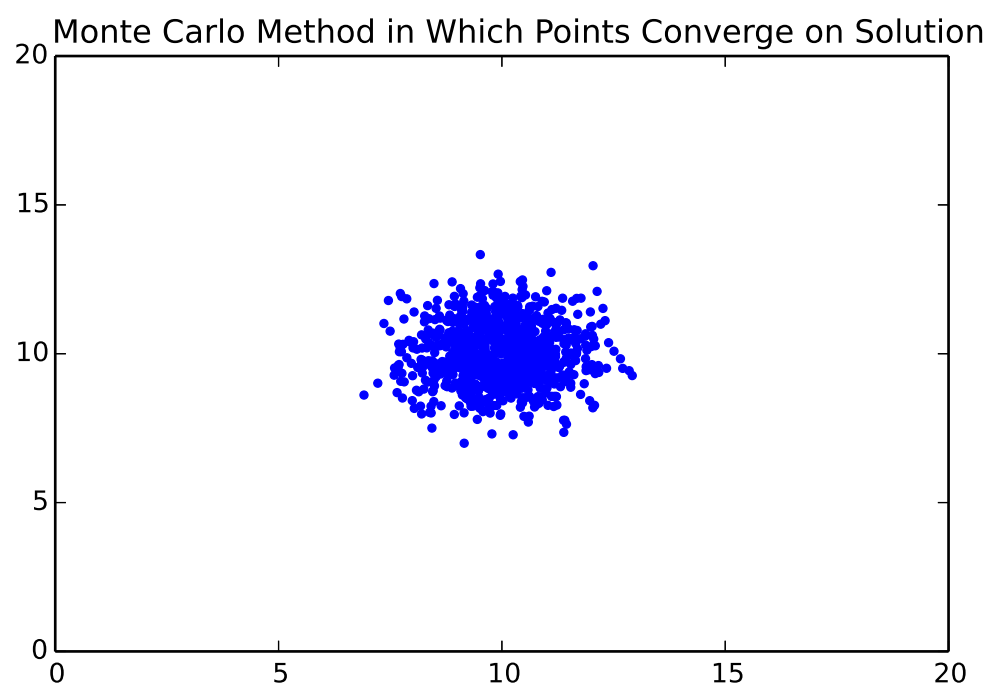
FIGURE 1.3: Markov Chain Monte Carlo method converging on solution at $(10, 10)$

Chapter 2

Theory

2.1 Neutrons

2.1.1 Particle Description of Neutrons

The neutron is a subatomic hadron particle that is present in the nucleus of every atom except ^1H. The neutron is composed of two down quarks and a single up quarks. This composition gives a neutral electric charge for the neutron making it an ideal candidate for sensitive experiments, however the downside is that neutrons are much more difficult to manipulate. The neutron is also therefore a fermion and by the Pauli exclusion principle only a single neutron is allowed in each quantum state. The free neutron is unstable and undergoes beta decay with a lifetime of just $881.5 \pm 1.5s$. The neutron has a rest mass of approximately $939.56 Mev$. Free neutrons are produced using either neutral fission or fusion although in practical experiments fission is almost always used. At the NIST Research Reactor free neutrons are produced from the fission of ^{235}U.

2.1.2 Thermal Neutrons

Neutron interferometry utilizes thermal neutrons which are free neutrons that follow a Boltzmann distribution. The neutrons at NIST are found in the kinetic energy range of 4-20meV around room temperature of $T = 293.15K$. This gives gives neutron velocities of $875 - 1956 \frac{m}{s}$ which gives $v << c$ and therefore relativistic affects do not play a role. Therefore thermal neutrons are in near thermal equilibrium with their surroundings. Neutrons are decelerated to a thermal state in the reactor by collisions with neutron moderators in the reactor. From de Broglie relations the wavelength of thermal neutrons is approximately $\lambda = \frac{h}{p} = $ 2.0-4.5$\mathring{A}$. After being emitted form the NIST reactor the

neutrons follow a wave-guide and using a wave splitter are sent into individual labs. As the strongest known phase space density of a neutron source is around 10^-14 it can be safely assumed that the probability of two neutrons interacting inside the wave-guide or interferometer is sufficiently low that it can be disregarded and therefore detected neutrons have no correlation between each-other.

2.2 Neutron Interferometry

The Neutron interferometer is similar to other forms of interferometry in which an incoming wave is split and than allowed to interfere at a later point which allows the two wave paths to be compared. The modern day neutron interferometer is functionally equivalent to an optical Mach-Zender (MZ) interferometer.[14]

2.2.1 Bragg Scattering

In neutron interferometry the crystal planes of the interferometer blades act as diffraction gratings. Incident waves that satisfy the Bragg condition 2.1 are coherently scattered.

$$n\lambda = 2dsin(\theta_b) \tag{2.1}$$

Where n is a positive integer, d is the distance between the atomic planes of the crystal lattice and θ_b is the angle between the incident beam and the atomic plane of the crystal. Not all of the amplitude of the neutron beam is scattered and a large portion is transmitted through the crystal with only its phase shifted. The amplitudes of the transmitted and the reflected beams are given by the coefficients t (transmitted) and r reflected. Under the assumption that the entire beam is either Bragg scattered or transmitted the coefficients t and r must have the following relationship.[2]

$$|r|^2 + |t|^2 = 1 \tag{2.2}$$

2.2.2 Quantum Scattering Theory

Starting with the assumption the Hamiltonian has the form of

$$\mathcal{H} = \mathcal{H}_0 + \mathcal{V} \quad \mathcal{H}_0 = \frac{\mathbf{p}^2}{2m} \tag{2.3}$$
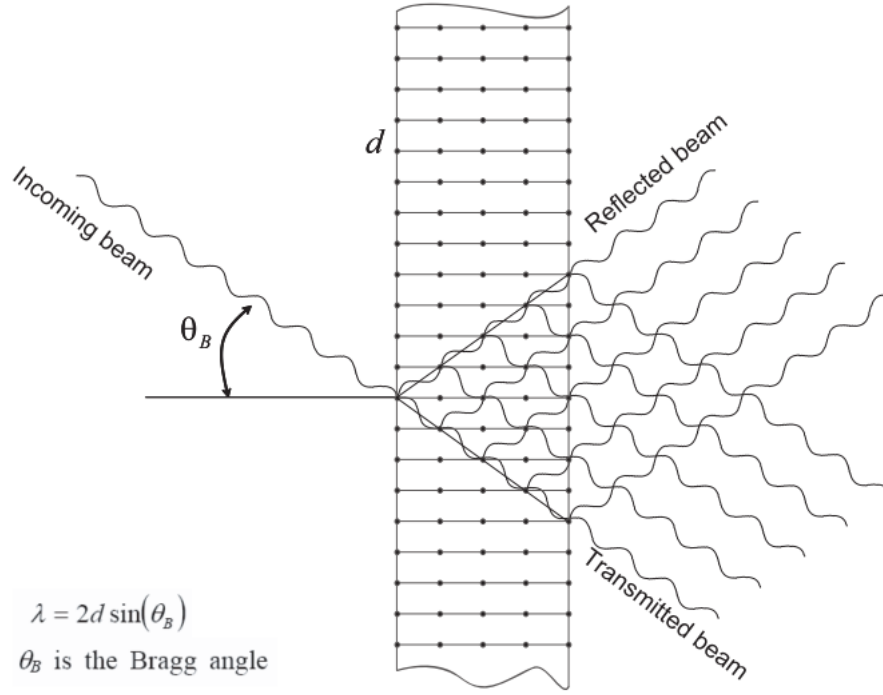
FIGURE 2.1: Bragg scattering off of a crystal lattice

The presence of the potential $\mathcal{V}$ causes the solution to be different than the free particle state

$$\mathcal{H}_0 \ket{\Phi} = E \ket{\Phi}$$

Therefore we are looking for solutions to the Schrdinger equation of the form

$$\mathcal{H}_0 + \mathcal{V} \ket{\Psi} = E \ket{\Psi} \tag{2.4}$$

A valid solution should have that $\ket{\Psi} \to \ket{\Phi}$ as $\mathcal{V} \to 0$. A solution that satisfies these requirements is known as the Lippmann-Schwinger equation.

$$\ket{\Psi^{\pm}} = \ket{\Phi} + \frac{1}{E - \mathcal{H}_0 \pm i\epsilon} \mathcal{V} \ket{\Psi^{\pm}} \tag{2.5}$$

Here the energy E was made slightly complex with the addition of $\pm\epsilon$ to deal with the singular nature of the operator $1/(E - \mathcal{H}_0)$. It can easily be seen that the application of the operator $E - \mathcal{H}_0$ reduces (2.5) to the desired solution (2.4) when neglecting the imaginary component. By taking the Lipmann-Schwinger equation to the position basis

explicitly it can be represented as

$$\langle \boldsymbol{x} \,|\, \Psi^{\pm} \rangle = \langle \boldsymbol{x} \,|\, \Phi \rangle - \frac{2m}{\hbar^2} \int d^3 x' \frac{e^{\pm ik\left|\boldsymbol{x} - \boldsymbol{x}'\right|}}{4\pi \left|\boldsymbol{x} - \boldsymbol{x}'\right|} \left\langle \boldsymbol{x}' \,\middle|\, \mathcal{V} \,\middle|\, \Psi^{\pm} \right\rangle \tag{2.6}$$

As our scattering potentials are a function of position only the assumption can be made that the potential is *local* such that it is diagonal in the position representation. Specifically the potential satisfies the requirement that

$$\langle \boldsymbol{x}' |\mathcal{V}| \boldsymbol{x}'' \rangle = \mathcal{V}(\boldsymbol{x}')\delta^{(3)}(\boldsymbol{x}' - \boldsymbol{x}'') \tag{2.7}$$

Utilizing this potential we obtain

$$\left\langle \boldsymbol{x} \,\middle|\, \mathcal{V} \,\middle|\, \Psi^{\pm} \right\rangle = \int d^3 x'' \, \langle \boldsymbol{x}' |\mathcal{V}| \boldsymbol{x}'' \rangle \, \langle \boldsymbol{x}'' |\Psi^{\pm} \rangle = \mathcal{V}(\boldsymbol{x}') \, \langle \boldsymbol{x}' |\Psi^{\pm} \rangle \tag{2.8}$$

With this result the Lippmann-Schwinger equation can be reduced to

$$\langle \boldsymbol{x} \,|\, \Psi^{\pm} \rangle = \langle \boldsymbol{x} \,|\, \Phi \rangle - \frac{2m}{\hbar^2} \int d^3 x' \frac{e^{\pm ik\left|\boldsymbol{x} - \boldsymbol{x}'\right|}}{4\pi \left|\boldsymbol{x} - \boldsymbol{x}'\right|} \mathcal{V}(\boldsymbol{x}') \left\langle \boldsymbol{x}' \,\middle|\, \Psi^{\pm} \right\rangle \tag{2.9}$$

Given that we are concerned with studying finite range scatters and that any observations that will be made will be made outside the range of the potential due to the macroscopic nature of neutron detectors the assumption can be made that $|\boldsymbol{x}| >> \left|\boldsymbol{x}'\right|$.
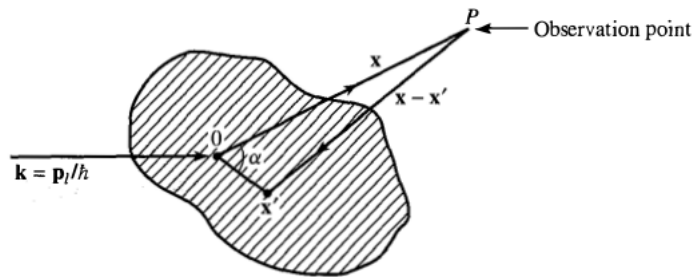
FIGURE 2.2: The finite range scattering potential. Any observations via detectors will be outside the range of the potential and therefore approximations can be made when evaluating (2.9).

Keeping in mind this result we can define

$$r = |\boldsymbol{x}|$$

$$r^{'} = \left| \boldsymbol{x}^{'} \right|$$

$$\alpha = \angle(\boldsymbol{x}, \boldsymbol{x}^{'})$$

$$\hat{\boldsymbol{r}} \equiv \frac{\boldsymbol{x}}{|\boldsymbol{x}|}$$

$$\left| \boldsymbol{x} - \boldsymbol{x}^{'} \right| \approx r - \hat{\boldsymbol{r}} \cdot \boldsymbol{x}^{'} \tag{2.10}$$

$$\boldsymbol{k}^{'} \equiv k\hat{\boldsymbol{r}} \tag{2.11}$$

Utilizing equations (2.10,2.11)

$$e^{\pm ik\left| \boldsymbol{x} - \boldsymbol{x}^{'} \right|} \approx e^{\pm ikr} e^{\mp i\boldsymbol{k}^{'} \cdot \boldsymbol{x}^{'}} \tag{2.12}$$

For the distant r at the observation point it is a useful approximation to say that

$$\frac{1}{\left| \boldsymbol{x} - \boldsymbol{x}^{'} \right|} \approx \frac{1}{r} \tag{2.13}$$

Now replacing our incident generic wave with an incident plane wave $|\Phi\rangle \rightarrow |\boldsymbol{p}\rangle$ and using $\boldsymbol{k} \equiv \boldsymbol{p_i}/\hbar$ To remove the $\hbar$'s from the expression. We obtain for the first term in (2.9)

$$\langle \boldsymbol{x} \,|\, \boldsymbol{k} \rangle = \int d^3 k^{'} \left\langle \boldsymbol{x} \,\middle|\, \boldsymbol{k}^{'} \right\rangle \left\langle \boldsymbol{k}^{'} \,\middle|\, \boldsymbol{k} \right\rangle = \int d^3 k^{'} \left\langle \boldsymbol{x} \,\middle|\, \boldsymbol{k}^{'} \right\rangle \delta^{(3)}(\boldsymbol{k}^{'} - \boldsymbol{k}) = \frac{e^{i\boldsymbol{k} \cdot \boldsymbol{x}}}{(2\pi)^{\frac{3}{2}}} \tag{2.14}$$

Using this result in (2.9) gives an expression for the scattered wave function at a relatively distant observation point for the positive Lippmann-Schwinger wavefunction.

$$\left\langle \boldsymbol{x} \,\middle|\, \Psi^{+} \right\rangle = \frac{1}{(2\pi)^{\frac{3}{2}}} \left(e^{i\boldsymbol{k} \cdot \boldsymbol{x}} + \frac{e^{ikr}}{r} f(\boldsymbol{k}^{'}, \boldsymbol{k}) \right) \tag{2.15}$$

$$f(\boldsymbol{k}^{'}, \boldsymbol{k}) = -m \left(\frac{2\pi}{h} \right)^2 \left\langle \boldsymbol{k}^{'} \,\middle|\, \mathcal{V} \,\middle|\, \Psi^{+} \right\rangle \tag{2.16}$$

It is very easy to see that the result wavefunction is a combination of the original incident plane-wave and an outgoing spherical wave with an amplitude described by (2.16). An obvious issue is that here scattering has only been treated for an incident plane-wave which is not a normalizable wavefunction. In reality to describe discrete particles such as neutrons wave packet solutions are used to describe the incident particles. However, provided the size of the wave packet is much larger than the range of the finite potential $\mathcal{V}$ it is sufficient to treat an incident packet as a plane-wave.

2.2.3 Differential Scattering Cross-Section

The scattering cross section is an important parameter for experimental scattering physics. It relates the number of particles scattering into the solid angle $d\Omega$ per unit time to the number of incident particles into an infinitesimal element $d\sigma$ of area per unit time. We search for a relation between $d\Omega$ and $d\sigma$ which we term the differential cross section given by $d\sigma/d\Omega$. Evidently the probability of an incident particle being within
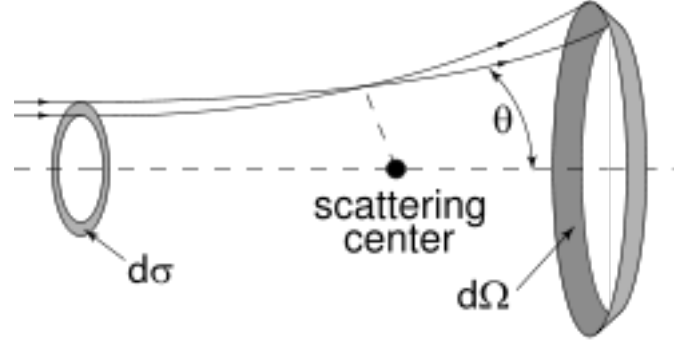
FIGURE 2.3: The differential cross section is the relationship between incident particles travelling through area $d\sigma$ to scattered particles crossing through the solid angle $d\Omega$

an area $d\sigma$ in time dt while travelling with velocity v is just

$$dP = |\Psi_i|^2 \, dV = \frac{1}{2\pi}^3 (vdt)d\sigma \tag{2.17}$$
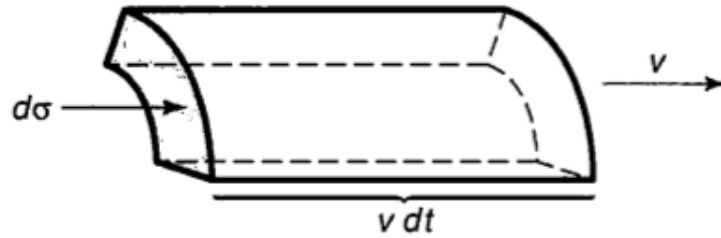
FIGURE 2.4: The volume element dV that a beam occupies passing through an area $d\sigma$ in time dt

Relating this probability to the probability of a particle being scattered into solid angle $d\Omega$ with equal velocity v per unit time dt.

$$dP = |\Psi_s|^2 \, dV = \frac{1}{2\pi}^3 \frac{\left|f(\boldsymbol{k}',\boldsymbol{k})\right|^2}{r^2}(vdt)r^2 d\Omega \tag{2.18}$$

Equations (2.17) and (2.18) can be solved for the differential cross section

$$\frac{d\sigma}{d\Omega} = \left|f(\boldsymbol{k}',\boldsymbol{k})\right|^2 \tag{2.19}$$

2.2.4 Scattering Amplitude

While equation (2.16) defines the magnitude of the outgoing spherical wave, it is defined implicitly in terms of the unknown ket $|\Psi^+\rangle$. The solution to this problem in the case of sufficiently weak scatterers is to use the first Born approximation

$$\left\langle \boldsymbol{x}' \,\middle|\, \Psi^+ \right\rangle \to \left\langle \boldsymbol{x}' \,\middle|\, \Phi \right\rangle = \frac{e^{i\boldsymbol{x}'}}{(2\pi)^{3/2}} \tag{2.20}$$

Combining (2.20) and (2.16) results in the first-order Born amplitude

$$f^{(1)}(\boldsymbol{k}', \boldsymbol{k}) = -\frac{m}{2\pi\hbar^2} \int d^3x' \, e^{i(\boldsymbol{k}-\boldsymbol{k}')\cdot\boldsymbol{x}'} \mathcal{V}(\boldsymbol{x}') \tag{2.21}$$

As the potentials that will be dealt with are spherically symmetrical, further approximations can be made utilizing $\boldsymbol{q} \equiv \boldsymbol{k} - \boldsymbol{k}'$ and

$$\left|\boldsymbol{k} - \boldsymbol{k}'\right| \equiv q = 2k\sin\left(\frac{\theta}{2}\right)$$

as seen in fig(2.5) The spherical symmetry can be used to integrate explicitly the angular component of the scattering magnitude.
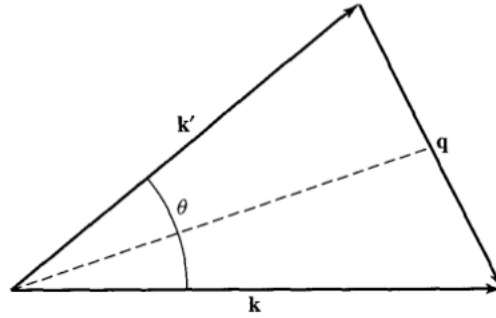
FIGURE 2.5: Scattering amplitude as a function of θ where $\boldsymbol{q} = \boldsymbol{k} - \boldsymbol{k}'$

$$f^{(1)}(\theta) = -\frac{m}{2\pi\hbar^2} \int_{r=0}^{\infty} \int_{\phi=0}^{2\pi} \int_{\theta'=0}^{\pi} e^{i|q||r|\cos(\theta')} \sin(\theta') r^2 \mathcal{V}(r) d\theta' d\phi dr$$

$$= -\frac{1}{2i}\frac{2m}{\hbar^2 q} \int_0^{\infty} r\mathcal{V}(r)(e^{iqr} - e^{-iqr})dr = -\frac{2m}{\hbar^2}\frac{1}{q} \int_0^{\infty} r\mathcal{V}(r)\sin(qr)dr \tag{2.22}$$

Given a potential that has spherical symmetry it is now much more simple to calculate the scattering amplitude using (2.22).

2.2.5 Neutron-nucleus Scattering

Generally there are two interactions that an incident neutron on a material will experience. The interaction with the nucleus of the material atoms and which is referred to as nuclear scattering and the scattering due to interaction with unpaired electrons and their magnetic moments which is known as magnetic scattering. In practice nuclear scattering is more common as it allows the structure of solids to probed.

Given the assumptions that an incoming neutron beam will be elastically scattered and that the nucleus is fixed, the scattering will depend on the potential $V(\boldsymbol{r})$ between the nucleus and neutron. As this is interaction is due to the strong-force it is naturally occurring over a very short range, and is approximately zero at a distance of the order $\boldsymbol{r} = 10^{-15}m$. As this is much shorter than the wavelength of thermal and cold neutrons which are used in almost all scattering experiments, the nucleus acts as a point scatterer. A neutron beam can be represented as a plane wave a wave function described by (2.14) and the scattered wavefunction will take the form of (2.15,2.16).

Due to the magnitude of the difference between the wavelength of the incident neutrons and the effective acting distance of the strong-force neutron-nucleus interaction and its approximate spherical symmetry it is an acceptable approximation to use the Fermi pseudo-potential

$$\mathcal{V}(\boldsymbol{x}') = \frac{2\pi\hbar^2}{m} b\delta(\boldsymbol{x}') \tag{2.23}$$

as a scattering potential. Where b is known as the neutron scattering length and his units of meters. For the case of multiple nuclei the potential takes the form

$$\mathcal{V}(\boldsymbol{x}') = \frac{2\pi\hbar^2}{m} \sum_j b_j\delta(\boldsymbol{x}' - \boldsymbol{x_j}) \tag{2.24}$$

Using the spherical approximate Born solution for the amplitude (2.21) and the Fermi pseudo-potential the scattering amplitude can be found to be

$$f^{(1)}(\theta) = -\int d^3x' e^{i(\boldsymbol{k} - \boldsymbol{k}')\cdot\boldsymbol{x}'} \sum_j b_j\delta(\boldsymbol{x}' - \boldsymbol{x_j}) = -\sum_j b_j e^{i(\boldsymbol{q} \cdot \boldsymbol{x_j}} \tag{2.25}$$

And in the case of a single scatterer at the origin reducing to

$$f^{(1)}(\theta) = -b \tag{2.26}$$

Therefore it can be seen that the completed neutron scattered wave form is

$$\langle \boldsymbol{x} \mid \Psi^+ \rangle = \frac{1}{(2\pi)^{\frac{3}{2}}} \left(e^{i\boldsymbol{k} \cdot \boldsymbol{x}} - \sum_j b_j e^{i(\boldsymbol{q} \cdot \boldsymbol{x_j}} \frac{e^{ikr}}{r} \right) \tag{2.27}$$

From this approximate solution it is evident that the only difference between individual scatterers is their neutron scattering length b. The value b varies greatly among even neighbouring elements in the periodic table. Unfortunately the outlined theory is not strong enough to predict the scattering length and the parameters must be determined experimentally.
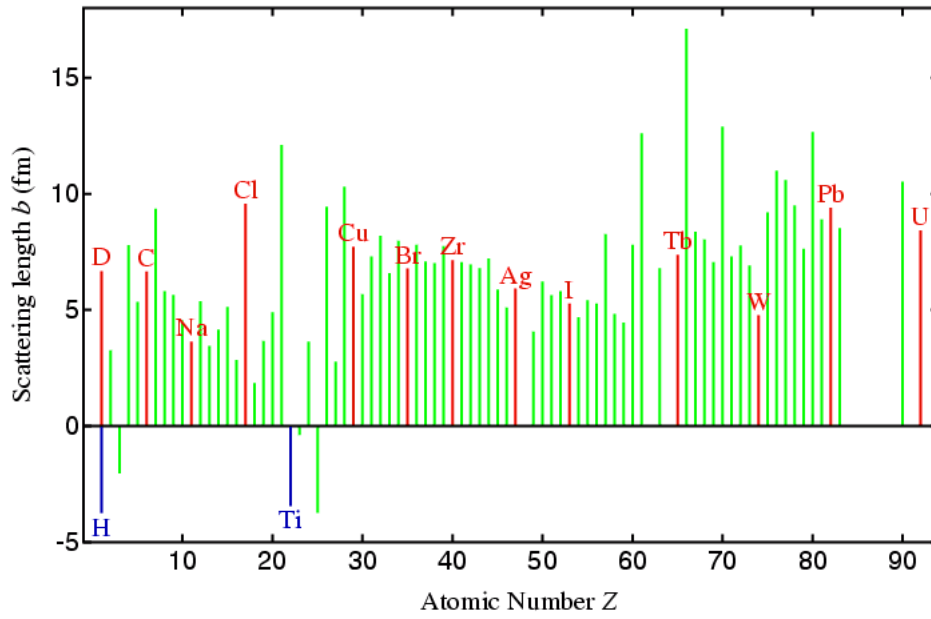
FIGURE 2.6: Neutron scattering lengths b for the elements of the periodic table

Inside a sufficiently large homogeneous material the Fermi pseudo-potential can be approximated to be

$$\mathcal{V}(\boldsymbol{x}') = \frac{2\pi\hbar^2}{m} \sum_j b_j \delta(\boldsymbol{x}' - \boldsymbol{x_j}) \approx \frac{2\pi\hbar^2}{m} bN \tag{2.28}$$

Where N is the atom number density.

2.2.6 Neutron Optics

As the neutron beam is a wavefunction many analogies from classical optics hold. The index of refraction is defined as the ratio of the speed of neutrons experiencing no

potential to the speed of neutrons affected by a potential. Compared to light the form is familiar

$$\frac{c}{n} = \frac{K}{k}$$

Where from Schrdinger's equation

$$\nabla^2 \Psi(\boldsymbol{x}) + \frac{2m}{\hbar^2}(E - \mathcal{V}(\boldsymbol{x}))\Psi(\mathbf{x}) = 0$$

$$K^2 = \frac{2m}{\hbar^2}(E - \mathcal{V}(\boldsymbol{x})) \tag{2.29}$$

$$n(\boldsymbol{x}) = \frac{K}{k} = \sqrt{\frac{E - \mathcal{V}(\boldsymbol{x})}{E}} = \sqrt{1 - \frac{\mathcal{V}(\boldsymbol{x})}{E}} \tag{2.30}$$

Given that the neutron scattering potentials are described by (2.28) the index of refraction can be approximated to

$$V(\boldsymbol{x}) = \sqrt{1 - \frac{\frac{2\pi\hbar^2}{m}bN}{E}} = \sqrt{1 - \frac{4\pi bN}{k^2}} \tag{2.31}$$

In the case of magnetic materials the magnetic potential

$$\mathcal{V}_{mag}(\boldsymbol{x}) = -\boldsymbol{u} \cdot \boldsymbol{B_{eff}}$$

must be accounted for. This results in an index of refraction for magnetic materials such as Fe,Ni and Co of

$$n_{\pm}(\boldsymbol{x}) = \sqrt{1 - \frac{\frac{2\pi\hbar^2}{m}bN \mp \boldsymbol{u} \cdot \boldsymbol{B_{eff}}}{E}} \tag{2.32}$$

2.2.7 Neutron in Material Phase-Shift

The phase shift due to a perturbing potential is given by [15]

$$\phi = \frac{1}{\hbar}\int \mathbf{p} \cdot \mathbf{dl} = \int \mathbf{K} \cdot dl = \int \mathbf{nk} \cdot \mathbf{dl} \tag{2.33}$$

Where $\mathbf{n}$ is the neutron index of refraction and $\mathbf{k}$ is the neutron wave vector. Under the assumption that $\mathbf{n}$ is isotropic and that the perturbation that creates $\mathbf{n}$ does not shift the path to a large degree, the phase difference between two optical paths may be written as

$$\delta\phi = \int (n_1 - n_2)\mathbf{k} \cdot \mathbf{dl} = Nb\lambda D \tag{2.34}$$

Where D is the thickness of the material the neutron beam travels through.

2.2.8 Mach-Zehnder Interferometer

The MZ utilizes a half-mirror to split the incoming electromagnetic wave and the resultant two beam paths are refocused on a second beam-splitter. The two interfered waveforms exit the second beam-splitter and are incident on two detectors that can be visualized as Detector 1 & 2 in fig(2.7),
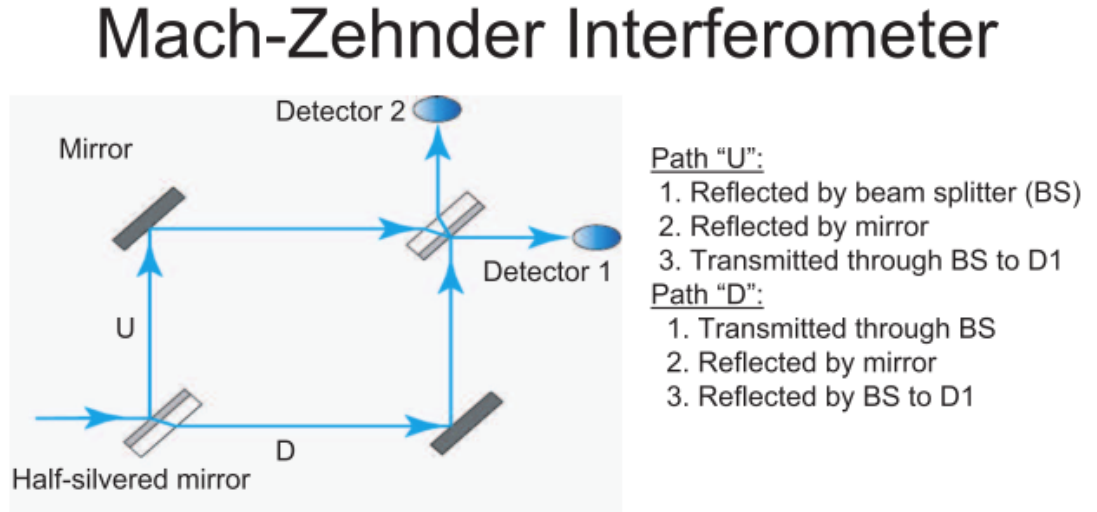
FIGURE 2.7: The Mach-Zehnder interferometer

As reflection results in a phase shift of π and assuming transmission through the half-mirrors results in a phase shift of δ we easily calculate the phase differences of the two paths at the two detectors. At detector 1 and path U there is a total of two reflections and a single transmission which results in a phase shift of $2\pi + \delta$. Similarly for path D the phase shift is also $2\pi + \delta$. therefore at detector 1 there is constructive interference. At detector 2 path U has a phase of $2\pi + 2\delta$ and path D has a phase of $\pi + 2\delta$. Therefore at detector 2 there is destructive interference.[2]

2.2.9 Three Blade Perfect Crystal Neutron Interferometer

When inside the neutron interferometer it is useful to analyse the wavefunction as the sum of the two possible neutron paths which will be refereed to as path one $\Psi_0 = |I\rangle$ and path two $|II\rangle$. The wavefunction inside the interferometer can be represented by[2]

$$\Psi = C_1 e^{i\phi_1} |I\rangle + C_2 e^{i\phi_2} II$$

Where ϕ_i are the phase of each component and C_i are the amplitudes of each component. Representing the amplitude of a Bragg transmitted wave as t and the reflected amplitude as r, where $|r|^2 + |t|^2 = 1$. The wavefunctions can be traced through the interferometer.

Consider an initial neutron with wavefunction $|I\rangle$ is incident on the first blade of the neutron interferometer. The resultant wavefunction is

$$te^{i\phi_{1_1}}|I\rangle + re^{i\phi_{2_1}}II$$

This wavefunction is than incident on the second blade interferometer. Discarding the transmitted waves as they exit the interferometer, the resultant wavefunction is

$$rte^{i\phi_{1_2}}|I\rangle + rre^{i\phi_{2_2}}II$$

At the third and final blade, the geometry of the wave splitting superimposes path one and two onto each-other the two resultant beams are incident on two detectors, the upper detector known as the O detector upon which the O beam is incident and the lower which is known as the H detector upon which the H beam is incident. The resultant wave functions are

$$|\Psi_O\rangle = (rrte^{i\phi_1} + trre^{i\phi_2})|I\rangle \quad |\Psi_H\rangle = (trte^{i\phi_1} + rrre^{i\phi_2})|II\rangle \qquad (2.35)$$

This results in an intensity at the O detector of[2]

$$I_O \propto P(O|\Delta\phi) = \langle\Psi_O\,|\,\Psi_O\rangle = |r|^4|t|^2(1 + cos(\psi_1 - \psi_2) = A(1 + cos(\Delta\phi)) \qquad (2.36)$$

Where $A = |r|^4|t|^2$ and $\Delta\phi = \phi_1 - \phi_2$. The resultant intensity for the H beam is therefore

$$I_H \propto P(H|\Delta\phi) = \langle\Psi_H\,|\,\Psi_H\rangle = (|t|^4|r|^2 + |r|^6) - |r|^4|t|^2(cos(\phi_1 - \phi_2)) = B - Acos(\Delta\phi) \qquad (2.37)$$

Where $B = |t|^4|r|^2 + |r|^6$.

The contrast of the neutron interferometer detectors is defined as

$$C = \frac{\max(I) - \min(I)}{\max(I) + \min(I)} \qquad (2.38)$$

Clearly than the contrast of the O beam will be 1 in the ideal case. However, the contrast of the H beam will depend on the size of the reflection coefficient r.

$$C_O = 1 \quad C_H = \frac{|r|^4|t|^2}{|t|^4|r|^2 + |r|^6} \qquad (2.39)$$

In (2.37,2.36) the intensities account for the loss of neutrons due to the transmitted neutrons in the second blade which exit the interferometer. This is apparent as $I_O + I_H \neq 1$. If the assumption is made that information is not lost with the exit of those neutrons the

intensities can be reformulated by treating the second blades as ideal mirrors resulting in the new intensity formulation of[2]

$$I_O \propto P(O|\Delta\phi) = 2|rt|^2(1 + cos(\Delta\phi)) \quad I_H \propto P(H|\Delta\phi) = (|r|^4 + |t|^4) - 2|rt|^2cos(\Delta\phi)$$

$$(2.40)$$

This revised intensity equation satisfies the property that $P(O|\Delta\phi) + P(H|\Delta\phi) = 1$.

2.2.10 Phase Flag

Utilizing the phase shift due to the neutron beam passing through a perturbing potential it is possible to analyse materials via the induced phase shift.
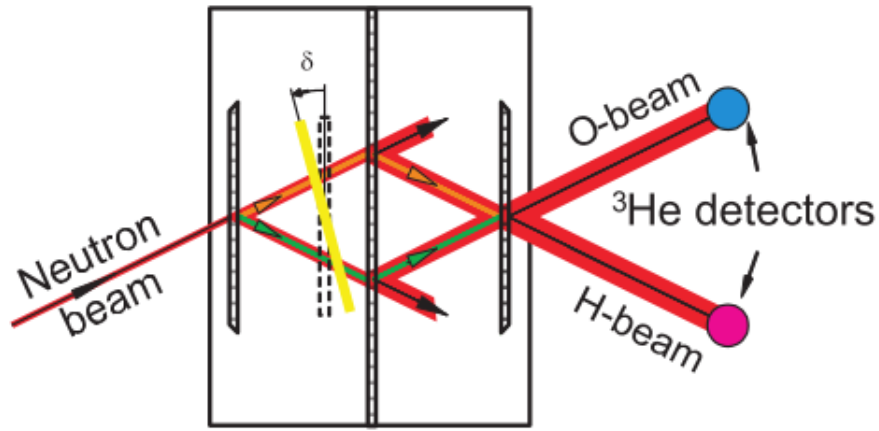
FIGURE 2.8: Neutron interferometer with a phase flag [2]

As seen in fig(2.8) the two paths will pass through the phase flag. Providing the angle of the phase flag δ is not equal to zero the paths D_1 and D_2 will accrue a phase shift due to the different path lengths inside the material. The path lengths of the up and down beams are given by

$$d_1 = \frac{D}{cos(\theta_B - \delta)} \quad d_2 = \frac{D}{cos(\theta_B + \delta)} \tag{2.41}$$

Where D is the thickness of the phase flag, θ_B is the scattering Bragg angle of the two neutron beams relative to the horizontal, and δ is the angle of the phase flag relative to the vertical. Using these paths lengths and (2.34) the phase shift along each path are

$$\delta\phi_1 = \frac{Nb\lambda D}{cos(\theta_B - \delta)} \quad \delta\phi_2 = \frac{Nb\lambda D}{cos(\theta_B + \delta)}$$

Which gives a relative phase between the two beams of

$$\Delta\phi = Nb\lambda D\left(\frac{1}{cos(\theta_B - \delta)} - \frac{1}{cos(\theta_B + \delta)}\right) =$$

$$= -2NbD\lambda\frac{sin(\theta_B)sin(\delta)}{cos^2(\theta_B) - sin^2(\delta)} \approx (CONSTANT) \times \delta = C\delta \qquad (2.42)$$

It should be noted that as (2.42) relies on a small angle approximation, the result only hold for approximately $\delta = 0..\pi/10$. Due to this restriction of phase flag angles to have full control of the phase we require that

$$C\times \geq \frac{2\pi}{\frac{\pi}{10}} = 20 \qquad (2.43)$$

Using this result the intensity at the O and H detectors become

$$I_O \propto P(O|\Delta\phi, \delta) = A(1 + cos(\Delta\phi) + C\delta) \quad I_H \propto P(H|\Delta\phi, \delta) = B - Acos(\Delta\phi + C\delta) \qquad (2.44)$$

The contrast of the neutron interferometer is a function of the phase flag angle and measured intensities at the neutron detectors[9]

$$C_0 = \frac{max[D_0(\delta)] - min[D_0(\delta)]}{max[D_0(\delta)] + min[D_0(\delta)]} \qquad (2.45)$$

Where D_0 is the detected neutron intensity at the O detector. Similarly to the case without a phase flag, the maximum contrast for the O beam is 1 and the H beam depends on the coefficient r.

2.2.11 Neutron Wave Guides

As neutrons are a wave, similarly to light the obey Snell's Law

$$n_1 sin(\theta_1) = n_2 sin(\theta_2) \qquad (2.46)$$

Total reflection will occur when

$$sin(\theta_2) = \frac{n_1}{n_2}sin(\theta_1) \leq 0 \quad \forall 0 < \theta < \pi$$

As $max(sin(\theta_1)) = 1$ occur when $\theta_1 = \frac{pi}{2}$, using (2.31) the condition for total reflection in air where $n_1 \approx 1$ is

$$V(x) > E_\perp \qquad (2.47)$$

Therefore given a material with large enough values for N and b it is possible to design a waveguide for neutrons as they will simply reflect back and forth inside the guide.
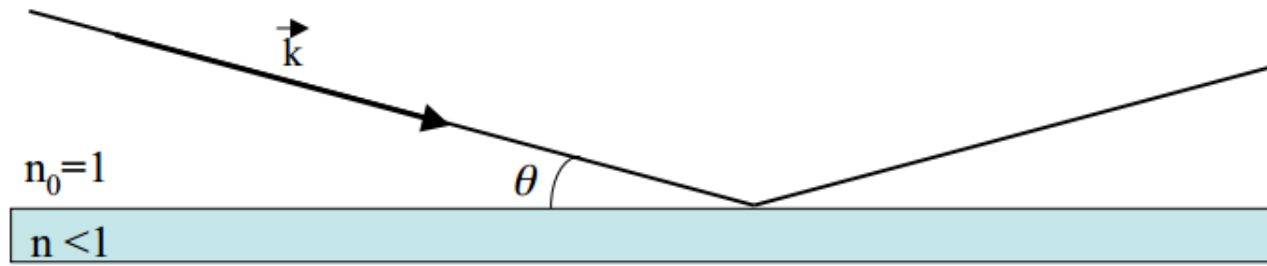
FIGURE 2.9: Neutron waveguide under the assumption that there is no Bragg scattering and the absorption is negligible [3]

Chapter 3

Contrast Improvement of Neutron Interferometry Measurements

3.1 Bayesian Markov Chain Monte Carlo Algorithms

Ever since Max Born presented what would become to be known as the Born interpretation, quantum mechanics has become a science of probabilities. The Born rule states that the probability of measuring an eigenvalue λ_i of an observable corresponding to a hermitian operator A will be

$$p(\lambda_i) = \langle \Psi \,|\, P_i \,|\, \Psi \rangle \tag{3.1}$$

This can be seen by applying the spectral theorem to A[16]

$$A = \lambda_1 P_1 + \lambda_2 P_2 \cdots \lambda_i P_i \tag{3.2}$$

Where P_i are the orthogonal projections of A onto the eigenspace corresponding to λ_i. As $I = P_1 + P_2 \cdots + P_i$ and $\langle \Psi \,|\, \Psi \rangle = 1$ given a probabilistic interpretation of the inner product.

$$\langle \Psi \,|\, I \,|\, \Psi \rangle = \langle \Psi \,|\, P_1 \,|\, \Psi \rangle + \langle \Psi \,|\, P_2 \,|\, \Psi \rangle + \cdots + \langle \Psi \,|\, P_i \,|\, \Psi \rangle = 1 \tag{3.3}$$

Given this result the interpretation may be made that

$$\langle \Psi \,|\, A \,|\, \Psi \rangle = \lambda_1 \langle \Psi \,|\, P_1 \,|\, \Psi \rangle + \lambda_2 \langle \Psi \,|\, P_2 \,|\, \Psi \rangle + \cdots + \lambda_i \langle \Psi \,|\, P_i \,|\, \Psi \rangle = \lambda_1 p(\lambda_1) + \lambda_2 p(\lambda_2) + \cdots + \lambda_i p(\lambda_i) = \bar{\lambda} \tag{3.4}$$

Therefore it is quite easy to see where the Born rule arises. It is therefore clear that the Born interpretation opens physics up to the world of statisticians and their probabilities. Using techniques from statistics it is possible that improvements in neutron interferometer contrast may be made.

3.1.1 Baye's Theorem

Baye's theorem is a very interesting statistical method that is useful for predicting the accuracy of statistical models given event data. The generic form of Baye's Thereom is

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{3.5}$$

What this says is that given an event B occurring from a set of events $\mathcal{E}$, the posterior probability that the model A describing it is the most likely model in the set of models $\mathcal{M}$ is the probability of event B occurring according to model A multiplied by the prior likelihood of model A and normalized by the total probability of event B occurring according to all models in set $\mathcal{M}$.[17]

This is an incredibly powerful theorem as it allows a set of models to be analysed statistically to determine the correct model with an actual likelihood of correctness given event data. A broad range of models can be tested and as more data rolls in the likelihood of the correct model to describe the system will rise.

There are discrete and continuous forms of Baye's theorem, however in this case only the discrete form is of interest.

$$P(A_i|B) = \frac{P(B|A_i)P(A_i)}{\sum\limits_{j} P(B|A_j)P(A_j)} \tag{3.6}$$
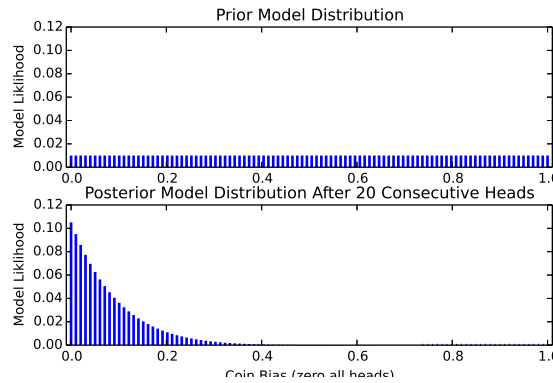
FIGURE 3.1: Example prior and posterior distribution for a set of coin flip models that samples 10 heads flips in a row

3.1.2 Markov Chain Monte Carlo Methods

Often when attempting to fit a model it is computationally infeasible to explore the entire model space. This can be due to having a large number of possible models, models having a large number of parameters and the fact that most physical process parameters are described by real numbers and it is therefore impossible to explore the entire parameter space. A way to deal with model selection in this case is known are Markov Chain Monte Carlo (MCMC) methods. Given a set of models

$$\mathcal{M} = \{\mathbf{M}(x_1, x_2, \cdots, x_i; y_1, y_2, \cdots, y_i) : y_j \in \mathcal{F}\} \tag{3.7}$$

where x_j are input variables to the model and y_j are model parameters, it is desired to find and approximate ideal model. MCMC methods will select an original set of possible model parameters

$$\mathbf{P} = \{(y_{11}, y_{12}, \cdots, y_{1i}), \cdots, (y_{j1}, y_{j2}, \cdots, y_{ji})\}$$

and assign a prior distribution to these models $\mathcal{D}_{prior}$. Given a set of inputs

$$\mathbf{I} = \{(x_{11}, x_{12}, \cdots, x_{1i}), \cdots, (x_{j1}, x_{j2}, \cdots, x_{ji})\}$$

the posterior distribution $\mathcal{D}_{pos}$ can be evaluated using (3.6). In order to explore the parameter space, the algorithm must select a new set of parameters. One method of doing this is to select some $\subset \mathbf{P}$ and generate a new set of models and posterior distributions according to some utility function[17]

$$\mathcal{M}_{pos}, \mathcal{D}_{pos_2} = f(\mathcal{M}, \mathcal{D}_{pos}) \tag{3.8}$$

It is useful to think of the set of selected model parameters as a large number of points in space, in practice these are often referred to as points. By re-sampling model parameters as in (3.8) it is possible to select points throughout the parameter space and ideally zero in on the optimal model. However, due to the nature of sampling normally being non-deterministic and numerical precision errors, it is crucial examine the convergence of the model to experimental data.

3.1.3 Fisher Information and the Cramer-Rao Lower Bound

The Cramer-Rao lower bound is hugely important in statistics as it provides a lower bound on the variance of an unbiased estimator

3.1.3.1 Covariance Matrix

The covariance matrix in statistics is a measure of how a random vector of variables changes together. Two random variables are related by their covariance $\sigma(x,y) = E_x\left[(x - E[x])(y - E[y])\right]$, where $E[x]$ denotes the expectation value of x. As an example, in quantum mechanics the expectation value according to the Born interpretation is as in (3.4). The covariance of two random variables can be generalized to that of a random vector of estimated coefficients $\mathbf{X}$ where [18]

$$cov(\mathbf{X})_{ij} = cov(X_i, X_j) = E[(X_i - E[X_i])(X_j - E[X_j])] \tag{3.9}$$

Which in matrix notation takes the form

$$\boldsymbol{\Sigma} = cov(\mathbf{X}) = [(\mathbf{X} - u)(\mathbf{X} - u)^T] = E(\mathbf{X}\mathbf{X}^T) - uu^T \tag{3.10}$$

Where $u = E(\mathbf{X})$. It is prudent to note that the elements of the diagonal of the covariant matrix $\sum ii$ are the variances of the elements of the vector $\mathbf{X}$. When $\sigma(x,y) > 0$ the relationship between the two variables tends to be similar, ie. when one grows the other grows and similarly when $\sigma(x,y) < 0$ when one variable grows the other will tend to show the opposite behaviour.[18]

3.1.3.2 Cramer-Rao Lower Bound

The Cramer-Rao lower bound states that the variance of an unbiased estimator $\hat{\theta}$ is lower bounded by the reciprocal Fisher Information $I(\theta)$.[19]

$$var(\hat{\theta}) \geq \frac{1}{I(\theta)} \tag{3.11}$$

The Fisher Information is the expectation value of the square of the first moment of the log likelihood function for some probability density function $p(\mathbf{X}; \theta)$. The log-likelihood of the probability density function is defined as $\updownarrow(\mathbf{X}; \theta) = log(p(\mathbf{X}; \theta))$ Therefore the Fisher information is

$$I(\theta) = E\left[\left(\frac{\partial}{\partial\theta}log(p(\mathbf{X}; \theta))\right)^2 \Big| \theta\right] = -E\left[\frac{\partial^2}{\partial\theta^2}l(\mathbf{X}; \theta)\Big|\theta\right] \tag{3.12}$$

The Fisher Information is the variance of the statistical score of the probability density function. Geometrically it is a measure of the amount of curvature of the log-likelihood function. It is a measure of the amount of information an observable $\mathbf{X}$ carries about an unknown parameter θ of which the probability of $\mathbf{X}$ relies on according to the function

$p(\mathbf{X}; \theta)$. In multiple dimensions the Fisher Information matrix takes the form

$$\mathbf{I}(\theta) = E_D[\nabla_\theta log(P(D|\theta)) \cdot \nabla_\theta^T log(P(D|\theta))] \qquad (3.13)$$

The Cramer-Rao lower bound establishes a lower limit on the amount of information an unknown parameter of a probability distribution carries.[20] This is very useful in determining the fit of a set of model parameters as shall be seen. It is important to keep in mind that this only holds for unbiased estimators and may be violated by biased estimators.

3.2   Robust Online Hamiltonian Learning

The construction of a large scale quantum information processor is an extremely difficult challenge. Before error correcting codes and algorithms can be applied to such a system, the full characterization of the system is required. This is normally achieved via statistical estimation of Hamiltonian parameters via tomography. In quantum tomography the process is normally carried out on the device as a *black box* and a set of measurements is chosen such that the parametrization of them will allow the output of any input state to be predicted.[21] Full tomography is computationally complex and requires the simulation of the behaviour of the quantum system. As the systems being simulated grow in complexity, it is no longer possible for classical computer to simulate.[22]

In practice there is often more information available about the form of the Hamiltonian of a given system and this allows the number of parameters that need be estimated to be reduced. This is known as partial process estimation or partial tomography. The goal of the experiment proposed in this thesis is to provide experimental verification of the algorithm proposed in the 2013 paper "Robust Online Hamiltonian Learning" by Chris Granade at the University of Waterloo.[23]

An additional benefit to the algorithm of focus is that in quantum mechanics, often future measurement results depend on prior measurements. The use of MCMC methods allows model parameters to be estimated for such time dependent measurements. As given a set of estimated models for the hamiltonian it is possible to predict the probability of a future measurement d_{N+1} given a future set of experimental controls c_{N+1}.

$$P(d_{N+1}|D; c_{N+1}, C) = E_{\mathbf{x}|D;C}[P(d_{N+1}|\mathbf{x}; c_{N+1}] \qquad (3.14)$$

3.2.0.3   Algorithm Specifics

3.2.0.4   Bounding the Error of an Estimator

Given a set of future experimental controls and their potential outcome probabilities it is useful to quantify the measurements usefulness. This is just the utility function as described in section (3.1.2). An often used measure is that of information gain, or in other words the expectation value of the logarithm of the estimator.

$$U(d_{N+1}; c_{N+1}) = E_{\mathbf{x}|dN+1,D;c_{N+1},C}[logP(\mathbf{x}|d_{N+1}, D; c_{N+1}, C)] \tag{3.15}$$

Maximizing the utility function is equivalent to minimizing the entropy of the posterior distribution.

The quality of an estimator $\hat{\mathbf{x}}$ for a set of observed data D and experimental controls C can be quantified by using the quadratic loss function

$$L_{\mathbf{Q}}(\mathbf{x}, \hat{\mathbf{x}}(X, X)) = (\mathbf{x}, \hat{\mathbf{x}}(D, C))^T \mathbf{Q}(\mathbf{x}, \hat{\mathbf{x}}(D, C)) \tag{3.16}$$

Where $\mathbf{Q}$ is a chosen positive definite matrix deciding the importance of model parameters in the quadratic loss function. The quadratic loss is used to define the risk of an estimator by taking the expectation value of the quadratic loss over all possible outcomes of the experiment.

$$R(\mathbf{x}, \hat{\mathbf{x}}; C) = E_{D|\mathbf{x};C}[L(\mathbf{x}, \hat{\mathbf{x}}(D, X))] \tag{3.17}$$

The Bayes risk is the average risk of the risk function with respect to a prior distribution $\pi(\mathbf{x})$

$$r(\pi; C) = E_{\mathbf{x}}[R(\mathbf{x}, \hat{\mathbf{x}}; C)] = \int \pi(\mathbf{x})R(\mathbf{x}, \hat{\mathbf{x}}; C)d\mathbf{x} \tag{3.18}$$

Here $\hat{\mathbf{x}}$ is taken to be the unique Bayes estimator, which is the estimator that minimizes the total risk. The utility function is chosen as the negative Bayes risk as this corresponds to minimizing the Bayes risk for chosen set of parameters.

Remembering the Fisher information matrix as defined in (3.13) we call the Bayesian information matrix[24]

$$\mathbf{J}(\pi; C) = E_{\mathbf{x}}[\mathbf{I}(\mathbf{x}; C)] = \int \pi(\mathbf{x})R(\mathbf{x}, \hat{\mathbf{x}}, C)d\mathbf{x} \tag{3.19}$$

From Bayesian information matrix a lower bound on the risk function called the Bayesian Cramer-Rao bound can be placed [25]

$$r(\pi; C) \geq \frac{1}{\mathbf{J}(\pi; C)} \tag{3.20}$$

This is used rather than the Fisher Information as in the case of an experiment giving no information about a parameter set, the Fisher Matrix I is singular and therefore its inverse is undefined, this is resolved via the Bayesian Cramer-Rao Lower Bound (3.20).

Given (3.20) the lower bound on the Bayes risk can be iteratively monitored.

$$r(\pi; C_{N+1}) \geq \frac{1}{\mathbf{J}_{N+1}(C_{N+1})} \tag{3.21}$$

where

$$\mathbf{J}_{N+1}(C_{N+1} = \mathbf{J}(\pi; C_{N+1}) + J_N(C_N) \tag{3.22}$$

3.2.0.5 Resampling

Utilizing sequential monte carlo methods the costly posterior distribution are approximately calculated by approximating the distribution as a set of particles or points distributed over the parameter space, each with weight w_i and with the sum of all weights being $\sum_i w_i = 1$. Therefore the probability distribution is

$$P(\mathbf{x}|D) \approx \sum_{k=1}^{n} w_k(D)\delta(\mathbf{x} - \mathbf{x}_k)$$

with the weights being iteratively updated according to

$$w_k(d_{j+1} \cup D) = \sum_{k=1}^{n} P(d_{j+1}|\mathbf{x}_k)w_k(d_j)$$

As with MCMC algorithms the accuracy can be made arbitrarily high by increasing the number of particles. The initial weights are chosen to simply be $w_k = 1/n$. Using this approximate distribution, Bayes updates can be performed and an accurate estimator approached.

Since any particle estimate will only ever approach the true parametrization, all weights will tend to zero. This can be postponed by re-sampling and updating weights at intervals. The basic idea is too select some n most likely particles and choose new particles around these likely particles according to some distribution to replace unlikely particles. Re-sampling will be performed when $n_ess = 1/\sum_i w_i^2 > 0.5$ where 0.5 is the re-sampling threshold. After re-sampling the weights are reset.

The algorithm to be tested adds the additional step of a random normally distributed perturbation vector $\epsilon = \mathcal{N}(0, \sum)$ to the position of a random selected point. This is to

further increase the size of the parameter space explored.[23]

$$pos(\mathbf{x}') = \sum_i w_i \frac{1}{\sqrt{(2\pi)^k |\sum|}} exp\left(-\frac{1}{2}(\mathbf{x}' - \mathbf{u_i})^T \mathbf{\Sigma}^{-1}(\mathbf{x}' - \mathbf{u_i})\right) \qquad (3.23)$$

Where k is the number of model parameters. This distribution is known as a mixture distribution. The mean u_i of each particle is chosen using a linear combination of the form

$$\mathbf{u}_i = a\mathbf{x}_i + (1-a)\mathbf{u} \qquad (3.24)$$

Where $\mathbf{u} = E[\mathbf{x}]$. Therefore re-sampling consists of sampling from the distribution described by (3.23) for the $k = N - n$ unselected particles and then resetting all weights to $w_i = 1/n$.

3.2.0.6　Region Estimation

It is not just important to have an accurate estimator for the system, but also crucial to be able to quantify the error in the estimator and how much it differs from the Bayesian Cramer-Rao lower bound. This is achieved by finding a region of the parameter space $\hat{\mathbf{X}}$ such that the probability of the true parameter set $\mathbf{x_0}$ is contained within $\hat{\mathbf{X}}$ is maximized, while the volume $V(\hat{\mathbf{X}})$ is minimized.[23]

The probability of the true model being within a region $\hat{\mathbf{X}}$ can be expressed as

$$P(\mathbf{x_0} \in \hat{\mathbf{X}}) = E[1_{\hat{\mathbf{X}}}] \qquad (3.25)$$

Where $1_{\hat{\mathbf{X}}}(x) = 1$ if $\mathbf{x} \in \hat{\mathbf{X}}$ and otherwise $1_{\hat{\mathbf{X}}}(x) = 0$. The probability of a region can then be calculated from the particle weights from section (3.2.0.5).

$$E[1_{\hat{\mathbf{X}}}] \approx \sum_i w_i 1_{\hat{\mathbf{X}}}(\mathbf{x}_i) = \sum_{i, \mathbf{x_i} \in \hat{\mathbf{X}}} w_i \qquad (3.26)$$

Therefore a region $\hat{\mathbf{X}}$ must be found such that (3.26) is sufficiently large, while $V(\hat{\mathbf{X}})$ is sufficiently small. Under the assumption that after a reasonable amount of data the posterior distribution is approximately normally distributed according to $\mathcal{N}(\mu, \mathbf{\Sigma})$, from the definition of the normal distribution it follows that the inverse covariance matrix Σ^{-1} describes an ellipse of dimension d where 0.682^d of the probability mass is contained within the ellipse. Where, d is the number of model parameters. The volume of the ellipse can be found to be

$$V(\mathbf{\Sigma^{-1}}) = \frac{\pi^{d/2}}{\Gamma(d/2+1)} det(\mathbf{\Sigma^{-1}}) \qquad (3.27)$$

In order to determine the fit of the region found, it is important to compare the found covariance ellipse to the optimal one according to the Bayesian Cramer-Rao lower bound

$$E_\pi[cov(\hat{\mathbf{x}})] \geq \mathbf{J}(\pi; C)^{-1}$$

3.3 Experimental Verification of the Robust Online Hamiltonian Learning Algorithm

While simulations predict that the algorithim introduced in sec(3.2) quickly approaches the lower bound on the error for a variety of physical systems.[23] It has yet to be verified experimentally. Therefore a proof of concept neutron interferometry experiment will be proposed which theoretically should allow the prediction of neutron interferometer model parameters and improve the maximum contrast of interferometer measurements. As control of the contrast of the neutron interferometer is desired, precise contrast control will only be obtainable when all system parameters are known.

3.3.1 Experimental Model

Take the O and H detectors as 0 and 1 respectively. We model the probability of the neutron striking the O detector similarly to (2.40) except now the model is non-ideal and has parameters that must be fitted to it as

$$P(0|A, b, \delta\phi, C, \delta) = (\frac{1}{2} + b) + \frac{A}{2}cos(\Delta\phi + C\delta) \tag{3.28}$$

where b is an unknown experimental offset, A is an unknown amplitude, $\Delta\phi$ is the induced phase due to the different paths, C is the phase flag constant and δ is the phase flag setting.

3.3.1.1 Humidity and Temperature Phase Dependence

As the neutron interferometry chamber is not humidity or temperature controlled we will attempt to increase the contrast of measurements by accounting for the induced phase shift due to the water vapour in the air. This phase shift will occur if there is a path length difference between the two interferometer paths. This path length difference will be induced by one of the paths spending a longer time in the phase flag and therefore will be a function of the phase flag angle. The model will now have the form

$$P(0|A, b, \Delta\phi, C; \delta) = (\frac{1}{2} + b) + \frac{A}{2}cos(\Delta\phi + C\delta + \Delta\gamma(\Phi, T, \delta)) \tag{3.29}$$

Where

$$\Delta\gamma(\Phi, T, \delta) = \frac{n}{v}\lambda b D$$

D is the path length difference and from sec(2.2.10) the phase shift as a function of the phase flag angle is the negative of (2.42) due to the extra path length in the water vapour occurring when the other pass is still in the phase flag. therefore the phase difference induced by the water vapour is

$$\Delta\gamma(\Phi, T, \delta) = 2\frac{n(\Phi, T)}{V}b_{water}\lambda D\frac{sin(\theta_B)sin(\delta)}{cos^2(\theta_B) - sin^2(\delta)} \approx \alpha \times \frac{n(\Phi, T)}{V}\delta \qquad (3.30)$$

Where α is the phase multiplier constant and δ is the angle of the phase flag.

By the ideal gas law

$$\frac{n(\Phi, T)}{V} = \frac{p_w(\Phi, T)}{RT} \qquad (3.31)$$

The partial pressure of water p_w at percent humidity Φ is given by

$$p_w(\Phi, T) = \Phi p_w^*(T) \qquad (3.32)$$

Where p_w^* is the saturation vapour pressure in bars.[26] The saturation pressure of water at temperature T can be fitted to the Antoine equation [27]

$$p_w^*(T) = 10^{h_A - \frac{h_B}{h_C + T}} \qquad (3.33)$$

Therefore h_A, h_B and h_C are empirical constants that must be fitted and are parameters of the estimator for the O detector model. In the range of $255.9K - 373K$ the parameters were fitted to values $h_A = 6.20963, h_B = 2354.731$ and $h_C = 7.559$.[27] The phase shift due to water vapour is can be found by combining (3.30),(3.32) and (3.33) to obtain

$$\Delta\gamma(\Phi, T, \delta) = \frac{\alpha'\Phi\delta}{T} \times 10^{h_A - \frac{h_B}{h_C - T}} \qquad (3.34)$$

In reality we do not have the exact values of Φ and T and therefore must marginalize over the distribution of $P(T, \Phi|\hat{T}, \hat{\Phi})$ conditioned on the estimator for the phase shift described by (3.34).

$$P(\Delta\gamma|\hat{\Phi}, \hat{T}) = \int_\Phi \int_T P(\Delta\gamma|\Phi, T)P(T, \Phi|\hat{T}, \hat{\Phi})dTd\Phi \qquad (3.35)$$

For simplicity we assume that $\hat{\Phi} \perp \hat{T}$ which reduces (3.35) to

$$P(\Delta\gamma|\hat{\Phi}, \hat{T}) = \int_\Phi \int_T P(\Delta\gamma|\Phi, T)P(T, \hat{T})P(\Phi, \hat{\Phi})dTd\Phi \qquad (3.36)$$
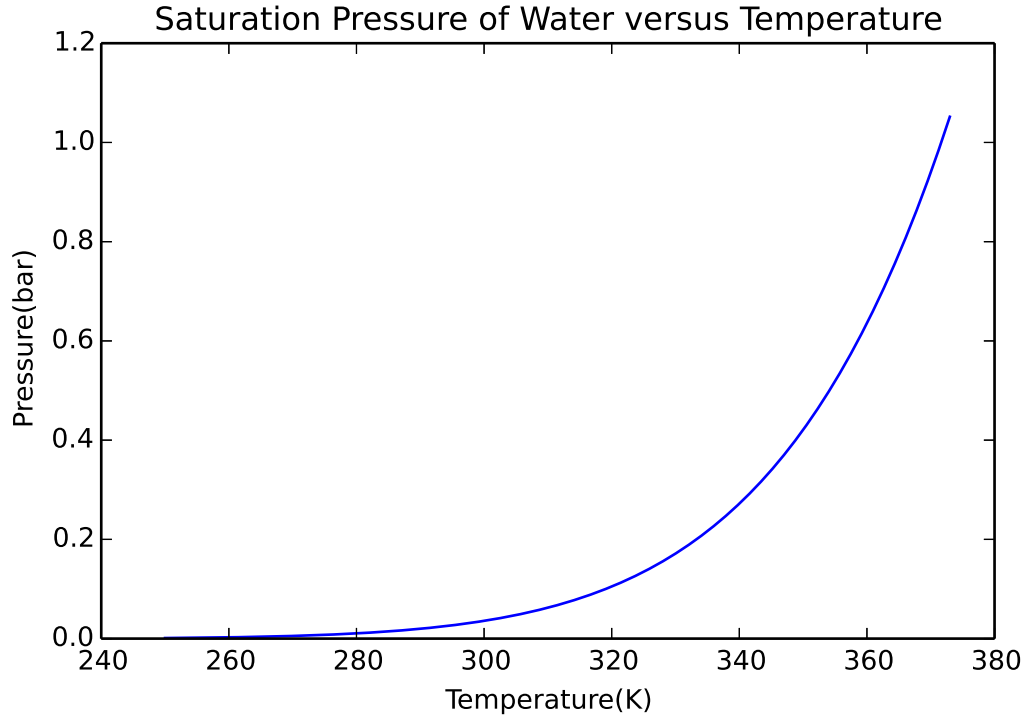
FIGURE 3.2: Saturation Pressure as a function of temperature

Taking account of the humidity and temperature dependence of the phase due to water vapour in the air, the updated model now takes the form

$$P(0|A,b,\Delta\phi,C,\alpha';\delta,\hat{\Phi},\hat{T}) = \int\limits_{\Delta\gamma} P(0|A,b,\Delta\phi,C,\delta,\alpha',\hat{\Phi},\hat{T},\Delta\gamma)d\Delta\gamma =$$

$$= \int\limits_{\Delta\gamma}\int\limits_{\Phi}\int\limits_{T} P(0|A,b,\Delta\phi,C,\alpha';\delta,\Phi,T,\Delta\gamma)P(T,\hat{T})P(\Phi,\hat{\Phi})dTd\Phi d\Delta\gamma =$$

$$= \int\limits_{\Phi}\int\limits_{T} \left[(\frac{1}{2}+b) + \frac{A}{2}cos\left(\Delta\phi + C\delta + \frac{\alpha'\Phi\delta}{T}\times 10^{h_A - \frac{h_B}{h_C+T}}\right)\right] P(T|\hat{T})P(\Phi|\hat{\Phi})dTd\Phi \quad (3.37)$$

The integral in (3.37) will be integrated using Monte Carlo sampling techniques in order to reduce the computational complexity involved in evaluating (3.37) many, many times. Provided the temperature and humidity data is normally distributed the required number of samples should be $n \approx 100-300$ and is tunable based on the desired accuracy. Similarly to the original model (??), due to the small angle approximation of the phase flag a lower bound must be put on the phase induced by the flag and water vapour, in order to allow full phase control.

$$C + \frac{\alpha'\Phi}{T}\times 10^{h_A - \frac{h_B}{h_C+T}} \geq 20 \quad (3.38)$$

3.3.2 Model Analysis

Due to unforeseen departmental issues it was not possible to experimentally verify the model (3.37) using the algorithim described in sec(3.2). However it was possible to simulate the model for test *true* parameters and simulate both naive and adaptive experiments to determine if it should be possible to reduce the Bayes risk to an acceptable level. A variety of *true* parameter sets were chosen, to test the affects of different parameters as seen in table (3.1). In fig(**??**) the likelihood of detection at the O-detector

Parameter Set	A	b	$\Delta\phi$	C	α	Description
1	0.9	0.0	0.0844	22.02	-1.02	Normal set of Parameters
2	0.9	0.0	0.0844	15.0	-1.02	No full phase control
3	0.9	0.0	1.5707	22.02	-1.02	Larger phase offset
4	0.7	0.0	0.0844	22.02	-1.02	Smaller A constant
5	0.1	0.0	0.0844	22.02	-1.02	Very small A constant
6	0.7	0.1	0.0844	22.02	-1.02	B is offset

TABLE 3.1: True parameter sets to simulate model for

for the system parameters in table (3.1) are plotted as a function of the phase flag angle δ at a temperature of $293.15K$ and humidity of 70%.

It can be seen that the likelihood of detection is never 100% due to the parameter A being less one in all test cases. In the case of parameter set five, A is very small and the result is a very low maximum probability of detection. Additionally, in the case of parameter set 2 it can be seen that full phase control does not exist due to the weakness of the settings of the phase flag constant α'. Uniform prior distributions were established around the true values as in table (3.2). The set of starting values were kept relatively near to the true values in order to reduce simulation time. It should be noted that the specifications of the system that the simulations were performed on can be found in appendix (**??**). From these initial uniform distributions the Bayes risk can be calculated

A	b	$\Delta\phi$	C	α
±0.02	±0.02	$[0, \frac{pi}{2}]$	±0.05	±0.05

TABLE 3.2: Range around true parameters that prior distributions were established for

as seen in fig(**??**). It is clear that the initial risk is very large. This is as to be expected since the initial distributions are simply uniform distributions. It should be noted that the risk is largest around the point when the sum of the phases is $\pm\pi/2$. This is because when the total phase offset is $\pm\pi/2$, $cos(\pi/2) = 0$ and there is no information about
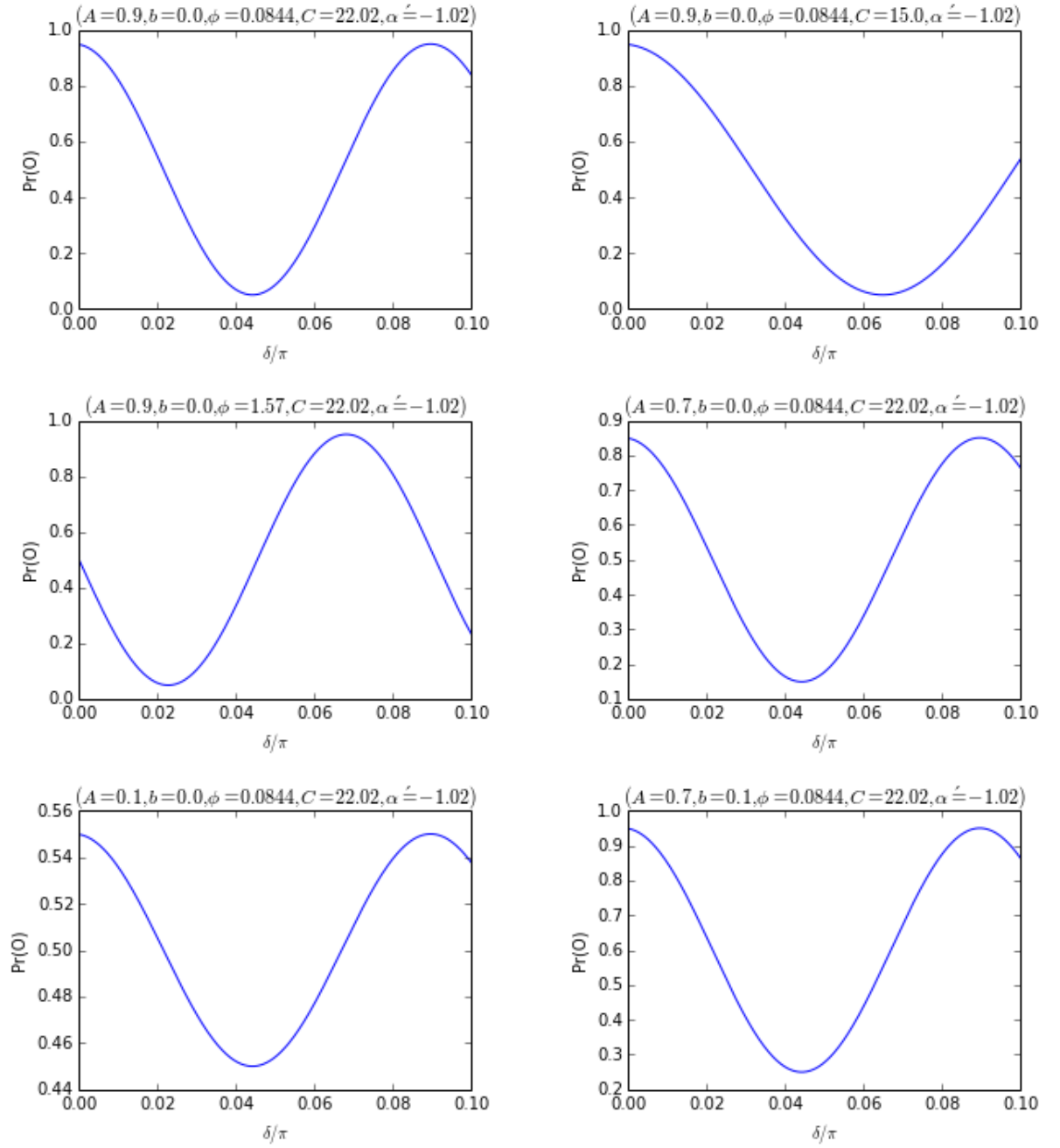
FIGURE 3.3: Liklihood of detection at O-detector for various system parameters as a function of δ, $T = 293.15$, $\Phi = 70\%$

any parameters except b. The risks of parameter set two are singularly peaked, due to the lack of full phase control.

We next examine the algorithm behaviour for an experiment in which $2^{1}3$ linearly spaced values for δ from $\delta = [0, \pi/10]$ are measured, with the assumption that at each interval 100 neutrons are detected. The Monte Carlo algorithm is using 20000 particles. Additionally the temperature is set to $293.15 \pm 0.5K$ and the humidity $70 \pm 1\%$, both are normally distributed. While this experiment is unrealistic as the number of neutrons
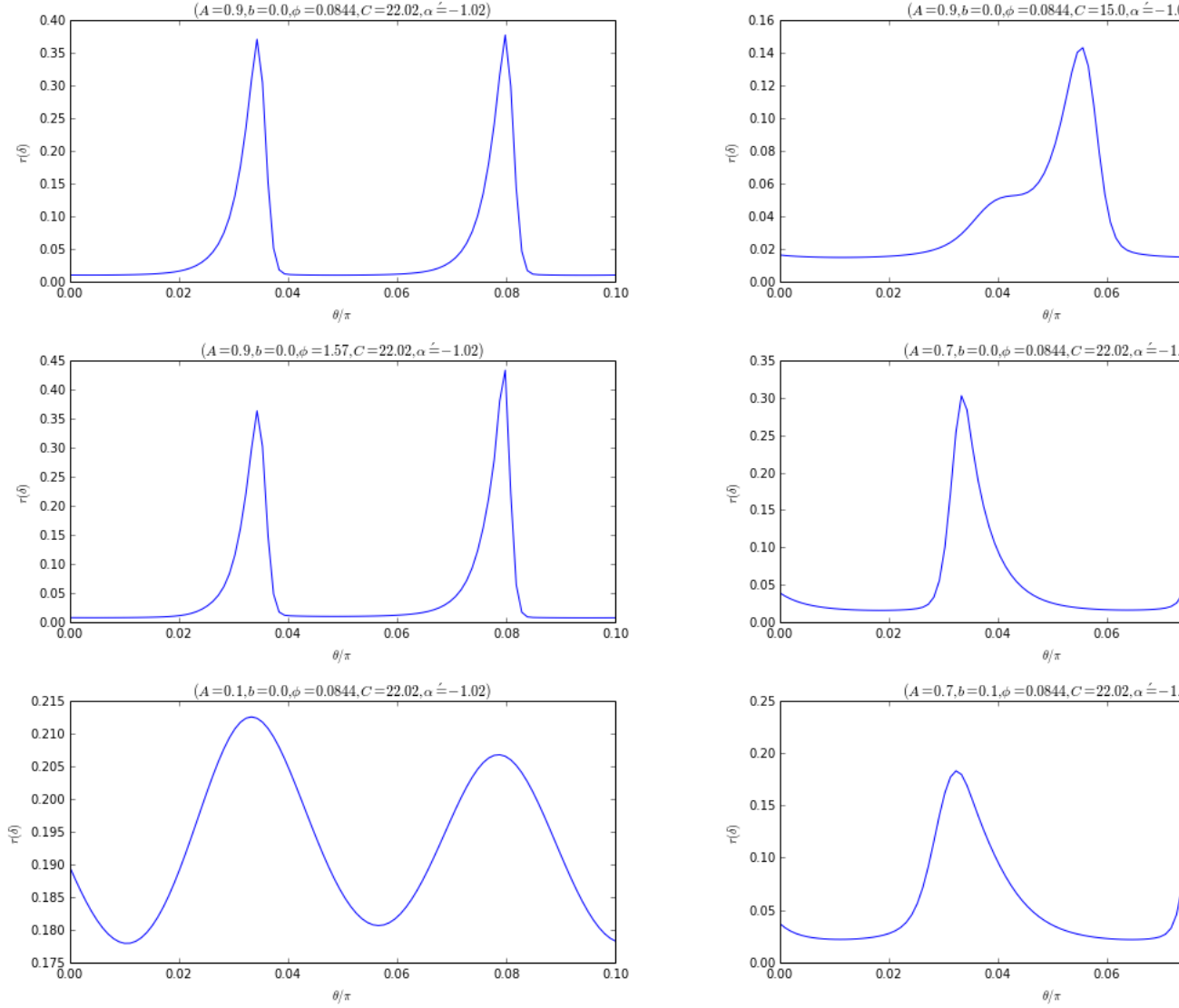
FIGURE 3.4: The Bayes Risk of the initial prior distributions

detected would be much larger than 100 and would be varying, it serves as a naive simulation. The results are as seen in fig(**??**). From the naive simulation it can be seen the risk is over two to three orders of magnitude lower than that of the uniform distribution for all parameter sets. Additionally, the risk is much more uniform in value over the entire phase flag range as all risks fluctuate only by order 10^{-7}. The approximate values for the system parameters calculated by the simulation can be found in table (**??**).
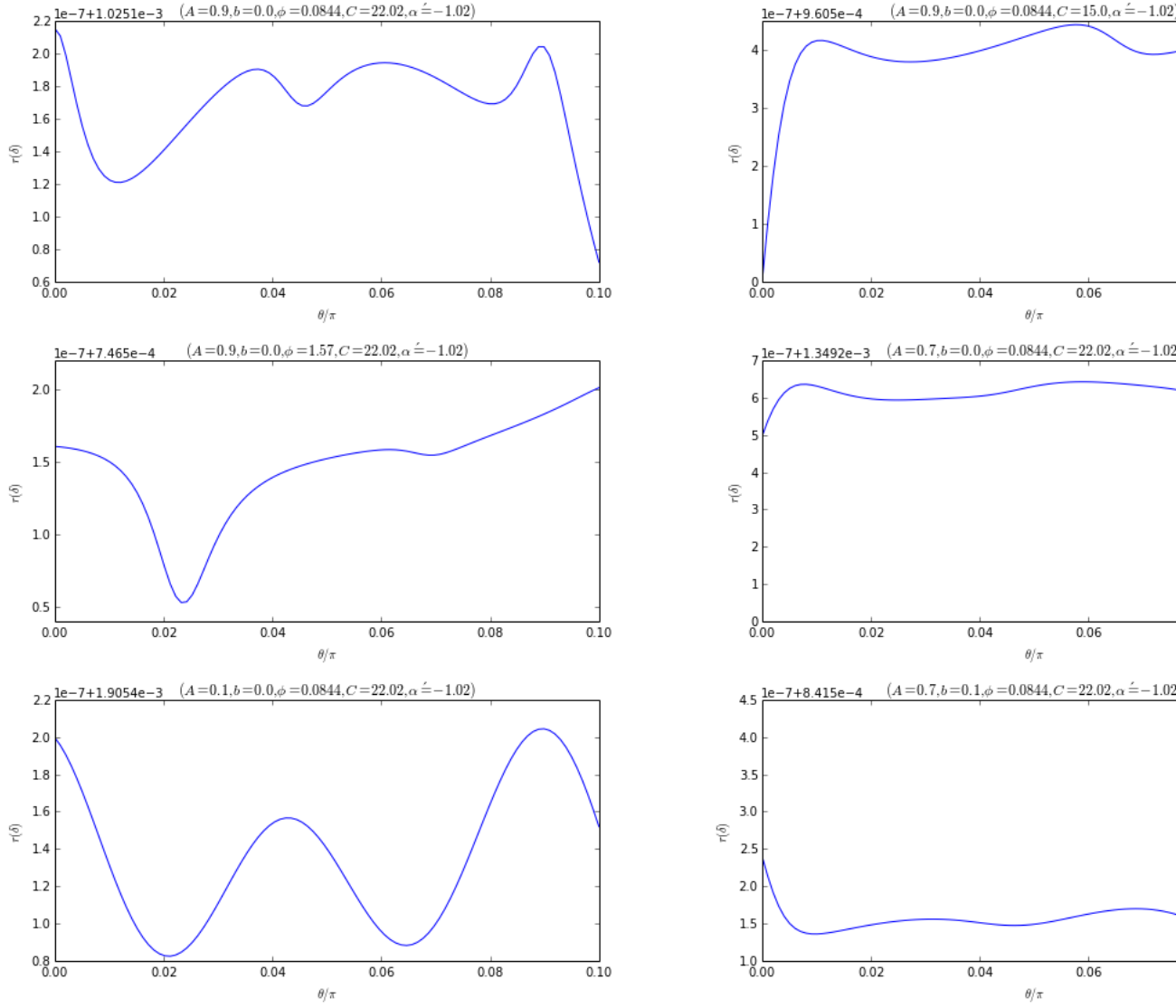
FIGURE 3.5: The Bayes Risk a naive algorithm that sweeps through all phase flag settings from 0 to 2π

Chapter 4

Experimental Setup

4.1 The Neutron Interferometer

The neutron interferometer that this thesis refers to is located at the Neutron Interferometry and Optics facility (NIOF) at the National Institute of Standards and Technology (NIST) in Gaithersburg, MD.

4.1.1 NIST

4.1.2 Reactor

NIST operates a 20MW split-core research reactor. Neutrons of approximate energy $1MeV$ are emitted during ^{235}U fission and then thermalized using heavy water (D_2O) as a moderator. This brings the neutrons to room temperature as discussed in (2.1.2). At the reactor core the peak thermal neutron flux is $4 \times 10^{14} neutrons/cm^2$. The reactor is operated on a seven week cycle during which it is operated at full power for 38 days and then followed by 11 days of refuelling and maintenance operations.

As the longer wavelength of cold neutrons ($\lambda > 1.8\mathring{A}$ and $E < 25meV$) is often desired for condensed matter study there is a cold moderator installed next to the core. The thermal neutrons scatter with liquid hydrogen at $20K$ and exit with a Maxwellian distribution of characteristic temperature of $34K$.

There are eight thermal neutron ports available for lab use. The neutrons are transported to the instruments in the NCNR hall using neutron guides. The neutron interferometer facility is located on the NG7 guide shown in figure (4.1). The guides are of a rectangular cross-section and are produced by gluing together meter long sections of $100nm$ thick

^{58}Ni optically-flat borated glass plates. ^{58}Ni is used due to its large neutron reflective potential, which allows it to totally reflect neutrons as in section (2.2.11).
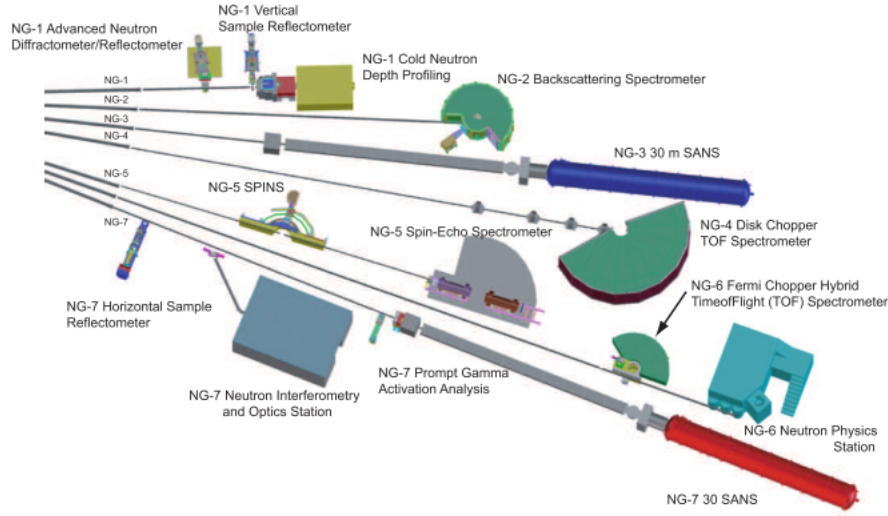
FIGURE 4.1: The Neutron labs at NIST Center for Neutron Research Guide Hall

4.1.3 Motors and Actuators

The neutron interferometry lab uses a variety of motors and actuators that allow experimental parameters to be controlled over the wire. Depending on the device communication is achieved via analog or digital protocols.

4.1.3.1 Newport 301

The Newport ESP301 is a three axis motion controller and driver.[28] It can drive both DC servo motors and 2-phase stepper motors at up to 3 amps. It has a 1000x microstep resolution per axis which allows very fine grained control of movements which is necessary for precision measurements. The Newport ESP 301 is primarily used to drive servo motors that orient the phase flag in the neutron interferometer. Precise angular control is a must as this is one of the primary experimental parameter. The device can also be used to control the interferometry stage to orient it.

The Newport device utilizes encoder feedback built into servos to obtain precise positional feedback. While most Newport brand motors will automatically supply their configuration information, the equipment at NIST is not necessarily compatible in such a way. Therefore advanced configuration must be supplied by the device programmer.

The ESP301 is controllable via a fairly extensive language of approximately 100 commands, a large portion of which are necessary for the device to be controlled successfully. Valid configuration information must be supplied. Commands are transmitted via ASCII characters according to the selected protocol. Supported protocols are serial RS232C, USB and IEEE488 with delays of $7-30ms$,$3.5ms$ and $1ms$ respectively.
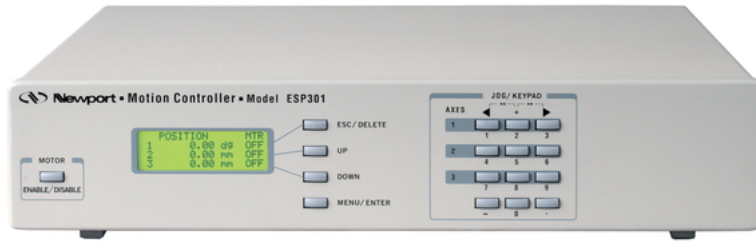
FIGURE 4.2: Newport ESP301 three axis motion controller

4.1.3.2 Kepco Power Supply

Kepco series ATE power supplies are used to control a variety of devices in the interferometry lab. Specifically the 15-6M and 36-15M models are used and are rack mounted. The models are rated at (0-15V,0-6A,50W) and (0-36V,0-15A,500W) respectively.[29] The models are controllable via an analog input line that sets the supply output as a linear function of the input from 0-10V. Additionally there is a crowbar voltage controller which allows a maximum voltage to be set in a similar manner. The Kepco supplies are controllable via DACs which are managed via a LabJack.

4.1.3.3 LabJack

The LabJack is a low cost measurement and automation platform. Specifically NIST will use the U3-LV variant. The U3-LV provides up to 16 analog inputs, 2 analog outputs and up to 20 digital I/O pins. The analog inputs accept voltages from 0-3.6V and the onboard DACs outputs from 0-5V. The LabJack devices are easily controlled via a Python API.

In Addition to the LabJack the LJTick-DACs, a DAC made to be digitally controlled by LabJack devices are used to provide control inputs to devices such as the Kepco power supplies. The LJTick-DAC outputs $\pm 10V$ controllable by the LabJack pins in which it is inserted too. Each LJTick-DAC provides two DACs.

FIGURE 4.3: Labjack U3 with LJTick-DAC

4.1.4 Sensors

As the likelihood method for the contrast measurement is a function of the temperature and humidity of the interferometer chamber, a variety of measurements must be taken using many different sensors.

4.1.4.1 EI1050 Temperature and Humidity Probe

The EI1050 is a digital temperature and humidity probe produced by LabJack. While its protocol is open source, it has been designed to be used with a LabJack device and sample code has been provided. The device is not especially accurate as it is rated at $\pm 0.5°$ and $\pm 3.5\%$ humidity at ranges from -40-$120C$ and 0-100% humidity. It is still useful to provide a quick and easy measurement.

4.1.4.2 Stanford Research Systems CTC100 Temperature Controller

The SRS CTC100 is a cryogenic temperature controller. It provides four sensor inputs, four analog outputs and six feedback control loops. Temperature readings are made by thermistor sensors and heating is provided by resistive heaters. The device is programmable using USB, Ethernet and either GPIB or RS-232 inputs. Commands are provided via ASCII commands. The thermistor temperature readings are very accurate with an accuracy of $\pm 0.25\Omega$ for a 300Ω thermistor. The CTC100 automatically supplies the mean and standard deviation of sampled temperatures.

Figure 4.4: Stanford Research Systems CTC100: Cryogenic Temperature Controller

4.2 NI-Engine

The current system at the NIST interferometry lab uses an Excel spreadsheet connected via ActiveX to closed source controller code. This system only controls the bare minimum of experimental hardware and is very inflexible. The system is not easily extensible and the software design principles are very poor. As the contrast improvement experiment requires many different sensor inputs and system controllability for many different measurement cases, it would have been a tremendous task to implement into the current system.

Ni-Engine attempts to solve the problems of the previous system for the proposed experiment and future lab researchers. NI-Engine is an experimental neutron interferometry control system programmed in Python using open source libraries and software design principles.

4.2.1 Design Requirements

The core design requirement of NI-Engine are to be a complete, easy to use neutron interferometry experimental control system. In order fulfil this core requirement the software had to fulfil these requirements:

- Be able to communicate with and control a large range of hardware in a manner that allows powerful functionality while hiding the inherent complexity.

- Handle experimental setup and configuration via configuration files such that an experimental setup is easily repeatable given an identical configuration.

- Handle the collection and storage of experimental data. Experimental data will be large, possible into the $10 + GB$ range. Should allow for multiple identical experiments to be performed sequentially.

- Execution of software functionality must be fast and allow for experiment running times to be dominated by hardware measurement and control, rather than software running time.

- Software must be able to run for long periods of time with consistent memory requirements.

- Data should be collected and stored with units available to avoid confusion.

- Should be modular. Outside researchers should be able to modify software to their own requirements. ie. addition of new hardware, file formats, data types.

- Software must have extensive documentation and example code. Researchers should be able to utilize without extensive outside help.

These design requirements had to be taken into account when designing and programming Ni-Engine

4.2.2   Language and Software Library Choices

4.2.2.1   Language

Given the design requirements outlined in (4.2.1) it is clear that the system required is quite extensive. Python[30] was selected as the development language for Ni-Engine. Python was selected for a variety of reasons including

- Python is very easy to learn and use. Its syntax style allows newcomers the ability to quickly write code, while still allowing experts powerful expressibility.

- Python's code is extremely legible and the nature of its syntax naturally generates readable code.

- Python has many powerful libraries available that when used allow rapid development of complex software without the complexity of rolling ones own solution.

- Python has a large and help community.

- It is very easy to interface with C based code if performance is desired.

- Python is object oriented which when used correctly allows simple creation of modular and extensible software.

- Python is dynamically typed and evaluated which allows rapid iteration and ease of use at the cost of performance and security.

While the use of python does come at some costs, namely memory and CPU performance, on a whole the choice of language provided enormous benefits.

4.2.2.2   Software Libraries

Due to the nature of an experimental system composed of heterogeneous hardware there is a lot of complexity in communicating with hardware over different communication channels. To overcome this challenge the library InstrumentKit[31] was used. InstrumentKit allows identical commands to be sent over a variety of different communication channels at a high level of abstraction. It was a much more simple matter to contribute communication code for necessary sensors and hardware to InstrumentKit and then just utilize the library. As Ni-Engine must handle large volumes of data and manipulate this data, Numpy[32] was used for its efficent handling of arrays in Python and the variety of methods it provides. To store this data the first file format to be implemented was HDF5[33], which is a version of the Hierarchical Data Format. HDF5 was designed for the storage of large amounts of numerical data. Additionally, it provides an easily navigable hierarchical file system. Creation and manipulation of HDF5 files was facilitated by the memory efficient H5py[34] library for Python. Configuration data is in the YAML[35] format and is manipulated using PyYaml[36]. As the main goal of the system is to handle experimental data that is often associated with physical units of measurement Python units[37] was used. Units allows the wrapping of Python data types and Numpy arrays in physical units, and performs the necessary unit operations when performing operations between data types of units.

As extensive documentation is a core design requirement of Ni-Engine, it was essential that a documentation generation library be used throughout development. Sphinx[38], the De facto Python standard for documentation pulls in-code comments and generates relational documentation in a variety of formats including html and pdf formats.

4.2.3   System Architecture

Ni-Engine is split into several major components controlled by the main NiEngine object. Major components are the configuration, hardware, controllers, sensors and storage sections.

4.2.3.1 Design Patterns

Due to the complexity of Ni-Engine it was necessary to use object oriented design patterns. The main NiEngine class controls the execution and control flow of the experimental system. During execution of an experiment all commands initially must go through the NiEngine class. Ni-Engine follows a chain of responsibility pattern and has a strict hierarchy with NiEngine sitting at the top.

Ni-Engine divides the software into five separate components, each of which attempts to be as modular as possible, allowing the modification of internals without the affecting the behaviour of the system as a whole. The main sections are the configuration, hardware, controller, sensors and storage sections. The NiEngine object reads in the experimental configuration via the Configuration object and supplied file. From there, it dispatches creation of experimental software components HardwareManager, ControllerManager, SensorManager and DataHandler objects which are responsible for their self-descriptive duties.
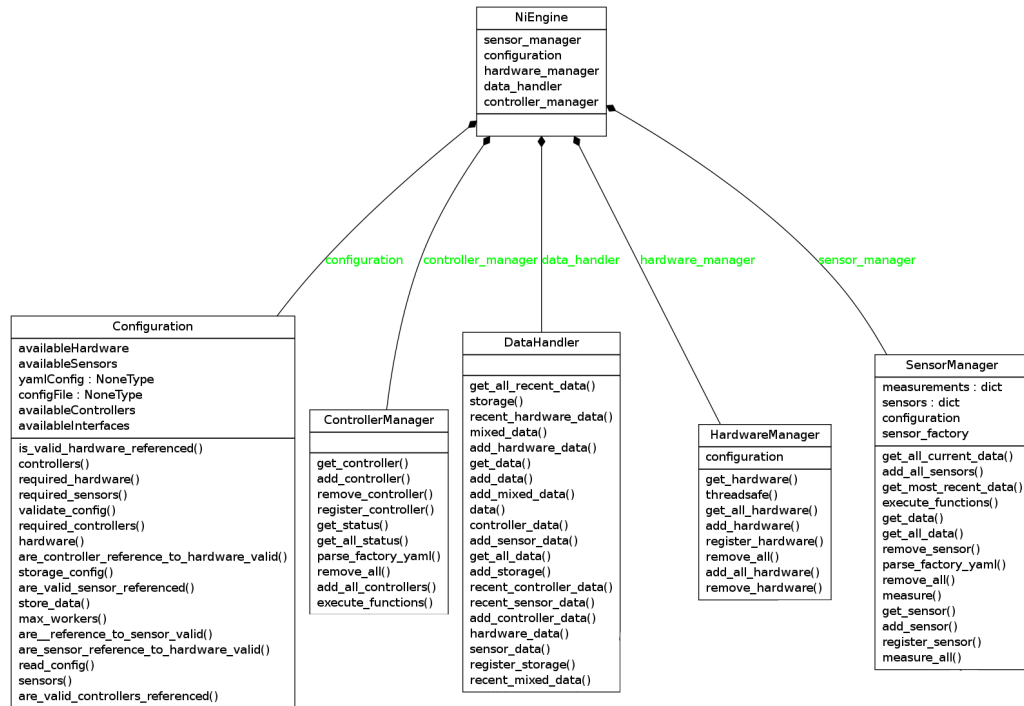

FIGURE 4.5: NiEngine object control hierarchy

4.2.3.2 Configuration

A core tenet of the design of Ni-Engine was that it has to make experiments easily scriptable and repeatable. As experimental hardware requires many different configuration values to be set, that if set incorrectly can result in disastrous consequences it is infeasible to supply configuration via the command-line or as python based configuration. Therefore configuration is provided to Ni-Engine via Yaml configuration files which allows the specification of all runtime configuration options for Ni-Engine. This allows experiments to be run with the same initial settings consistently simply by passing Ni-Engine a predefined configuration file. Configuration is made simple thanks to Yaml's hierarchical data format, which is converted into a very simple to use Python dictionary by PyYaml.

4.2.3.3 Data Storage

Ni-Engine utilizes a storage format agnostic API. All storage communication is channelled through a DataHandler instance. At initialization time DataHandler is a passed its required configuration information as defined in the experiments configuration file. It is possible to utilize many different file formats provide a class that implements the AbstractPhysicalStorage is provided and it is registered with the main Ni-Engine class. This Builder pattern allows future extensibility and interaction, although at current time only the HDF5 file format is supported.

In Ni-Engine all measurements reside in memory as Data objects. Data objects are interesting in that they measurement meta-information with such fields such as measurement names, measurement types and the time the measurement was taken at. All the while Data objects behave as their value data-types in code. This allows measurements to be created with essential meta-information and than to be manipulated in a normal fashion as would integers,floats, arrays and other basic data-types.

Due to the slow nature of permanent storage devices, data to be written to storage are batched in order to limit the amount of write calls. When the buffer is filled it is flushed to the disk. A balance must be established between the amount of data stored in memory compared to how often it is written to disk. Of course any data in memory is ethereal and power failure / program crashes will cause the data to be lost.
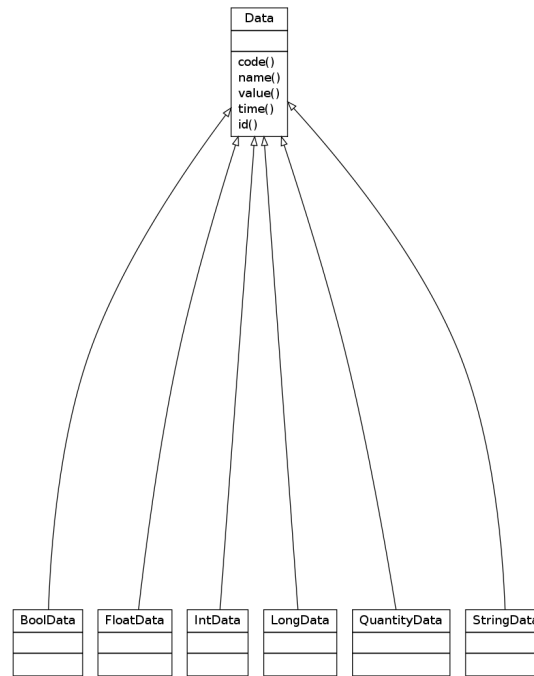
FIGURE 4.6: Datatype measurement class

4.2.3.4 Manager Objects

In Ni-Engine all control of physical hardware is managed by the HardwareManager,ControllerManager and SensorManager objects. Managers are responsible for initialization, configuration, data acquisition and object destruction. Upon initialization these manager classes are passed configuration information that contains all necessary information for the setup and control with all experimental hardware. Each manager object creates its own singleton factory object that is responsible for constructing either the hardware, controllers or sensors for the system depending on its type. Every controller and sensor object interacts with physical hardware through hardware objects. This is to ensure that each physical device is only instantiated once and prevents race conditions. Upon creation of these experimental equipment objects they are placed under management by their respective managers.

Measurement calls to devices are made via manager objects. Manager objects have the ability to measure individual devices or all devices at once. As differing devices have different latencies, sequential measurement can take an excessive amount of time. Therefore measurement calls are threaded to allow measurements to be performed in parallel. Of course certain sensors and controllers may utilize the same hardware and therefore race conditions can result. It is therefore important for these to devices to implement thread locking at the hardware level for these objects. After measurements

are made they are passed onto the DataHandler object for storage as well as returned
to the function call.

4.2.3.5 Documentation

As the purpose of Ni-Engine is to be a general neutron interferometry experimental
engine and not just to be used one time the neutron contrast experiment it is important
that documentation be available for future researchers that may not have the luxury of
a hands on introduction by the software designers. To increase the software ease of use

4.3 Q-Infer

Q-Infer is a software library written in python that implements the Hamiltonian finding
algorithm described in sec(**??**).[39] The library is open-source and was written by Chris
Granade and Dr. Chris Ferrie at the Institute for Quantum Computing. It allows the
Hamiltonian Learning algorithm to be used with minimal coding.

4.3.1 Interaction with NI-Engine

Ni-Engine was designed to allow the experimental testing of Q-Infer. It is simply a
by-product of good software design that it in theory should be usable for many other
projects at the NIST interferometry labs. As both Q-Infer and NI-Engine are separate
libraries they do not share any dependencies and all interaction between the two libraries
is through generic python code. The anticipated workflow for experimental testing of
Q-Infer is seen in fig(4.7).

4.3.2 GPU Implementations of Likelihood functions

One of the most interesting features of Q-Infer and its algorithm is that online adaptive
parameter set learning is possible. This is to say that Q-Infer is able to dynamically
chose experimental parameters to attempt to maximize information gain of the unknown
parameter set. The downside to this is that online evaluation requires that a large num-
ber of potential experimental parameters be evaluated to minimize the Bayes risk of the
experiment. Therefore adaptive selection of experimental parameters to minimize risk
is only viable when the time of selecting the optimal set of parameters is much less than
the time required to perform additional experiments. If the adaptive simulation time
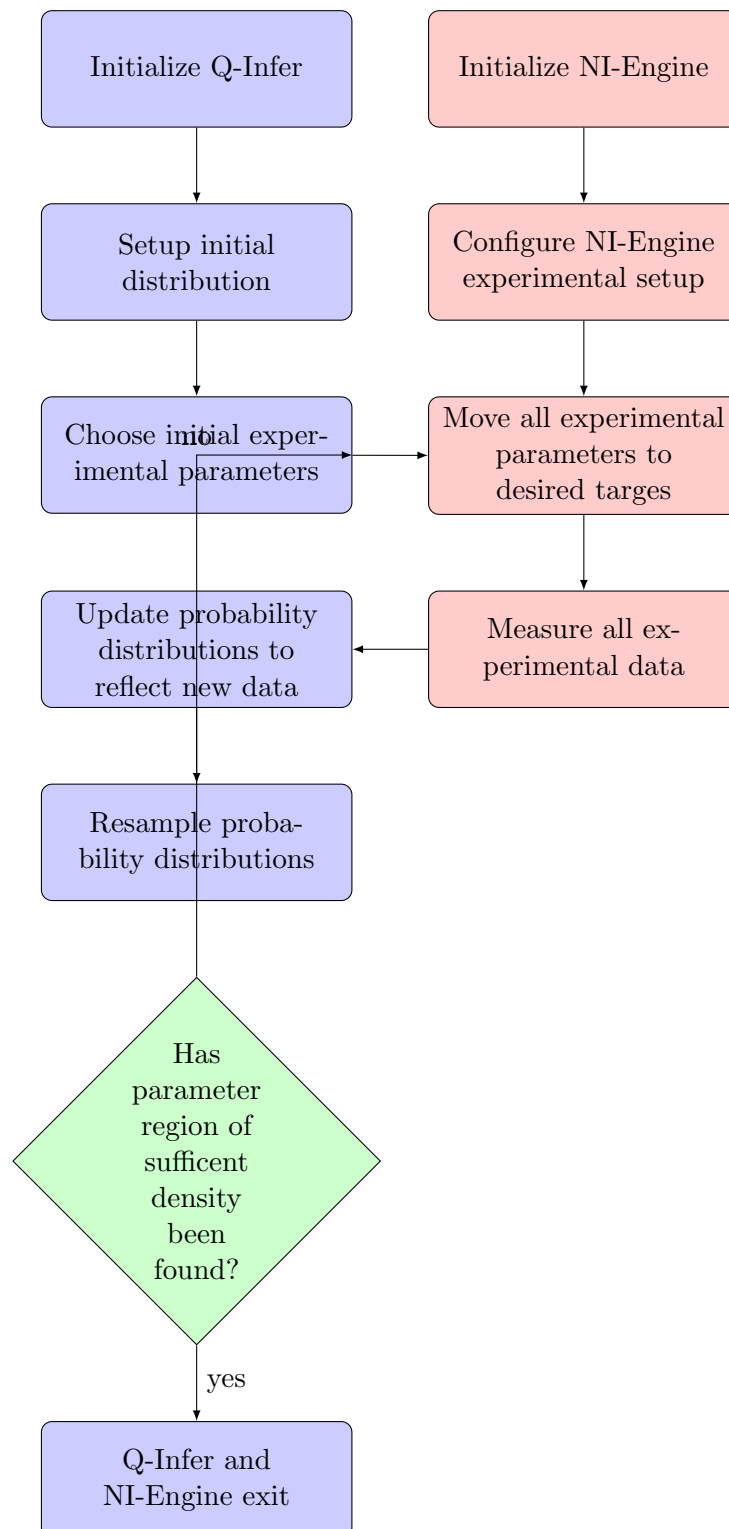
FIGURE 4.7: Experiment flow with NI-Engine and Q-Infer

is too large, it would be quicker to perform additional experiments using experimental parameters selected via some additional means.

In order to increase the feasibility of adaptive measurements we explored the possibility of decreasing simulation time, via faster hardware and algorithms. The likelihood function for the model will be called approximately mp^3n times, when attempting to find the next set of experimental parameters. Where m is the number of iterations the minimization algorithm makes, p is the number of experimental parameters and n is the number of Monte Carlo particles the algorithm is using. when attempting to find the next set of experimental parameters. As a result, the majority of adaptive simulation time is spent evaluating the likelihood function. Currently the form of the calls to the function is

$$\# \text{ Calls} = \sum_{i=0}^{m} \sum_{\mathbf{j=j_i}}^{\mathbf{p}} \sum_{k=0}^{n} P_{i\mathbf{j}k}(0|A, b, \Delta\phi, C, \alpha'; \delta, \hat{\Phi}, \hat{T}) \tag{4.1}$$

In recent years the power of graphics cards have been made available for general-purpose computing on graphics processing units (GPGPU). GPGPU is an entirely different programming paradigm in which all calculations are done in parallel. The benefit of GPGPU is that graphics cards have been able to scale up the amount of computational power available per card by using hundreds to thousands of computational cores, whereas by comparison CPU have been relatively stagnant as they have reached the end of an era of easy performance gains.[40] The downside of GPGPU programming is that only "embarrassingly parallel" problems can easily be computed on a graphics card. The execution of likelihood calls per Bayes update in Q-Infer is performed sequentially currently. In order to speed up the execution of Q-Infer for adaptive measurements we implemented the likelihood function calls in parallel on the GPU using OpenCl.[41] Prior to panellization the $\sum_{k=0}^{n} P_{i\mathbf{j}k}(0|A, b, \Delta\phi, C, \alpha'; \delta, \hat{\Phi}, \hat{T})$ component of (4.1) was executed sequentially on a single thread on the CPU. After panellization each likelihood function $P_{i\mathbf{j}k}(0|A, b, \Delta\phi, C, \alpha'; \delta, \hat{\Phi}, \hat{T})$ was executed simultaneously on the GPU. Speeds up we're observed in the execution time of the SMC algorithm, increasing with the number of particles used. The specifications of the test system can be found in appendix (**??**).

Unfortunately speed-ups were not as large as expected. This is most likely due to the relatively small number of Monte Carlo particles used. As the accuracy of the neutron interferometry model is quite high with $10000 - 20000$ particles there is generally no need to utilize the high number of particles required to achieve a larger speed up from the GPU. This could be due to the naive implementation

Chapter 5

Discussion and Conclusion

5.1 Results

While it was not possible to carry out our proposed experiment to test the algorithm described in section (3.2) it was still possible to model and simulate the behaviour of the algorithm for the proposed system. Our model model attempted to take into account imperfections in the experimental setup, such as variable offsets, temperatures, humidities and phase constants. The derived model for probability of detection at the O-detector is given by (3.37).

Simulation of the behaviour of the MCMC algorithm was performed on a variety of chosen parameter sets for the model, in order to verify if in theory the algorithm would be able to adequately converge on a valid approximate solution. Simulations demonstrated that the algorithm would be able to greatly decrease the Bayes risk and converge on the true parameter set in all tested cases.

The most interesting aspect of the algorithm and our results is that of experimental design. The Robust Online Hamiltonian Learning algorithm allows future experimental parameters to be chosen to maximize the amount of information gain and thus reduce the Bayes risk to a much greater degree with each additional experiment. This is especially useful when experiments require a long time to perform for different experimental parameters and the time of simulation is much less than that of performing additional experiments. As could be seen in fig(**??**) the algorithm could rapidly converge on the true parameter set in as little as 50 experiments, rivalling the accuracy of the naive test of the ridiculous $2^1 5$ phase flag settings experiment. Analysis of the probability density functions, shows that the majority of probability density is concentrated around the true values of parameters.

A downside of the experimental adaptive design is the simulation time required greatly increases. The selection of experimental parameters to maximize information gain greatly increases the computational time. Therefore adaptive experimental design will only actually increase the average information gain per time unit if the cost of simulation is less than the time taken to perform enough experiments to equal the information gained through measurement of choice experimental parameters. In section (4.3.2) we explored implementing the likelihood functions on graphics processing units for GPGPU calculations. Although the speed-up experienced was lower than expected, the are almost certainly large optimizations to be made as our approach was relatively naive and experimental. Additional ways in which the likelihood evaluation speed could be increased is through field programmable gate arrays or application specific circuits (ASIC).

5.2 Ni-Engine

Ni-Engine was programmed to be usable with Q-Infer in order to perform the experimental verification of our simulations. Unfortunately due to scheduling conflicts and the priorities of the research group, both Q-Infer and Ni-Engine have to yet to be put to use at NIST. However, it is anticipated that at some point in the near future, the software and hardware will be installed and we will be able to perform our experimental verification.

Ni-Engine was extensively tested at IQC with the real hardware. While, experiments were not able to be performed on at the neutron interferometry lab, simulated experiments were performed with backups of the lab equipment. This allowed real experiments to be scripted, data measured and stored. This along with the thorough documentation provided allows me to believe that Ni-Engine will have widespread use in the lab when installed due to the benefits it provides over the current experimental system.

5.3 Application to Quantum Information and Fundamentals

The algorithm implemented by Q-Infer has the potential to be very useful in the field of experimental quantum mechanics. Most quantum experimental setups in existence have been modelled mathematically, however all models require that the parameters of the particulars of the system be discovered. This is normally a very experimentally and computationally challenging task as many experiments must be done to gain sufficient information to fit the parameters to the system, and fitting itself is a computationally

intensive task. Our algorithm solves this issue and even though it is using approximate probabilistic methods allows the error in approximated parameters to be calculated. The amount of experiments required with this algorithm are often much less than would be required to characterise the system with quantum tomography. Provided the system has a model the proposed algorithm appears to be an excellent solution to the problem of discovering system parameters.

Bibliography

[1] R. Colella, A. W. Overhauser, and S. A. Werner. Observation of gravitationally induced quantum interference. *Phys. Rev. Lett.*, 34:1472–1474, Jun 1975. doi: 10.1103/PhysRevLett.34.1472. URL http://link.aps.org/doi/10.1103/PhysRevLett.34.1472.

[2] Dmitry A Pushin. *Coherent control of neutron interferometry.* PhD thesis, 2007. URL http://hdl.handle.net/1721.1/39292.

[3] Chupp T. Neutron optics and polarization, 2009. URL http://www.ncnr.nist.gov/summerschool/ss09/pdf/Chupp_FP09.pdf.

[4] A. A. Michelson and E. W. Morley. On the relative motion of the earth and the luminiferous ether. *American Journal of Science*, 34:333–345, 1887. doi: doi:10.2475/ajs.s3-34.203.333. URL http://dx.doi.org/10.1007/s10701-010-9529-9.

[5] Yuji Hasegawa, Rudolf Loidl, Gerald Badurek, Matthias Baron, and Helmut Rauch. Violation of a bell-like inequality in single-neutron interferometry. *Nature*, 425, 2003. ISSN 6953. URL http://dx.doi.org/10.1038/nature01881.

[6] A.G. Klein. Adventures in neutron interferometry. *Foundations of Physics*, 42 (1):147–152, 2012. ISSN 0015-9018. doi: 10.1007/s10701-010-9529-9. URL http://dx.doi.org/10.1007/s10701-010-9529-9.

[7] D. A. Pushin, M. G. Huber, M. Arif, and D. G. Cory. Experimental realization of decoherence-free subspace in neutron interferometry. *Phys. Rev. Lett.*, 107:150401, Oct 2011. doi: 10.1103/PhysRevLett.107.150401. URL http://link.aps.org/doi/10.1103/PhysRevLett.107.150401.

[8] Yuji Hasegawa, Rudolf Loidl, Gerald Badurek, Stefan Filipp, Jürgen Klepp, and Helmut Rauch. Evidence for entanglement and full tomographic analysis of bell states in a single-neutron system. *Phys. Rev. A*, 76:052108, Nov 2007. doi: 10.1103/PhysRevA.76.052108. URL http://link.aps.org/doi/10.1103/PhysRevA.76.052108.

[9] Christopher J. Wood, David G. Cory, Mohamed O. Abutaleb, Michael G. Huber, Muhammad Arif, and Dmitry A. Pushin. Quantum correlations in a noisy neutron interferometer. Jun 2013. URL http://arxiv.org/abs/1304.6935.

[10] Cozmin Ududec, Howard Barnum, and Joseph Emerson. Three slit experiments and the structure of quantum theory. *Foundations of Physics*, 41(3):396–405, 2011. ISSN 0015-9018. doi: 10.1007/s10701-010-9429-z. URL http://dx.doi.org/10.1007/s10701-010-9429-z.

[11] Neutron interferometry and optics facility. http://physics.nist.gov/MajResFac/InterFer/text.html. Accessed: 2014-01-27.

[12] Sam Werner. Neutron interferometry: From missouri to nist. http://webster.ncnr.nist.gov/nran/talks/Werner.pdf. Accessed: 2014-01-27.

[13] Diaconis Persi. The markov chain monte carlo revolution. *American Mathematical Society*, 46(2):179–205, April 2009. URL http://web1.sph.emory.edu/users/hwu30/teaching/statcomp/diaconis.pdf.

[14] K P Zetie, S F Adams, and R M Tocknell. How does a mach-zehnder interferometer work. *Physics Education*, 35(1), 06 1999. URL http://www.cs.princeton.edu/courses/archive/fall06/cos576/papers/zetie_et_al_mach_zehnder00.pdf.

[15] Daniel M. Greenberger. The neutron interferometer as a device for illustrating the strange behavior of quantum systems. *Reviews of Modern Physics*, 55(4):875–905, 1983. URL http://link.aps.org/doi/10.1103/RevModPhys.55.875.

[16] Stephen H. Friedberg, Arnold J. Insel, and Lawrence E. Spence. *Linear Algebra*. Pearson Education, Inc., 4th edition, 2013. ISBN 978-81-203-2606-4.

[17] John K. Kruschke. *Doing Bayesian Data Analysis: A Tutorial with R and BUGS*. Elsevier, 1st edition, October 2010. ISBN 0123814855.

[18] Ben Danforth. Variance-covariance matrix. pages 1–4, June 2009. URL http://www.unc.edu/~jjharden/methods/vcv_week3.pdf.

[19] Frank Nielsen. Cramer-rao lower bound and information geometry. *CoRR*, abs/1301.3578, 2013.

[20] Konstantinos G. Derpanis. Cramer-rao bound. September 2006. URL http://www.cse.yorku.ca/~kosta/CompVis_Notes/cramer-rao.pdf.

[21] J.F. Poyatos, J.I. Cirac, and P. Zoller. Complete characterization of a quantum process: The Two bit quantum gate. *Phys. Rev. Lett.*, 78:390–393, 1997. doi: 10.1103/PhysRevLett.78.390.

[22] B. P. Lanyon, C. Hempel, D. Nigg, M. Mller, R. Gerritsma, F. Zhringer, P. Schindler, J. T. Barreiro, M. Rambach, G. Kirchmair, M. Hennrich, P. Zoller, R. Blatt, and C. F. Roos. Universal digital quantum simulation with trapped ions. *Science*, 334(6052):57–61, 2011. doi: 10.1126/science.1208001. URL http://www.sciencemag.org/content/334/6052/57.abstract.

[23] Christopher E Granade, Christopher Ferrie, Nathan Wiebe, and D G Cory. Robust online hamiltonian learning. *New Journal of Physics*, 14(10):103013, 2012. URL http://stacks.iop.org/1367-2630/14/i=10/a=103013.

[24] Justin Dauwels. Computing bayesian cramer-rao bounds. In *Information Theory, 2005. ISIT 2005. Proceedings. International Symposium on*, pages 425–429. IEEE, 2005.

[25] Richard D. Gill and Boris Y. Levit. Applications of the van trees inequality: a bayesian cramr-rao bound. *Bernoulli*, 1(1-2):59–79, 03 1995. URL http://projecteuclid.org/euclid.bj/1186078362.

[26] Ramis Climent, Romero Romualdo, and Alonso Sergio. Relative humidity. URL http://www.uib.es/depart/dfs/meteorologia/ROMU/formal/relative_humidity/relative_humidity.pdf.

[27] NIST. Water, 2011. URL http://webbook.nist.gov/cgi/cbook.cgi?ID=C7732185&Mask=4&Type=ANTOINE&Plot=on#ANTOINE.

[28] Esp301 integrated 3-axis motion controller/driver, 2013. URL http://assets.newport.com/webDocuments-EN/images/14294.pdf.

[29] INC. Kepco. Series ate. URL http://www.kepcopower.com/specs/atespecs.pdf.

[30] Guido van "Benevolent Dictator For Life" Rossum. Python, 2013. URL http://www.python.org/.

[31] Chris Galvant and Chris Granade. Instrument kit, 2014. URL https://github.com/Galvant/InstrumentKit.

[32] Numpy Developers. Numpy, 2013. URL http://www.numpy.org/.

[33] The HDF Group. Hierarchical data format version 5, 2000-2010. URL http://www.hdfgroup.org/HDF5.

[34] Andrew Collete. Hdf5 for python, 2013. URL http://www.h5py.org/.

[35] Oren Ben-Kiki, Clark Evans, and Ingy dot Net. Yaml aint markup language, October 2009. URL http://www.yaml.org/spec/1.2/spec.html.

[36] Pyyaml, 2014. URL http://pyyaml.org/wiki/PyYAML.

[37] Dale Darren. Python-quantities, 2014. URL https://github.com/python-quantities/python-quantities.

[38] Sphinx Team. Sphinx: Python documentation generator, 2014. URL http://sphinx-doc.org/.

[39] Chris Granade and Chris Ferrie. Python q-infer, 2014. URL https://github.com/csferrie/python-qinfer.

[40] Tony King-Smith and Alexandru Voica. The rise of gpu compute, August 2012. URL http://www.computer.org/portal/web/computingnow/high%20performance/content?g=53319&type=article&urlTitle=the-rise-of-gpu-compute.

[41] Khronos Group. Opencl: The open standard for parallel programming of heterogeneous systems, 2014. URL https://www.khronos.org/opencl/.