

A PLAN OF CONCENTRATION IN THE NATURAL SCIENCES

Crystal Diffraction and Quantum Wave Packet Scattering

Author:

Joshua LANDE

Plan Sponsors:

Dr. Travis NORSEN

Dr. James MAHONEY

MARLBORO COLLEGE

Marlboro, Vermont

May, 2008

Table of Contents

Table of Contents	3
I Introduction	9
II A Discussion of Reflection and Transmission...	13
1 A Derivation of the Reflection and Transmission Coefficients	17
1.1 Definitions	17
1.2 Normalization of the Energy Eigenstates	18
1.3 A Gaussian Wave Packet	21
1.4 ψ as a Superposition of Energy Eigenstates	22
1.5 The Time Evolution of ψ	23
1.6 The R Term	24
1.7 The T Term	24
1.8 Interpretation of Results	25
1.9 Arbitrary Wave Packets	26
2 A Special Case – The Step Potential	29
3 Another Example: the Finite Potential Barrier	33
3.1 The Eigenfunctions	33
3.2 A Cute Argument	35
III A Theoretical Discussion of Crystal Diffraction...	39
4 Crystal Diffraction	43
4.1 The Bravais Lattice	43
4.2 The Reciprocal Lattice	44
4.3 Crystal Diffraction	46
4.4 Visualization of Diffraction	49

5	Powder Diffraction	51
6	Microwave Optics	55
IV	Area Diffraction Machine Manual	59
7	Tips and Tricks	61
7.1	Calibration	61
7.2	Masking	62
7.3	Caking	63
7.4	Integrate	64
7.5	Macro	64
8	An Example	65
9	Viewing Diffraction Data	79
9.1	File Formats	81
9.2	Loading Multiple Images	81
9.3	Saving the Diffraction Image	81
10	Detector Geometries	83
10.1	The Three Tilt Angels	85
10.2	The β Tilt	85
10.3	The α Roll	86
10.4	The R Rotation	88
10.5	Relationship to Pixel Coordinates	89
10.6	Inverting the Equations	89
10.7	Q , 2θ , and χ	89
11	Calibration	91
11.1	The Calibration Algorithm	91
11.2	The Fitting	92
11.3	Calibrating With the Program	94
11.4	The “Number of Chi?” and “Stddev?” Input	96
11.5	Work in λ	97
11.6	Fixing Calibration Parameters	97
11.7	Displaying Constant Q Lines	97
11.8	Displaying Constant ΔQ Lines	99
11.9	Displaying Peaks	99
11.10	Masking Peaks	100
11.11	Saving the Peak List	101
11.12	Handling Calibration Data	102
11.13	Handling Q Data	102

11.14	The “Get From Header?” Input	103
12	Pixel Masking	105
12.1	Threshold Masking	105
12.2	Polygon Masking	108
12.3	Masking Caked Plots	111
13	Caking	113
13.1	The Caking Algorithm	113
13.2	Caking with the Program	114
13.3	AutoCake	114
13.4	Displaying Q and ΔQ Lines	116
13.5	Displaying Peaks	118
13.6	Polarization Correction	118
13.7	Working in 2θ	118
13.8	Saving Cake Images	119
13.9	Saving Cake Data	119
14	Intensity Integration	123
14.1	The Integration Algorithm	123
14.2	Integrating with the Program	124
14.3	The Integration Window	126
14.4	Working in 2θ	126
14.5	AutoIntegrate	126
14.6	Constraining the Inputs	127
14.7	Masking	127
14.8	Saving Integrated Data	127
15	Macros	129
15.1	Record Macros	129
15.2	Run Macros	129
15.3	The Macro File Format	130
15.4	Looping Over Diffraction Data	131
15.5	The PATHNAME and FILENAME Commands	132
15.6	Loops Over Multiple Images	133
15.7	The FOLDERPATH and FOLDERNAME commands	134
15.8	Setting Colors in a Macro	134
15.9	Little Tidbits	135
15.10	Macro Commands	135
15.11	What You Can’t Do With Macros	143
16	Software Licensing	145

V	Plan Exams	147
17	Comprehensive Exam, part1	149
17.1	A solid spherical ball of uniform mass density (e.g., a pool ball) rolls without	149
17.2	A planet orbits the sun under the influence of the gravitational force. Suppose	150
17.3	Alice and Bob are at opposite ends of a spaceship whose rest length $L = 20cs$	153
17.4	The point of suspension of a pendulum (mass m , length L) is allowed to move	156
17.5	A bucket full of water rotates at uniform angular velocity ω . It is near the surf	157
18	Comprehensive Exam, part2	159
18.1	A charge $+Q$ is distributed uniformly along the z axis from $z = a$ to $z = +a$.	159
18.2	A long coaxial cable is made from two conducting cylindrical shells of radius	162
18.3	An AC voltage source (amplitude V_0 , angular frequency ω) drives a circuit con	164
18.4	Finn has a toy magnifying glass designed to look at bugs. Its a cylinder whose	165
18.5	Find the transmission coefficient for light waves passing through a pane of gla	167
18.6	A circular coil of wire (radius R) carries current I and lies in the $x - y$ plane	170
19	Comprehensive Exam, part3	173
19.1	What is Compton scattering? What role did it play in the early days of quant	173
19.2	Calculate the lifetime (in seconds) for each of the four $n = 2$ states of hydrog	175
19.3	Estimate (or really: put a bound on) the ground state energy of Hydrogen usi	181
19.4	You've learned about the Born approximation in the context of 3D scattering	182
19.5	Let ... be the matrix representation of the Hamiltonian for a three-state syste	184
19.6	What is Bell's Theorem and what does it prove? (No need to recapitulate the	186
20	Comprehensive Exam, part4	189
20.1	Here is a very simplified model of the unwinding of two-stranded DNA molecu	189
20.2	A cold white dwarf is held up against gravitational collapse by the pressure of	190
20.3	Suppose a star were made of an ideal gas composed of molecules of mass m at	193
20.4	The latent heat (or 'heat of fusion') for the ice-water phase transition is 80 cal	194
20.5	In The Physical Universe, Shu discusses the 'Missing-Mass Problem' on pages	195
20.6	The heat capacity of non-metallic solids at sufficiently low temperatures is pro	197
20.7	Considering the earth as a thermodynamic system, it's clear that over geologic	198
VI	Appendix	199
A	How and Why to Think about Scattering...	201
A.1	Introduction	201
A.2	The plane-wave account and its problems	203
A.3	Scattering probabilities and packet widths	207
A.4	Gaussian wave packet scattering from a step potential	210
A.5	Discussion	213

Part I

Introduction

The largest portion of my plan is software that I wrote to perform an analysis of X-Ray Powder diffraction analysis. I became interested in diffraction initially last the summer. I participated in the Science Undergraduate Laboratory Internships and interned at the Stanford Synchrotron Radiation Laboratory which is at the Stanford Linear Accelerator Center. My advisor was Dr. Samuel Webb, a beamline scientist who is involved with diffraction research performed at the Synchrotron. My project was initially born out of frustration with available that does powder diffraction analysis.

I would like to acknowledge all those who helped me write this plan. This research was partially funded by my family. Thanks!

Part II

A Discussion of the Reflection and Transmission Coefficients for Wave Packet Scattering off an Arbitrary Potential Barrier

Abstract

In this paper, we will derive the reflection and transmission coefficients for a Gaussian wave packet to travel through an arbitrary potential barrier. These equations should be thought of as the fundamental equations governing reflection and transmission. We will show that our equations reduce in the wide wave packets limit to the plane wave reflection and transmission coefficients derived for plane waves should be in turn thought of as approximate value valid for wide wave packets. We will extend this argument to arbitrary wave packets and show how to generalize the argument. We will then take as a special case the step potential and perform a Taylor expansion of the reflection and transmission coefficients to derive approximate (but more accurate than plane wave) expressions for the coefficients. We will then examine the rectangular barrier potential and show that our expression for the reflection and transmission coefficients are qualitatively different from those from the plane wave approximation. This provides a good example of why our equations for the reflection and transmission coefficients should be thought of as fundamental.

Chapter 1

A Derivation of the Reflection and Transmission Coefficients

1.1 Definitions

We can characterize an arbitrary potential that we would like a wave packet to scatter off as

$$V(x) = \begin{cases} 0 & x < 0 \\ V_m(x) & 0 \leq x \leq a \\ V_0 & x > a. \end{cases} \quad (1.1)$$

Here, we assume that the potential is zero until $x = 0$ and that it ends at some constant value V_0 as $x = a$. Other than that, it can do anything between $x = 0$ and $x = a$. Figure 1.1 shows a sketch of this potential for some particular $V_m(x)$.

We are interested in finding the energy eigenstates for this potential. We will do this by finding general solutions to the time independent Schrödinger equation. For this potential, wavefunctions have the form

$$\psi_k = \begin{cases} Ae^{ikx} + Be^{-ikx} & x < 0 \\ \psi_m(x) & 0 \leq x \leq a \\ Ce^{i\kappa x} & x > a. \end{cases} \quad (1.2)$$

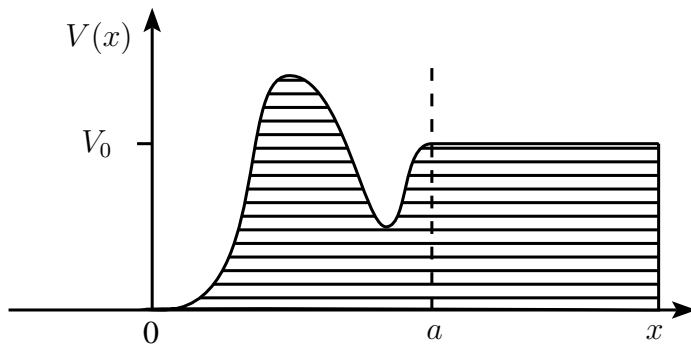


Figure 1.1: Here is a plot of the potential defined in equation 1.1. The potential is 0 for values less than 0 and V_0 for values greater than a . The potential is some arbitrary unspecified function for values in between.

with

$$\kappa^2 = k^2 - \frac{2mV_0}{\hbar^2} = k^2 - p^2. \quad (1.3)$$

This is only a formal solution because we do not know what $\psi_m(x)$ is. All we can say is that it is some arbitrary solution to the Schrödinger equation for the arbitrary potential $V_m(x)$ that ensures that both ψ and ψ' are continuous. We can do quite a lot without actually specifying exactly what $\psi_m(x)$ is. We write the probability current for the incoming and outgoing waves as

$$j = -\frac{i\hbar}{2m} \left(\psi^* \frac{\partial \psi}{\partial x} - \psi \frac{\partial \psi^*}{\partial x} \right). \quad (1.4)$$

For a plane wave with $\psi = Ae^{ikx}$, the probability current is

$$j = \frac{\hbar k}{m} |A|^2. \quad (1.5)$$

Because the probability current must be conserved, the probability current for the incoming wave must equal the probability current for the reflected and transmitted waves:

$$\frac{\hbar k}{m} |A|^2 = \frac{\hbar k}{m} |B|^2 + \frac{\hbar \kappa}{m} |C|^2. \quad (1.6)$$

This simplifies to

$$\left| \frac{B}{A} \right|^2 + \frac{\kappa}{k} \left| \frac{C}{A} \right|^2 = 1. \quad (1.7)$$

1.2 Normalization of the Energy Eigenstates

Because ψ_k involves infinite plane waves, the particular normalization that we pick is in some sense arbitrary. Will pick following normalization¹

$$\psi_k(x) = \frac{1}{\sqrt{2\pi}} \left[\left(e^{ikx} + \frac{B}{A} e^{-ikx} \right) \theta(-x) + \psi_{k,m}(x) \theta(x) \theta(a-x) + \frac{C}{A} e^{i\kappa x} \theta(x-a) \right]. \quad (1.8)$$

This convention is picked so that

$$\int \psi_{k'}^* \psi_k dx = \delta(k - k') \quad (1.9)$$

¹ Here, we have defined

$$\theta(x) = \begin{cases} 0 & x < 0 \\ 1 & x > 0. \end{cases}$$

It follows that

$$\theta(x) \theta(a-x) = \begin{cases} 0 & x < 0 \\ 1 & 0 < x < a \\ 0 & x > a. \end{cases}$$

We can prove this as follows:

$$\begin{aligned} \int \psi_{k'}^*(x) \psi_k(x) dx = & \frac{1}{2\pi} \int \left[\left(e^{-ik'x} + \frac{B'^*}{A'^*} e^{+ik'x} \right) \theta(-x) + \psi_{k',m}^*(x) \theta(x) \theta(a-x) + \frac{C'^*}{A'^*} e^{-i\kappa'x} \theta(x-a) \right] \\ & \times \left[\left(e^{ikx} + \frac{B}{A} e^{-ikx} \right) \theta(-x) + \psi_{k,m}(x) \theta(x) \theta(a-x) + \frac{C}{A} e^{i\kappa x} \theta(x-a) \right] dx. \end{aligned} \quad (1.10)$$

The primes on A' , B' , C' , and κ' are necessary because they are functions of k (really k' in this case). Since the θ functions are orthogonal and square to themselves, we have

$$\begin{aligned} \int \psi_{k'}^*(x) \psi_k(x) dx = & \frac{1}{2\pi} \int_{-\infty}^0 e^{i(k-k')x} dx + \frac{1}{2\pi} \int_{-\infty}^0 \frac{B}{A} e^{-i(k+k')x} dx \\ & + \frac{1}{2\pi} \int_{-\infty}^0 \frac{B'^*}{A'^*} e^{i(k+k')x} dx + \frac{1}{2\pi} \int_{-\infty}^0 \frac{B'^* B}{A'^* A} e^{i(k'-k)x} dx \\ & + \frac{1}{2\pi} \int_0^a \psi_{k',m}^* \psi_{k,m} dx + \frac{1}{2\pi} \int_a^\infty \frac{C'^* C}{A'^* A} e^{i(\kappa-\kappa')x} dx. \end{aligned} \quad (1.11)$$

The first term is equal to $\delta(k-k')/2$.² The second and third term are each proportional to $\delta(k+k')$. Since the eigenfunctions we are dealing with are only for incoming waves, all our waves have positive wave vector. It must be that $k+k'$ is positive so $\delta(k+k') = 0$. The second and third term integrate to 0. The fourth term is equal to $(B'^* B/A'^* A) \times \delta(k-k')/2$. The sixth term is a more complicated. It is equal to

$$\frac{1}{2\pi} \int_a^\infty \frac{C'^* C}{A'^* A} e^{i(\kappa-\kappa')x} dx = e^{-(\kappa-\kappa')a} \frac{1}{2\pi} \int_0^\infty \frac{C'^* C}{A'^* A} e^{i(\kappa-\kappa')x} dx \quad (1.13)$$

$$= e^{-(\kappa-\kappa')a} \frac{C'^* C}{A'^* A} \frac{\delta(\kappa-\kappa')}{2} \quad (1.14)$$

$$= \frac{C'^* C}{A'^* A} \frac{\delta(\kappa-\kappa')}{2}. \quad (1.15)$$

In the second step, we have changed variables from x to $x+a$. In the last step, the phase was ignored since it is unity whenever the other terms have support. This simplifies to³

$$\frac{\kappa'}{k} \frac{C'^* C}{A'^* A} \delta(\kappa-\kappa') \quad (1.22)$$

² To prove this, we use the identity

$$\delta(k-k') = \frac{1}{2\pi} \int_{-\infty}^\infty e^{i(k-k')x} dx. \quad (1.12)$$

³ To prove this, we need to write $\delta(\kappa-\kappa')$ in terms of $\delta(k-k')$. We can do so using the identity

$$\delta(g(x)) = \sum_i \frac{\delta(x-x_i)}{dg(x_i)/dx}. \quad (1.16)$$

Plugging all the terms in, we get

$$\int \psi_{k'}^*(x) \psi_k(x) dx = \frac{\delta(k - k')}{2} + \frac{B'^* B}{A'^* A} \times \frac{\delta(k - k')}{2} + \int_0^a \psi_{k',m}^* \psi_{k,m} dx + \frac{\kappa' C'^* C}{k' A'^* A} \frac{\delta(k - k')}{2}. \quad (1.23)$$

Since these delta functions are non-zero only when $k = k'$, we can without loss of generality replace the primed values in the coefficients with their unprimed values. We get

$$\int \psi_{k'}^*(x) \psi_k(x) dx = \frac{\delta(k - k')}{2} + \frac{|B|^2}{|A|^2} \times \frac{\delta(k - k')}{2} + \int_0^a \psi_{k',m}^* \psi_{k,m} dx + \frac{\kappa |C|^2}{k |A|^2} \frac{\delta(k - k')}{2} \quad (1.24)$$

$$= \delta(k - k') + \int_0^a \psi_{k',m}^* \psi_{k,m} dx \quad (1.25)$$

Our wave functions must be orthogonal in order for them to be an eigenstate of the Hamiltonian. For this to be true, it must be the case that

$$\int_0^a \psi_{k',m}^* \psi_{k,m} dx \Big|_{k \neq k'} = 0. \quad (1.26)$$

Here x_i are the real roots of $g(x)$. To use this identity, we can think of $\kappa - \kappa'$ as $g(x)$. The slightly confusing thing is that we have been thinking of both κ and κ' as variables while the identity works with only one variable. We will ‘think’ of κ as the variable and κ' as a constant. When we do this, we note that the real roots of our function are $k = k'$ and $k = -k'$.

$$\delta(\kappa - \kappa') = \delta(\sqrt{k^2 - 2mV_0/\hbar^2} - \sqrt{k'^2 - 2mV_0/\hbar^2}) \quad (1.17)$$

$$= \delta(g(k)) \quad (1.18)$$

$$= \frac{\delta(k - k')}{dg(k')/dk} + \frac{\delta(k + k')}{dg(-k')/dk}. \quad (1.19)$$

Since we are only dealing with incident plan waves where k is positive, the second part of this equation is equal to 0. We can calculate the denominator of the first term as

$$\frac{dg(k)}{dk} = \frac{1}{2} \frac{1}{\sqrt{k^2 - 2mV_0/\hbar^2}} \times 2k \quad (1.20)$$

$$= k/\kappa \quad (1.21)$$

It follows that $\delta(\kappa - \kappa') = (\kappa'/k') \times \delta(k - k')$.

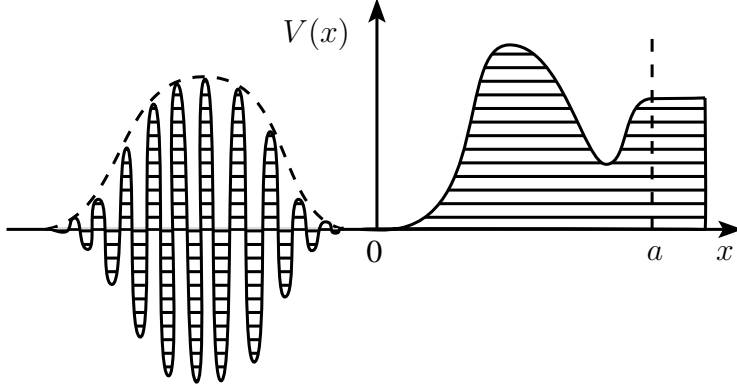


Figure 1.2: This is a plot of the real part of $\psi(x, 0)$ defined in equation 1.32. The plot also shows $V(x)$ on top of it. This plot is possibly misleading because $V(x)$ and $\psi(x, 0)$ have different units and are therefore not comparable. They are simply plotted on top of each other. The scale of one can not be compared to the scale of another.

Furthermore, the value of this integral when $k = k'$ must be finite. Thus, when $k = k'$, we have⁴

$$\int_0^a \psi_{k',m}^* \psi_{k,m} dx \Big|_{k=k'} = \infty + [\text{finite value}] = \infty \quad (1.30)$$

We see that

$$\int_0^a \psi_{k',m}^* \psi_{k,m} dx = \delta(k - k'). \quad (1.31)$$

1.3 A Gaussian Wave Packet

We are interested in calculating the reflection and transmission coefficients of a wave packet through an arbitrary potential. Because of its simplicity, we will first work with a wave packet that is Gaussian. We write our initial wavefunction as

$$\psi(x, 0) = (\pi\sigma^2)^{-1/4} e^{ik_0(x+a)} e^{-(x+a)^2/2\sigma^2}. \quad (1.32)$$

Our wave packet is centered at $-a$ in position space and k_0 in k space. Figure 1.2 shows a plot of $\psi(x, 0)$ and a plot of $V(x)$. We are interested in calculating $\lim_{t \rightarrow \infty} \psi(x, t)$. The

⁴We can make this argument a little bit more rigorous by integrating this function in k space over an infinitesimal range from $k' - \epsilon$ to $k' + \epsilon$

$$\int_{k' - \epsilon}^{k' + \epsilon} \int_{-\infty}^{\infty} \psi_{k'}^* \psi_k dx dk = \int_{k' - \epsilon}^{k' + \epsilon} \left(\delta(k - k') + \int_0^a \psi_{k',m}^* \psi_{k,m} dx \right) dk \quad (1.27)$$

$$= 1 + \int_0^a \int_{k' - \epsilon}^{k' + \epsilon} \psi_{k',m}^* \psi_{k,m} dk dx \quad (1.28)$$

The second term integrates to 0 for any wave function that is finite at all points. This will always hold so long as the potential doesn't do anything funny like go off to infinity. By definition then, we see that

$$\int \psi_{k'}^*(x) \psi_k(x) dx = \delta(k - k'). \quad (1.29)$$

fraction of the wave packet to the left of $x = 0$ is the reflection probability and the fraction of the wave packet to the right of $x = 0$ is the transmission probability. To calculate this, we will write out our wave function as a linear combination of the energy eigenfunction. Then, we will advance ψ in time (by multiplying each eigenfunction by an energy phase). We will be able to take the large t limit to find the percent of the wave to the left and to the right.

1.4 ψ as a Superposition of Energy Eigenstates

We are interested in writing $\psi(x, 0)$ as a superposition of plane wave states

$$\psi(x, 0) = \int \phi(k) \psi_k(x) dk. \quad (1.33)$$

We have to solve for $\phi(k)$:

$$(\pi\sigma^2)^{-1/4} e^{ik_0(x+a)} e^{-(x+a)^2/2\sigma^2} = \int \psi_k(x) \phi(k) dk \quad (1.34)$$

We can multiply each side by $\psi_{k'}^*(x)$ and integrate over all x :

$$\int \psi_{k'}^*(x) (\pi\sigma^2)^{-1/4} e^{ik_0(x+a)} e^{-(x+a)^2/2\sigma^2} dx = \int \int \psi_{k'}^*(x) \psi_k(x) \phi(k) dx dk \quad (1.35)$$

$$= \phi(k') \quad (1.36)$$

Here, we have used the orthogonality of the ψ_k , Since there are no corresponding unprimed variables, we can get rid of the primes in this equation

$$\phi(k) = (\pi\sigma^2)^{-1/4} \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} \left[\left(e^{ikx} + \frac{B}{A} e^{-ikx} \right) \theta(-x) + \psi_{k,m}(x) \theta(x) \theta(a-x) + \frac{C}{A} e^{ikx} \theta(x-a) \right] \times e^{ik_0(x+a)} e^{-(x+a)^2/2\sigma^2} dx. \quad (1.37)$$

We can deal with the four terms separately. We assume that the wave packet comes in from far to the left so $w \ll a$. This means that the term $\exp(-(x+a)^2)$ has vanishing support for $x > 0$. The first term in the equation is

$$\frac{(\pi\sigma^2)^{-1/4}}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-ikx} e^{ik_0(x+a)} e^{-(x+a)^2/2\sigma^2} \theta(-x) dx. \quad (1.38)$$

Since the Gaussian has vanishing support for $x > 0$, we can approximate this integral by removing the $\theta(-x)$ term We have

$$\frac{(\pi\sigma^2)^{-1/4}}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-ikx} e^{ik_0(x+a)} e^{-(x+a)^2/2\sigma^2} dx. \quad (1.39)$$

We make the change of variables $x' = x + a$

$$\frac{(\pi\sigma^2)^{-1/4}}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{i(k_0 x' - k x' + ka)} e^{-x'^2/2\sigma^2} dx' = e^{ika} \frac{(\pi\sigma^2)^{-1/4}}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{i(k_0 - k)x'} e^{-x'^2/2\sigma^2} dx'. \quad (1.40)$$

Solving this integral, we find that our first term is⁵

$$\left(\frac{\sigma^2}{\pi}\right)^{1/4} e^{-(k-k_0)^2\sigma^2/2} e^{ika} \quad (1.41)$$

We can now deal with the second term. We can get rid of the $\theta(-x)$ term since it is equal to 1 when the rest of the integrand has support. What is left is the ordinary Fourier transform integral which gives the $-k$ component of the original ψ . But we assume that our incoming wave packet has no left moving wave components and thus this integral must be 0.⁶

The third and fourth terms both integrate to 0. This is because the exponential term in the integrals have vanishing support for $x > 0$ whereas the $\theta(x)\theta(a-x)$ and $\theta(x)$ terms have vanishing support for $x < 0$. The function has no support for all values. The integral must be 0.

From this, we see that

$$\phi(k) = \left(\frac{\sigma^2}{\pi}\right)^{1/4} e^{-(k-k_0)^2\sigma^2/2} e^{ika}. \quad (1.42)$$

1.5 The Time Evolution of ψ

We can write the wave function for our Gaussian wave packet at any arbitrary time by multiplying all the eigenstates in equation 1.33 by the phase associated with their time evolution:

$$\psi(x, t) = \int \phi(k) \psi_k(x) e^{-iE(k)t/\hbar} \quad (1.43)$$

$$= \int \left(\frac{\sigma^2}{\pi}\right)^{1/4} e^{-(k-k_0)^2\sigma^2/2} e^{ika} \times \frac{1}{\sqrt{2\pi}} \left[\left(e^{ikx} + \frac{B}{A} e^{-ikx} \right) \theta(-x) + \right. \\ \left. \psi_{k,m}(x) \theta(x) \theta(a-x) + \frac{C}{A} e^{i\kappa x} \theta(x-a) \right] e^{-i\hbar k^2 t/2m} dk. \quad (1.44)$$

⁵ Here we are using the identity

$$\int_{-\infty}^{\infty} e^{-ax^2+bx} dx = e^{b^2/4a} \sqrt{\frac{\pi}{a}}.$$

⁶This is actually an approximation. The wave packet's distribution in k -space is Gaussian so there is some amplitude for the packet to have any k value—even negative values! The amplitude for the packet to have a negative k values and therefore the Fourier component for the packet to be moving to the left will be negligible so long as the width of the wave packet in k space (roughly $1/\sigma$) is small in comparison to the central value of k . This is a reasonable assumption of a well defined wave packet incident on the barrier.

The first term in this equation corresponds to the incoming wave. It dies out for large t . The second term is the reflected wave.

$$\psi_R(x, t) = \left(\frac{\sigma^2}{4\pi^3} \right)^{1/4} \int e^{-i\hbar k^2 t/2m} e^{-(k-k_0)^2 \sigma^2/2} e^{ika} \left(\frac{B}{A} \right) e^{-ikx} \theta(-x) dk. \quad (1.45)$$

The third term is the part of the wave in the potential $V(x)$. It also dies out for large t . The fourth term is the transmitted wave.

$$\psi_T(x, t) = \left(\frac{\sigma^2}{4\pi^3} \right)^{1/4} \int e^{-i\hbar k^2 t/2m} e^{-(k-k_0)^2 \sigma^2/2} e^{ika} \left(\frac{C}{A} \right) e^{i\kappa x} \theta(x - a) dk \quad (1.46)$$

1.6 The R Term

In order to calculate the reflection coefficient, we examine the reflected part of the wave packet $\psi_R(x, t)$. For large t , this wave will exist only for $x < 0$. We can remove the $\theta(-x)$ term. If we make the substitution $x = -x$, we can now write $\psi_R(x, t)$ as

$$\psi_R(x, t) = \int \frac{e^{ikx}}{\sqrt{2\pi}} \phi_R(k) dk. \quad (1.47)$$

We have⁷

$$\psi_R(x, t) = - \left(\frac{\sigma^2}{4\pi^3} \right)^{1/4} \int e^{-i\hbar k^2 t/2m} e^{-(k-k_0)^2 \sigma^2/2} e^{ika} \left(\frac{B}{A} \right) e^{ikx} dk. \quad (1.48)$$

Therefore,

$$\phi_R(k) = - \left(\frac{\sigma^2}{\pi} \right)^{1/4} e^{-i\hbar k^2 t/2m} e^{-(k-k_0)^2 \sigma^2/2} e^{ika} \left(\frac{B}{A} \right). \quad (1.49)$$

We write the reflection coefficient as

$$R = \lim_{t \rightarrow \infty} \int |\phi_R(k)|^2 dk. \quad (1.50)$$

$$R = \left(\frac{\sigma^2}{\pi} \right)^{1/2} \int e^{-(k-k_0)^2 \sigma^2} \left| \frac{B}{A} \right|^2 dk. \quad (1.51)$$

1.7 The T Term

In order to calculate the transmission coefficient, we examine the transmitted part of the wave packet $\psi_T(x, t)$. For large t , the transmitted packet is entirely to the right of $x = a$

⁷Remember that when we make the change of variables, the limits of integration change. The limits are changed back at the cost a minus sign.

and we can remove the $\theta(x - a)$ term. When we make the change of variables from k to $\sqrt{k^2 + p^2}$, we can write $\psi_T(x, t)$ as

$$\psi_T(x, t) = \int \frac{e^{ikx}}{\sqrt{2\pi}} \phi_T(k) dk \quad (1.52)$$

where

$$\psi_T(x, t) = \left(\frac{\sigma^2}{4\pi^3} \right)^{1/4} \int e^{-i\hbar(k^2+p^2)t/2m} e^{-(\sqrt{k^2+p^2}-k_0)^2\sigma^2/2} e^{ika} \left(\frac{C}{A} \right) e^{ikx} \frac{k}{\sqrt{k^2+p^2}} dk. \quad (1.53)$$

Therefore,

$$\phi_T(k) = \left(\frac{\sigma^2}{\pi} \right)^{1/4} e^{-i\hbar(k^2+p^2)t/2m} e^{-(\sqrt{k^2+p^2}-k_0)^2\sigma^2/2} e^{ika} \left(\frac{C}{A} \right) \frac{k}{\sqrt{k^2+p^2}}. \quad (1.54)$$

We can then write the total probability of transmission as

$$T = \lim_{t \rightarrow \infty} \int |\phi_T(k)|^2 dk. \quad (1.55)$$

For our wave packet, we have

$$T = \left(\frac{\sigma^2}{\pi} \right)^{1/2} \int e^{-(\sqrt{k^2+p^2}-k_0)^2\sigma^2} \left| \frac{C}{A} \right|^2 \frac{k^2}{k^2+p^2} dk. \quad (1.56)$$

We can then change variables back to $k = \sqrt{k^2 - p^2}$

$$T = \left(\frac{\sigma^2}{\pi} \right)^{1/2} \int e^{-(k-k_0)^2\sigma^2} \left| \frac{C}{A} \right|^2 \frac{\sqrt{k^2 - p^2}}{k} dk. \quad (1.57)$$

We get⁸

$$T = \left(\frac{\sigma^2}{\pi} \right)^{1/2} \int e^{-(k-k_0)^2\sigma^2} \frac{\kappa}{k} \left| \frac{C}{A} \right|^2 dk. \quad (1.58)$$

1.8 Interpretation of Results

When we take the wide wave packet limit, our wavefunction approaches a plane wave with wave vector k_0 . In this limit, we can approximate $|C/A|$ and $|B/A|$ by their value at $k = k_0$. When we do so, we find that

$$\lim_{t \rightarrow \infty} R = \left| \frac{B}{A} \right|^2 \quad \lim_{t \rightarrow \infty} T = \frac{\kappa}{k} \left| \frac{C}{A} \right|^2 \quad (1.59)$$

⁸There is a one part of this derivation that I glossed over. A and C are functions of k so they actually change during our change of variables. Technically, we should probably denote them with a new name after the change. But we then change variables back and the A and C term revert to their previous value. No harm is caused by this omission.

These are what are typically called the reflection and transmission coefficients R_k and T_k for wave packet scattering off a step barrier. We see that these terms are approximate and valid in the wide wave packet limit whereas the equations we derive are exact.

Furthermore, equation 1.42 can be interpreted to show that the probability amplitude for our wave packet to have wave vector k is

$$P(k) = |\phi(k)|^2 = e^{-(k-k_0)^2\sigma^2}. \quad (1.60)$$

For our Gaussian wave packet, we can write the reflection and transmission coefficients as

$$R = \int P(k)R_k dk \quad T = \int P(k)T_k dk. \quad (1.61)$$

These equations are just what we would expect. The reflection coefficient of a plane wave is just the sum (or technically integral) of the probability of the incoming wave packet having a particular wave vector times the reflection coefficient for a wave having that wave vector.

1.9 Arbitrary Wave Packets

Our derivation of equations 1.61 was actually much less general than it needs to be. There is no reason that we need to assume that the incoming wave packet is Gaussian. Instead, we can just say that there is some incoming wave packet $\psi(x, t)$ which initially is far to the left of $x = 0$. Next, we write

$$\psi(x, 0) = \int \psi_k(x)\phi(k)dk. \quad (1.62)$$

So,

$$\phi(k)dk = \int \psi_k^* \psi(x, 0)dx. \quad (1.63)$$

It is still true that

$$P(k) = |\phi(k)|^2. \quad (1.64)$$

Although we cannot work out an analytic expression for $\phi(k)$, we can still carry through with the calculations.

$$\psi(x, t) = \int \phi(k)\psi_k(x)e^{-iE(k)t/\hbar} \quad (1.65)$$

Or,

$$\psi(x, t) = \int \phi(k)\psi_k(x)e^{-iE(k)t/\hbar} \quad (1.66)$$

$$= \int \phi(k) \times \frac{1}{\sqrt{2\pi}} \left[\left(e^{ikx} + \frac{B}{A} e^{-ikx} \right) \theta(-x) + \psi_{k,m}(x) \theta(x) \theta(a-x) + \frac{C}{A} e^{i\kappa x} \theta(x-a) \right] e^{-i\hbar k^2 t/2m} dk. \quad (1.67)$$

Just like before, the reflected part of the wave function is

$$\psi_R(x, t) = \frac{1}{\sqrt{2\pi}} \int \phi(k) \left(\frac{B}{A} \right) e^{-ikx} \theta(-x) e^{-i\hbar k^2 t/2m} dk \quad (1.68)$$

Again, in the large t limit we can ignore the $\theta(-x)$. We can write this equation as

$$\psi_R(x, t) = \int \frac{e^{ikx}}{\sqrt{2\pi}} \phi_R(k) dk \quad (1.69)$$

if we make the same $x = -x$ change of variables. We get

$$\psi_R(x, t) = -\frac{1}{\sqrt{2\pi}} \int \phi(k) \left(\frac{B}{A} \right) e^{-i\hbar k^2 t/2m} dk. \quad (1.70)$$

We have

$$\phi_R(k) = -e^{-i\hbar k^2 t/2m} \phi(k) \left(\frac{B}{A} \right). \quad (1.71)$$

As before

$$R = \lim_{t \rightarrow \infty} \int |\phi_R(k)|^2 dk = \int |\phi(k)|^2 \left| \frac{B}{A} \right|^2 = \int P(k) R_k dk \quad (1.72)$$

We can do the same for the transmission term

$$\psi_T(x, t) = \frac{1}{\sqrt{2\pi}} \int \phi(k) \left(\frac{C}{A} \right) e^{ikx} \theta(x - a) e^{-i\hbar k^2 t/2m} dk. \quad (1.73)$$

In the large t limit, we can ignore the $\theta(x - a)$ term. If we make the change of variables from k to $\sqrt{k^2 + p^2}$, we can write $\psi_T(x, t)$ as

$$\psi_T(x, t) = \int \frac{e^{ikx}}{\sqrt{2\pi}} \phi_T(k) dk \quad (1.74)$$

where

$$\phi_T(k) = e^{-i\hbar(k^2 + p^2)t/2m} e^{ika} \phi(\sqrt{k^2 + p^2}) \left(\frac{C}{A} \right) \frac{k}{\sqrt{k^2 + p^2}}. \quad (1.75)$$

As before,

$$T = \lim_{t \rightarrow \infty} \int |\phi_T(k)|^2 dk = \int |\phi(\sqrt{k^2 + p^2})|^2 \frac{k}{\sqrt{k^2 + p^2}} \left| \frac{C}{A} \right|^2 dk \quad (1.76)$$

We can change variables back from k to $\sqrt{k^2 - p^2}$. When we do so, we get

$$T = \int |\phi(k)|^2 \frac{\kappa}{k} \left| \frac{C}{A} \right|^2 dk = \int P(k) T_k dk \quad (1.77)$$

Our derivation of equation 1.61 are perfectly general. They hold for an arbitrary potential and an arbitrary incoming wave packet.

Chapter 2

A Special Case – The Step Potential

In this section, we will discuss as a practical example the simplest example of a potential barrier—the step potential

$$V(x) = \begin{cases} 0 & x < 0 \\ V_0 & x > 0. \end{cases} \quad (2.1)$$

This can be thought of as the potential in equation 1.1 with $a = 0$. Our wave function is

$$\psi_k(x) = \frac{1}{\sqrt{2\pi}} \left[\left(e^{ikx} + \frac{B}{A} e^{-ikx} \right) \theta(-x) + \frac{C}{A} e^{ikx} \theta(x) \right]. \quad (2.2)$$

We know that ψ and ψ' must be continuous at $x = 0$. These conditions imply that

$$A + B = C \quad kA - kB = \kappa C. \quad (2.3)$$

Or,

$$\frac{B}{A} = \frac{k - \kappa}{k + \kappa} \quad \frac{C}{A} = \frac{2k}{k + \kappa}. \quad (2.4)$$

Therefore, the reflection and transmission coefficients across this step barrier are

$$R = \left(\frac{\sigma^2}{\pi} \right)^{1/2} \int e^{-(k-k_0)^2 \sigma^2} \left(\frac{k - \kappa}{k + \kappa} \right)^2 dk \quad (2.5)$$

and

$$T = \left(\frac{\sigma^2}{\pi} \right)^{1/2} \int e^{-(k-k_0)^2 \sigma^2} \frac{\kappa}{k} \left(\frac{2k}{k + \kappa} \right)^2 dk. \quad (2.6)$$

These equations have no analytic solution.¹ Nevertheless, we can approximate the solution by Taylor expanding the $(B/A)^2$ and $(C/A)^2$ factors around $k = k_0$ and doing each Gaussian integral individually. Note that both equations are of the form

$$G = \left(\frac{\sigma^2}{\pi} \right)^{1/2} \int e^{-(k-k_0)^2 \sigma^2} f(k) dk. \quad (2.7)$$

¹At least, no obvious analytic solution.

Expanding $f(k)$ gets us

$$f(k) = f(k_0) + f'(k_0)(k - k_0) + \frac{f''(k_0)}{2}(k - k_0)^2 + \dots \quad (2.8)$$

Plugging into the equation above gets

$$G = \left(\frac{\sigma^2}{\pi}\right)^{1/2} \int e^{-(k-k_0)^2\sigma^2} \left(f(k_0) + f'(k_0)(k - k_0) + \frac{f''(k_0)}{2}(k - k_0)^2 + \dots \right) dk. \quad (2.9)$$

The first term in the integral is²

$$f(k_0) \left(\frac{\sigma^2}{\pi}\right)^{1/2} \int e^{-(k-k_0)^2\sigma^2} dk = f(k_0). \quad (2.10)$$

Since $x \exp(-ax^2)$ is an odd function, we have

$$\int_{-\infty}^{\infty} x e^{-ax^2} dx = 0. \quad (2.11)$$

The second term in the integral is 0. The third term in the integral is³

$$f''(k_0) \left(\frac{\sigma^2}{\pi}\right)^{1/2} \int (k - k_0)^2 e^{-(k-k_0)^2\sigma^2} dk. \quad (2.12)$$

The third term in the integral is

$$\frac{1}{2} \frac{1}{\sigma^2} \times f''(k_0). \quad (2.13)$$

Our integral becomes

$$G = f(k_0) + \frac{1}{2} \frac{1}{\sigma^2} f''(k_0) + \dots \quad (2.14)$$

For the T term, we have

$$f(k) = \frac{\kappa}{k} \left(\frac{C}{A}\right)^2 \quad (2.15)$$

² We can calculate this by using the identity

$$\int_{-\infty}^{\infty} e^{-ax^2} dx = \sqrt{\frac{\pi}{a}}$$

and change variables to $x = (k - k_0)$.

³We calculate this by using the identity

$$\int_{-\infty}^{\infty} e^{-ax^2} x^2 dx = \frac{1}{2} \sqrt{\frac{\pi}{a^3}}$$

and doing the same change of variables as above.

We can work out the derivatives of this function⁴

$$f(k) = \left(\frac{C}{A}\right)^2 \frac{\kappa}{k} = \frac{4k\kappa}{(k+\kappa)^2} \quad (2.16)$$

$$f'(k) = \frac{(k+\kappa)^2 4(\kappa + k^2/\kappa) - 4k\kappa \times 2(k+\kappa)(1+k/\kappa)}{(k+\kappa)^4} \quad (2.17)$$

$$= \frac{4\kappa + 4k^2/\kappa - 8k}{(k+\kappa)^2} \quad (2.18)$$

$$f''(k) = \frac{(k+\kappa)^2 (4k/\kappa + 4 \times (\kappa \times 2k - k^2 \times k/\kappa)/\kappa^2 - 8)}{(k+\kappa)^4} \quad (2.19)$$

$$= \frac{-4k^3/\kappa^3 - 8k^2/\kappa^2 + 28k/\kappa - 16}{(k+\kappa)^2} \quad (2.20)$$

$$= -\left(\frac{4k}{\kappa^3} + \frac{16}{\kappa^2}\right) \left(\frac{k-\kappa}{k+\kappa}\right)^2. \quad (2.21)$$

It follows that, including the first non-vanishing correction, the transmission coefficient is

$$T = \frac{4k_0\kappa_0}{(k_0+\kappa)^2} - \left(\frac{2k_0}{\kappa_0^3} + \frac{8}{\kappa_0^2}\right) \left(\frac{k_0-\kappa_0}{k_0+\kappa_0}\right)^2 \frac{1}{\sigma^2}. \quad (2.22)$$

As was discussed in section 1.8, in the limit of large σ , the transmission coefficient reduces to the classical formula.

The R term can be worked out in the same manner

$$f(k) = \left(\frac{B}{A}\right)^2 \quad (2.23)$$

$$= \left(\frac{k-\kappa}{k+\kappa}\right)^2 \quad (2.24)$$

$$f'(k) = \frac{(k+\kappa)^2 2(k-\kappa)(1-k/\kappa) - (k-\kappa)^2 2(k+\kappa)(1+k/\kappa)}{(k+\kappa)^4} \quad (2.25)$$

$$= -\frac{4}{\kappa} \left(\frac{k-\kappa}{k+\kappa}\right)^2 \quad (2.26)$$

$$f''(k) = \frac{4}{\kappa^2} \frac{k}{\kappa} \left(\frac{k-\kappa}{k+\kappa}\right)^2 + \frac{16}{\kappa^2} \left(\frac{k-\kappa}{k+\kappa}\right)^2 \quad (2.27)$$

$$= \left(\frac{4k}{\kappa^3} + \frac{16}{\kappa^2}\right) \left(\frac{k-\kappa}{k+\kappa}\right)^2 \quad (2.28)$$

Including the first non-vanishing correction, we have

$$R = \left(\frac{k_0-\kappa_0}{k_0+\kappa_0}\right)^2 + \left(\frac{2k_0}{\kappa_0^3} + \frac{8}{\kappa_0^2}\right) \left(\frac{k_0-\kappa_0}{k_0+\kappa_0}\right)^2 \frac{1}{\sigma^2}. \quad (2.29)$$

⁴We use the fact that $d\kappa/dk = k/\kappa$.

Chapter 3

Another Example: the Finite Potential Barrier

3.1 The Eigenfunctions

We will now example the next most simple potential – the finite potential barrier. It is defined as

$$V(x) = \begin{cases} 0 & x < 0 \\ V_0 & 0 \leq x \leq a \\ 0 & x > a. \end{cases} \quad (3.1)$$

We can solve the time independent Schrödinger equation to find eigenstates of energy E :

$$\psi = \begin{cases} Ae^{ikx} + Be^{-ikx} & x < 0 \\ De^{i\kappa x} + Ee^{-i\kappa x} & 0 \leq x \leq a \\ Ce^{ikx} & x > a. \end{cases} \quad (3.2)$$

We can impose continuity of ψ and ψ' to find the relationship of the plane wave coefficients.

$$A + B = D + E \quad (3.3)$$

$$kA - kB = \kappa D - \kappa E \quad (3.4)$$

$$De^{i\kappa a} + Ee^{-i\kappa a} = Ce^{ika} \quad (3.5)$$

$$\kappa De^{i\kappa a} - \kappa Ee^{-i\kappa a} = kCe^{ika}. \quad (3.6)$$

We can combine equations 3.5 and 3.6 to get

$$D = \frac{\kappa + k}{2\kappa} e^{i(k-\kappa)a} C \quad (3.7)$$

$$E = \frac{\kappa - k}{2\kappa} e^{i(k+\kappa)a} C. \quad (3.8)$$

We can combine equations 3.3 and 3.4 to get

$$A = (k + \kappa)D + (k - \kappa)E. \quad (3.9)$$

Using equations 3.7 and 3.8, we get

$$A = \left(\frac{(k + \kappa)^2}{4k\kappa} e^{i(k-\kappa)a} - \frac{(k - \kappa)^2}{4k\kappa} e^{i(k+\kappa)a} \right) C. \quad (3.10)$$

Squaring gets us

$$|A|^2 = \left(\frac{(k + \kappa)^2}{4k\kappa} e^{-i\kappa a} - \frac{(k - \kappa)^2}{4k\kappa} e^{i\kappa a} \right) \left(\frac{(k + \kappa)^2}{4k\kappa} e^{i\kappa a} - \frac{(k - \kappa)^2}{4k\kappa} e^{-i\kappa a} \right) |C|^2 \quad (3.11)$$

$$= \frac{(k + \kappa)^4 + (k - \kappa)^4 - (k + \kappa)^2(k - \kappa)^2 2 \cos(2\kappa a)}{16k^2\kappa^2} |C|^2 \quad (3.12)$$

$$= \frac{2k^4 + 8k^2\kappa^2 + 2\kappa^4 + 4k^2\kappa^2 - (2k^4 - 4k^2\kappa^2 + 2\kappa^4)(1 - 2\sin^2(\kappa a))}{16k^2\kappa^2} |C|^2 \quad (3.13)$$

$$= \left(1 + \frac{(k^2 - \kappa^2)^2}{4^2\kappa^2} \right) |C|^2. \quad (3.14)$$

Therefore, we calculate the transmission coefficient for a plane wave with wave vector k to transmit across a rectangular barrier

$$T_k^{(\text{rect})} = \left| \frac{C}{A} \right|^2 = \frac{1}{1 + (k^2 - \kappa^2) \sin(\kappa a) / 4k^2\kappa^2}. \quad (3.15)$$

We have to be a little clear here about notation so as to avoid confusion. I will refer to the transmission coefficient for a plane wave of wave vector k for a potential step as T_k and the transmission coefficient for a plane wave of wave vector k for a rectangular barrier as $T_k^{(\text{rect})}$. The same applies for the reflection coefficients. We can rewrite $T_k^{(\text{rect})}$ in terms of R_k and T_k :

$$T_k^{(\text{rect})} = \frac{1}{1 - 4R_k \sin^2(\kappa a) / T_k^2}. \quad (3.16)$$

Using the fact that

$$T_k^{(\text{rect})} + R_k^{(\text{rect})} = 1, \quad (3.17)$$

we see that¹

$$R_k^{(\text{rect})} = \frac{4R_k \sin^2(\kappa a) / T_k^2}{1 - 4R_k \sin^2(\kappa a) / T_k^2}. \quad (3.19)$$

¹Of course, we could calculate the transmission term using the equation

$$R_k^{(\text{rect})} = \left| \frac{B}{A} \right|^2 \quad (3.18)$$

and calculate the reflection coefficient explicitly, but it is just more work and gets the same answer.

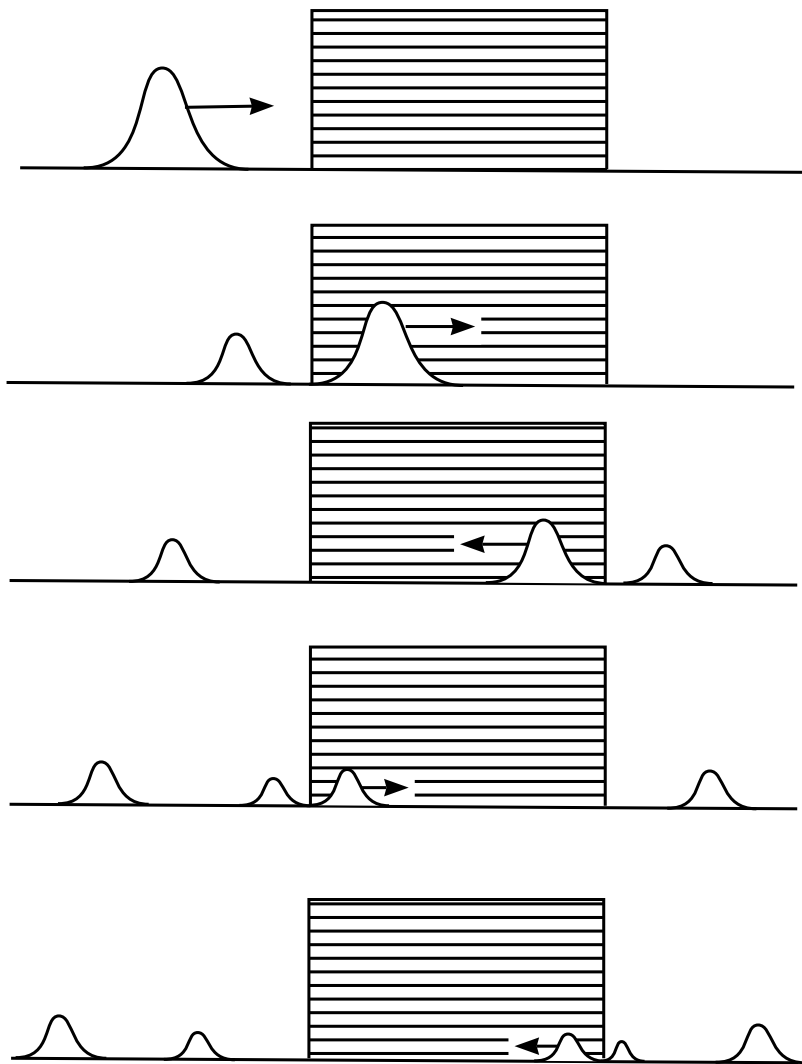


Figure 3.1: Here is a schematic diagram of the cute argument for deriving the reflection and transmission coefficients for a wave packet incident on a rectangular barrier. There is some amplitude for the wave to reflect or transmit at the left side of the barrier. This is represented by some part of the wave transmitting and some part reflecting. The same thing happens at the right side of the barrier and so on. Some part of the wave will be perpetually stuck in the barrier and successive parts of the wave will leave on either side.

3.2 A Cute Argument

To calculate the actual reflection from the rectangular barrier, we must first find the momentum space representation of the incoming wave packet. We can then write the reflection as a

$$R = \int P(k) R_k^{(\text{rect})} dk. \quad (3.20)$$

Although this integral is terribly ugly, it is the exact expression for the reflection of a wave off of a rectangular barrier.

There is a cute argument that can be used to derive this same reflection and transmission coefficients for the rectangular potential. We imagine the incoming wave as a localized packet moving to the right. Suppose that its wave vector distribution is centered on k_0 . This is shown in figure 3.1. When the wave arrives at the left of the potential, there is some amplitude for it to reflect into the barrier and some amplitude for it to transmit through

the barrier. These coefficients are, at least approximately, just the plane wave coefficients for a plane to transmit through a step potential. Next, there is some amplitude for the wave packet which transmitted the first time to reflect off the barrier on the right and then to transmit through the barrier on the left.² This would add a term $T_{k_0} R_{k_0} T_{k_0}$ to the reflection probability. Of course, there is some amplitude for the wave to transmit the first time, reflect off the barrier on the right, reflect off the barrier on the left, reflect off the barrier on the right, and to finally transmit through the barrier on the left. This will add a term $T_{k_0} R_{k_0}^3 T_{k_0}$ to the total reflection. The total reflection will be a sum of all the possible ways that the wave could be reflected:

$$R^{(\text{total})} = R_{k_0} + T_{k_0} R_{k_0} T_{k_0} + T_{k_0} R_{k_0}^3 T_{k_0} + \dots \quad (3.21)$$

$$= R_{k_0} + R_{k_0} T_{k_0}^2 (1 + R_{k_0}^2 + R_{k_0}^4 + \dots) \quad (3.22)$$

$$= 2R_{k_0} / (1 + R_{k_0}). \quad (3.23)$$

Of course, this expression will not valid generally because of many of the objects brought up earlier in this paper about using plane waves to calculate reflection and transmission coefficients. But it should be valid under certain assumptions which we can write formally. In particular, we insists that the wave packet is sufficiently close to a plane wave. This is true when the width of the wave packet in k space is sufficiently narrow:

$$\Delta k \ll k_0. \quad (3.24)$$

We also require that the wave wave packet is sufficiently narrow in comparison to the square barrier that it interacts with only one side of the barrier at any one time:

$$a \gg w \approx 1/\Delta k. \quad (3.25)$$

Formally, we expect this expression for the reflection coefficient to be valid when

$$1/a \ll \Delta k \ll k_0. \quad (3.26)$$

We can prove that the exact result for the reflection coefficient (equation 3.19) will reduce in the proper limit (equation 3.26) to equation 3.23. We have

$$R = \int P(k) \frac{4R_k \sin^2(\kappa a)/T_k^2}{1 - 4R_k \sin^2(\kappa a)/T_k^2} dk. \quad (3.27)$$

First, we note that the limit holds only when the average energy of the incoming beam k_0 is large. This is true only when $R_{k_0} \approx 0$ and $T_{k_0} \approx 1$. In this limit, the denominator is just 1 and we can ignore the other factor of T_k :

$$R = \int P(k) 4R_k \sin^2(\kappa a) dk. \quad (3.28)$$

²This argument is a bit lacking because we have not worked out the reflection and transmission coefficients for a plane wave going the other way across a potential step. It is completely trivial to work it out and it turns out that the reflection and transmission coefficients are exactly the same.

To good approximation, we can model the incoming wave packet in k space as relatively constant over some range from $k_0 - \Delta k$ to $k_0 + \Delta k$. The probability is

$$P(k) = \begin{cases} 0 & k < k_0 - \Delta k \\ 1/2\Delta k & k_0 - \Delta k \leq k \leq k_0 + \Delta k \\ 0 & k > k_0 + \Delta k. \end{cases} \quad (3.29)$$

Our integral becomes

$$R = \int_{k_0 - \Delta k}^{k_0 + \Delta k} \frac{1}{2\Delta k} 4R_k \sin^2(\kappa a) dk. \quad (3.30)$$

In the limit $\Delta k \ll k_0$, we know that R_k and T_k do not vary substantially over our integral range and can be taken outside of the integral and replaced with their value at k_0 :

$$R = 4R_{k_0} \frac{1}{2\Delta k} \int_{k_0 - \Delta k}^{k_0 + \Delta k} \sin^2(\kappa a) dk. \quad (3.31)$$

Since $a \gg 1/\Delta k$, the function $\sin(\kappa a)$ oscillates many time as k varies from $k_0 - \Delta k$ to $k_0 + \Delta k$. We may carry out the integral by multiplying the width of the integral by the average value $1/2$ of sin squared:

$$R = 4R_{k_0} \frac{1}{2\Delta k} \times 2\Delta k \frac{1}{2} = 2R_{k_0}. \quad (3.32)$$

Of course, this is not exactly the same as equation 3.23 derived using the cute argument. But since that formula is only valid in the large k_0 limit, that equation also approaches $2R_{k_0}$. The formalism introduced in this paper correctly predicts the reflection coefficient for the rectangular potential. We can also understand where the $2R_{k_0}$ comes from in an even more intuitive way. Refer back to figure 3.1. Since the R_{k_0} term is small, we see that the only appreciable terms which will contribute to the reflection of the plane wave are the first reflection R_{k_0} off the left part of the barrier and the term where the wave packet transmits, reflects, and transmits. It has a value $T_{k_0} R_{k_0} T_{k_0} \approx R_{k_0}$. Any of the higher order terms require at last three reflections and will be negligible, so the total reflection is approximately $2R_{k_0}$.

On the other hand, if we naively used the plane wave approximation for a plane wave of wave vector k , we would have thought that the reflection coefficient was

$$R_{k_0}^{(\text{rect})} = \frac{4R_{k_0} \sin^2(\kappa_0 a)/T_{k_0}^2}{1 - 4R_{k_0} \sin^2(\kappa_0 a)/T_{k_0}^2}. \quad (3.33)$$

Of course, taking the same limits as in equation 3.26, we find the denominator to again be approximately 1 and can again ignore the T_{k_0} term. But even so, we would predict the reflection probability to be

$$R_{k_0}^{(\text{rect})} \rightarrow 4R_{k_0} \sin^2(\kappa_0 a). \quad (3.34)$$

This function oscillates from 0 to $4R_{k_0}$ and is qualitatively different from the exact result derived above. We see that this the plane wave approximation is qualitatively different in this case from the exact result.

Part III

A Theoretical Discussion of Crystal Diffraction and an Experimental Investigation of Microwave Diffraction

Abstract

We will describe the theory of crystal diffraction. We will then describe the theory of powder diffraction. The theoretical discussion in this paper and several of the figures closely follows chapter 6 of [8]. We will then describe an experimental investigation into crystal diffraction that was done using the Pasco microwave optics kit.

Chapter 4

Crystal Diffraction

4.1 The Bravais Lattice

Crystals are very regular structures. A Bravais lattice is a mathematical device used to describe the regularity and self similarity of a crystal. A Bravais lattice is an array of lattice points. Each lattice point's position is of the form

$$\mathbf{R} = n_1 \mathbf{a}_1 + n_2 \mathbf{a}_2 + n_3 \mathbf{a}_3. \quad (4.1)$$

Here, \mathbf{a}_1 , \mathbf{a}_2 , and \mathbf{a}_3 are three linearly independent basis vectors and n_1, n_2, n_3 can be any possible integer.

The simplest Bravais lattice is the cubic lattice. It represents a cubic crystal structure. A lattice cell of this structure is shown in figure 4.1. The lattice vectors are defined as:

$$\mathbf{a}_1 = a\hat{\mathbf{x}}, \quad \mathbf{a}_2 = a\hat{\mathbf{y}}, \quad \mathbf{a}_3 = a\hat{\mathbf{z}}. \quad (4.2)$$

There are two other very common Bravais lattices. One is the face-centered cubic lattice. It is a cubic lattice where each side of the cubes have a lattice point in the middle. A

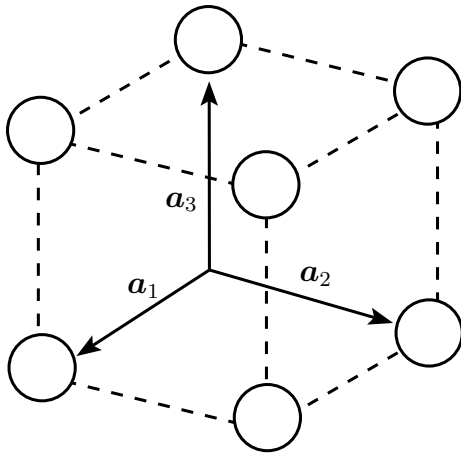


Figure 4.1: The cubic lattice. This is the simplest Bravais lattice.

face-centered cubic lattice is shown in figure 4.2a. We can pick as a set of Bravais lattice vectors:

$$\mathbf{a}_1 = \frac{a}{2}(\hat{\mathbf{y}} + \hat{\mathbf{z}}), \quad \mathbf{a}_2 = \frac{a}{2}(\hat{\mathbf{z}} + \hat{\mathbf{x}}), \quad \mathbf{a}_3 = \frac{a}{2}(\hat{\mathbf{x}} + \hat{\mathbf{y}}). \quad (4.3)$$

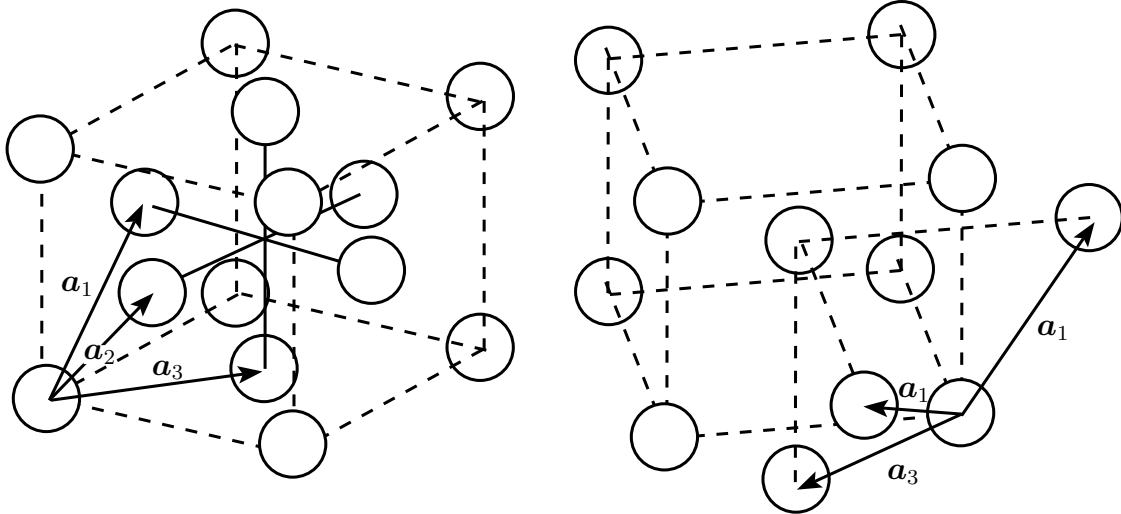
These are labeled in the figure. The other common structure is the body-centered cubic. It can be thought of as a cubic lattice where each cube has another lattice point in the middle of it. Part of this crystal is shown in figure 4.2b. We could pick as our set of Bravais lattice vectors

$$\mathbf{a}_1 = a\hat{\mathbf{x}} \quad \mathbf{a}_2 = a\hat{\mathbf{y}} \quad \mathbf{a}_3 = \frac{a}{2}(\hat{\mathbf{x}} + \hat{\mathbf{y}} + \hat{\mathbf{z}}). \quad (4.4)$$

But there is a more useful set of Bravais lattice vectors

$$\mathbf{a}_1 = \frac{a}{2}(\hat{\mathbf{y}} + \hat{\mathbf{z}} - \hat{\mathbf{x}}), \quad \mathbf{a}_2 = \frac{a}{2}(\hat{\mathbf{z}} + \hat{\mathbf{x}} - \hat{\mathbf{y}}), \quad \mathbf{a}_3 = \frac{a}{2}(\hat{\mathbf{x}} + \hat{\mathbf{y}} - \hat{\mathbf{z}}). \quad (4.5)$$

These vectors are shown in the figure.



(a) A cubic cell of the face-centered cubic lattice. (b) A cubic cell of the body-centered cubic lattice.

4.2 The Reciprocal Lattice

We will introduce the reciprocal lattice. It will be an important tool in discussing crystal diffraction. The reciprocal lattice for a Bravais lattice is as all wave vectors \mathbf{Q} that have the periodicity of a Bravais lattice. Mathematically, this means that

$$e^{i\mathbf{Q} \cdot (\mathbf{r} + \mathbf{R})} = e^{i\mathbf{Q} \cdot \mathbf{r}} \quad (4.6)$$

for all \mathbf{R} in the Bravais lattice. This condition is equivalent to

$$e^{i\mathbf{Q}\cdot\mathbf{R}} = 1. \quad (4.7)$$

The reciprocal lattice is also a Bravais lattice so a reciprocal lattice is generated by reciprocal lattice vectors. We can construct reciprocal lattice vectors from Bravais lattice vectors as follows

$$\mathbf{b}_1 = 2\pi \frac{\mathbf{a}_2 \times \mathbf{a}_3}{\mathbf{a}_1 \cdot (\mathbf{a}_2 \times \mathbf{a}_3)} \quad \mathbf{b}_2 = 2\pi \frac{\mathbf{a}_3 \times \mathbf{a}_1}{\mathbf{a}_2 \cdot (\mathbf{a}_3 \times \mathbf{a}_1)} \quad \mathbf{b}_3 = 2\pi \frac{\mathbf{a}_1 \times \mathbf{a}_2}{\mathbf{a}_3 \cdot (\mathbf{a}_1 \times \mathbf{a}_2)} \quad (4.8)$$

Then, any vector in the reciprocal lattice vector \mathbf{Q} can be written as

$$\mathbf{Q} = q_1 \mathbf{b}_1 + q_2 \mathbf{b}_2 + q_3 \mathbf{b}_3. \quad (4.9)$$

We can prove that this vector satisfies equation 4.7 as follows. Since $\mathbf{a}_2 \times \mathbf{a}_3$ is normal to \mathbf{a}_2 and \mathbf{a}_3 , it follows that $\mathbf{a}_2 \cdot \mathbf{b}_1 = \mathbf{a}_3 \cdot \mathbf{b}_1 = 0$. Furthermore,

$$\mathbf{a}_1 \cdot \mathbf{b}_1 = \mathbf{a}_1 \cdot \left[2\pi \frac{\mathbf{a}_2 \times \mathbf{a}_3}{\mathbf{a}_1 \cdot (\mathbf{a}_2 \times \mathbf{a}_3)} \right] = 2\pi \quad (4.10)$$

It follows that¹

$$\mathbf{b}_i \cdot \mathbf{a}_j = 2\pi \delta_{ij}. \quad (4.12)$$

For any Bravais lattice vector of the form $\mathbf{R} = n_1 \mathbf{a}_1 + n_2 \mathbf{a}_2 + n_3 \mathbf{a}_3$ we have

$$\mathbf{Q} \cdot \mathbf{R} = 2\pi q_1 n_1 + q_2 n_2 + q_3 n_3 \quad (4.13)$$

from which it follows that equation 4.7 is only satisfied if and only if q_1 , q_2 , and q_3 are all integers. From this, we see that \mathbf{b}_1 , \mathbf{b}_2 , and \mathbf{b}_3 are reciprocal lattice vectors.

As an example, we can determine the reciprocal lattice vectors for the body-centered cubic lattice from equation 4.5. The denominator of equation 4.8 is

$$\mathbf{a}_1 \cdot (\mathbf{a}_2 \times \mathbf{a}_3) = \frac{a}{2}(\hat{\mathbf{y}} + \hat{\mathbf{z}} - \hat{\mathbf{x}}) \cdot \left[\frac{a}{2}(\hat{\mathbf{z}} + \hat{\mathbf{x}} - \hat{\mathbf{y}}) \times \frac{a}{2}(\hat{\mathbf{x}} + \hat{\mathbf{y}} - \hat{\mathbf{z}}) \right] = a^3/2 \quad (4.14)$$

¹ Here, we use the identity

$$\mathbf{a}_1 \cdot (\mathbf{a}_2 \times \mathbf{a}_3) = \mathbf{a}_2 \cdot (\mathbf{a}_3 \times \mathbf{a}_1) = \mathbf{a}_3 \cdot (\mathbf{a}_1 \times \mathbf{a}_2). \quad (4.11)$$

We calculate each of the terms

$$\mathbf{b}_1 = 2\pi \frac{\mathbf{a}_2 \times \mathbf{a}_3}{\mathbf{a}_1 \cdot (\mathbf{a}_2 \times \mathbf{a}_3)} \quad (4.15)$$

$$= 2\pi \frac{\frac{a}{2}(\hat{\mathbf{z}} + \hat{\mathbf{x}} - \hat{\mathbf{y}}) \times \frac{a}{2}(\hat{\mathbf{x}} + \hat{\mathbf{y}} - \hat{\mathbf{z}})}{a^3/2} \quad (4.16)$$

$$= \frac{2\pi}{a}(\hat{\mathbf{y}} + \hat{\mathbf{z}}) \quad (4.17)$$

$$\mathbf{b}_2 = 2\pi \frac{\mathbf{a}_3 \times \mathbf{a}_1}{\mathbf{a}_1 \cdot (\mathbf{a}_2 \times \mathbf{a}_3)} \quad (4.18)$$

$$= 2\pi \frac{\frac{a}{2}(\hat{\mathbf{x}} + \hat{\mathbf{y}} - \hat{\mathbf{z}}) \times \frac{a}{2}(\hat{\mathbf{y}} + \hat{\mathbf{z}} - \hat{\mathbf{x}})}{a^3/2} \quad (4.19)$$

$$= \frac{2\pi}{a}(\hat{\mathbf{z}} + \hat{\mathbf{x}}) \quad (4.20)$$

$$\mathbf{b}_3 = 2\pi \frac{\mathbf{a}_1 \times \mathbf{a}_2}{\mathbf{a}_1 \cdot (\mathbf{a}_2 \times \mathbf{a}_3)} \quad (4.21)$$

$$= 2\pi \frac{\frac{a}{2}(\hat{\mathbf{y}} + \hat{\mathbf{z}} - \hat{\mathbf{x}}) \times \frac{a}{2}(\hat{\mathbf{z}} + \hat{\mathbf{x}} - \hat{\mathbf{y}})}{a^3/2} \quad (4.22)$$

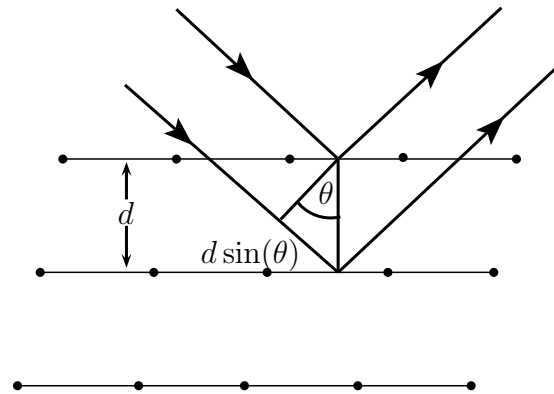
$$= \frac{2\pi}{a}(\hat{\mathbf{x}} + \hat{\mathbf{y}}). \quad (4.23)$$

$$(4.24)$$

The reciprocal lattice for a body-centered cubic crystal is a face-centered cubic lattice. It is also true that the reciprocal lattice for a face-centered cubic is a body-centered cubic. The easiest way to see this is to note from the symmetry of equation 4.7 that the reciprocal lattice of a reciprocal lattice is the original Bravais lattice.

4.3 Crystal Diffraction

Figure 4.2: The Bragg condition for constructive interference and therefore intensity maxima. We divide the crystal into Bragg planes. Bragg said that there would be constructive interference where the light reflected off the Bragg planes was in phase.



Crystal diffraction is an important way to learn about the internal structure of matter. Bragg proposed a rather unphysical model to explain wave diffraction off crystals. He said that we can divide a crystal into Bragg planes. This is shown in figure 4.2. We can think of

the incoming waves as scatter off of the Bragg planes. There will be constructive interference only when the light leaving the Bragg planes is in phase. The condition for constructive interference is known as Bragg's Law:

$$n\lambda = 2d \sin \theta. \quad (4.25)$$

Here, d is the distance between Bragg planes and θ is the incident angle of the waves. This condition holds for any integral n . There are many ways of dividing up a crystal into Bragg planes and each one can lead to diffraction peaks.

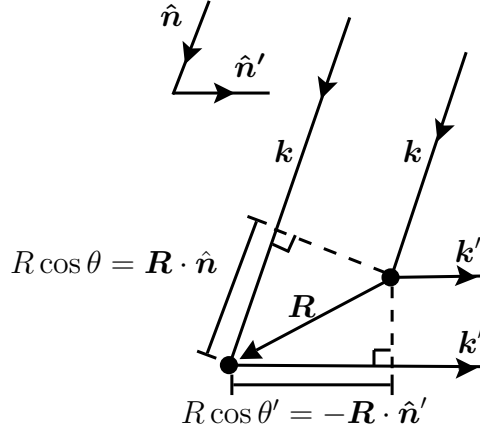


Figure 4.3: This figure shows two atoms separated by a Bravais lattice vector R . This diagram shows the path difference taken by two beams of light that diffract through the crystal.

The Von Laue approach to studying diffraction is more physical but leads to the same conclusion. We model crystal scattering by having waves scatter off of each atom in the crystal separately. The condition for constructive interference is that the path difference of the waves for all the atoms is only different by integers times the wavelength. Constructive interference for a particular pair of atoms separated by a Bravais lattice vector R is shown in figure 4.3. Two beams of light enter from the top, are scattered by the atoms, and leave to the right. Our beam is monochromatic so the incoming waves have the same momentum \mathbf{k} . We assume that the diffraction is elastic so the waves all leave with the same wave vector ($|\mathbf{k}'| = |\mathbf{k}|$).

The path difference is

$$R \cos \theta + R \cos \theta' = \mathbf{R} \cdot (\hat{\mathbf{n}} - \hat{\mathbf{n}}'). \quad (4.26)$$

The Von Laue condition for constructive interference is that

$$\mathbf{R} \cdot (\hat{\mathbf{n}} - \hat{\mathbf{n}}') = m\lambda \quad (4.27)$$

for some integer m . If we multiply by $2\pi/\lambda$, we get

$$\mathbf{R} \cdot (\hat{\mathbf{k}} - \hat{\mathbf{k}}') = 2\pi m. \quad (4.28)$$

Or,

$$e^{i(\mathbf{k}' - \mathbf{k}) \cdot \mathbf{R}} = 1. \quad (4.29)$$

This condition must hold for all Bravais lattice vectors R , $\mathbf{Q} = \mathbf{k}' - \mathbf{k}$ must be an element of the reciprocal lattice.

We can prove that this condition is equivalent to Bragg's law. Since our beam is monochromatic, $|\mathbf{k}| = |\mathbf{k}'|$. From this, it follows that $k = |\mathbf{k} - \mathbf{Q}|$. Squaring both sides of the equation

$$k^2 = |\mathbf{k} - \mathbf{Q}|^2 \quad (4.30)$$

$$k^2 = k^2 - 2\mathbf{k} \cdot \mathbf{Q} + Q^2 \quad (4.31)$$

$$\mathbf{k} \cdot \mathbf{Q} = \frac{1}{2}Q^2 \quad (4.32)$$

$$\mathbf{k} \cdot \hat{\mathbf{Q}} = \frac{1}{2}Q. \quad (4.33)$$

the component of \mathbf{k} parallel to \mathbf{Q} is exactly half way along the reciprocal lattice vector \mathbf{Q} . \mathbf{k} must lie on a plane which is the perpendicular bisect of \mathbf{Q} . This is a Bragg plane. This is shown in figure 4.4.

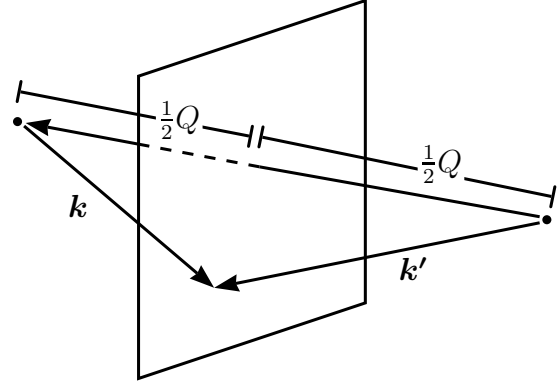


Figure 4.4: This diagram shows $\mathbf{Q} = \mathbf{k}' - \mathbf{k}$ when $k = k'$.

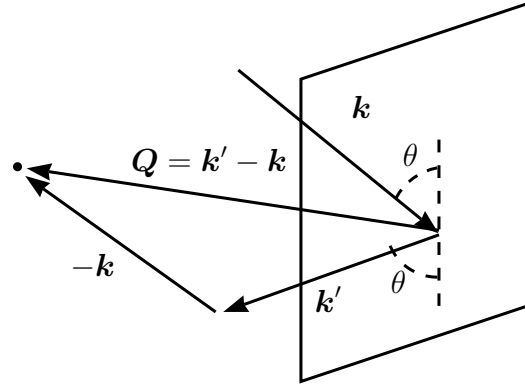


Figure 4.5: This figure shows figure 4.4. with \mathbf{k}' moved onto the Bragg plane and the head of $-\mathbf{k}$ moved onto the head of \mathbf{k}' . This diagram shows that we can think of diffraction as being reflected off some plane in the crystal. We see from this figure that the incident and reflected angles are equal.

Now, redraw figure 4.4 by moving the reciprocal lattice vector \mathbf{k}' onto the Bragg plane and moving \mathbf{k} onto its head. This is shown in figure 4.5. Here, the angle between \mathbf{k} and the Bragg plane and \mathbf{k}' and the Bragg plane must both be θ . Furthermore, for Bragg planes a distance d apart, the reciprocal lattice vectors parallel to them all have distances of the form

$Q = 2\pi n/d$ with n an integer. From the diagram, we see that $Q = 2k \sin(\theta)$. Using this, we have

$$k \sin \theta = \pi n/d. \quad (4.34)$$

Since $k = 2\pi/\lambda$, we derive Bragg's law.

4.4 Visualization of Diffraction

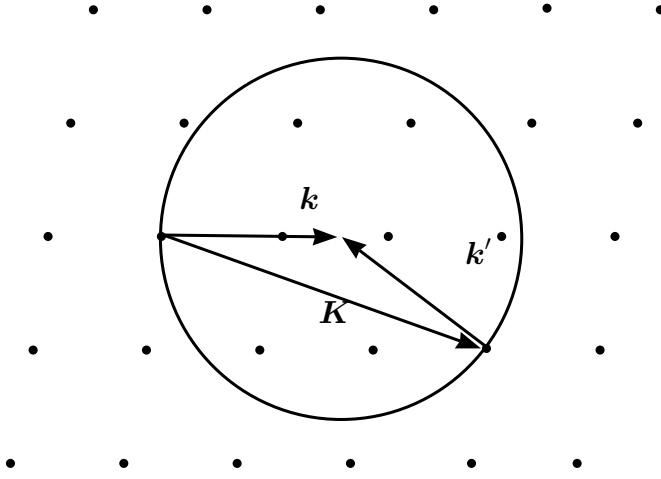


Figure 4.6: The Ewald sphere construction. We draw the incoming wave vector \mathbf{k} starting at some reciprocal lattice point. We draw a circle of radius k centered at the head of \mathbf{k} (really, it should be a sphere). If the Ewald sphere intersects some other reciprocal lattice point, we can draw a new vector \mathbf{k}' from that point to the center of the circle such that $\mathbf{k} - \mathbf{k}'$ is in the reciprocal lattice. This is the condition for light to preferentially scatter. So, whenever the Ewald sphere intersects a reciprocal lattice point, we will have constructive interference.

We introduce a new construction called the Ewald sphere to help think about diffraction. Figure 4.6 show a diagram of an Ewald sphere. We begin by placing the tail of the incoming wave vector on a reciprocal lattice vector. We then draw a sphere of radius $|\mathbf{k}|$ centered on the head of \mathbf{k} . This is called the Ewald sphere. Whenever another reciprocal lattice vector intersects the Ewald sphere, we can draw a reflected vector \mathbf{k}' that begins at the other reciprocal lattice vector and ending at the head of \mathbf{k} such that $\mathbf{k}' - \mathbf{k}$ is in the reciprocal lattice. This is the condition for constructive interference. So only when a reciprocal lattice point intersects that Bragg plane will there be constructive interference, and we can use the Ewald construction to determine the angle of scattering.

Chapter 5

Powder Diffraction

Although a presentation of powder diffraction will not be needed for the following discussion of microwave optics, it will be discussed here because it provides a background for the powder diffraction software part of my plan.

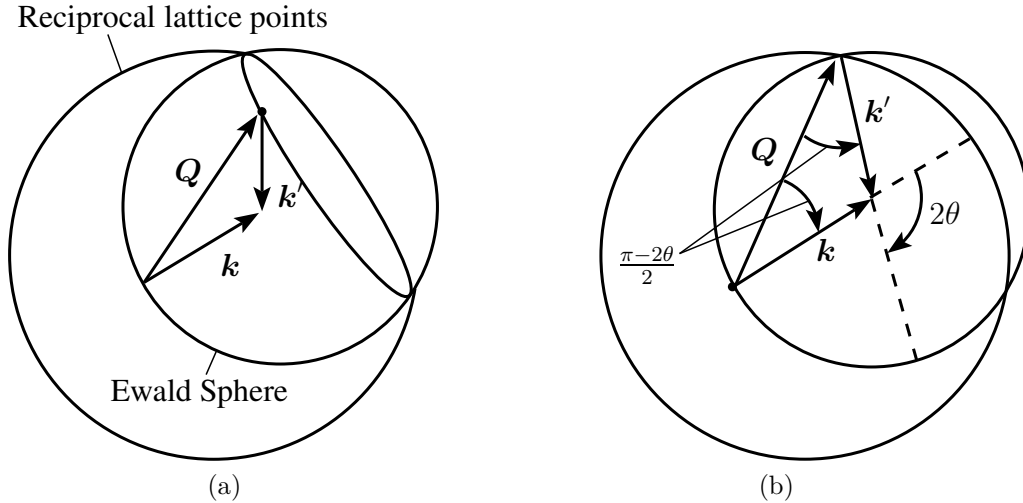


Figure 5.1: These figures show the Ewald sphere for powder diffraction. Because powder diffraction is diffraction off of many small crystals with different orientations, there can be constructive interferences for reciprocal lattice vectors rotate at any angle. In these figures, we draw an Ewald sphere and then one particular reciprocal lattice vector \mathbf{Q} which is rotated through all possible angles. The intersection of these two spheres leads to constructive interference (since $\mathbf{Q} = \mathbf{k} - \mathbf{k}'$). Therefore, for each reciprocal lattice vector there will be associated scattering in a cone. 2θ is the angle between \mathbf{k} and \mathbf{k}' .

Our previous discussion of crystal diffraction assumed that the crystal that we are imaging represents a Bravais lattice and is self similar over very large distances. But powder diffraction is different because what is imaged is a crystalline powder, where there are many pieces of crystal which are large on a microscopic scale but small on a macroscopic scale.

Powder diffraction is achieved experimentally by grinding a crystal with a mortar and pestle until it is very fine.

When we perform diffraction on a powder, we effectively scatter some light off of crystals with every possible orientation. This corresponds to crystals with reciprocal lattice vectors that are rotated in all possible directions.

We can draw an Ewald sphere to analyze this situation. The reason why our Ewald sphere is different is because each reciprocal lattice vector will generate a sphere (with different points on the sphere corresponding to different crystals in the powder). The Ewald sphere will intersect the sphere of one of the reciprocal lattice vectors in a circle¹ Figure 5.1 shows a figure of these two spheres.

This circle of intersection will correspond to a cone of light emanating from the scattering powder. The scattering angle 2θ can be calculated from figure 5.1b as follows. Since we have drawn an equilateral triangle, it must be that

$$\cos\left(\frac{\pi - 2\theta}{2}\right) = \frac{Q/2}{k} \quad (5.1)$$

Or,

$$Q = 2k \sin(2\theta/2). \quad (5.2)$$

Since $k = 2\pi/\lambda$, we have

$$Q = \frac{4\pi \sin(2\theta/2)}{\lambda}. \quad (5.3)$$

Therefore, if we know the magnitudes of the reciprocal lattice vectors for a particular crystal, we can use equation 5.3 to calculate the scattering angles that would be found when performing powder diffraction. Alternately, we could measure the scattering angles due to powder diffraction and use those to calculate the magnitude of the reciprocal lattice vectors of the crystal. These values often be used to reconstruct the structure of the reciprocal lattice and subsequently the actual Bravais lattice. Powder diffraction therefore provides an experimental technique to determine the structure of crystals.

The primary purpose of the diffraction software that was written as part of this plan is to infer the Q values from powder data by measuring the scattering angle.² The reason why the software is so complicated is because what is directly measured experimentally is an area diffraction pattern. The cones of light intersect the area detector in conic sections and it is difficult in practice to work from the area data to a list of Q values. We will explain how we can use the list of Q values to determine the crystalline structure of some powder sample. For example, below is a a list of the crystal Lanthanum Hexaboride's Q values

¹This only happens for reciprocal lattice vectors smaller than $2k$.

²Actually, all the program really does is produce a plot of intensity as a function of Q and another program must be used to calculate the actual Q values based on the peaks of the intensity plot.

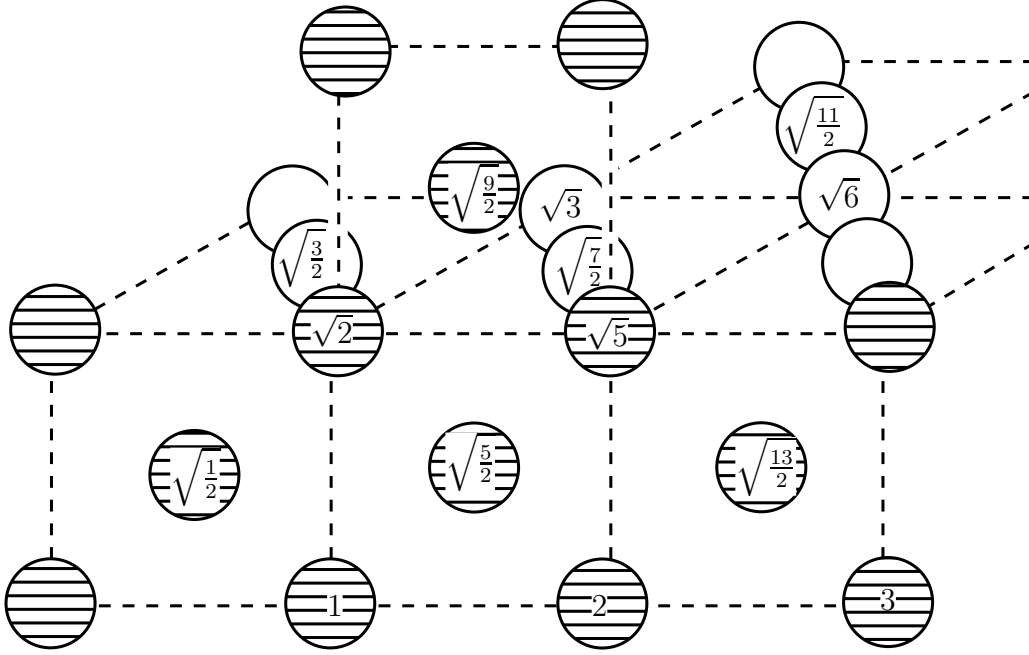


Figure 5.2: A picture of of a face-centered cubic Bravais lattice. On top of many of the lattice points is the distance to the lattice point from the bottom left lattice point if the space of a lattice cube is 1.

1.511543809	$= 0.707 \times 2.137646823 \approx$	$\sqrt{1/2} \times 2.137646823$
2.137646823	$= 1.000 \times 2.137646823 =$	1×2.137646823
2.618102966	$= 1.225 \times 2.137646823 \approx$	$\sqrt{3/2} \times 2.137646823$
3.023087619	$= 1.414 \times 2.137646823 \approx$	$\sqrt{2} \times 2.137646823$
3.379873753	$= 1.581 \times 2.137646823 \approx$	$\sqrt{5/2} \times 2.137646823$
3.702525225	$= 1.732 \times 2.137646823 \approx$	$\sqrt{3} \times 2.137646823$
4.275148198	$= 2.000 \times 2.137646823 \approx$	2×2.137646823
4.534631428	$= 2.121 \times 2.137646823 \approx$	$\sqrt{9/2} \times 2.137646823$
4.77990514	$= 2.236 \times 2.137646823 \approx$	$\sqrt{5} \times 2.137646823$
5.013313099	$= 2.345 \times 2.137646823 \approx$	$\sqrt{11/2} \times 2.137646823$
5.23603139	$= 2.449 \times 2.137646823 \approx$	$\sqrt{6} \times 2.137646823$
5.44989618	$= 2.549 \times 2.137646823 \approx$	$\sqrt{13/2} \times 2.137646823$

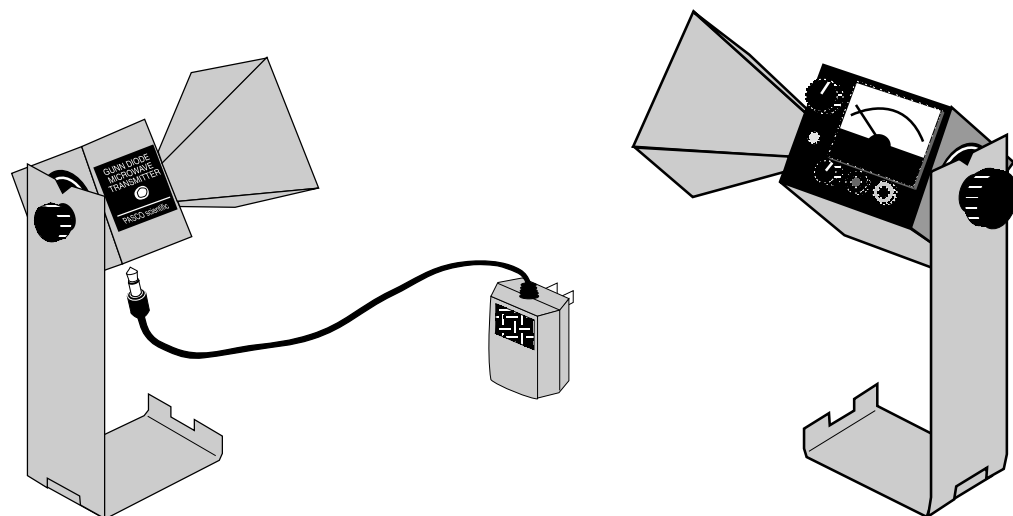
These Q values are given in units of inverse angstrom. If we examine the ratio of the Q values, we see from figure 4.2a that it is the same as the ratio of the lengths of the face centered cubic Bravais lattice.³ It must be that the reciprocal lattice is face-centered

³Actually, I suspect that this list of Q values did not come from real data because the numbers come out a little too perfect. I was given these Q values to use during diffraction image calibration. These values are used to measure parameters of the experimental setup when Lanthanum Hexaboride is imaged. These values are probably calculated based on our best guess at what the lattice spacing is because that is what the data should be calibrated off of. Presumably, experimentally measured Q values are not quite so nice.

cubic with a cubic cell of width $2.137646823/\text{\AA}$. This means that the Bravais lattice is body-centered cubic with a cubic cell of width 2.93\AA .

Chapter 6

Microwave Optics



(a) The Pasco microwave transmitter. This picture is from [10].

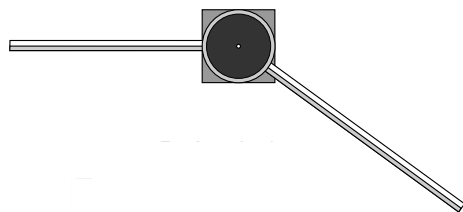
(b) The Pasco microwave receiver. This picture is from [10].

Figure 6.1:

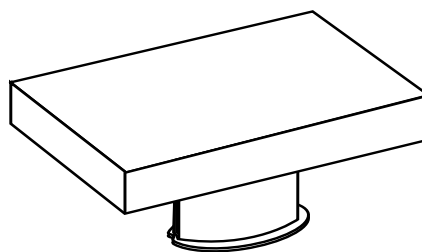
Because of Bragg's law, to measure a crystal with spacing of order d , one must use light whose wavelength is of similar size. This is so the scattering angle θ is neither too large nor too small. Typically, x-ray diffraction (wavelength of order 1 angstrom) is used to study solids whose crystalline structure typically is of order 1 angstrom.

Pasco Scientific manufactures an experimental kit that can be used to study diffraction at a much different scale. It creates electromagnetic waves in the microwave spectrum. Pasco claims that their transmitter produces microwaves of wavelength 2.85cm. Because of this, the crystal that should be diffracted should have a characteristic spacing of centimeters, which is easily visible and constructable.

To realize this experimentally, the Pasco microwave diffraction kit comes with a transmitter, a receiver, a goniometer, and a rotating table. Figures of the transmitter and receiver



(a) The goniometer. The circular middle allows for the measurement of the angle between the arms. The transmitter and the receiver can slide directly onto the two arms. This picture is from [10].



(b) The rotating table. This table fits on top of the circular middle of the goniometer. A crystalline structure can be placed on top of it. This picture is from [10].

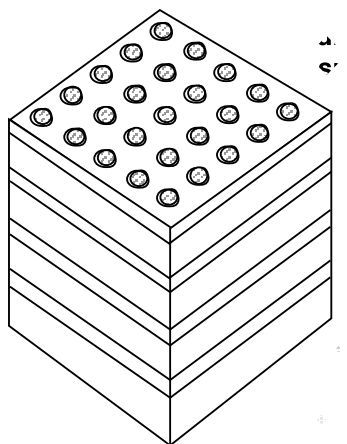
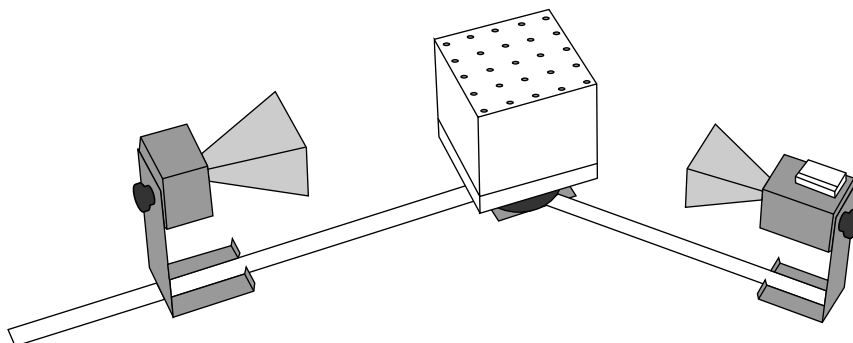


Figure 6.2: The cubic lattice that came with the microwave diffraction kit. The cubic lattice contains 100 metal spheres. It is a $5 \times 5 \times 4$ array. Picture from [10].

Figure 6.3: The equipment setup. Here, the receiver and transmitter are attached to the arm of the goniometer and the rotating table is placed on the middle of it. The crystalline structure is placed on top of the table. Picture from [10].



are shown in figure 6.1. Figures of the goniometer and rotating table are shown in figure 6. The goniometer acts as a base which attaches the transmitter to the receiver. The middle of the goniometer holds the rotating table and on top of that rests the crystalline structure which will be images. A figure of the crystalline structure which the Pasco microwave optics kit comes with is shown in figure 6.2. It is a $5 \times 5 \times 4$ cubic structure. A figure of the total assembly is shown in figure 6.3.

Using this experimental setup, we can pick a particular Bragg plane and measure the intensity of the scattered microwaves as a function of angle θ , just as in figure 4.2.

The particular experiment that done using the microwave optics kit involved measuring the diffraction off of the (100) plane shown in figure 6.4.

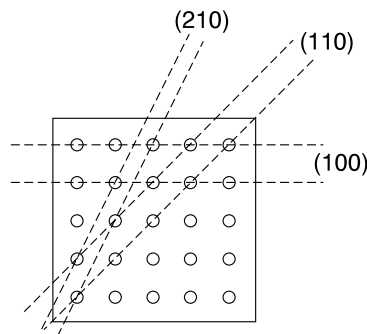


Figure 6.4: Several possible Bragg planes for a cubic crystal. Picture from [10].

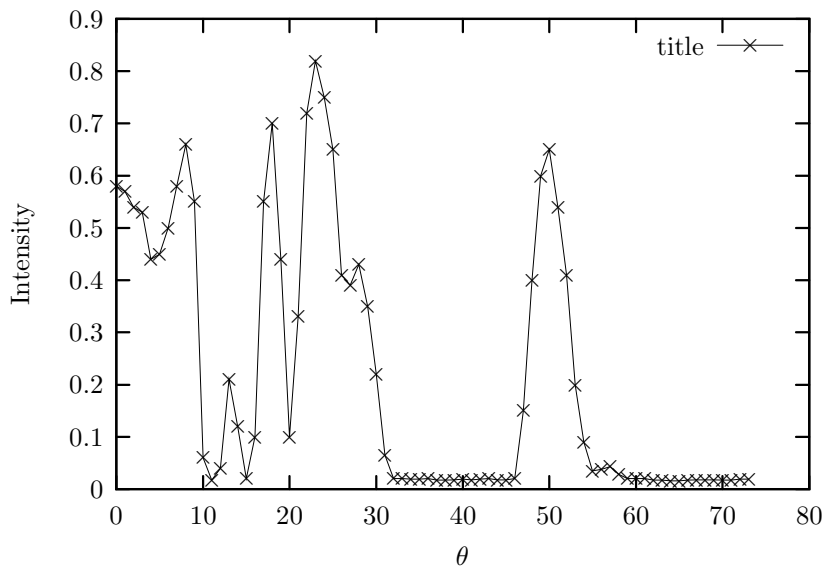


Figure 6.5: The intensity as a function of scattering angle off of the (100) plane shown in figure 6.4.

Figure 6.5 shows a plot of the diffraction data that was collected using the experiment. There is reason to believe that all of the data for θ below about 15 degrees should not be believed because the intensity that is recorded is coming from microwaves which don't diffract at all but instead go directly through the crystal. For θ above about 15 degrees, we see that there are three diffraction peaks at 18° , 23° , and 50° .

According to Pasco, the characteristic atomic spacing of this crystal is 3.8cm and the wavelength microwaves are 2.85cm. From this, it follows that we should see diffraction peaks at 22° and 49° . The diffraction peak at 18° evidently came from diffraction off of a different crystal. The other peaks are in good agreement with the theory.

Part IV

Area Diffraction Machine Manual

Chapter 7

Tips and Tricks

7.1 Calibration

The “Calibration” tab can be used to load diffraction data into the program. The tab can be used to calibrate diffraction data to determine the experimental parameters that characterize the experiment. The data can be loaded using the “Data File:” input. The program recognizes “mar2300”, “mar3450”, “mccd”, and “tiff”, and “edf” data. Multiple files can be loaded into the program by selecting multiple files with the file selector. The sum image will be used.

This program characterizes a diffraction experiment according to the parameters:

- “xc:”, “yc:” - the x and y coordinates on the detector where the incoming x-ray beam would have hit the detector were there no sample in the way (in pixels).
- “d:” - the distance from the sample to detector (in mm).
- “E:” - the energy of the incoming beam (in eV).
- “alpha:”, “beta:” 2 tilt parameters of the detector (in degrees).
- “R:” - the rotation of the detector around the center (in degrees).
- “pl:” - the pixel length of the image. The width of one pixel (in microns).
- “ph:” - The pixel height of the image. The height of one pixel (in microns).

Before calibrating an image, three things must be done. First, the calibration data must be loaded. Second, a Q data file with the standard Q values for the sample must be loaded. Third, an initial guess of the calibration parameters must be loaded. This can be done with the “Parameters” inputs. A decent guess at the calibration parameters can sometimes be found in the header of a diffraction file. These values can be loaded into the program using the “Get From Header” button to. The “Do Fit” button will perform the calibration and find a best guess at the real experimental parameters.

The “**Work in Lambda**” selection in the “**Calibration**” menu can be used to switch the program to work with the x-ray’s wavelength instead of its energy. The relationship between these values is $E = hc/\lambda$. The calibration parameter “**E:**” will be replaced with “**λ:**” and the current value will be converted.

The “**Q Data:**” input can be used to load in standard Q data files. This program stores several standard Q files. The can be selected through the “**Standard Q**” menu in the “**Calibration**” menu.

The calibration fit can be modified in a couple of ways. The calibration algorithm will look the diffraction data to find diffraction peak. It does so by running from the center of the image out. The number of peaks that the program tries to find can be set with the “**Number of Chi?**” input. This tells the program how many of these radial slices from the center of the image should be done. The “**Stddev?**” input tells the program what ratio higher the peak must be then the standard deviation of the background near the peak in order for the peak to be considered real. The higher the value, the more picky the program is about finding legitimate peaks.

If some of the experimental parameters are known exactly, pushing the “**Fixed?**” check box will fix the associated variable so that it will be not refined when fitting. The pixel length and pixel height can never be refined so this does not apply for them.

To see how good the current calibration parameters are at characterizing the loaded data, the “**Draw Q Lines?**” check box can be used to make the program draw on the diffraction image lines of constant Q specified by the Q data file. The ΔQ ranges specified in the Q file can also be drawn using the “**Draw dQ Lines?**” check box. The “**Draw Peaks?**” check box can be used to display on top of the diffraction image all of the peaks that were found while doing the fit.

The diffraction image can be zoomed into by left clicking in the image, dragging the mouse, and then releasing. The image can be zoomed out of by right clicking on the image. The image can be panned across by shift clicking on the image and dragging. The image can be made bigger or smaller by resizing the window.

In the file menu, the “**Save Image**” option can be used to save the current diffraction file in several popular image formats. The image will be saved with the current zoom level and any Q lines, ΔQ lines, peaks, or masks drawn on top of it.

7.2 Masking

The program can ignore certain pixels in an image when performing diffraction analysis. This is done on the “**Masking**” tab. Threshold masking can be used to ignore pixels above or below a certain value.

All pixels larger than a certain value can be ignored by checking the “**Do Greater Than Mask?**” check box and specifying the value in the “**(Pixels Can’t Be) Greater Than Mask:**” input. All pixels less than a certain value can be ignored by checking the “**Do Less Than Mask?**” check box and specifying the value in the “**(Pixels Can’t Be) Less Than Mask:**” input. The overloaded or underloaded pixels will show up as a different color on the

diffraction and cake displays. That color can be specified by the color inputs next to the check boxes. When a threshold mask is applied, masked pixels will not be used during an intensity integration.

The program can mask certain areas of the diffraction image using polygon masks. The “Do Polygon Mask?” check box will enable polygon masking. Any masks in the program will be displayed over the diffraction data and cake data. Any masked pixels will not be used during an intensity integration. The “Add Polygon” button can be used to draw new polygon masks. To draw a mask, simply push the button, then left click all the nodes on the diffraction image except the last one, and finally right click the final node. This will create the polygon. The “Remove Polygon” button can be used to remove polygons from the diffraction image. Simply push the button, then click on the polygon that should be removed. The “Clear Mask” button will remove all the polygons from the program. The “Save Mask” button will save all the polygons in the program to a file. The “Load Mask” button will load into the program all of the polygons in a file.

7.3 Caking

A caked image is a plot of diffraction data in Q vs χ space. χ is a measure of the angle around the incoming x-ray beam. By convention, χ is equal to 0 degrees to the right of the center of the image. It increases in a counterclockwise direction. The program needs to know a range and bin size in Q and χ in order to make a caked plot. The “Do Cake” button will create a caked plot of the data. The program will present a new window with the caked data in it. The caked window can be interacted with just like the diffraction window. Any Q lines, ΔQ lines, and peaks that are drawn on top of the diffraction image will also be displayed on top of the cake image. The Q and ΔQ lines are just vertical lines on the caked image. The “Save Data” button will save a caked plot as plain text. The “Save Image” button will save the caked plot as a popular image format. The image will have any Q lines or peaks saved drawn on the caked plot saved on top of it.

The “Do Polarization Correction?” button will apply a polarization correction to the caked plot. The polarization of the incoming beam can be specified with the “P?” input. The formula for calculating the polarization correction is

$$I = Im/PF \quad (7.1)$$

$$PF = P(1 - (\sin(2\theta) \sin(\chi - 90))^2) + (1 - P)(1 - (\sin(2\theta) \cos(\chi - 90))^2) \quad (7.2)$$

with Im the measured intensity.

There is a convenient button called “AutoCake” which automatically picks the smallest cake range so that the whole image shows up in the cake. It then picks the bin size so that each pixel displayed on the screen is a single bin. It then caked the data. This button can be used to quickly make a good cake.

7.4 Integrate

An intensity integration is a plot of average intensity vs Q , χ , or 2θ . By default, the option is to integrate in Q or χ . The “**Work in 2theta**” select in the file menu can be used make the program integrate in 2θ instead of Q .

The program needs to know a range (both a lower and upper value) and a bin size in order to perform an intensity integration. When these values are loaded, the “**Integrate**” button will perform an integration. A new window will open up with the data in it. By default, the integration will be over all possible values of the other variable. For example, if you integrate in Q , it will be over all χ . This can be changed using the constraint check boxes.

For example, selecting the “**Constraint With Range on Right?**” check box and setting the “**Chi Lower?**” input to 0 and the “**Chi Upper?**” into to 90 will cause the integration in Q to be only of pixel values with χ values between 0 and 90.

Just like a caked plot, a polarization can be applied during an intensity integration. The “**Save Data**” button can be used to save out the intensity integration data as two column ASCII.

7.5 Macro

Macros can often be used to greatly speed up the data analysis. The “**Start Record Macro**” option in the “**Macro**” menu will begin recording a macro. After the desired tasks have been recorded, the “**Stop Record Macro**” option will stop the recording and save the commands to a file. The “**Run Saved Macro**” option will run a macro file.

Small edits to a macro file can make them much more versatile. Most macro commands are just the name of the GUI item possibly followed by whatever the GUI would want (such as a filename or a number). The macro command to load a diffraction file is “**Data File:**”. It must be followed by a line with a filename. It can also be followed by a list of filenames, a directory containing diffraction data, or some combination of each. The program will run the subsequent macro lines on every file in the list and all diffraction files found in any folders in the list. The loop will end with a subsequent “**Data File:**” command, a “**END LOOP**” line, or the end of the macro file.

When looping over diffraction files, there is special markup which makes it easy to save files in a loop to useful places with useful names. They are “**BASENAME**” and “**FILENAME**”. Whenever the program finds “**BASENAME**” in a macro file, it will be replaced with the path of the current diffraction file that has been loaded. “**FILENAME**” will be replaced with the filename of the current diffraction file. You could recreate a diffraction file (if’s extension was mar3450) with the macro command “**PATHNAME/FILENAME.mar3450**”. An exmaple of these keywords being used would be the macro line “**Save Integration Data**” followed by the line “**PATHNAME/FILENAME_int.dat**”. The macro would always save the intensity integrated data right next to the diffraction file with a name similar to the diffraction file.

Chapter 8

An Example

This section will present a pedagogically interesting example which demonstrates several of the programs important features. The purpose of this chapter is neither to be comprehensive nor to be particularly detailed. It will instead give a sense of the type of analysis that can be done with this program. It will motivate the rest of the manual. Further details and information on any of the things described below can be found in the appropriate sections of the manual.

David, a user of the program, was studying iron thin films using powder diffraction. He was particularly interested in measuring the shifts in diffraction peaks of a sample. To realize this experimentally, he captured the image of the standard calibration crystal Lanthanum Hexaboride (LaB6). Without changing the experimental parameters, he then imaged many samples for which he wanted to measure the shift.

The steps that are needed to do this analysis will be described. First, we will calibrate the diffraction detector. This is to say that we want to determine the precise experimental parameters that characterized the diffraction machine when the images were captured (for example, the distance between sample and detector, the energy of the x-rays, etc). Since the image of the standard calibration crystal was taken at the same time as the images of interest, the calibration parameters inferred from the standard crystal can be used to analyze the rest of data.

To perform this calibration, we first opened up the Area Diffraction Machine. Figure 8.1 shows what we are first presented with.

From the “Data File” input, we load into the program the LaB6 file. Once the file is loaded in, a new window opens up which shows the diffraction data. This window is shown in figure 8.2.

To do the detector calibration, the program must know the Q values associated with the standard crystal. Since LaB6 is so common, it is a preset default in the program. We go into the menu bar, into the “calibration” menu, into the “Standard Q ” menu, and then selected Lanthanum Hexaboride. This is shown in figure 8.3.

(More standard Q files might be added in the future). In order to perform image calibration, the program finally needs to know an initial guess at the calibration parameters. Although one could enter these parameters by hand, often times decent guesses at the ex-

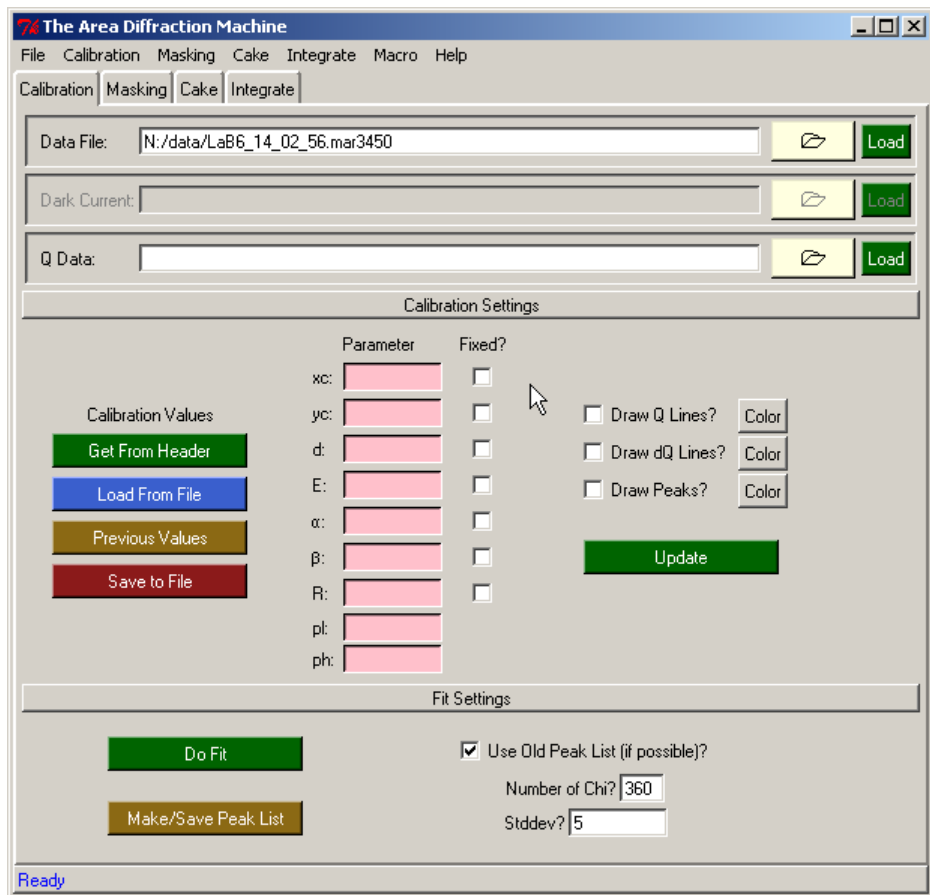


Figure 8.1: The calibration tab.

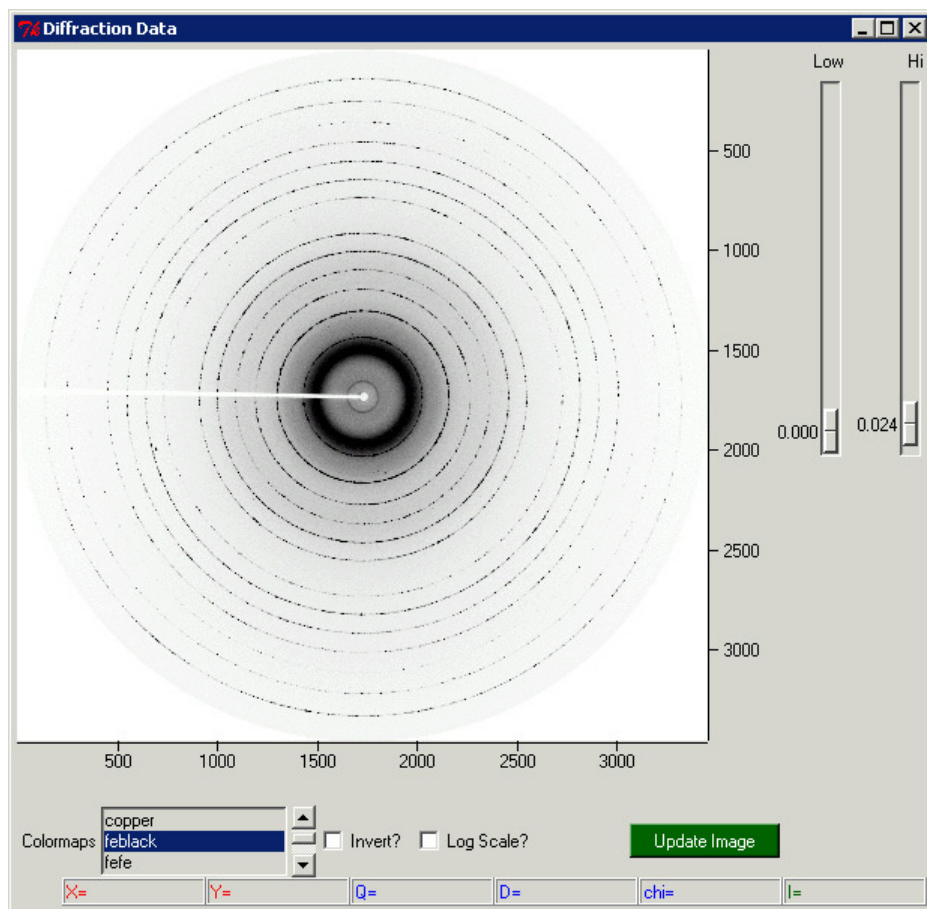


Figure 8.2: The data window.

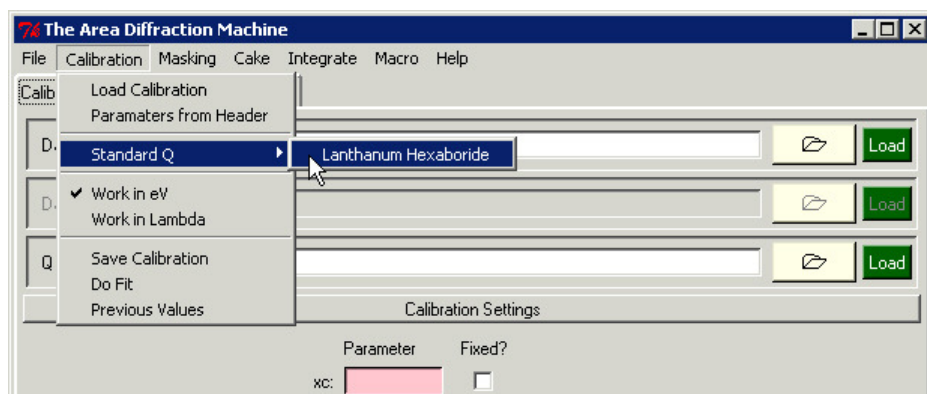


Figure 8.3: Loading a standard Q file.

perimental parameters are stored in the header data inside of the diffraction image. The program can try to find these header calibration values and put them into the inputs in the program. To do this, we could push the “Get From Header” button. With the image, the Q values, and an initial guess in the program, we are ready to do the calibration.

But first, we want to examine how good the initial guess is. To do so, we can select the “Draw Q Lines?” check box on the Calibration tab. When this is selected, the program will draw on top of the diffraction image red lines corresponding to what diffraction pattern should show up on the detector (for the given calibration parameters and Q values). Figure 8.4 shows what the program displays for our example.

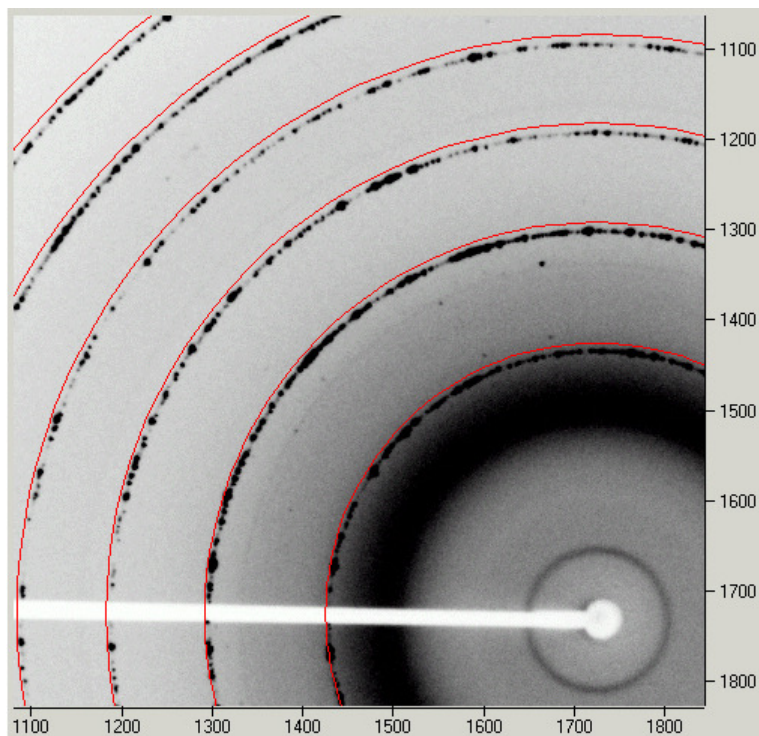


Figure 8.4: The diffraction image with constant the Q lines displayed upon it. These lines are calculated for the calibration parameters found in the header of the image. They are not particularly accurate.

Of course, our initial guess isn’t great so the red lines don’t match too well with the loaded pattern. The data will look like

We can do a cake of the data. A caked plot is a presentation of the data in a different parameter space. The x axis is Q and the y axis is χ . Ideally, if the calibration parameters are known exactly, the caked data will show up as many vertical lines. We can cake the data by going to the cake tab. This tab is shown in figure 8.5.

On this tab, we have to push the “AutoCake”. When we do so, a new cake window opens up. Figure 8.6 shows what the program displays for our example.

We see that for the caked data with the initial guess calibration parameters, our diffraction lines have a systematic wiggle. It might be hard to see with the full image, but by zooming into just one line we find the difference to be much more obvious. A zoomed in range is shown in figure 8.7.

This means that our initial guess at calibration parameters is not great. We can now do

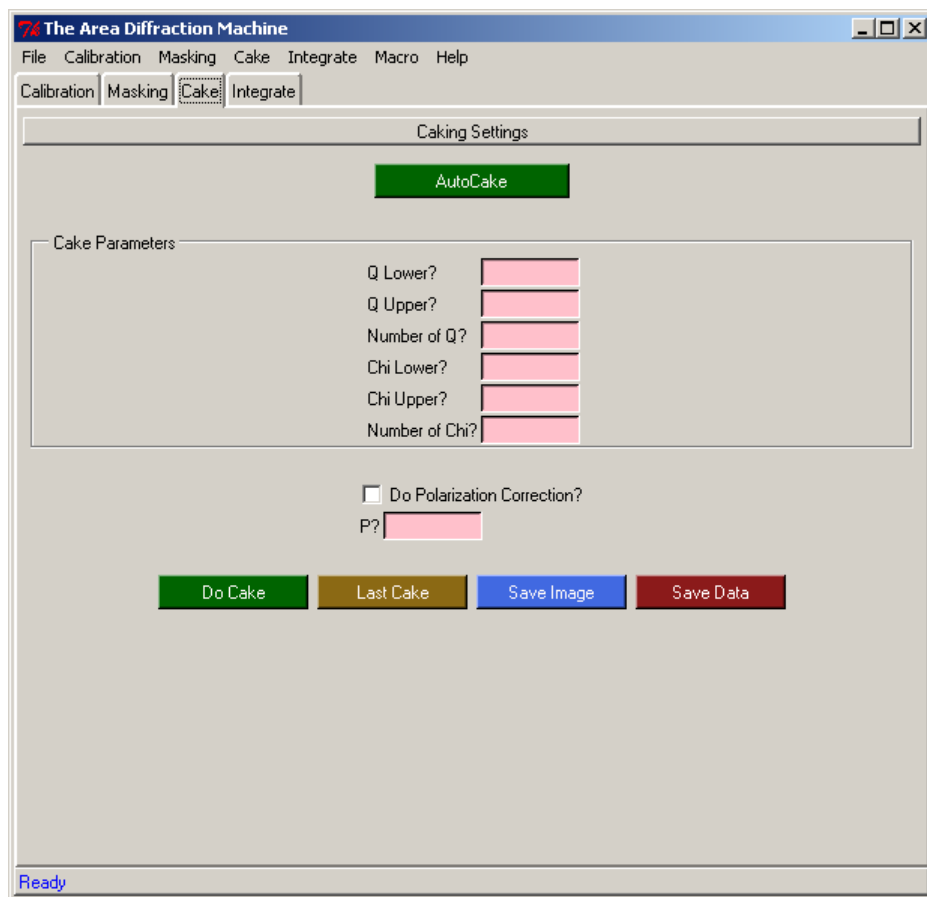


Figure 8.5: The calibration tab.

Figure 8.6: A caked plot done with the calibration parameters found in the header of the image. The header parameters are not particularly obvious and the diffracton peaks are not particularly straight. Calibratin helps improve the strightness of the diffracton peaks.

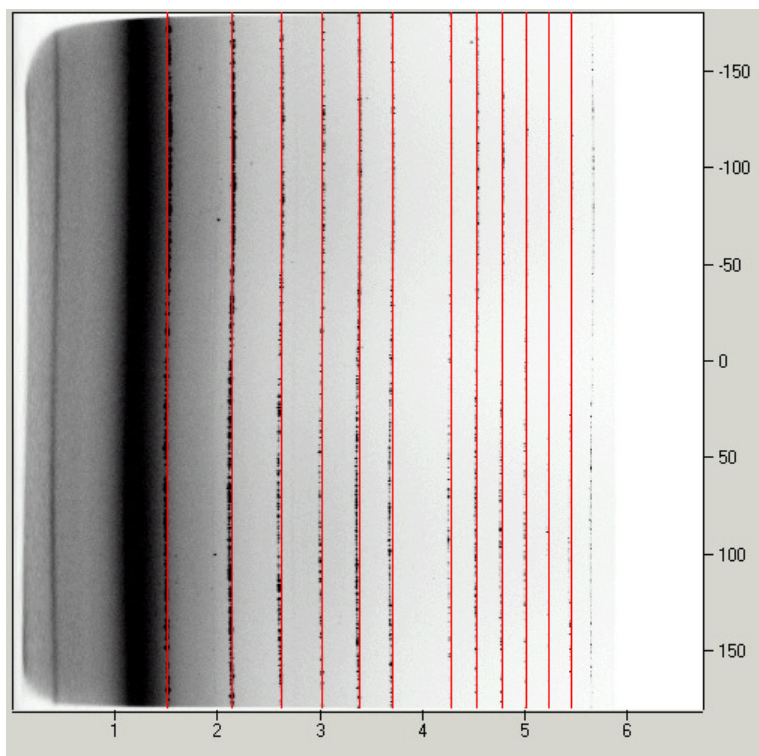
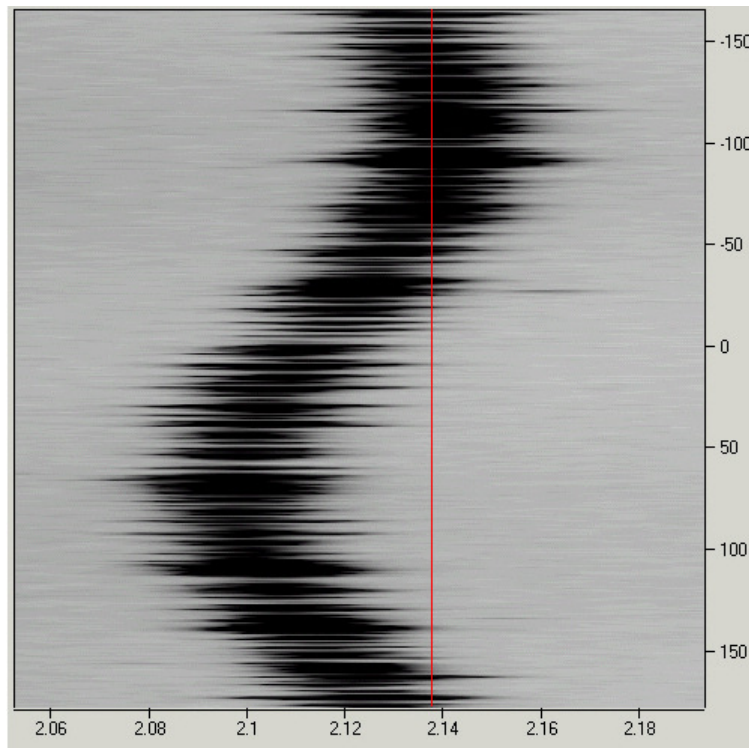


Figure 8.7: A zoom in of the cake shown in figure 8.6. When zoomed into diffraction image, the poor calibration becomes much more obvious.



the calibration. To do so, we push the “Do Fit” button on the “Calibration” tab. If the calibration did a good job, the constant Q lines drawn on the diffraction image move so that they are entirely over the diffraction pattern. This is shown in figure 8.8.

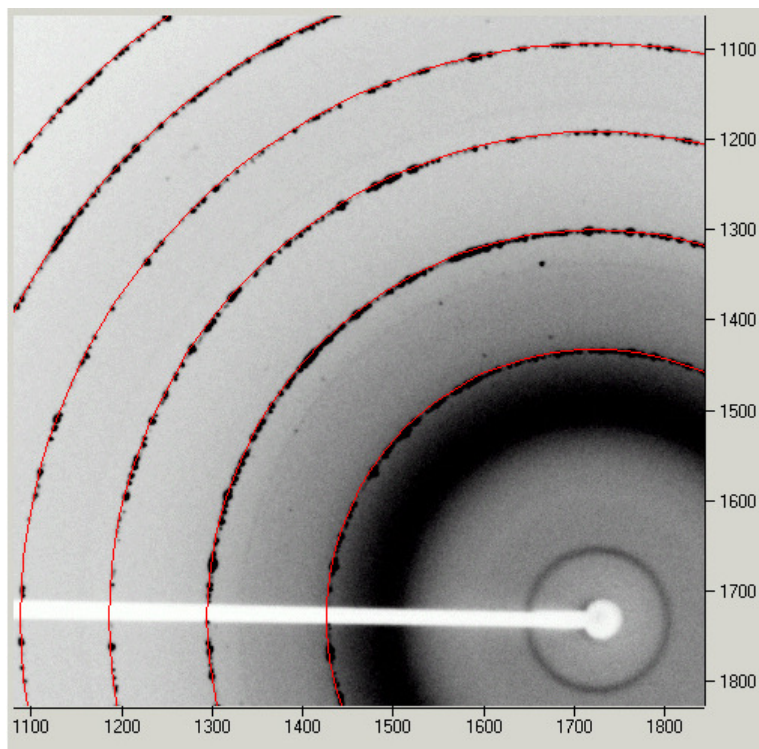


Figure 8.8: The diffraction window after being calibrated. The constant Q lines fall well on top of the diffraction peaks.

The diffraction peaks on the caked image become much straighter. The caked window after calibration is shown in figure 8.9.

They look good even when zoomed in. A corresponding zoom in of the caked window is shown in figure 8.10.

After caking the calibrated data and convincing ourselves that our calibration parameters are good, we can save the calibration parameters to a file for later use. We can do so using the “Save to File” button on the “Calibration” tab. After selecting the location “C:/Data/LaB6_cal.dat”, the calibration file gets saved as

Listing 8.1: ‘The Calibration Parameters File’

1	xc	1722.966078	0
2	yc	1724.227970	0
3	D	122.691351	0
4	E	12707.219316	0
5	alpha	-0.052910	0
6	beta	0.130553	0
7	rotation	-41.523477	0
8	pixelLength	100.000000	
9	pixelHeight	100.000000	

Figure 8.9: The cake window after calibration. The lines are much straighter than the lines in figure 8.6 before calibration.

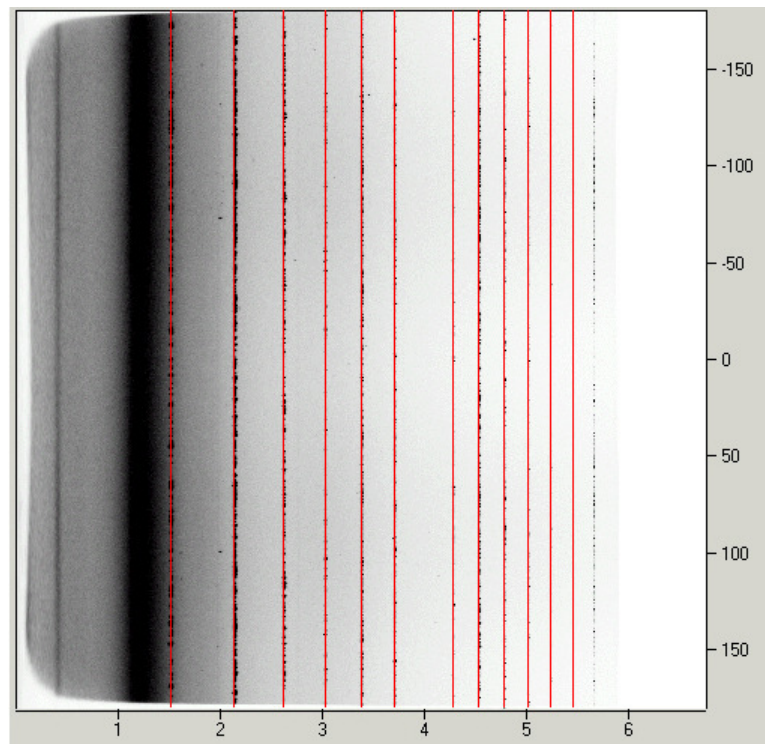
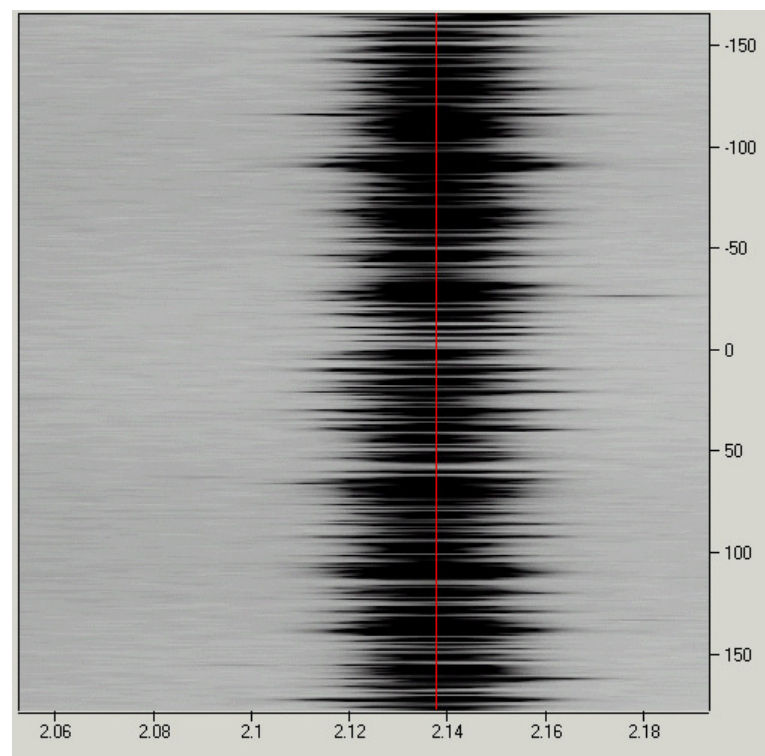


Figure 8.10: A zoomed in part of figure 8.9. Even at a large zoom in, the line remains very straight.



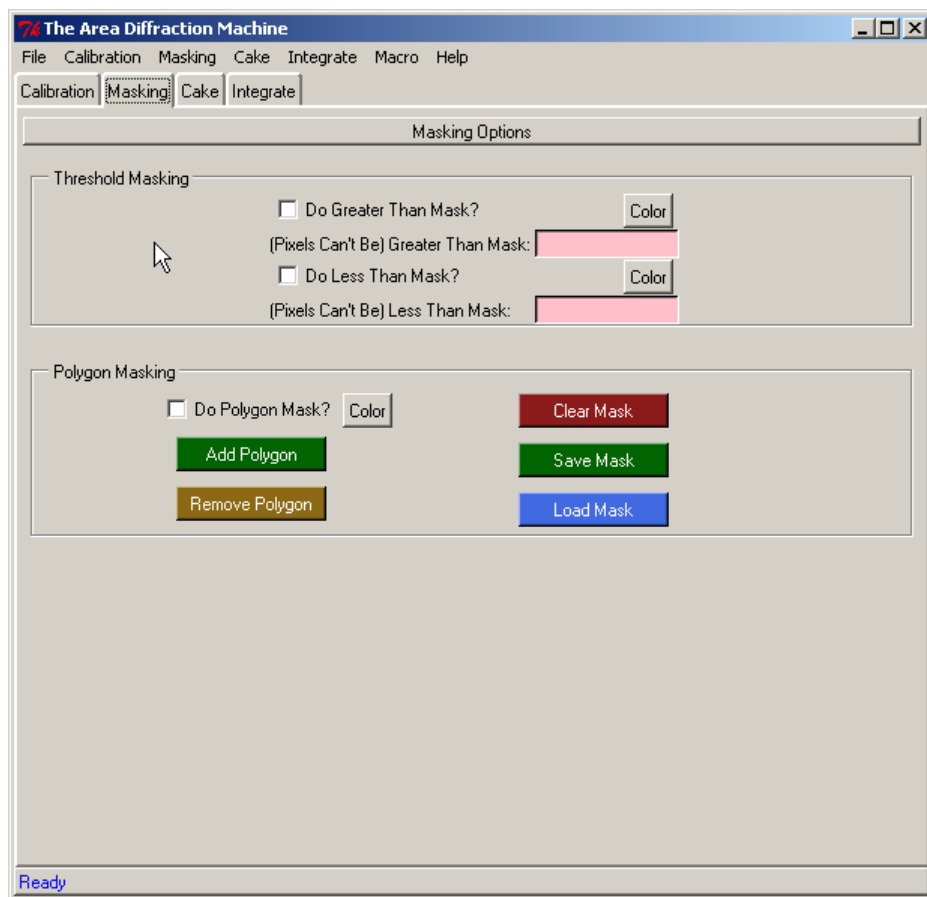


Figure 8.11: The pixel masking tab.

As can be seen in figure 8.2, there is a beam stop on the left side of the image which is obstructing part of the image. We know that none of the pixels blocked by the beam stop contain any interesting information so we are going to want to tell the program to ignore any pixels blocked by the mask. We can do so with a polygon mask. All polygon masking is done on the “masking” tab. A screenshot of this tab is shown in figure 8.11. We want to add a rectangular polygon mask on top of the beam stop in the image. To do so, we push the “Add Mask” button. We then move to the diffraction image and draw the beamstop on the image by left clicking nodes on the screen. We add the final node by right clicking. After having drawn the polygon mask, our diffraction image is shown in figure 8.12.

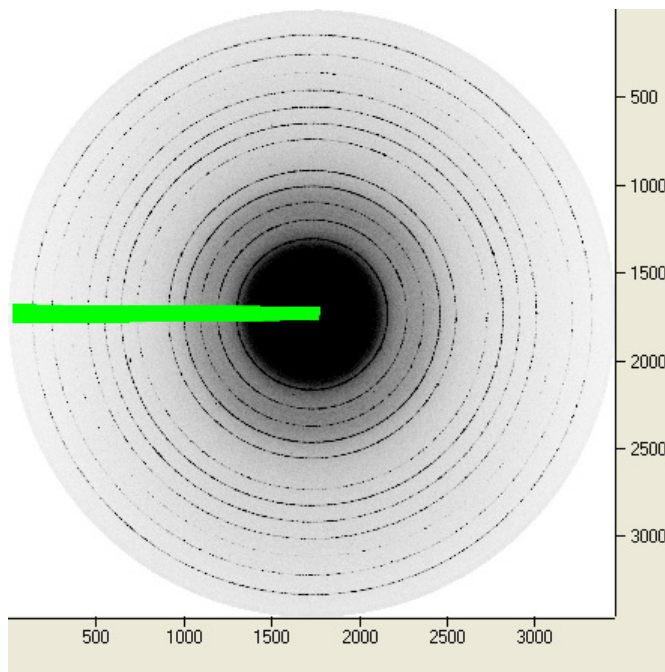


Figure 8.12: Here is the same diffraction data as in figure 8.2 but with a polygon mask drawn over the beam stop. This polygon mask will stop the beam stop below it from being used in subsequent data analysis.

Once we decide we are happy with our polygon mask, we can save it to a file using the “Save Mask” button. The file gets saved out as

Listing 8.2: ‘beam_stop_mask.dat’

```

1 # Polygon(s) drawn on Mon Apr 14 00:33:12 2008
2 25.6749379653    1634.63771712
3 42.7915632754    1814.36228288
4 1959.85359801    1857.15384615
5 1959.85359801    1626.07940447

```

We can then load in this mask when we do the rest of our analysis. The mask will make sure that none of the the pixels within the beam stop are used for any subsequent analysis.

Now, we are going to want to perform an intensity integration of the rest of our data. We can use the intensity integrate data to look for peaks in the data. The steps for doing the rest of this analysis are as follows. Load in particular file we are interested in. Load

in these calibration parameters using the “Load From File” button on the “Calibration” tab.¹ Next, we can load in our previously recorded beam stop mask using the “Load Mask” button on the “Masking” tab. We also have to make sure polygon masks are used in the analysis by making sure the “Do Polygon Mask?” check box is selected. With everything loaded into the program, we can perform a Q integration by going to the “Integrate” tab. The integration tab is shown in figure 8.13

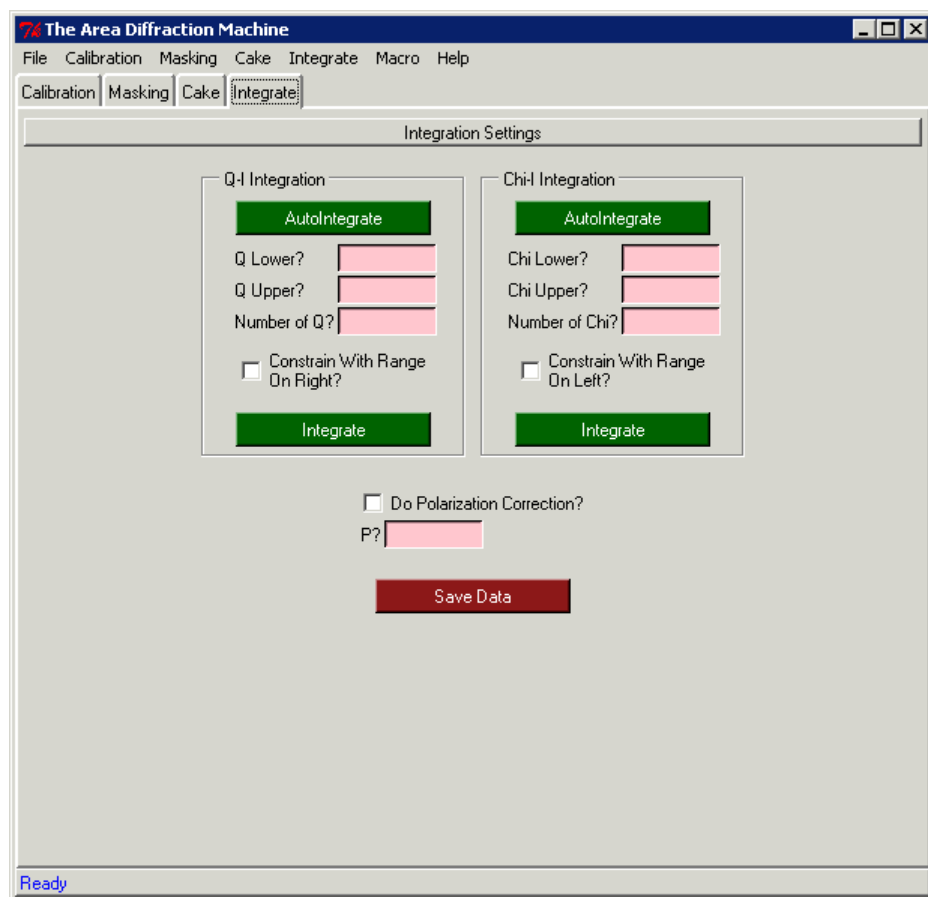


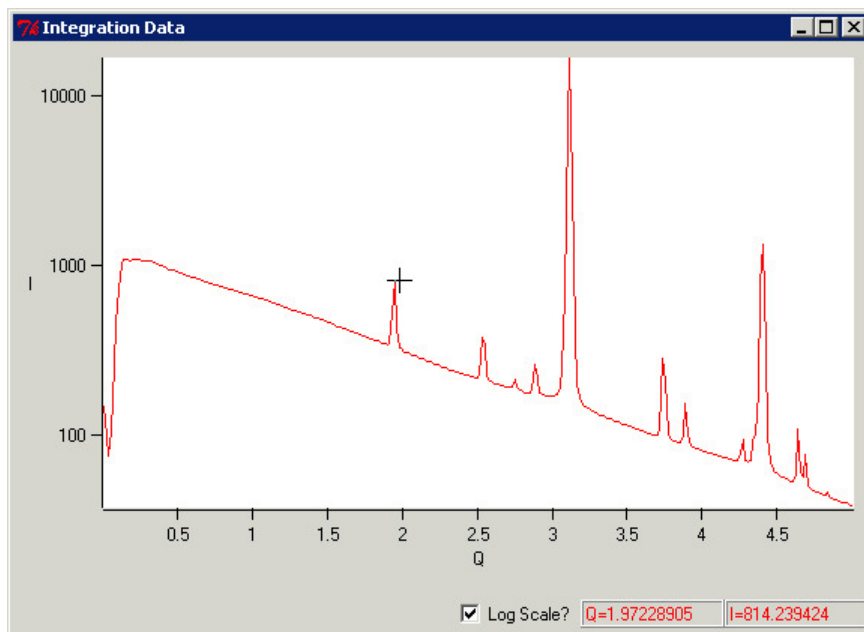
Figure 8.13: The integration tab.

We set the range of the Q integration by setting “Q Lower?” to 0 and “Q Upper?” to 5. We then set the precision of the integration, or the bin size, by setting the “Number of Q?” input to 300. Finally, we push the left “Integrate” button and a window showing the diffraction data opens. For a particular iron sample, this window is shown in figure 8.14.

We can save this data to a file with the “Save Data” button on the “Integration” tab. This data is saved out as two column ASCII. After doing this for all the different files that we have, we can load all the data into another program, such as Microsoft Excel, and compare the peaks.

¹If you just did the calibration, the parameters should already be in the inputs. The point is just that you could load the parameters into the program if you were, say, to open the program at some later point in time.

Figure 8.14: The intensity integration window for a particular iron sample.



But if there are a lot of files to analyze, this whole process can be very time consuming. Instead of doing this analysis by hand, we can automate the process by writing a macro to analyze all the files one at a time. First, we put all of our data into “C:/Data/”. The macro that we can run is

Listing 8.3: 'A macro to automate the analysis'

```

1 Data File:
2     C:/Data/
3 Load From File
4     C:/Data/LaB6_cal.dat
5 Load Mask
6     C:/Data/beam_stop_mask.dat
7 Do Polygon Mask?
8     Select
9 Integrate Q Lower?
10    0
11 Integrate Q Upper?
12    5
13 Integrate Number of Q?
14    300
15 Integrate Q-I
16 Save Integration Data
17     PATHNAME/FILENAME_int.dat

```

The first command loads into the program all of the diffraction files in the folder “C:/Data/” one at a time and runs the rest of the analysis on that particular file. The program then loads in the calibration file that we saved earlier and sets the integration bounds. Then the

program loads in the beam stop mask. Then, the program does a Q vs intensity integration and saves the intensity integrated data to a file. The PATHNAME keyword gets replaced with the path leading up to the particular file and the FILENAME keyword gets replaced with the particular file's name. For example, the file "FeL2_d070.mar3450" in the folder "C:/Data/" would be replaced with "C:/Data/FeL2_d070_int.dat". This command will let us save out of our intensity integrated data next to the corresponding diffraction file with a useful filename.

After we run this macro, all of our data will be saved out into text files. We can, for example, open the files in Excel and plot the different diffraction patterns on the same graph. If we did this, we would obtain a plot that looked something like the graph shown in figure 8.15.

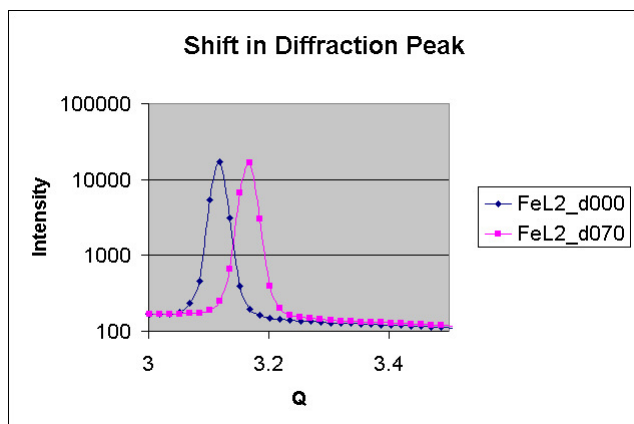


Figure 8.15: An example of what the shift in peaks might look like when two diffraction patterns were plotted in Excel on top of one another.

Chapter 9

Viewing Diffraction Data

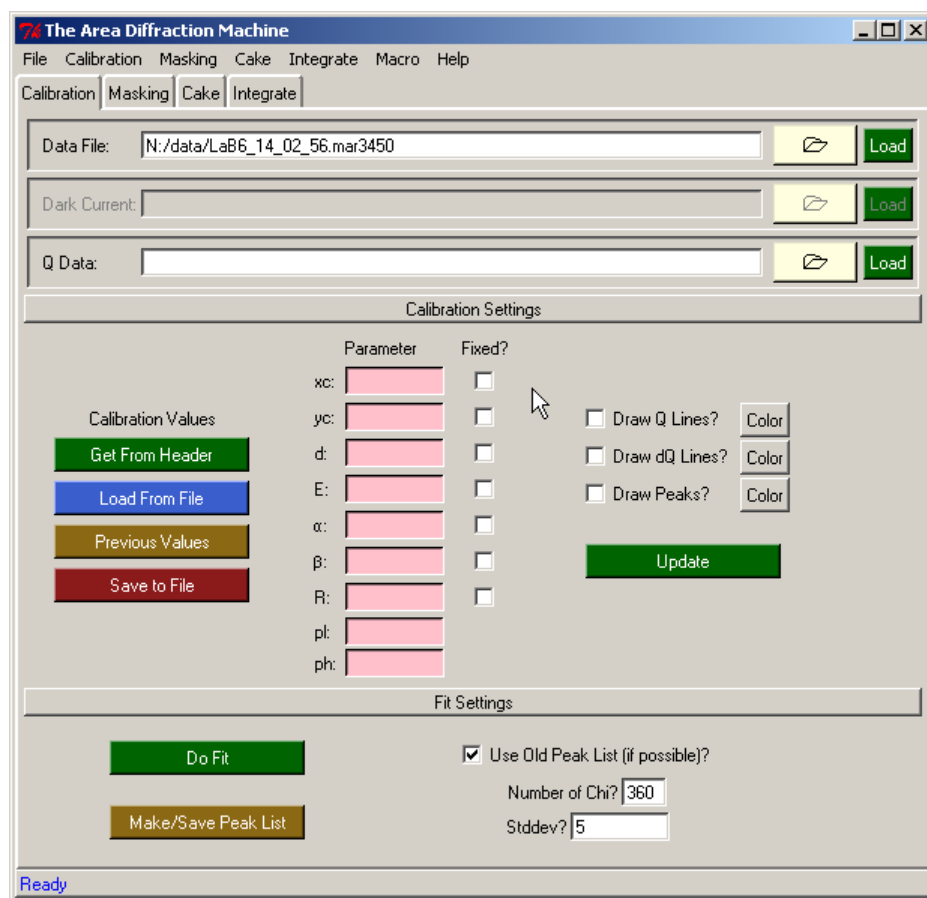
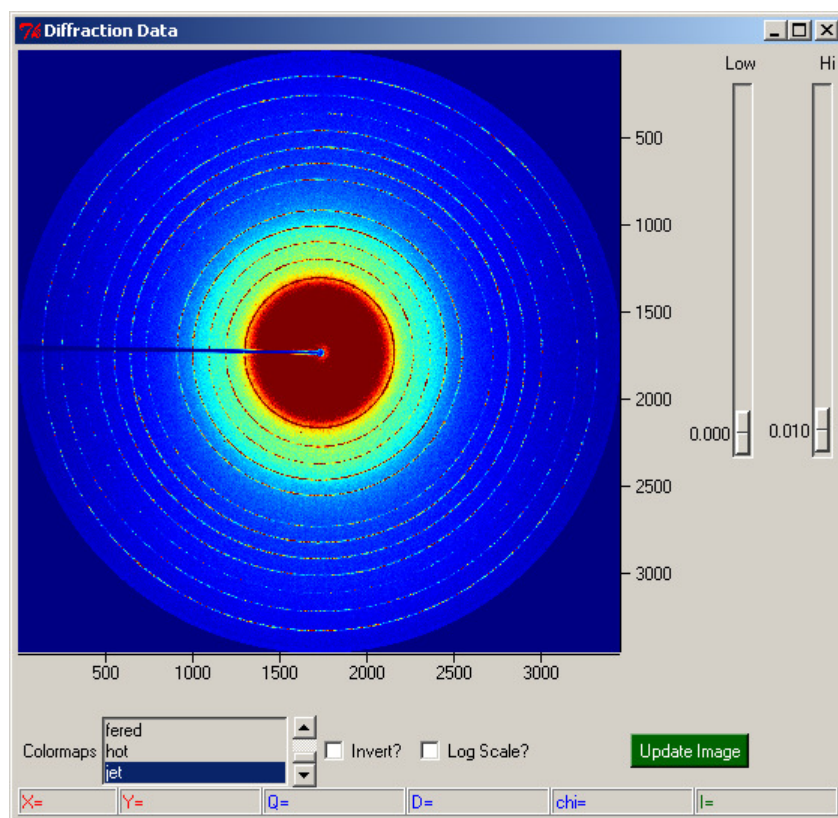


Figure 9.1: The calibration tab. This is what you see when you first open the program. This tab allows you to load diffraction data into the program.

When you first open the Area Diffraction Machine, you will see the calibration tab. It is shown in figure 9.1. The first thing you will probably want to do is load diffraction data into the program. This can be done with the “Data File:” input either by typing in the filename by hand and pushing the load button or clicking on the folder icon and using a file selector. After the file is loaded, a diffraction data window will open. This window is shown

in figure 9.2.

Figure 9.2: The diffraction data window. This window will open after a file is loaded. This window allows you to interact with diffraction data.



You can use the diffraction data window to interact with your diffraction data. you can:

- *Zoom into the data* – left click on the data and hold down on the mouse. When the mouse is moved around, the program will create a resizing square. When the mouse is released, the program will zoom into the selected range.
- *Zoom out of the data* – right click on the data.
- *Pan across the data* – hold shift, push down either mouse button, and then move the mouse around and the image will move with it. Let go of the mouse to stop panning.
- *Resize the window* – click on the bottom right corner of the window and drag. The window will reszie just like any other window and the data will become larger or smaller.
- *Read coordinates for a selected point* – when mousing over the image, the x , y , Q , χ , and I values for that pixel will be displayed at the bottom of the window. Q and χ will only be displayed if valid calibration data is loaded into the program. See chapter 11.
- *Change the Color Map* – the “Colormaps” selector can be used to change the particular color map used to display the data.

- *Invert the Color Map* – The “**Invert?**” checkbox can be used to invert the colors of the color map.
- *Low & Hi Pixels* – The sliders to the right of the image can be used to change the intensity scaling of the image. The low value corresponds to the intensity value that will be mapped to the lowest part of the color map and the hi value corresponds to the intensity value that will be mapped to the highest part of the color map.¹ This feature is useful because it can help make visible certain intensity ranges in the image.
- *Log Scaling* – By default, intensity values are linearly mapped to colors in the color map. The “**Log Scale?**” checkbox can be selected to instead apply a log scale mapping of the intensity values to the color map.

9.1 File Formats

The program can load in Mar data: “.mar2300”, “.mar3450”, and the “.mccd” Mar CCD format. It can load in standard “.tiff” data. It can load in the ESRF Data Format “.edf”. The program can only display square data. Whenever non-square data is loaded into the program, the program will simply pad out the image until it is a square with pixels whose intensity is 0.

9.2 Loading Multiple Images

Using the same file input, you can load multiple files into the program at the same time. If multiple files are put in the “**Data File:**” text input and separated by spaces, they will all be loaded in. Alternately, the diffraction data file selector can be used to select multiple files at the same time. All of the selected files will be loaded. When several files are loaded at the same time, the program will add the intensities of the images pixel by pixel and work with the combined image. This can be useful for analyzing several images taken of the same sample. The program can only add together files of the same format.

9.3 Saving the Diffraction Image

You can save diffraction data in the program as a popular image format. The data can be saved by doing to the “**File**” menu bar and selecting the “**Save Image**” option. The formats currently allowed are “jpg”, “gif”, “eps”, “pdf”, “bmp”, “png”, “tiff”, and the ESRF data format “.edf”.

Images saved as a popular image format will be saved with whatever threshold masks, polygon masks, Q lines, ΔQ lines, and peaks are currently displayed over the data in the

¹Technically, what is set is the percentage of the most intense pixel in the image should be mapped to the lowest or highest value in the color map.

diffraction data window. And it will be saved at whatever the current zoom level is.² See chapter 11 for a discussion of the Q lines, ΔQ lines, and peaks. See chapter 12 for a discussion of threshold masks and polygon masks.

Because the program will pad any non-square data when it is loaded to. The program will always save out all images as squares. If this is undesirable, the saved images will need to be cropped using another program.

²This is not the case with ESRF data. When an image is saved as an ESRF file, it will be saved un-zoomed with none of the lines or masks on top of it.

Chapter 10

Detector Geometries

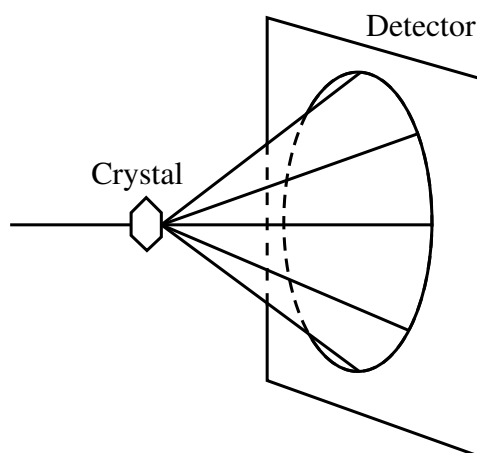


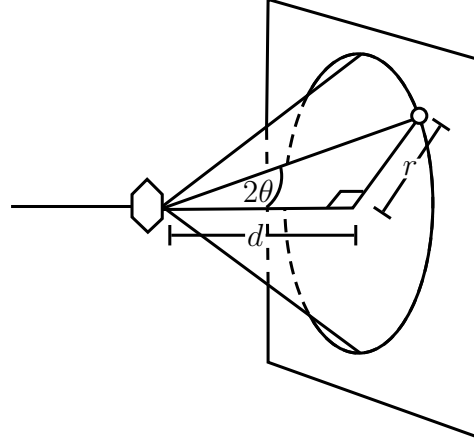
Figure 10.1: An X-Ray diffraction setup. X-rays scatter from a 3-D sample and are captured by a 2-D detector. In this setup, the detector is perpendicular to the incoming x-ray beam.

X-ray diffraction can be modeled as in figure 10.1. Cones of light leave the crystal at particular angles to the incoming beam. These cones of light are captured by a detector. By convention, the scattering angle of the x-rays measured with respect to the incoming beam is called 2θ . Usually, the interesting thing to measure by doing x-ray diffraction is the scattering angles of these cones of light. If we placed a detector perpendicular to the incoming beam, the cones of light would be detected as circles of high intensity. If we knew the distance from the sample to the detector and the distance from the center of the detector to a particular ring (or really any point on the detector), we could easily calculate the scattering angle of the light. If the distance from the crystal to the detector is d and the distance from the center of the detector to our particular point on the detector is r , then the scattering angle is

$$\tan 2\theta = \frac{r}{d}. \quad (10.1)$$

This is shown in figure 10.2. Life is not always so simple. The detector is never exactly perpendicular to the incoming beam. In practice, the detector will always be slightly offset with respect to the incoming beam. Failing to account for this would introduce a systematic error in a measurement of scattering angles.

Figure 10.2: The same setup as in figure 10.1. We are now interested in some particular point on the detector. 2θ is the scattering angle of the light that gets to this point, d is the distance from the crystal to the detector, and r is the distance from the center of the detector to some particular point (which 2θ is associated with). By center of the detector, we mean the point on the detector where the beam would hit if it did not interact with the crystal.



There is a need to analyze diffraction data on detectors that are not perpendicular to the incoming x-rays. We will present a theory of tilted detectors first developed by Abhik Kumar in [5]. Our derivation will result in different formulas because of a different assumption about how the detector is tilted.

What we are interested in is mathematically describing position coordinates on a tilted detector by relating them to more theoretically motivated quantities such as the scattering angles that would lead to a beam hitting that particular point on the detector. In order to do this, we must first work out the transformation of points on a tilted detector to points on an untilted detector. This is to say that we want to figure out where on an untilted detector the beam would have hit were it to hit that untilted detector instead of the tilted detector. The point on the titled detector can be thought of as the shadow of the point on the untilted detector. We will call the point on the untilted detector as measured on the untilted detector (x, y) and the corresponding point on the tilted detector as measured on the tilted detector as (x''', y''') . The reason for three primes will become obvious shortly. This is shown schematically in figure 10.3. Another way to think about this problem is to imagine putting your head at the sample and then looking directly at some point (x''', y''') on the real tilted detector. What we want to figure out is some corresponding point (x, y) on an imagined untilted detector which would appear to eye to be in the same direction.

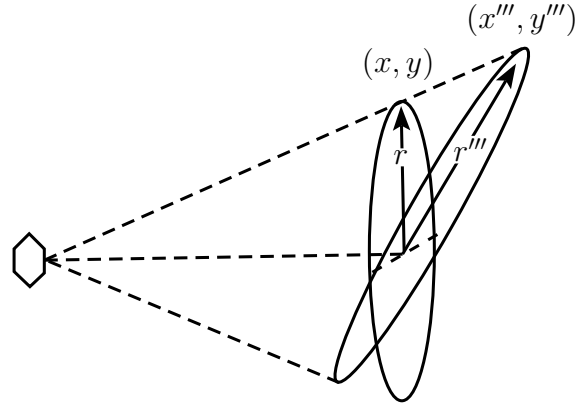


Figure 10.3: Here, the detector is titled by some arbitrary angle with respect to the incoming beam. We will call some arbitrary point on the tilted detector (x''', y''') . We are interested in relating this point to the point (x, y) on some imagined untilted detector where a scattered beam would have hit were that tilted detector in place instead of the tilted detector.

10.1 The Three Tilt Angels

In order to relate these points, we need to find a way to describe some arbitrary tilt. To do so, we will characterize a detector tilt in terms of 3 independent detector rotations. We will use two orthogonal rotations about the x and y axis followed by one rotation about the center of the detector. These three angles are shown in figure 10.4. We can solve our original problem much easier if we deal with each rotation separately.

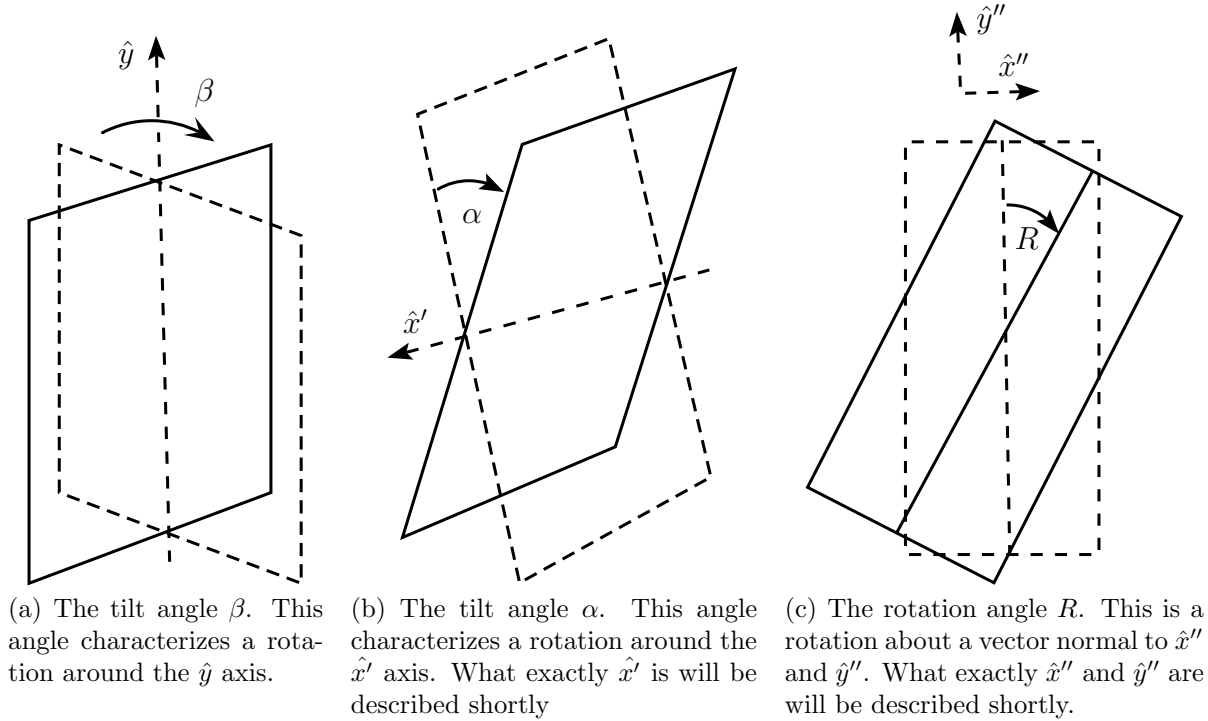


Figure 10.4: Any detector tilt can be characterized as a rotation by β followed by a rotation by α followed by a rotation around the center of the image by R .

10.2 The β Tilt

We will first apply a rotation around \hat{y} by angle β . To do this, we will first consider a point (x, y) on an untilted detector and project it onto some point (x', y') on this rotated detector. This is to say that we will figure out where on the detector rotated by angle β a beam would hit were it to hit the tilted detector instead of the untilted detector. A diagram of this is shown in figure 10.5. We can use the geometry of these diagrams to figure out the relationships between the coordinates. Using the property of similar triangles, we see that

$$\frac{x}{d} = \frac{x' \cos \beta}{d + x' \sin \beta}. \quad (10.2)$$

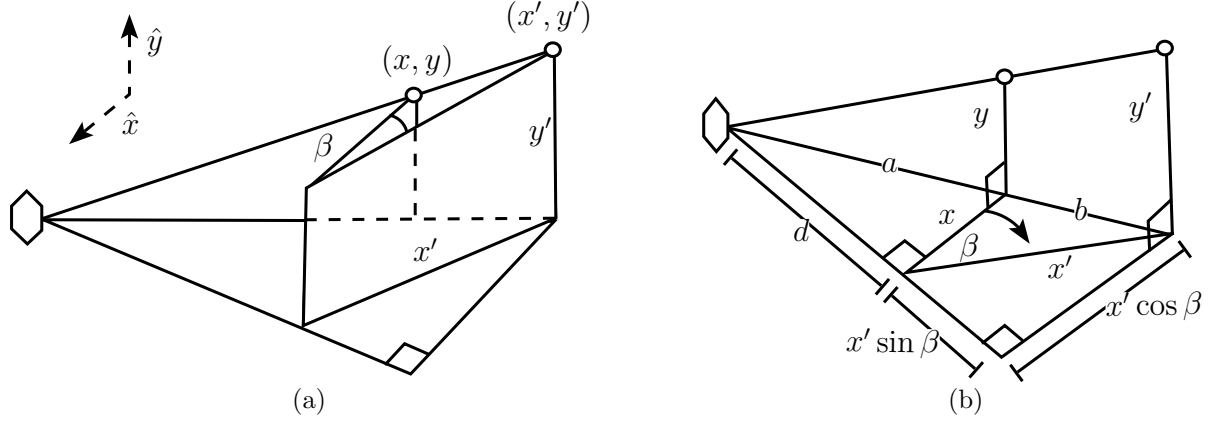


Figure 10.5: A diagram of the situation depicted in figure 10.3 where only the β rotation about \hat{y} has been applied.

From this it follows that

$$x = \frac{dx' \cos \beta}{d + x' \sin \beta}. \quad (10.3)$$

Using similar triangles again, we see that

$$\frac{y}{a} = \frac{y'}{a + b} \quad (10.4)$$

$$\frac{d}{a} = \frac{d + x' \sin \beta}{a + b}. \quad (10.5)$$

from which it follows that

$$y = \frac{dy'}{d + x' \sin \beta}. \quad (10.6)$$

So, equation 10.3 and 10.6 give us the proper geometrical equations for relating a point on the untilted plane (x, y) to the corresponding point (x', y') on the first plane.

10.3 The α Roll

We can now take this point (x', y') on the tilted plane and project it onto another plane which has been tilted by β about \hat{y} and a rolled by α around \hat{x}' . To do so, we take the plane which is rotated by an angle β around \hat{y} and then rotate it around the line x' . This is diagrammed in figure 10.6. A more geometric diagram can be seen in figure 10.7 and a cross section of the $y = 0$ plane can be seen in figure 10.8.

We can use these figures to determine the equations that we need. We see from figure 10.8 that $f = y'' \sin \alpha \cos \beta$. From figure 10.7, we see that $h = y'' \cos \alpha$. Using the property of similar triangles, we see that

$$\frac{y}{a} = \frac{y'' \cos \alpha}{a + b + c} \quad (10.7)$$

Using similar triangles again, we see that

$$\frac{d}{a} = \frac{d + x'' \sin \beta + f}{a + b + c} \quad (10.8)$$

From which we can deduce that

$$y = \frac{dy'' \cos \alpha}{d + x'' \sin \beta + y'' \sin \alpha \cos \beta}. \quad (10.9)$$

Figure 10.8 shows that $g = y'' \sin \alpha \sin \beta$ and that $x'' \cos \alpha = l + g$. Using similar triangles again, we see that

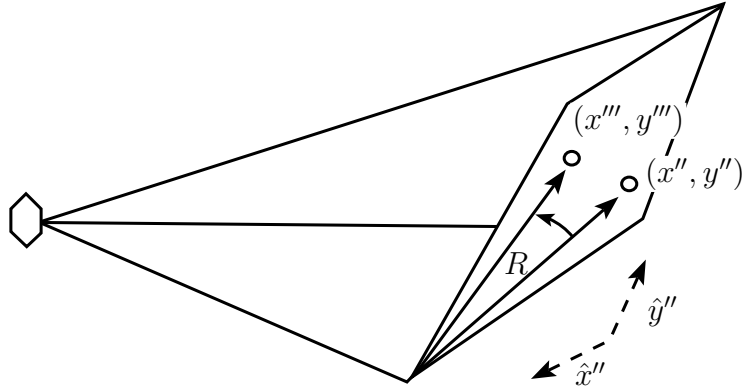
$$\frac{x}{d} = \frac{l}{d + x'' \sin \beta + y'' \sin \alpha \cos \beta} \quad (10.10)$$

Plugging in and simplifying, we get

$$x = \frac{d(x'' \cos \beta - y'' \sin \alpha)}{d + x'' \sin \beta + y'' \sin \alpha \cos \beta}. \quad (10.11)$$

10.4 The R Rotation

Figure 10.9: Here, we take a point on a plane rotated by angle β about \hat{y} and by angle α about \hat{x}' . We then rotated this point about a line normal to the plane going through the origin by angle R . Rotating the point is equivalent to rotating the plane.



We have to deal with the final rotation. We will rotate the coordinate (x'', y'') on the previous detector about a line perpendicular to the plane that goes through the center of the detector. We will call this final point (x''', y''') . This is shown schematically in figure 10.9. The equation for this rotation is

$$x'' = x''' \cos R + y''' \sin R \quad (10.12)$$

$$y'' = y''' \cos R - x''' \sin R \quad (10.13)$$

Applying equation 10.12 onto equation 10.11 and 10.9 give us the relationship that we wanted all along.

10.5 Relationship to Pixel Coordinates

(x''', y''') is suppose to represent what we actually measure on a real detector. Unfortunately, things are not quite so easy. We do not actually measure these values. The whole formalism assumes that we are measuring distances from the point on the detector where the beam would hit were it not to be diffracted. Unfortunately, it is not at all clear what this point is. A discussion of how to find this center center will be given in section 11, but for now lets simply state that there is some point on the detector that is the center and call it (x_c, y_c) We are interested in some other pixel reading on the detector which corresponds to the point (x''', y''') . Lets call it (x_d, y_d) . There is some material property of the detector describing the distance between each pixel (e.g. 1000 mm/pixel). We will call this width ps . We can relate these quantities using:

$$x''' = (x_d - x_c) \times ps \qquad y''' = (y_d - y_c) \times ps \qquad (10.14)$$

This means that, in terms of (x_c, y_c) and ps , we can relate (x, y) and (x_d, y_d) which are directly measurable experimental quantities.

10.6 Inverting the Equations

We can invert these formula to learn what x'' and y'' are in terms of x and y . We have:

$$x'' = \frac{dx}{d \cos \beta - x \sin \beta - \cos \beta (x \cos \beta + d) / (\frac{x}{y} \cot \alpha + 1)} \qquad (10.15)$$

and

$$y'' = \frac{dx \cos \beta / (\frac{x}{y} \cos \alpha + \sin \alpha)}{d \cos \beta - x \sin \beta - \cos \beta (x \cos \beta + d) / (\frac{x}{y} \cot \alpha + 1)}. \qquad (10.16)$$

10.7 Q , 2θ , and χ

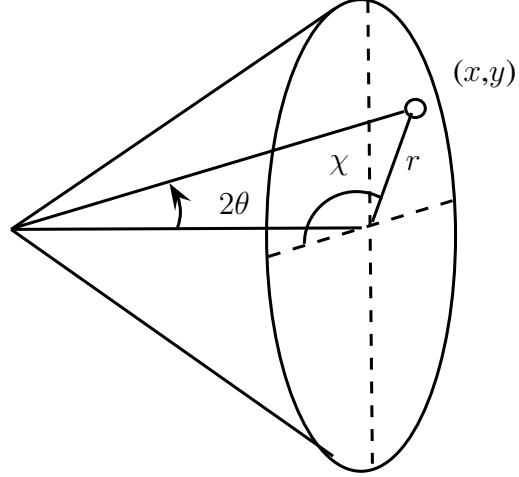
We now have a way of relating (x''', y''') , a point on a detector with a pitch β , tilt α , and a roll R applied to it, to a point on an untilted detector (x, y) where a beam of light would have intersected were it not to hit the tilted detector. With this relationship, we can now relate these quantities to theoretically motivated quantities. In particular, the angle of scattering of a beam is by convention called 2θ and a quantity measuring the scattering angle around the incoming beam is called χ . These quantities are shown in figure 10.10. We can see that the relationship between (x, y) and 2θ and χ is

$$\tan 2\theta = \frac{r}{d} = \frac{\sqrt{x^2 + y^2}}{d} \qquad (10.17)$$

and

$$\tan \chi = \frac{y}{x} \qquad (10.18)$$

Figure 10.10: For a particular point (x, y) , we always associate two quantities: 2θ and χ . 2θ is the angle of scattering of the beam, or the angle that an incoming beam is deflected by when it diffracts off the crystal. χ is a measure of the azimuthal angle around the beam. It tells you in what direction radially outwards (with respect to the undeflected beam) the outgoing beam was scattered.



The quantity Q is often used instead of 2θ . they are related by

$$Q = 4\pi \sin(2\theta/2)/\lambda \quad (10.19)$$

The reason for using Q instead of 2θ is because diffraction theory shows that the Q values of preferential scattering of a crystal is a material property independent of the experimental setup (such as d and λ).

Alternately, energy could be used in this formula. To do so, energy can be related to wavelength using the De Broglie's formula

$$E = hc/\lambda \quad (10.20)$$

Finally, sometimes people use the quantity D instead. D is related to Q by

$$D = 2\pi/Q \quad (10.21)$$

Using equation 10.14, we now have a way of relating pixel coordinates (x_d, y_d) read directly off of a detector to the theoretically motivated coordinates (Q, χ) . In order to do this conversion, we must use the values x_c , y_c , ps , d , λ , α , β , and R . A discussion of how these values can be determined so that this transformation can in practice be done will be given in section 11

Chapter 11

Calibration

One of the most common types of analysis of diffraction data is to perform an intensity integration in Q . This will create a plot of average intensity as a function of Q . Since powder diffraction procedures cones of light, this means that the intensity should be uniformly large for some Q values and uniformly low for others, leading to Q values where the intensity sharply peaks. The Q values that lead to these peaks can be used to learn structural information about the crystals that are being diffracted. So in principle, using the transformations just described, it should be easy to convert all of the pixel coordinates (x_d, y_d) into Q values and then plot average intensity as a function of Q . The only problem we would face is that in order to do the transformation, we would need to know the values of the parameters that characterize an experiment. These are x_c , y_c , d , λ , α , β , and R .¹ Calibration then is the process used to find what we will now call the calibration values.

11.1 The Calibration Algorithm

Although in principle all the calibration values could be experimentally measured, in practice they can not be directly measured to an acceptable level of precision. Instead, a standard calibration procedure is used to infer these values from real diffraction data. The trick to doing this calibration is to image a standard while performing the diffraction analysis of an unknown sample. Assuming that the diffraction machine was not changed between the collection of the standard crystal and the diffraction of the unknown sample, the calibration data corresponding to the two images will be the same. So, if we can figure out the calibration values of the standard crystal, we can use these values when analyzing the unknown crystal. This is exactly what is done in practice.

What it means to use a standard crystal is to know the particular Q values for which the crystal preferentially scatters light. With this information, and the calibration values for some particular experiment, we could in principle figure out exactly what diffraction pattern we should find. To do this, we could, for each Q value, vary χ and calculate the (x_d, y_d)

¹The pixel scale ps is usually known in advance as a uniform property of the detector being used.

coordinate corresponding to that (Q, χ) pair. After using enough χ values, we would be able to fill in the rings as they would show up on the detector.

In fact, my program can do just this. If you load in a set of Q values (see section 11.13) and then put into the program some calibration values, and then push the “Draw Q Values?” check box, you can then see what the particular diffraction image would have shown up on the detector. This is described thoroughly in section 11.7

Being able to do this still leaves us with a hard problem to solve. For particular calibration values, we can easily calculate what the diffraction pattern should look like. But what we really know is what the calibration values are for the known diffraction pattern of a standard crystal. In order to perform the real calibration, then, we can vary the calibration values until they make the pattern that can be calculated to show up to match the pattern that was actually captured. The process of image calibration then is a procedure to ‘fit’ the calibration values to a diffraction pattern with known Q values.

11.2 The Fitting

In order for the fitting algorithm to work, the program must already have an initial guess of the real calibration parameters. This initial guess does not have to be perfect, but it should be somewhat close. The algorithm then requires a list of the known Q values. And it additionally requires a range for each of these Q values. In order for the algorithm to work properly, inside of this Q range (as calculated by the initial calibration value guess) there should be the peaks that we are interested in and no spurious other peaks that would confuse the computer.

With the Q values specified along with Q ranges, we can divide up any diffraction image several regions, where within each region we know there is a unique peak. An example of this is shown in figure 11.1.

Our algorithm first requires finding (x, y) coordinates of many diffraction peaks. To do so, the algorithm will pick some χ value and then spread radially out from the center of the diffraction image in this χ direction.² Between the given Q range (for each of the Q ranges), the program stores an array of all the data point on the line. It then fits a Gaussian to the data and the (x, y) coordinate of the center of this Gaussian (x, y) is taken to be the peak. A diagram showing this algorithm is shown in figure 11.2. This method is then done for many different evenly spaced χ values and the particular value can be selected by the user for increased accuracy.

The only really tricky part about this step is that there is not always a consistent diffraction ring around the image and therefore some of these fits should not find peaks. Whenever this occurs, the program just ignores the current fit and moves to the next. But figuring out when some particular peak is bad is not particularly obvious. The method that this program uses is to ensure each peak passes a few tests. The first test is that the fit peak was too close to the edge of the image. So any peak where the Gaussian fit’s center plus or minus

²Remember that the center is specified by the initial calibration values

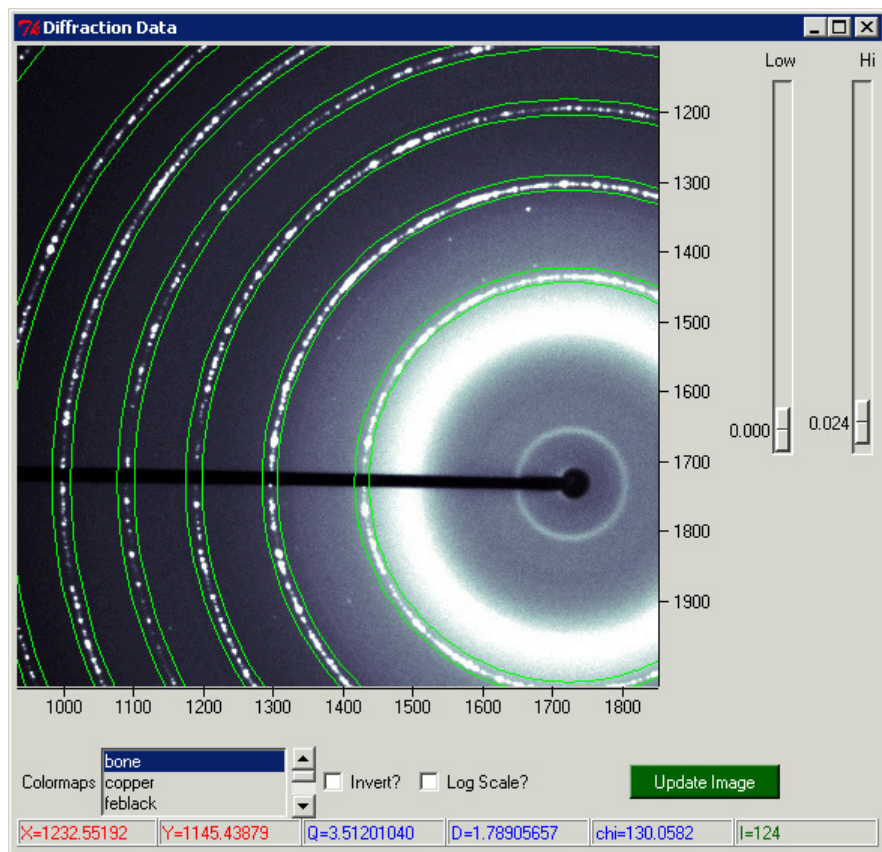


Figure 11.1: A division of a diffraction image into Q ranges where each diffraction peak falls uniquely inside one Q range.

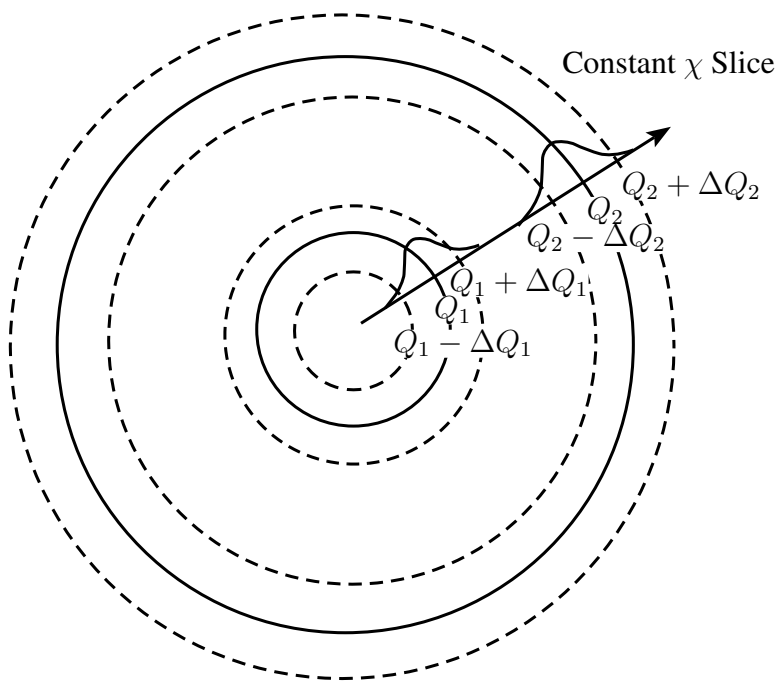


Figure 11.2: Here is a diagram of the peak finding algorithm. The solid circular black lines represent diffraction peaks on the image. The dotted lines represent the Q ranges used to find the peaks. The diffraction peaks are entirely within the ranges. Finally, the radial line represents the program picking a particular χ value and looking for peaks inside of the Q ranges. Finally, the Gaussian peaks represent the program fitting a gaussian to the intensity profile inside of each of the ranges.

twice the fit’s standard deviation gets outside of the Q range is considered too close to the edge of the image. The next test that is done is to calculate is the standard deviation of the data outside of the peak is significant when compared to the height of the peak fit. To do this, the code calculates the standard deviation of all the pixels that are farther then twice the peak’s fit standard deviation away from the center of the peak. If the height of the peak divided by this calculated background standard deviation is smaller then some particular value, the peak is considered bad. This value is called by the program “Stddev” and can be specified by the user from user. Presumably, the higher that “Stddev” is, the more picky the program is about what a good peak looks like. This isn’t the most robust method for finding peaks, but it seems to work pretty well and it should be easy in principle to add new tests to the algorithm.

After compiling a list of diffraction peaks in the image, the program can then define a residual function which we can minimize to find the best fit calibration values. To do so, we can convert the (x, y) coordinate of each of the peaks into a $(Q_{\text{peak}}, \chi_{\text{peak}})$ pair. For each of these (x, y) coordinates, we also know what the input Q list says the experimental Q value for this peak should be (which we will call Q_{exp}). We can therefore define the residual function as

$$\text{Residual}(x_c, y_c, d, \lambda, \alpha, \beta, R) = \sum_{x, y \text{ pairs}} (Q_{\text{peak}} - Q_{\text{exp}})^2 \quad (11.1)$$

The functional dependence comes from calculating Q_{peak} from a known (x, y) coordinate. We see that the smaller the Residual is, the closer we have come to finding the real calibration values which characterized the diffraction experiment. If we had perfect calibration parameters, the residual should be equal to zero. But it is well defined for any calibration parameters. So we can take this function of 7 variables and minimize it. The value of this function at its minimized is the best guess calibration values. There are plenty of computer algorithm that can minimize arbitrary multi-variable functions. The one that this code uses is called the Levenberg-Marquardt nonlinear least squares algorithm and the particular implementation that is used to to perform the calibration is Manolis Lourakis’s levmar library[6]. Ideally, once the minimization is done, a good guess at the calibration values is found.

11.3 Calibrating With the Program

Diffraction image calibration is done with the calibration tab of the program. This tab is shown in figure 9.1 on tab 79.

As described above, to calibrate an image you must have already loaded into the program a diffraction data file, a Q data file for the particular sample that was taken, and an initial guess at the calibration data.

Once you have done these three things, you can simply push the “Do Fit” button to calibrate the diffraction data. The program will then perform the calibration algorithm as described in section 11.1. Once the program finds a best guess for the new calibration values, it will put those values into the inputs.

While fitting the program will print to the console some useful things. Most interesting, the program will calculate the residual function divided by the number of 11.1 and display print the value to the terminal before and after the calibration is done³ The output will look like

Listing 11.1: Displaying the Residual

```

1 - Before fitting, the calculated residual is 5.336138e-04
2 - Doing the fitting
3 - After fitting, the calculated residual is 6.532131e-06

```

The program will then display the reason why the fitting algorithm decided to quit doing the fitting and decided that it found its best guess. For example, the program might print out

Listing 11.2: Reason For Quitting

```

1 - Reason for quitting the fit: 2-stopped by small gradient J^T e

```

The different reasons are told to me by the levmar fitting algorithm. I am taking them directly from the levmar website <http://www.ics.forth.gr/~lourakis/levmar/> That website says that the different reasons why the fitting can stop are:

- stopped by small gradient $J^T e$
- stopped by small Dp
- stopped by itmax
- start from current p with increased μ
- no further error reduction is possible. Restart with increased μ
- stopped by small $\|e\|_2$ ^[6]

I think that the first reason to quit (stopped by smallw gradient) means that the program found its way to the bottom of the hill and is convinced that it did its best job minimizing the function. I think that (stopped by itmax) means that the program was forced to quit by a hard coded limit to the number of loops through the fitting. So if you come across this message, you should probably do the fit again with the current values. I honestly don't know enough about the levmar fitting algorithm to know what the other messages really mean. If you need to know, you should go into the fitting algorithm's documentation and see what you can find out.

The fitting algorithm also provides a covariance matrix that it finds while fitting. I do now know how it calculates this matrix or what it exactly what it means physically. Nevertheless, I print it out after the fitting is done.

³Actually, the program calculates the residual function divided by the number of peaks. So it really displays the residual per peak, which is a more useful quantity because it would not change if more peaks were used in the fit

Listing 11.3: Display of A Covariance Matrix

```

1 Covariance Matrix
2 [[ 9.43e-04 -1.53e-04 5.36e-05 3.27e-03 -1.77e-03 3.64e-03 2.10e+00]
3 [ -1.53e-04 1.17e-03 -1.40e-04 -8.58e-03 3.91e-05 -2.02e-04 -1.25e-01]
4 [ 5.36e-05 -1.40e-04 2.07e-04 1.38e-02 -1.45e-04 3.12e-04 1.78e-01]
5 [ 3.27e-03 -8.58e-03 1.38e-02 9.49e-01 -6.44e-03 1.40e-02 8.02e+00]
6 [ -1.77e-03 3.91e-05 -1.45e-04 -6.44e-03 4.01e-01 -8.42e-01 -4.76e+02]
7 [ 3.64e-03 -2.02e-04 3.12e-04 1.40e-02 -8.42e-01 1.77e+00 9.99e+02]
8 [ 2.10e+00 -1.25e-01 1.78e-01 8.02e+00 -4.76e+02 9.99e+02 5.65e+05]]

```

The rows (from top to bottom) correspond to “xc”, “yc”, “d”, “E”, “alpha”, “beta”, and “rotation”. The columns (from left to right) also correspond to “xc”, “yc”, “d”, “E”, “alpha”, “beta”, and “rotation”. I think that the square root of the diagonal elements of the covariance matrix are supposed to correspond to uncertainties, but I do not know enough about the minimization algorithm to be really comfortable saying that these are the true uncertainties in the fit parameters. Your mileage may vary. Anyway, I print out the root of the diagonals. The printout by the program

Listing 11.4: Display of the root of the diagonals

```

1 Root of the diagonal of the covariance matrix ...
2 xc: 0.0307145820046
3 yc: 0.0341970790239
4 d: 0.0143735880013
5 E: 0.97393322373
6 alpha: 0.633295666676
7 beta: 1.32940880588
8 rotation: 751.595873785

```

If you do not like the guess for the calibration parameters, you can always undo to the previous calibration values before the fit using the “Previous Values” input.

11.4 The “Number of Chi?” and “Stddev?” Input

The calibration algorithm requires starting at the center and moving across the image in constant χ slices (see section 11.1 or figure 11.2 for a graphical representation). The number of these slices around the image that should be done is user selectable using the “Number Of Chi?” input. The default value is 360. The more χ slices that are used, the slower the fit will be.

Section 11.2 describes how the program uses a parameter to determine how picky it should be in allowing peaks that it finds. Roughly, this parameter corresponds to how many times larger the peak has to be than the background noise outside of the peak. This parameter can be set using the “Stddev?” input. The default value is 5. The higher the value, the less likely the program will be to find and use bad peaks but the more likely it will be to ignore valid good peaks.

11.5 Work in λ

Often times, one wishes to deal with the wavelength of the incoming beam of light instead of the energy of the beam. Of course, the energy and wavelength are intimately related by the formula

$$E = hc/\lambda \quad (11.2)$$

If you wish to work with wavelength in units of nanometers instead of energy in units of electron volts, you can change the state of the program so that the program works with wavelength instead of energy. To do so, you have to go into the menu bar and change the radio select from “Work in eV” to “Work in Lambda”. Once you do that, the calibration parameter input will be labeled λ . Any number in that input will then be converted. After the parameter is modified during a fit, the program will put the wavelength value into the input. Finally, when the calibration parameters are saved to a file, the wavelength will be saved to the file instead of the energy.

11.6 Fixing Calibration Parameters

When fitting calibration parameters, it is not always desirable to allow the program to vary all of the calibration parameters. For example, the energy of the beam used during the diffraction experiment might be already very well known already so there would be no reason to calibrate the energy. If you wish to fix any of the calibration parameters values so that it does not vary during a calibration fit, you can use the check boxes under the “Fixed?” label to fix the parameter. When the corresponding check box is checked, the parameter will not vary during the fit. When it is not checked, the parameter will vary during the fit. You can not fix the pixel length and pixel height because they are always held fixed. This is because these are never the short of thing that one would want to vary. They are some property of the detector that is known in advance.

11.7 Displaying Constant Q Lines

After the program has been given a diffraction file, a list of the constant Q lines, and some calibration parameters, the program has a very useful feature where it can display on top of the image the diffraction pattern that should show up for the particular Q lines and the particular calibration parameters.

The “Draw Q Lines?” button on the “Calibration” tab enables this. Figure 11.3 shows what the diffraction image looks like with the Q lines drawn on it.

Drawing these lines is actually very easy. For each Q value, the program picks a lot of χ values. We know that each of the Q , χ values is in the constant Q line so we can use the calibration parameters to convert them to x , y values and connect all the pixel coordinates to make the line.

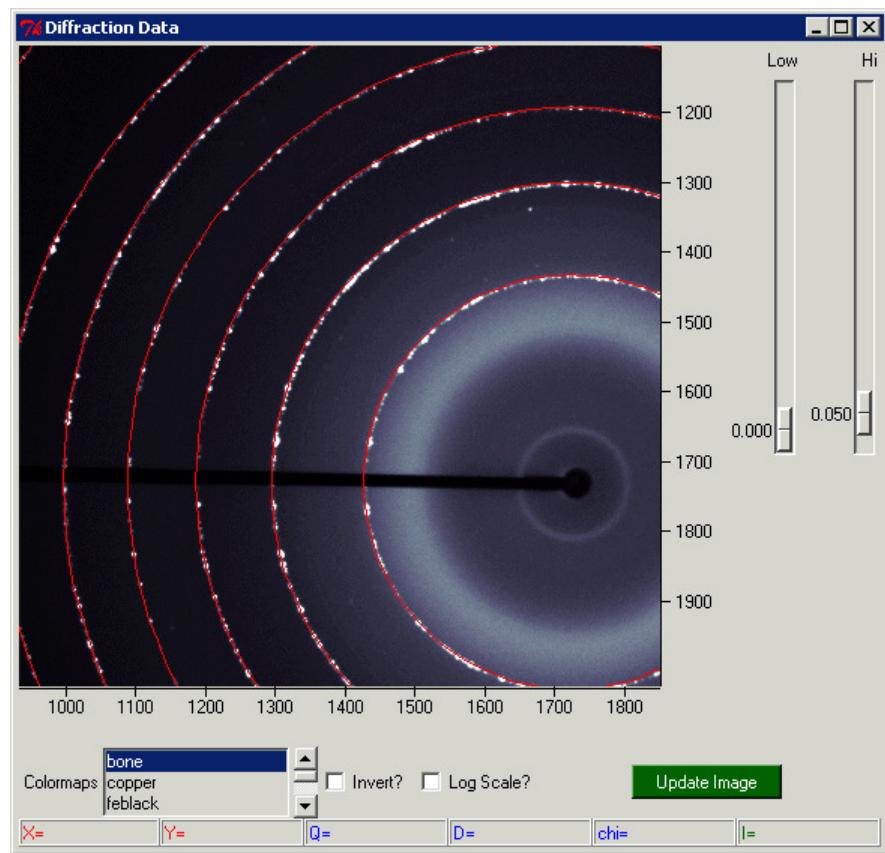


Figure 11.3: A diffraction image with constant the Q lines displayed on it.

Constant Q lines can also be drawn on top of the caked data. This is described in section 13.4. The color of the Q lines can be changed using the “Color” button next to the “Draw Q Lines?” button.

11.8 Displaying Constant ΔQ Lines

The program needs in addition to the Q values a range in Q to find the peaks. See section 11.2 for more details. Because the program has this range, it can also display the ΔQ range on top of the image. This can be done with the “Draw dQ Lines?” button and the color of these lines can be changed with the corresponding “Color” button. Figure 11.4 shows what the diffraction image looks like with the ΔQ lines drawn on it.

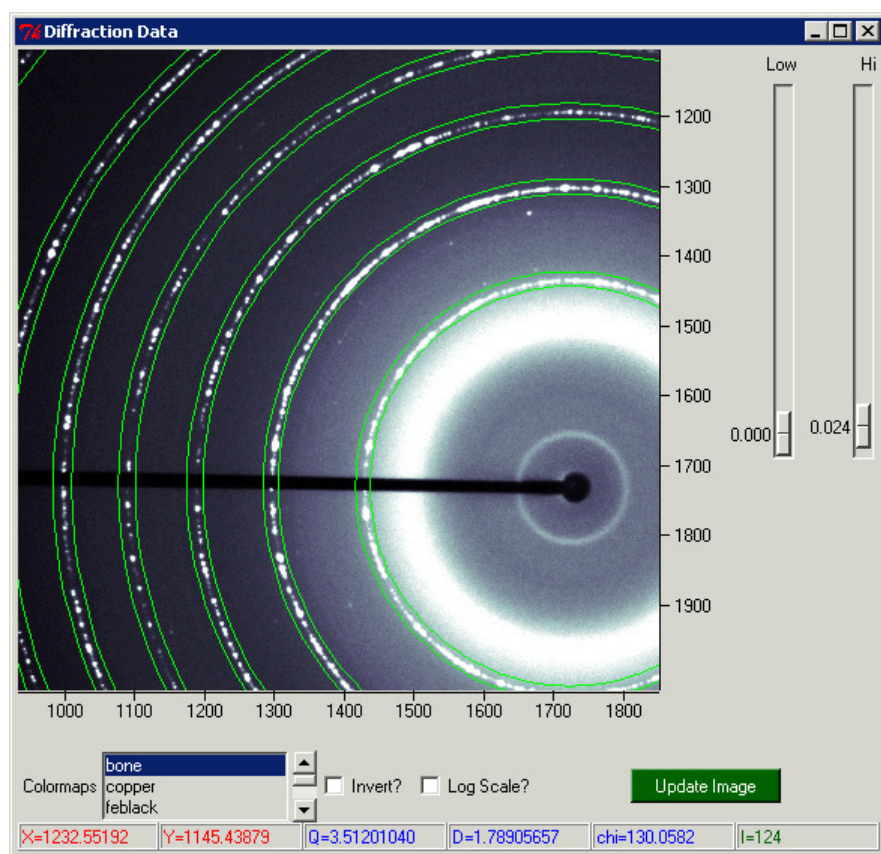


Figure 11.4: A diffraction image with the constant ΔQ lines displayed upon it.

Constant ΔQ lines can also be drawn on top of the caked data. This is described in section 13.4.

11.9 Displaying Peaks

Section 11.2 describes how the program has to find a bunch of peaks on the diffraction image in order to perform the calibration. After the program has found all the peaks, it can

conveniently display them on top of the diffraction image. This can be done with the “**Draw Peaks?**”. The peaks will be displayed as crosses and the color of the peaks can be changed with the corresponding “**Color**” button. Figure 11.5 shows what the diffraction image looks like with the peaks drawn on it. This feature is useful because you can use it to see if the program is actually finding real peaks corresponding to diffraction maxima. If many of the peaks that the program finds do not correspond to diffraction maxima, it is less likely that the program would do a good job calibrating the diffraction image.

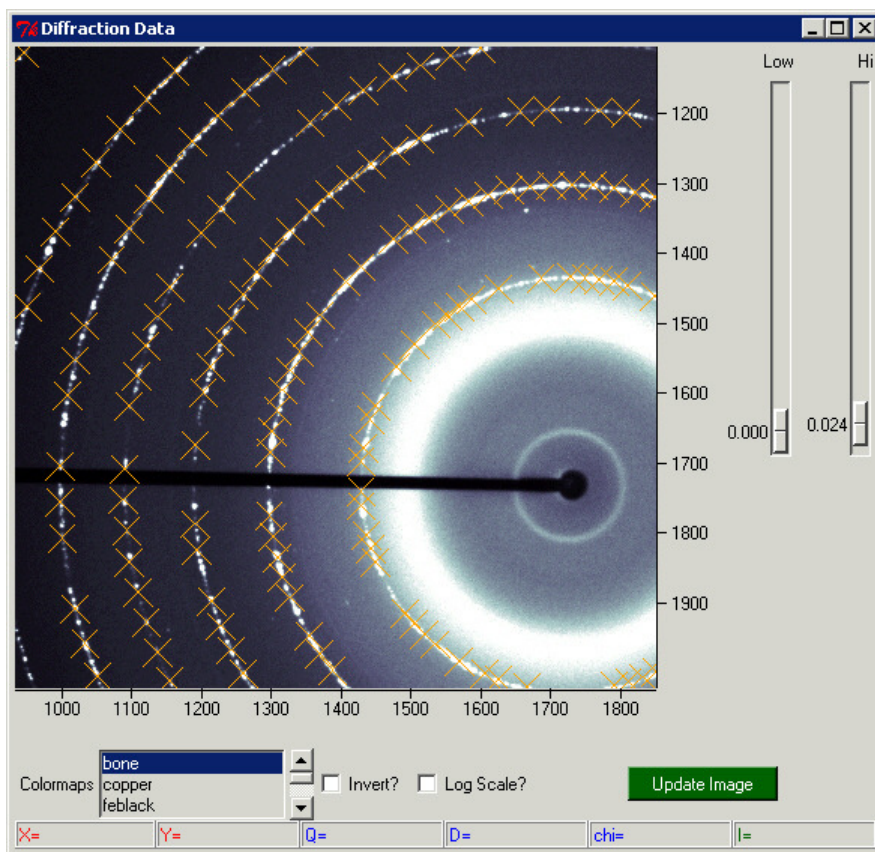


Figure 11.5: A diffraction image with the peaks displayed upon it.

Peaks can also be drawn on top of the caked data. This is described in section 13.4.

11.10 Masking Peaks

The general idea behind masking peaks is allow polygon masks (see chapter 12) to be used as a way to forbid the program from using any peaks found within a certain region. So if a polygon mask covers a certain area of the image, none of the peaks found within that area will be used while calibrating. Also, none of the peaks will be displayed on top of the diffraction image or cake image. An example of this is shown in figure 11.6a. See section 11.9 for a discussion of displaying peaks on a diffraction image. Figure 11.6b shows the same effect on top of the corresponding caked plot. See chapter 13 for a discussion of caking. In particular,

see section 13.5 for more information on displaying peaks on a caked image. This feature was added in version 2.0.0.

11.11 Saving the Peak List

The program has a feature where it can generate a list of diffraction peaks that it finds the diffraction image (just like when it is calibrating) but then instead of calibrating the image, the program saves out all of the peaks to a data file. This can be useful, for example, if you need a list of pixel coordiantes where diffraction peaks are for some further data analysis. The “Make/Save Peak List” button can be used to save out the peak list. Just as in calibrating, the program requires in advance for a diffraction file to be loaded, for a standard Q file to be loaded, and for a guess at the calibration parameters to be in the inputs.

A typical peak list file looks like⁴

Listing 11.5: A Peak List File, basicstyle=

```

1 # A list of peaks found in the diffraction image.
2 # Calculated on Sun Apr 6 18:06:56 2008
3 # Calibration data used to find peaks:
4 #   x center:      1725.0000000 pixels
5 #   y center:      1725.0000000 pixels
6 #   distance:      122.5040000 mm
7 #   energy:        12714.2388941 eV
8 #   alpha:         0.0000000 degrees
9 #   beta:          0.0000000 degrees
10 #  rotation:       0.0000000 degrees
11 #  pixel length:    100.0000000 microns
12 #  pixel height:    100.0000000 microns
13 #  x y RealQ FitQ chi width intensity 2theta
14 2016.15 1724.44 1.511 1.50 0.11 0.0075 5564.32 13.36
15 2016.68 1719.33 1.51 1.50 1.11 0.0093 1662.72 13.39

```

First, the file contains the calibration parameters used to generate the peaks. Then it has a comment string describing each of the numbers in each of the rows that follow. Each row corresponds to a unique peak. The first two numbers x and y are the x and y pixel coordinate corresponding to a location in the diffraction image of the peak. RealQ is the Q value found in the Q list that is already known. FitQ is the Q value calculated from the (x, y) coordinate using the calibration parameters. χ is also calculated from the pixel coordinate using the calibration parameters. Intensity is intensity value found in the data at this peak. 2θ is calculated at the (x, y) coordinate using the calibration parameters.

⁴I have modified what a real file looks like a bit. The numbers are really tab separated but I show them space separated for brevity.

11.12 Handling Calibration Data

There are inputs in the calibration tab of the program for input of the calibration parameters. “xc” is for the x center, “yc” is for the y center, “d” is for the distance, “E” (or “ λ :”) is for the energy or wavelength. The α , β , and R inputs are for the three angles. “pl” stands for the pixel length and “ph” stands for the pixel height.

You can directly input calibration data using the inputs and once the data is in the inputs it can be used by the program to do the calibration (or the caking or anything else).

But there are a couple of other ways to deal with calibration data. You can load and save calibration program from the program using the “Load From File” and “Save To File” buttons. This is nice because it can be used, for example, to save the data that was found by calibration data for future reference. As you will see, the calibration data files can handle information about whether the parameters should be fixed (see section 11.6).

The format for a calibration data file is pretty simple. Below is an example

Listing 11.6: Calibration Parameters

1	#	Calibration File		
2	xc	1725.000000	0	
3	yc	1725.000000	0	
4	D	125.296000	0	
5	E	12735.395772	0	
6	alpha	0.000000	0	
7	beta	0.000000	0	
8	rotation	0.000000		0
9	pixelLength	100.000000		
10	pixelHeight	100.000000		

Comment lines beginning with a # and are ignored. Each of the parameters gets its own line. Each parameter name is followed by some spaces or tabs and then the value. The value can be followed by an optional second number which is either zero or one. The second number corresponds to whether or not the parameter should be fixed while fitting. One means fix the parameter. Zero means let it vary. If no number is given, the default is to not fix the parameter.

Instead of energy, the wavelength of the incoming beam of light can be stored in a calibration file. The wavelength line would look like “wavelength 0.973540”. When the program is in wavelength mode, the program will save out calibration parameters with this line instead of the one above. The program will load in a file containing either no matter what mode the program is in. It will do the conversion if it has to do put the right value into the input. See section 11.5.

11.13 Handling Q Data

Q data is always loaded into the program from files. Q data can be loaded into the program with the “Q Data:” input on the calibration tab. You can either type in the filename of the

Q file by hand and push the load button or click on the folder icon to the side and use the file selector to pick the file that you want.

The Q data file format is pretty simple. Below is an example

Listing 11.7: Lanthanum Hexaboride.dat

```

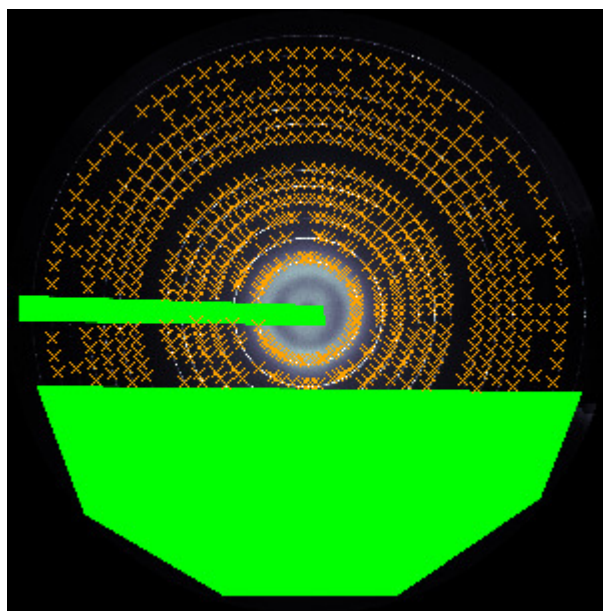
1  # This is Q Data for Lanthanum Hexaboride
2  Q    dQ
3  1.511543809 .05
4  2.137646823 .05
5  2.618102966 .05
6  3.023087619 .05
7  3.379873753 .05
8  3.702525225 .05
9  ...

```

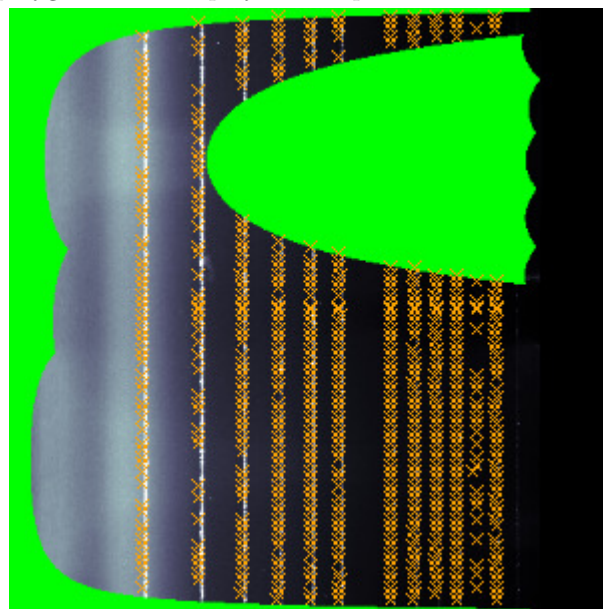
Comment lines beginning with a $\#$ and are always ignored. The first line in the file should be of the form " Q d Q " or " Q delta Q " to specify that this is a list of Q values. The rest of the file should have Q values followed a ΔQ range. All Q values must be larger than 0. None of the Q ranges can overlap. Instead of inputting Q values, the program can input D values if the first line is instead " D d D " or " D delta D " The values should be given instead in D space and the values will be converted using 10.21.

11.14 The “Get From Header?” Input

Often times guesses at the experimental parameters are stored in the header data inside of the diffraction image. The program can try to find these header calibration parameters and put them into the calibration parameters inputs in the program. This can be done with the “Get From Header” button.



(a) A diffraction image with diffraction peaks and two polygon masks displayed on top of it.



(b) A caked plot with diffracton peaks and two polygon masks displayed on top of it.

Figure 11.6: Polygon masks can be used to block out certain regions of the image. Whenever a polygon mask is loaded into an image, none of the peaks found in the mask's region will be used while calibrating the image. Furthermore, none of the peaks within the masks will be displayed on the diffraction image or caked plot.

Chapter 12

Pixel Masking

When analyzing diffraction data, not all of the pixels in an image should be used in the analysis. In order to make the program ignore certain pixels when doing the analysis, this program allows for two types of pixel masking: threshold masking and polygon masking. You can apply either of these from the “**Masking**” tab. figure 12.1 shows this tab.

12.1 Threshold Masking

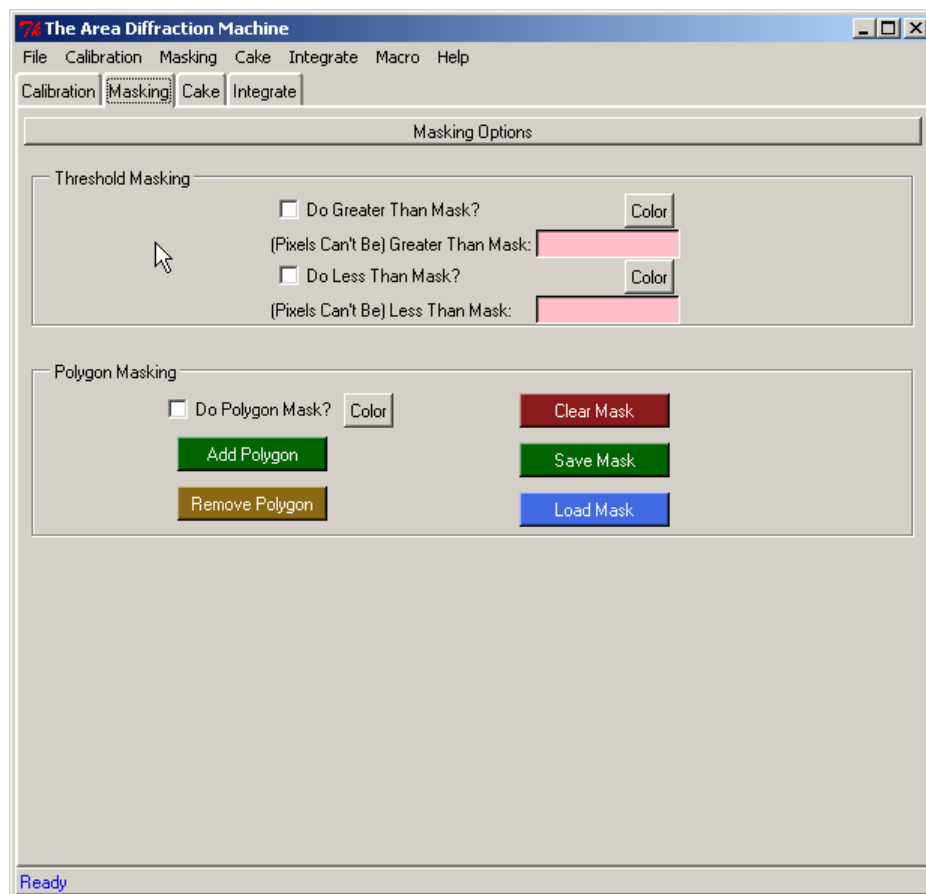
The top half of the “**Masking**” tab is devoted to threshold masking. Threshold masking allows all pixels, either above a certain intensity or below a certain intensity, to be ignored when doing the diffraction analysis. The “**Do Greater Than Mask?**” check box can be used to apply a mask that will cause all pixels greater than a certain value to be ignored. The “**(Pixel’s Can’t Be) Greater Than Mask**” input can be used to specify the maximum pixel value. Correspondingly, the “**Do Less Than Mask**” check box can be used to make the program ignore all pixels below a certain value. The particular value can be specified with the “**Less Than Mask**” input.

When you apply a threshold mask, the pixels over this threshold will all be colored differently on the diffraction and cake image. You can specify what you want these masked to be colored with the “**Color**” button next to the greater than and less than masks. Figure 12.2 shows what a diffraction image looks like when all pixels with intensity above 5000 are colored green and all pixels below 30 are colored red.

When caked data is saved out to a file, any of the pixels that are larger than the greater than mask are saved as -2. Any of the pixels smaller than the less than mask are saved as -3. If you need to analyze caked data outside the program, this behaviour needs to be accounted for.

When an intensity integration is saved to a file, any of the too high or too low pixels are simply ignored when calculating average intensity.

Figure 12.1: The pixel masking tab. It allows for threshold masking and polygon masking.



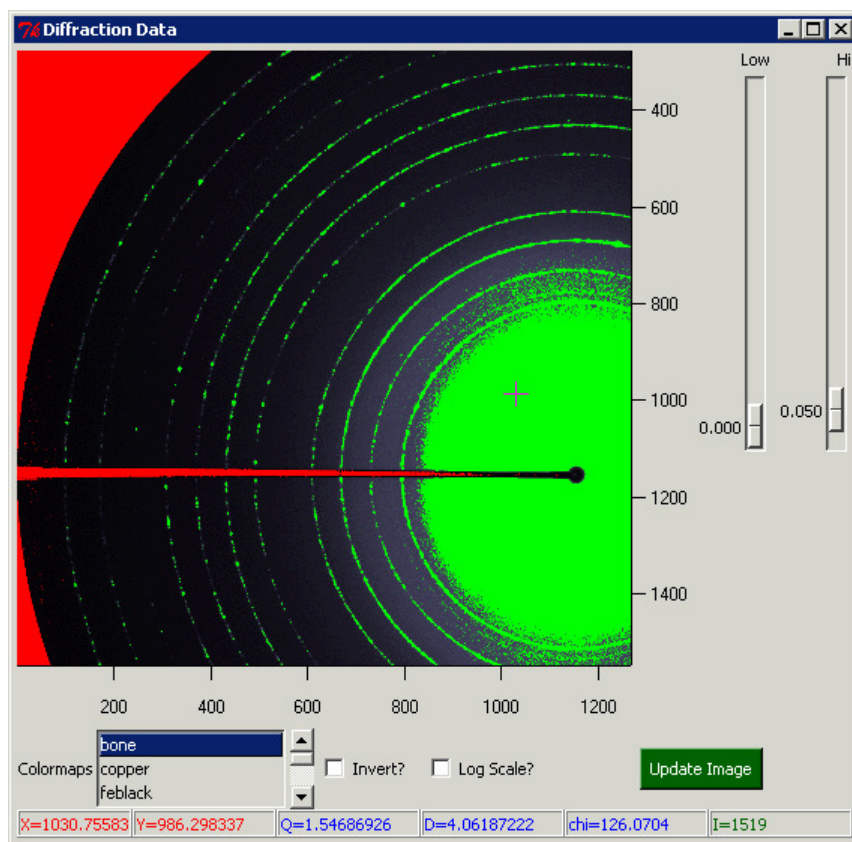


Figure 12.2: A diffraction image with a greater than mask and less than mask. All pixels with intensity greater than 5000 have been colored green. All pixels with intensity less than 30 have been colored red. Applying an intensity mask can be a useful way to see if a detector's pixels have been overloaded. They can also be a used to ensure that no overloaded pixels are used in subsequent data analysis.

12.2 Polygon Masking

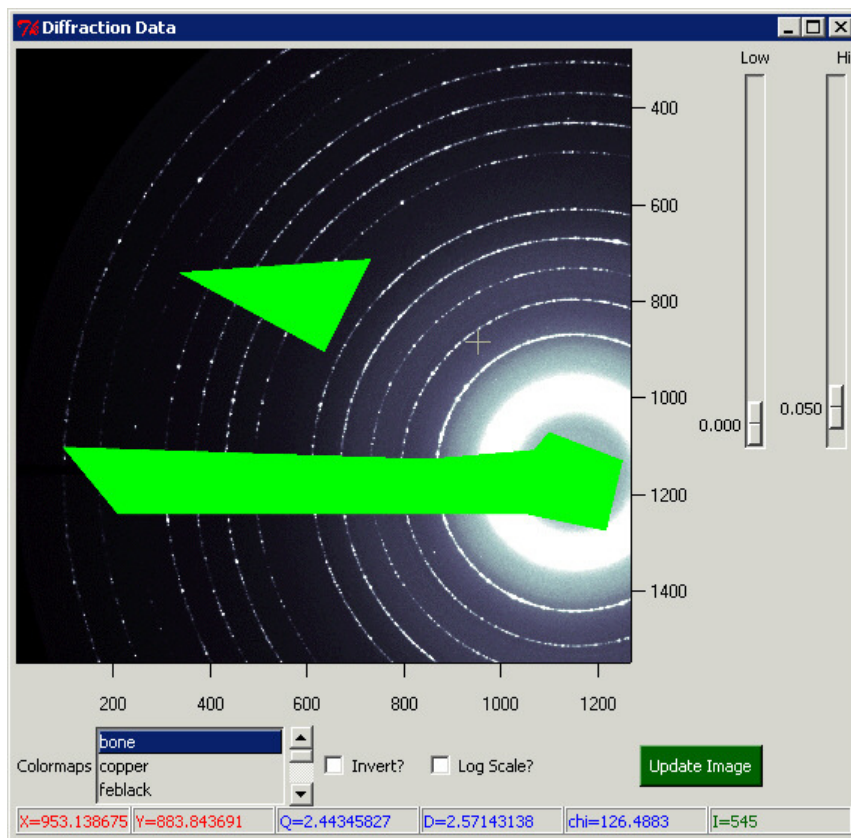


Figure 12.3: Here are two polygon masks that have been applied to a diffraction image. One of them blocks the beam stop.

Sometimes, large areas of a diffraction image should not be included in any data analysis. For example, often a beam stop blocks part of the detector and the pixels behind the beam stop should be ignored. To allow for this sort of masking, the program has a polygon masking feature. Polygons can be drawn around certain parts of the diffraction image and those parts of the image will not be used in any subsequent analysis. This program can handle multiple polygons at the same time.

So long as the “Do Polygon Mask?” check box is selected, the polygon masks will be used when performing subsequent analysis. The polygons will be displayed on the diffraction and cake image. Any pixel in the diffraction or cake image that is inside one of the polygons will have a different color. An example of polygons on a diffraction image are shown in figure 12.3. The color of the polygon masks can be changed using the “Color” button next to the “Do Polygon Mask?” check box. When caked data is saved out, any pixels inside polygon masks will be given an intensity value of -4. During an intensity integration masked pixels will be ignored.

A polygon mask can be added to the image by pushing the “Add Polygon” button on the “Masking” tab. This button will stay down when pushed. Pushing it puts the program in polygon drawing mode. In this mode, the diffraction image will behave differently. The

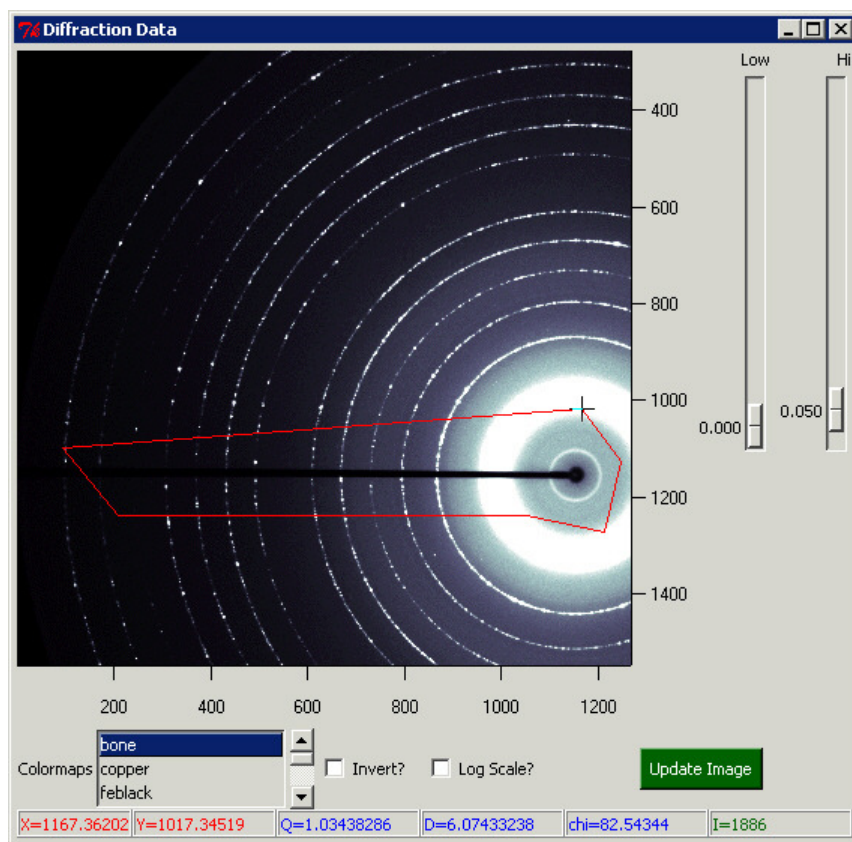
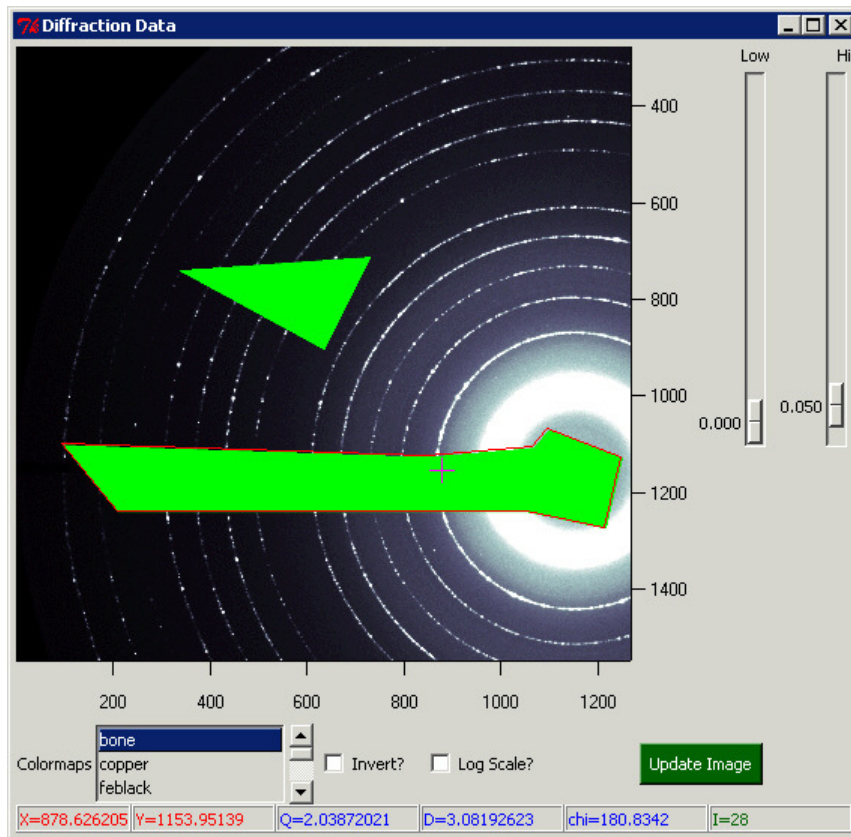


Figure 12.4: Here is the interface for adding a new polygon mask to the program. This particular mask will cover the beam stop so that the beam stop does not affect the intensity integration.

diffraction image can no longer be zoomed or panned. Instead, left clicking on the diffraction image will make the program draw the polygon. The first left click adds the first vertex. Each success left click add another vertex. The drawing can be finished by right clicking (this will also create a final vertex). Right clicking will make the program exit the drawing mode, return to its original state, and add the polygon into the program. Multiple polygons can be added using the “Add Polygon” button. Figure 12.4 shows the program when a polygon is being drawn. Drawing a polygon can be aborted without saving the mask by unpushing the “Add Polygon” button.

Figure 12.5: Here is the diffraction image window as a polygon is about to be removed. When mousing over a polygon to remove it, the program will display a red border around it.



The “Remove Polygon” button can be used to remove a polygon in the program. Like the “Add Polygon” button, this button will stay pushed and change the behavior of the diffraction image. After the “Remove Polygon” button is pushed, clicking over a particular polygon will remove it. After the polygon is removed, the program will return to its normal state. Figure 12.5 shows what the diffraction window looks like when a polygon is about to be removed. The program can be returned to its normal state without removing a polygon by unpushing the “Remove Polygon” button.

The “Clear Mask” button can be used to remove all the polygons at once. The “Save Mask” button can be used to save all the polygons to a file. A file of polygons can be added to the program using the “Load Mask” button. The file for polygon files is very simple. For the polygons in figure 12.3, the following file would be saved:

Listing 12.1: 'polygons.dat'

```

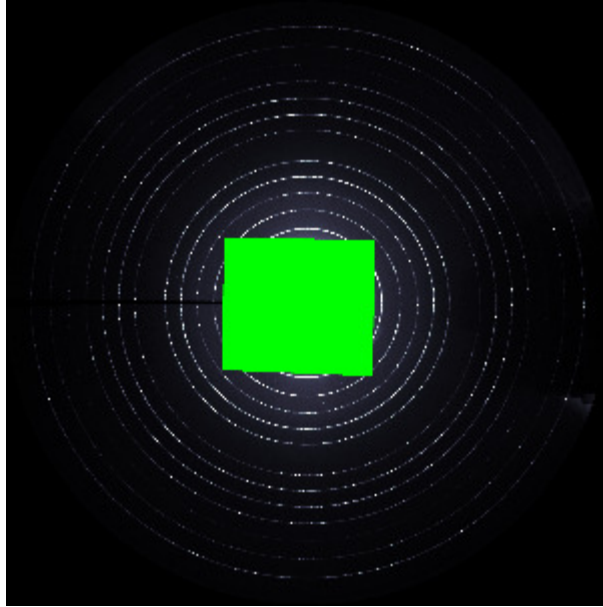
1  # Polygon(s) drawn on Thu Feb 07 00:00:21 2008
2  93.140587183      1098.06704199
3  208.013978042     1237.77792276
4  1052.48863517     1237.77792276
5  1213.93231962     1271.92947139
6  1248.08386825     1126.00921814
7  1095.95424252     1067.02017959
8  1064.90738013     1104.27641447
9  847.579343365     1122.9045319
10
11 332.201427619      737.923438212
12 633.355992844      902.471808902
13 729.601266267      709.981262058

```

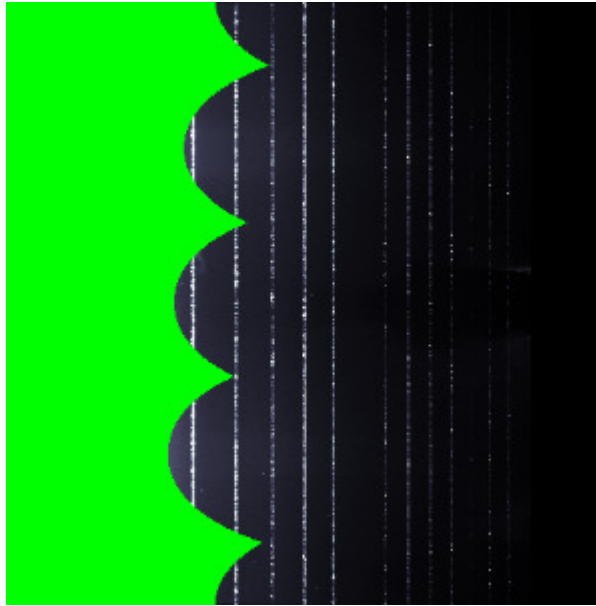
Each line is an (x,y) coordinate for one of the nodes of a polygon. The coordinates are separated by spaces. Each polygon is separated by a newline. Comment lines beginning with `#` are ignored.

12.3 Masking Caked Plots

Any polygon mask or threshold mask will also show up on the caked plot. Polygons on the diffraction image can look very distorted on caked plots. Figure 12.6 shows an example.



(a) A rectangular polygon mask in the middle of a diffraction image



(b) The same rectangular mask on a caked plot

Figure 12.6: An example of how a relatively simple shape on a diffraction image will can look very different on a caked plot.

Chapter 13

Caking

13.1 The Caking Algorithm

A caked plot is like a radial (r vs θ) plot of the diffraction data as it would appear if it were captured on an untitled detector. A radial plot will make circles of constant r become straight lines. Caked plots are important because diffraction peaks will also be straight lines. A caked plot is actually a plot of Q vs χ . Equation 10.19 shows that Q is related by the sin function to 2θ and 2θ is just the scattering angle of the diffraction peak. From equation 10.17, 2θ is related to the radius r by a tangent function. Although the relationship is not linear, Q increase as r increases and therefore Q is a similar quantity to r . χ corresponds to the angle radially around the center of the image. So a cake plot of Q and χ is really analogous to a radial plot.

Cakes plots are calculated with the following algorithm. The program must first bin Q and χ space. The user can specify the bin range and bin size with inputs. Alternately, the code can try to pick a range that is large enough to encompass the whole region. Once the bin size is specified, the program has to fill each bin an intensity value. Since each bin has some particular Q and χ value¹ we can calculate the corresponding (x''', y''') pixel coordinate for this Q and χ value using equation 10.15 and 10.16. The intensity value for the pixel coordinate x''' and y''' is the intensity that should be put in the bin. (x''', y''') is generally not a whole number so a bilinear interpolation of the intensity around this coordinate is used to get a best estimate.

In principle, the caking algorithm could be implemented differently. The algorithm currently runs a loop over each bin. One could alternately loop over all the pixels of diffraction data. Each pixel has a particular (x''', y''') coordinate. Equations 10.9 and 10.11 could be used to calculate the Q and χ value for each pixel in the image, and each pixel could be put into its corresponding bin. After doing this for all the pixels, we could average the intensity in all the bins. This implementation does not necessarily put an intensity value into all the bins. This could be overcome by applying the previous algorithm only to the bins for which

¹Technically, each bin has a Q and χ range. We will take the middle of the bin to be the particular Q and χ value for the bin.

nothing was added. This method would in some ways be more accurate because each of the pixels in the diffraction image would be used in the analysis whereas they are aren't all used in the above algorithm. But the biggest downside of this alternate algorithm is that it is substantially slower because there are usually significantly more pixels in the diffraction image than bins used in a cake. For example, mar3450 data holds 3450×3450 pixels while cakes typically have a resolution of 1000×1000 . This alternative algorithm was not implemented for this reason.

Caked data can be masked with pixel masks. Whenever the program finds an intensity value that should be masked (either because it is too large, too small, or in a polygon mask), it fills in that part of the caked array with a particular negative value. When the caked data is displayed, these negative values are given special colors.

The program can perform a polarization correction of the caked data. The polarization correction formula is

$$I = Im/PF \quad (13.1)$$

$$PF = P(1 - (\sin(2\theta) \sin(\chi - 90))^2) + (1 - P)(1 - (\sin(2\theta) \cos(\chi - 90))^2) \quad (13.2)$$

with Im the measured intensity. The 2θ and χ values correspond to the particular value that is being corrected. All pixels have their intensity corrected by this formula before they are put into a cake bin.

13.2 Caking with the Program

Figure 13.1 shows the “Caking” tab. This is where caking is done. The program can only cake data after one or more diffraction files has been loaded into the program and after calibration values for the particular diffraction image are loaded. In order to cake, this program needs to know a range in Q and χ space that should be caked. This can be inputted with the “Q Lower?”, “Q Upper?”, “Chi Lower?”, and “Chi Upper?” inputs. The program will also need to know how many Q and χ bins to create when caking data. This can be inputted with the “Number of Q?” and “Number of Chi?” inputs. Once this is done, the “Do Cake” button will cake the data.

After the cake finishes, the program will open a cake data window which displays the cake data interactively. The cake data window acts just like the diffraction data window so everything in Chapter 9 carries over. The only real difference is that whenever the caked data is zoomed into, the program will take the selected zoom range and put it into the inputs on the cake tab and the recake the image. The caked data can be taken to the previous zoom level either by right clicking on the caked plot or by pushing the “Last Cake” button

13.3 AutoCake

The program has a convenience button “AutoCake”. “AutoCake” will guess a good range of Q and χ values, put them into the input, and then push the “Do Cake” button automatically.

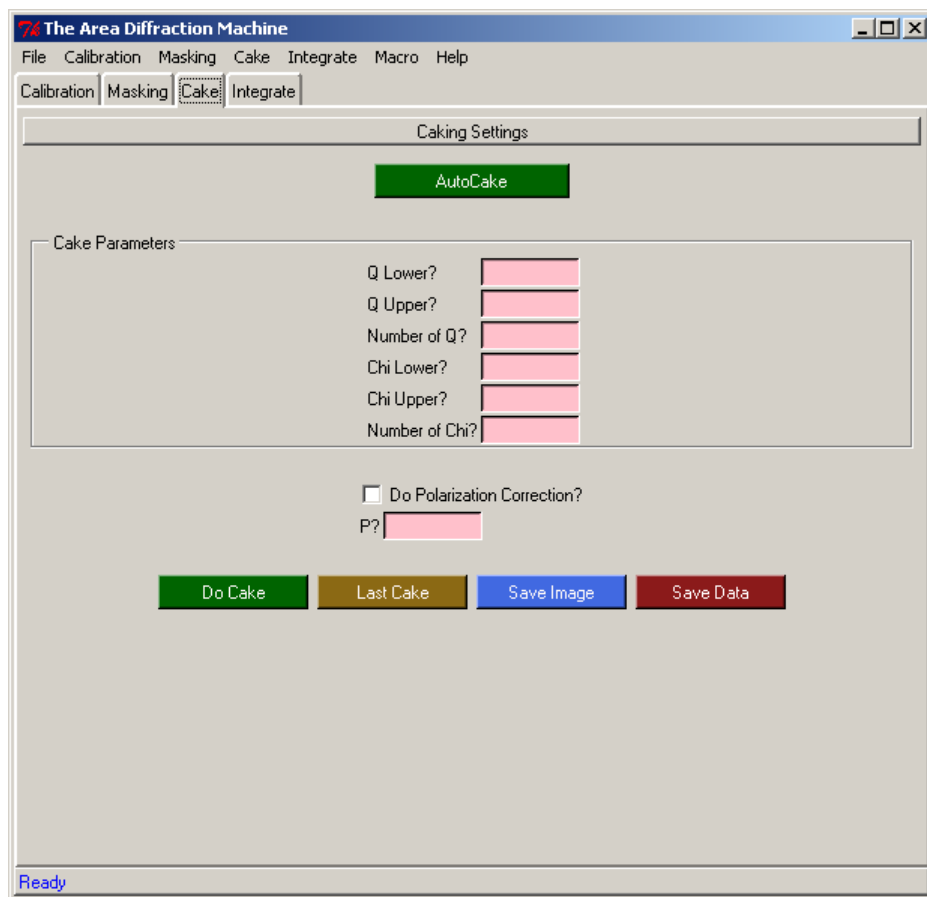
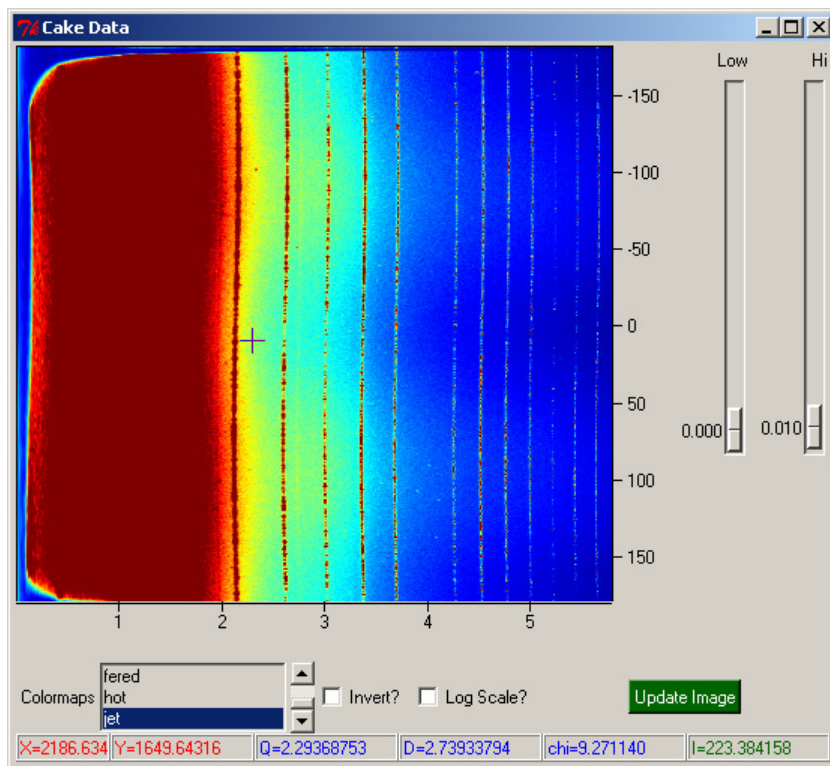


Figure 13.1: The caking tab of the program. This is where caking is done.

Figure 13.2: The cake data window for the program. This window will open up after the data is caked. This window behaves exactly like the diffraction data window.



This will create a cake without much work. The program will pick a range that puts every pixel from the diffraction image into the cake. It will pick a bins sizes so that each pixel of the displayed cake data will correspond to one bin. This will ensure that the cake looks as sharp as the computer can draw it. After the display is resized, the number of bins will change correspondingly. The next time “AutoCake” is pushed, the cake window will again look sharp.

13.4 Displaying Q and ΔQ Lines

If a Q list has been loaded into the program, constant Q lines or ΔQ lines can be displayed on top of the cake data. Remember that constant Q lines on the diffraction image are straight vertical lines on the caked plot. The program will display constant Q lines or ΔQ lines on the caked plot whenever they should be displayed on the diffraction image. See section 11.7 and section 11.8 for a discussion of displaying constant Q lines on diffraction data. Figure 13.3 shows constant Q lines displayed on a caked plot and figure 13.4 shows constant ΔQ lines displayed on a caked plot.

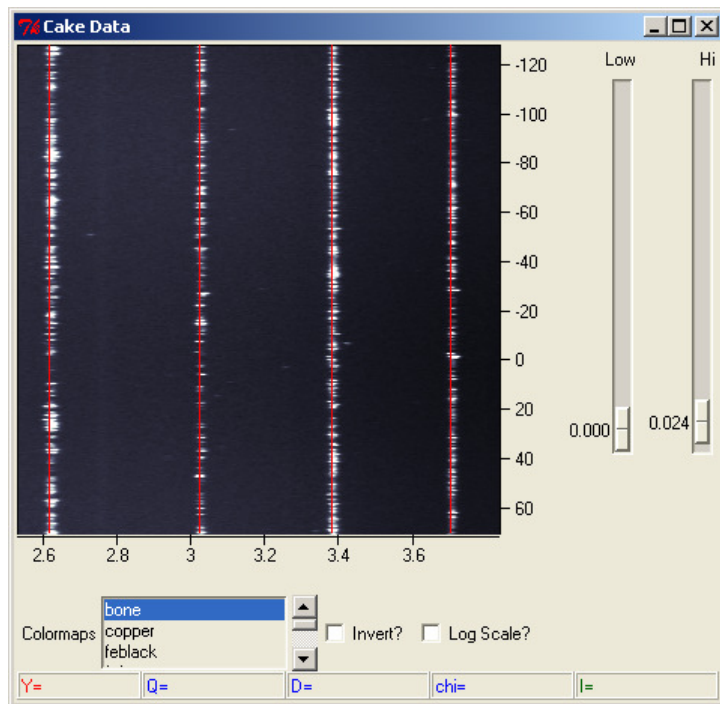


Figure 13.3: The caked data window with constant Q lines drawn on top of it.

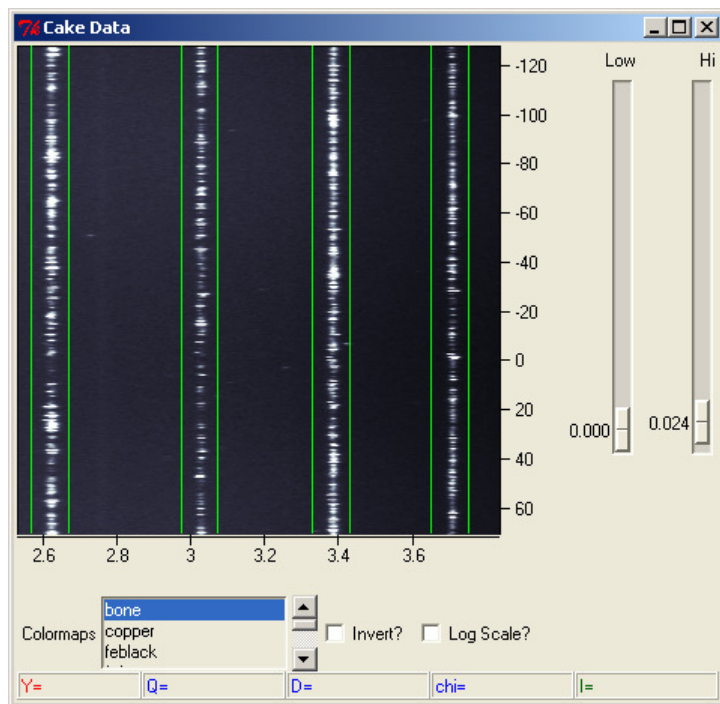


Figure 13.4: The caked data window with constant ΔQ lines drawn on top of it.

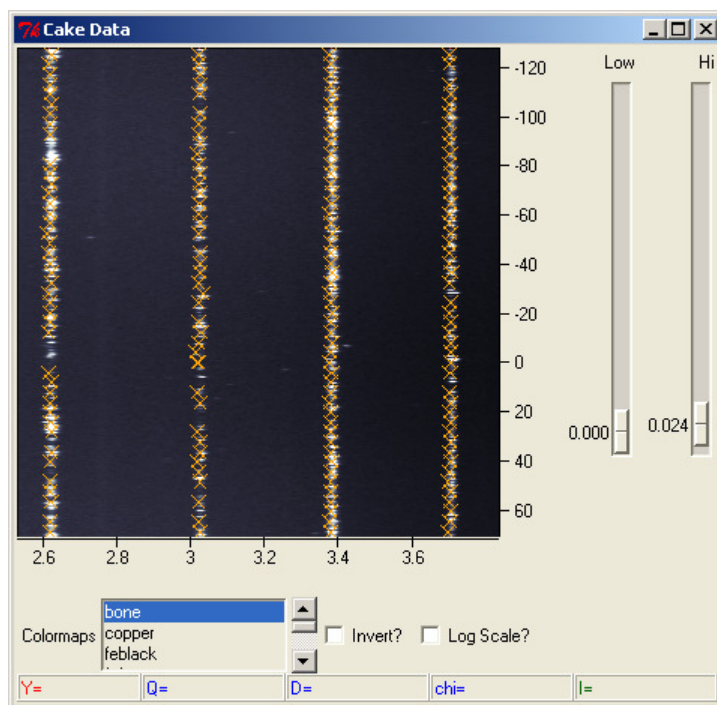


Figure 13.5: The caked data window with diffraction peaks drawn on top of it.

13.5 Displaying Peaks

Any peaks that the program finds when performing a calibration can be displayed on top of the caked data. The peaks will be displayed as crosses. Figure 13.5 shows peaks displayed on a caked plot. Peaks will be displayed on the caked plot whenever they should be displayed on the diffraction image. See section 11.9 for a discussion of displaying peaks on diffraction data. Being able to display Q lines and peaks can be very useful for checking if a calibration was done properly. Figure 13.6 illustrates this principle.

13.6 Polarization Correction

The program can apply a polarization correction to the cake. The “Do Polarization Correction?” check box can be used to apply a polarization and the polarization value can be set with the “P?” input.

13.7 Working in 2θ

Caked plots can have 2θ instead of Q as one of the axis. This can be done by changing the program to 2θ mode by doing into the file menu and selecting the “Work in 2theta” option. When this is selected, all the names in the program will change from Q to 2θ . For example, the program will have “ 2θ Lower”, “ 2θ Upper”, “Number of 2θ ”. The program will display the cake image with 2θ as its axis. The “Work in Q ” option in the file menu

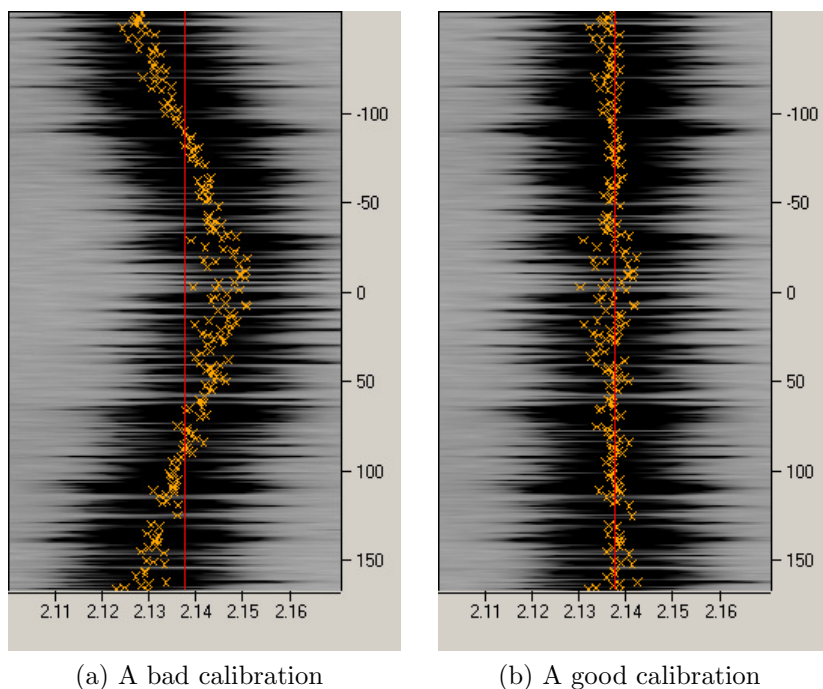


Figure 13.6: displaying peaks and constant Q lines on top of the caked data can be used to tell if the data is properly calibrated. If the calibration is good, all the peaks will cluster very close to a particular value of Q line and there will be no systematic variation of the diffraction peak. If the calibration is bad, the diffraction peaks will have a systematic distortion around some value of Q . This can be used to see if the program is properly calibrating the data.

can be used to return the program to caking with Q as one of the axis. This feature was introduces in version 2.0.0 of the program.

13.8 Saving Cake Images

You can save caked data out as one of many popular image formats. The program can save caked images as “jpg”, “gif”, “eps”, “pdf”, “bmp”, “png”, or “tiff”. When caked data is saved as an image, it will be saved out with whatever threshold masks, polygon masks, Q lines, ΔQ lines, and peaks were displayed over the caked data in the program.

13.9 Saving Cake Data

Caked data can also be saved as a plain text data file. This can be done by pushing the “Save Data” button and selecting a destination. The format for caked files is just a long comment string followed by the data as rows of numbers. Here is an example:

Listing 13.1: ‘caked_data.dat’

```

1 # Cake of: N:/data/LaB6_14_02_56.mar3450
2 # Data Caked on Wed Mar 12 21:30:55 2008
3 # Calibration data used to make the cake:
4 #   x center:      1725.0000000 pixels
5 #   y center:      1725.0000000 pixels
6 #   distance:      125.2960000 mm
7 #   energy:        12735.3957721 eV
8 #   alpha:         0.0000000 degrees
9 #   beta:          0.0000000 degrees
10 #  rotation:       0.0000000 degrees
11 #  pixel length:    100.0000000 microns
12 #  pixel height:    100.0000000 microns
13 # A Polarization correction was applied
14 #   P = 0.500000
15 # A greater than mask was applied
16 #   Greater than mask = 1000.000000
17 # A Less Than Mask was applied
18 #   Less than mask = 10.000000
19 # Polygon mask(s) were applied
20 # Polygon(s) used in the analysis:
21 #   2400.10912343      1073.5706619
22 #   962.511627907      2282.88014311
23 #   2850.51520572      2572.86762075
24 #
25 #   1573.33631485      1215.47942755
26 #   1820.13416816      2893.70483005
27 #   2906.04472272      1573.33631485
28 # Cake range:
29 #   Q Lower = 0.000000
30 #   Q Upper = 6.726544
31 #   Number of Q = 560.000000
32 #   Q Step = 0.012012
33 #   chi Lower = -180.000000
34 #   chi Upper = 180.000000
35 #   Number of Chi = 560.000000
36 #   chi Step = 0.642857
37 # Note: pixels outside the diffraction image are saved as -1
38 #   Pixels greater than the greater than mask are saved as -2
39 #   Pixels less than the less than mask are saved as -3
40 #   Pixels inside of a polygon masks are saved as -4
41 # chi increased down. Q increases to the right

```

the comment string describes what state the program was in when the cake was done. It first lists the name of the diffraction file(s) that were caked. Next it lists the calibration parameters used when caking the data. Then is the polarization correction, the greater than

mask and the less than mask that were used. It has the pixel coordinates of any polygons that were used when caking. It then lists the range of the cake and the number of bins that were used. The program sets the value of certain bins in the data to special values. Bins that are outside of the diffraction image are saved as -1. Bins that were masked because they were too large are saved as -2. Bins that were masked because they were too small are saved as -3. Bins that were inside a pixel mask are saved as -4. This is written in the comment string.

The program tries to be smart about the comment string. If no masks were used, the comment string instead contains lines like

Listing 13.2: 'Alternate Header'

```

1 # No greater than mask was applied
2 # No less than mask was applied
3 # No polygon masks were applied

```

If the program is working in 2θ mode, the comment string will instead say something like

Listing 13.3: 'Another Alternate Header'

```

1 # 2theta Lower = 0.000000
2 # 2theta Upper = 62.814525
3 # number of 2theta = 560.000000
4 # 2theta Step = 0.112169

```

Then comes the data. As the header describes, each line in the file is of constant χ and contains many numbers separated by spaces. Each column is of constant Q . χ increases down and Q increases to the right. The top left bin corresponds to Q lower and χ upper.

Chapter 14

Intensity Integration

14.1 The Integration Algorithm

An intensity integration is a plot of average intensity as a function either Q , 2θ , or χ . The calibration values for the diffraction data must be known before the integration is done. A range and bin size for the integration must be give. For example, a $Q - I$ integration might have a range from 2 to 5 with 100 bins.

The algorithm for performing the intensity integration is as follows: loop over every pixel in the image. Add its intensity to a bin if it Q , 2θ , or χ value falls within the bin's range. We need to know the calibration values because they are used to calculate Q , 2θ and χ from the pixel's coordinates using using equations 10.9 10.11, 10.18, 10.17, and 10.19. After binning all the pixel, the bins are then averaged.

This program can constrain the integration range. This means that you can perform, for example, a Q integration of only those pixels with some particular χ range. Or, you can constrain your χ integration to a particular Q range. This could be used, for example, to perform a χ integration of only one diffraction peak. The algorithm for performing the constraint isn't different. You just only bin intensity values which are allowed by by the constraint.

The program can perform a polarization correction to the integration. The polarization correction formula is

$$I = Im/PF \quad (14.1)$$

$$PF = P(1 - (\sin(2\theta) \sin(\chi - 90))^2) + (1 - P)(1 - (\sin(2\theta) \cos(\chi - 90))^2) \quad (14.2)$$

with Im the measured intensity. The 2θ and χ values correspond to the particular value that is being corrected. If this option is selected, all pixels have their intensity corrected by this formula before they are binned.

14.2 Integrating with the Program

The program requires one or more diffraction images and calibration parameters to be loaded into the program before an intensity integration can be done. Figure 14.1 shows the “Integrate” tab. This is where integration is done. There are two sets of inputs on the tab. The inputs on the left is titled “Q-I Integration” and can be used for performing Q integration. The “Q Lower?” and “Q Upper?” inputs on the left can be used to specify an integration range in Q . The number of bins in Q space can be specified with the “Number of Q?” input. The “Integrate” button on the left can be used to perform a Q integration.

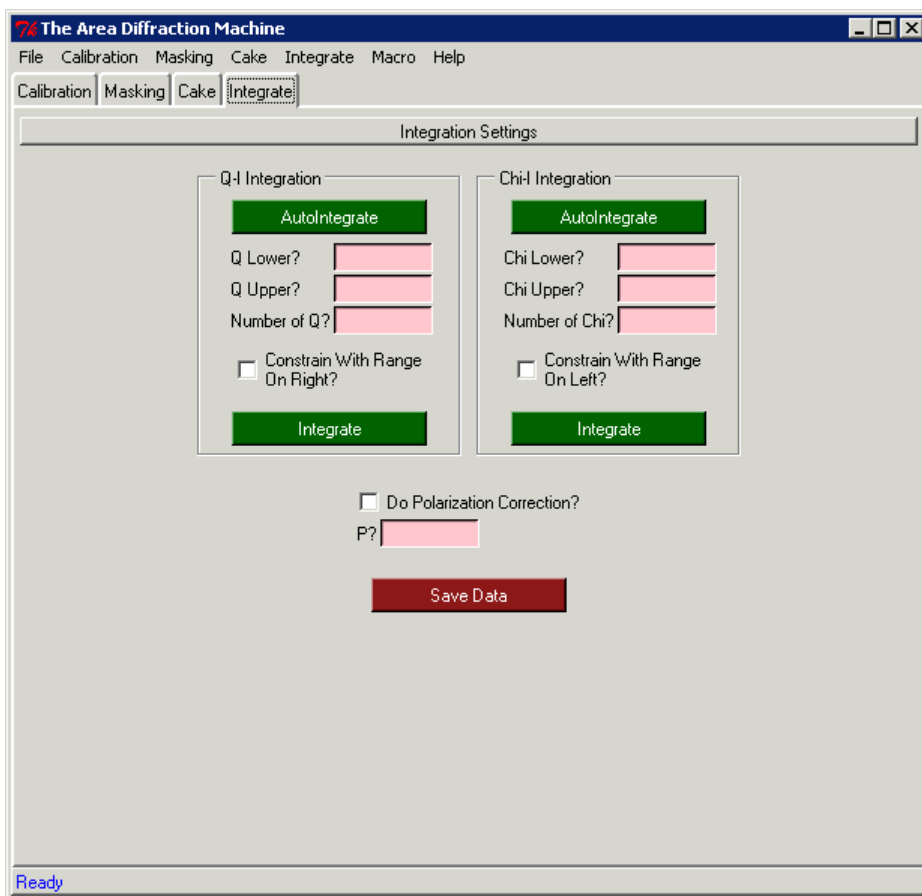


Figure 14.1: The integration tab. This is where intensity integration is done.

The inputs on the right is titled “Chi-I Integration” and can be used for performing a χ integration. The “Q Lower?” and “Q Upper?” inputs on the right can be used to specify an integration range in χ . The number of bins in χ space can be specified with the “Number of Chi?” input. The “Integrate” button on the right can be used to perform χ integration.

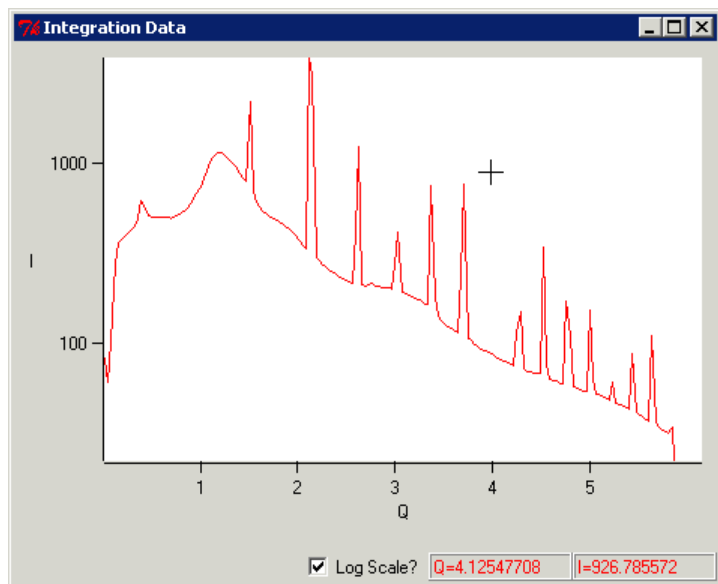


Figure 14.2: The integration window that opens up after an intensity integration is performed.

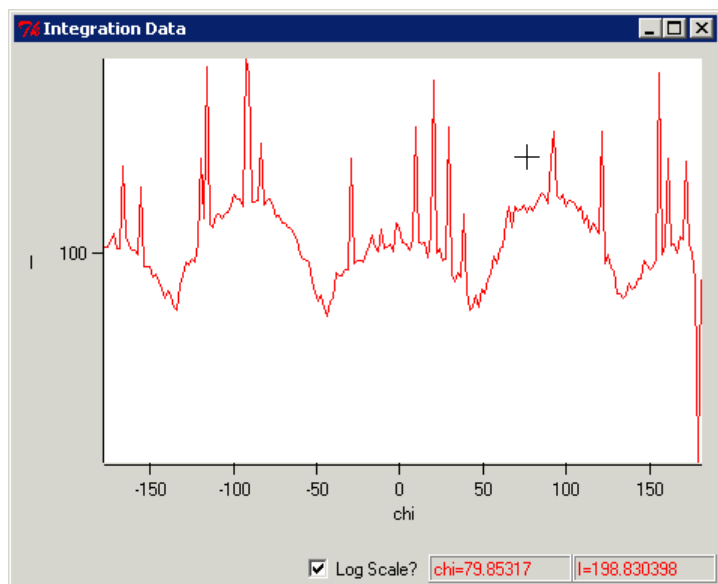


Figure 14.3: The integration window that opens up after you perform an intensity integration.

14.3 The Integration Window

After the program finishes integrating, a line plot of the integrated data will be displayed in a new window. Figure 14.2 shows the integration window displaying $Q - I$ integrated data and figure 14.3 shows the window displaying $\chi - I$ integrated data. This window has a couple of nice features for interacting with the data:

- *Zoom into the data* – left click on the plot and hold down on the mouse. When the mouse is moved around, the program will create a resizing rectangle. When the mouse is released, the program will zoom into the selected range.
- *Zoom out of the data* – right click on the plot.
- *Resize the window* – click on the bottom right corner of the window and drag. The window will resize just like any other window and the plot will become larger or smaller.
- *Read coordinates for a selected point* – when mousing over certain the plot, the selected Q , χ or 2θ and intensity value will be displayed on the bottom of the window.
- *Log Scaling* – the “Log Scale?” check box will toggle whether to display a log scale of the data.

14.4 Working in 2θ

This program can integrate in 2θ instead of Q . This The “Work in 2theta” option in the menu bar can be used to change the way that integration is done. This option will make the label on the left to say “ 2θ -I Integration”. The inputs below will change to “ 2θ Lower”, “ 2θ Upper”, and “Number of 2θ ”. The “Integrate” button will then perform an integrate in 2θ . The diffraction window will display average intensity as a function of 2θ . If there are any values in the “ Q Lower” or “ Q Upper”, they will be convert from Q to 2θ values when the program switches. The “Work in Q ” option in the menu bar can be used to change the program back to working with Q . Any values in the “ 2θ Lower?” or “ 2θ Upper?” will be converted back.

14.5 AutoIntegrate

There is a convenience function called “AutoIntegrate” that is similar to the “AutoCake” button. “AutoIntegrate” will try to pick a nice integration range and then do the integration. The AutoIntegrate button on the left will guess at a nice range of Q (or 2θ) and then do the Q (or 2θ) integration. It will always make the lower Q or 2θ value 0 and the upper value large enough to include all the data. It will set the number of Q or 2θ to 200. The “AutoIntegrate” button on the right will guess a nice range of χ and do the χ integration. It will always set “Chi Lower” to -180, “Chi Upper” to 180, and “Number of Chi” to 200.

14.6 Constraining the Inputs

As was described in section 14.1, an integration of one parameter can be constrained by another parameter. For example, a Q or 2θ integration can be done only of values in a particular χ range. χ integration can only be done of a particular Q or 2θ range. Of course, it would be pointless to constrain Q to a certain range of 2θ or vice versa.

To constrain the integration using the program, there are two convenient “Constrain With Range On Right?” and “Constrain With Range On Left?” check boxes.

When “Constrain With Range On Right?” is selected, the Q or 2θ integration being done will be constrained in χ by the chi range specified by “Chi Lower” and “Chi Upper”. When “Constrain With Range On Left?” is selected, the χ integration will be constrained by either the Q range specified by “Q Lower?” and “Q Upper?” or the 2θ range specified by “ 2θ Lower?” and “ 2θ Upper?”.

14.7 Masking

The program allows for masking of certain pixels while integrating. Masking of intensity integrated data is done whenever the “Do Greater Than Mask?”, “Do Less Than Mask?”, or “Do Polygon Mask?” check boxes are selected. Whenever the program finds an intensity value that should be masked (either because it is too large, too small, or in a polygon mask), the program will ignore the pixel and not bin it. Refer to Chapter 12 for a discussion of masking.

14.8 Saving Integrated Data

The intensity integrated data can be saved to a file using the “Save Data” button. A typical integration file looks like:

Listing 14.1: 'A Cake Data File'

```
1 # Q vs I Intensity Integration
2 # Intensity integration of: C:/data/LaB6_14_02_56.mar3450
3 # Data Integrated on Fri Mar 21 17:59:16 2008
4 # Calibration data used:
5 #   x center:      1725.000000 pixels
6 #   y center:      1725.000000 pixels
7 #   distance:      125.296000 mm
8 #   energy:        12735.3957721 eV
9 #   alpha:         0.000000 degrees
10 #  beta:          0.000000 degrees
11 #  rotation:       0.000000 degrees
12 #  pixel length:   100.000000 microns
13 #  pixel height:   100.000000 microns
14 # A polarization correction was applied
```

```

15 # P = 1.000000
16 # A greater than mask was applied
17 # Greater than mask = 10000.000000 (All pixels above 10000.000000 were
18 # A Less Than Mask was applied.
19 # Less than mask = 50.000000 (All pixels below 50.000000 were ignored)
20 # Polygon mask(s) were applied
21 # Polygon(s) used in the analysis:
22 # 647.844364937 1369.72808587
23 # 1449.93738819 3226.88193202
24 # 2535.84794275 1449.93738819
25 #
26 # 1258.66905188 641.674418605
27 # 1215.47942755 999.531305903
28 # 1505.46690519 1116.76028623
29 # 1653.54561717 777.413237925
30 # Integration performed with a chi constraint
31 # chi constraint lower: 90.000000
32 # chi constraint upper: 270.000000
33 # Integration Range:
34 # Q Lower = 0.000000
35 # Q Upper = 6.726544
36 # Number of Q = 200.000000
37 # Q Step = 0.033633
38 # Q Avg Intensity
39 0.016901 0.000000
40 0.050703 0.000000
41 0.084504 0.000000
42 0.118306 0.000000
43 0.152108 0.000000
44 0.185910 0.000000
45 ...

```

The header is a bunch of lines that begin with `#`. The header describes the state that the program was in when the intensity integration was performed. The first line describes what type of integration was performed. For example, if a $\chi - I$ integration was performed, the header file will say “`# Chi vs I Intensity Integration`”. The header then contains the name(s) of the diffraction files that were integrated. The header contains the calibration parameters that were used when integrating. The header contains information about any polarization correction, greater or less than mask that was applied, polygon mask that was applied. It describes any constraints on the integration and finally the integration range and step size. Following the header is the line “`# Q Avg Intensity`” (or “`# Chi Avg Intensity`” or “`# 2theta Avg Intensity`”). Following it is the data. Each line contains two numbers corresponding to one bin. The first number is the middle Q (or χ or 2θ value) in the bin and the second number is the average intensity.

Chapter 15

Macros

This program is almost fully automatable with macros. Macros can be used to perform data analysis as quickly as possible. The program is capable of recording macros and running macros. The macro file format is simple enough that it is easy to write or modify macro files by hand.

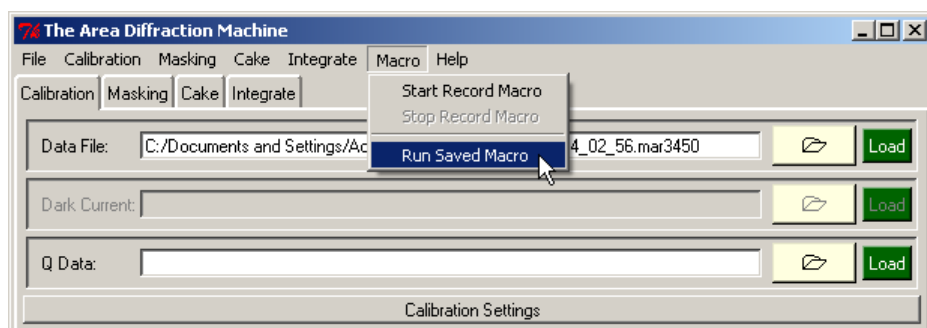


Figure 15.1: The “Macro” menu bar. This is where macros are recorded and run.

15.1 Record Macros

The easiest way to create a macro is to record it. A macro can be recorded by selecting the “Start Record Macro” option in the “Macro” menu bar. Figure 15.1 shows the “Macro” menu bar. After all of the steps that should be recorded are finished, pushing “Stop Record Macro” will save the macro file to a selectable file.

15.2 Run Macros

The “Run Saved Macro” option in the “Macro” menu to run a macro file. The program will run all the steps in the macro file and then return control of the program. This is how analysis can be done with macro files.

15.3 The Macro File Format

A macro file contains a list of commands which tell the program what to do. Each command in the GUI is on its own line. The syntax for macro commands is pretty straightforward. Macro commands are the text corresponding to the part of the GUI that does the command. For example, to make the macro get the calibration data from the header of the image, the macro command is “**Get From Header**”. To fit the calibration data from within a macro, the command is “**Do Fit**”.

Things get more interesting when the GUI item requires requires doing more then just pushing a button. For example, to deselect the “**Draw Q Data?**” check box, the macro needs to specify that the check box gets deselect instead of selected. For these, the macro commands need to be followed by a second line with the particular. For this example, we would write

Listing 15.1: ‘Draw the Q Lines on the Display’

```
1 Draw Q Data?
2     Select
3 # Or, to not display them:
4 Draw Q Data?
5     Deselect
```

It is the same when numbers should be set. To change a calibration values, the macro would look like:

Listing 15.2: ‘Input a Number’

```
1 xc:
2     1752.3
3 beta:
4     5.23
```

These are treated just the same. The following macro command would save the cake as an image:

Listing 15.3: ‘Save the Caked Image’

```
1 Save Caked Image
2     C:/data/cake_output.jpg
```

If you look on the first tab, there are three inputs at the top: “**Get From Header:**”, “**dark current:**”, and “**Q data:**”. The macro command to load any of these is a little bit ambiguous. When using the actual GUI, you would, at least in principle, type in the name of a file and then press load. But there is no reason to make the GUI so redundant. So to load in any of these using a macro command, all you have to do is give the name of the input and then the filename. It will automatically load the file without you explicitly giving the “load” line. So, for example, to load in the Q data, you would include the following lines:

Listing 15.4: 'Load the Q Data'

```

1 Q Data:
2   C:/data/q_data.dat

```

15.4 Looping Over Diffraction Data

To analyze a file, the command is just

Listing 15.5: 'Load the Diffraction Data'

```

1 Data File:
2   C:/data/first.mar3450
3 Get From Header
4 # ...

```

But macros files also allow for an easier way to loop over many files and perform the same analysis on all of them. To loop over multiple diffraction images at once, you could simply give more files after the first one. The loop will end when one of the 3 things in the macro file happens: a subsequent line in the macro file reads “END LOOP”, more diffraction data is loaded using the command “Data File:”, or the macro file ends. For example, if we look at this macro file.

Listing 15.6: 'Loop Over Diffraction Data'

```

1 Data File:
2   C:/data/first.mar3450 C:/data/second.mar3450
3 Integrate Q Lower?
4   .25
5 Integrate Q-I
6 END_LOOP
7 Draw Q Lines?
8   Select
9 # ...

```

We see that it would get evaluated just like this macro file:

Listing 15.7: 'An Equivalent Macro'

```

1 Data File:
2   C:/data/first.mar3450
3 Integrate Q Lower?
4   .25
5 Integrate Q-I
6 Data File:
7   C:/data/second.mar3450
8 Integrate Q Lower?
9   .25

```

```

10 Integrate Q-I
11 Draw Q Lines?
12     Select
13 # ...

```

You can even give it whole directories. When you give it a directory to loop over, the program will (non-recursively) look for all the diffraction files in that directory and include them in the list. For example, if the folder “C:/data/” contains only the file “first.mar3450” and “second.mar3450”, an equivalent way of looping over these files would be to issue the command

Listing 15.8: ‘Load the Diffraction Data’

```

1 Data File:
2     C:/data/
3 # ...

```

You can put as many folder and files after a “Data File:” line as you wish. Just make sure to put them all on the same line or the program will complain.

15.5 The PATHNAME and FILENAME Commands

Finally, there is a convenience markup which can help you make fancy macros. Whenever you have loaded data in, you can refer to the part name of the current diffraction file that is loaded using the string “PATHNAME” and you can refer to the file name itself using the string “FILENAME”. So, in our previous example, if we had loaded the file “C:/data/second_file.mar3450”, “PATHNAME” would get changed into “C:/data” and “PATHNAME” would get evaluated to “second_file” without the extension. In effect, you can imagine building back the full name from “PATHNAME” and “FILENAME” using an equation line

C:/data/second_file.mar3450= FILENAME/PATHNAME.mar3450

These commands are useful because they allow you to loop over many files at once but still save things in useful places and with useful names. It would be easy, for example, to save the intensity data you calculate for each file being looped over using the macro command:

Listing 15.9: ‘Using the FILENAME and PATHNAME Markup’

```

1 Save Integration Data
2     FILENAME/PATHNAME\_int.dat

```

This would save, for example, “C:/data/first.mar3450”’s intensity data to “C:/data/first_int.dat”, “C:/data/second.mar3450”’s intensity data to “C:/data/second_int.dat”, and the same for all the others. This feature lets you have the macro to save each of the files to the right place and give it a useful name.

15.6 Loops Over Multiple Images

We know from chapter 9 that you can load in multiple diffraction images and add them together. But we have not yet talked about how this can be done inside of a macro. The syntax is pretty straight forward. We introduce a new macro command named “**Multiple Data Files**” which signifies that many files should be loaded. To load in multiple files and have their intensities added, this command must be followed by a list of filenames enclosed within [and] brackets. Keeping with the same example above, we could load in “C:/data/first.mar3450” and “C:/data/second.mar3450” and have their intensities added using the command

Listing 15.10: 'Add the intensities'

```
1 Multiple Data Files:
2   [C:/data/first.mar3450 C:/data/second.mar3450]
3 # ...
```

The program enforces that all of the files are in the same folder. This is done so that that the “PATHNAME” variable remains meaningful when looping over multiple images.

You can then incorporate this into a loop in one of two ways. First, you can simply put several of these bracketed lists into the macro and each of the lists will be analyzed separately. For example,

Listing 15.11: 'Loop Over the Analysis'

```
1 Multiple Data Files:
2   [C:/first.mar3450 C:/second.mar3450] [C:/third.mar3450 C:/fourth.mar3450]
3 # ...
```

This will separately loop over “first.mar3450” and “second.mar3450” added together and then “third.mar3450” and “fourth.mar3450” added together. But this gets cumbersome. Alternately, you can simply take all of the files that you want to be added together and analyzed and group them into subfolders. Each of the subfolders will contain only files that should be added together and analyzed. If you then give the macro the name of the folder containing all these subfolders, it will loop over all the subfolders.

For example, suppose we created the folder “C:/data”. Inside of this folder is the subfolder “A” containing the files “first.mar3450” and “second.mar3450”. Also inside of the data folder is the subfolder “B” containing the files “third.mar3450” and “fourth.mar3450”. We could do the exact same data analysis as above by issuing the macro command with only the data folder name.

Listing 15.12: 'Using the Folder Syntax'

```
1 Multiple Data Files:
2   C:/data
3 # ...
```

You can also put as many folders and lists separated by [and] as you wish onto the line and it will loop over all of them.

Since the macro function insists that all files that are added together are in the same folder, the “PATHNAME” command will properly expand to the path that all of the files that were added together have. But since they all have different file names, the “FILENAME” command will always be replaced by the string “MULTIPLE_FILES” to avoid ambiguity.

15.7 The FOLDERPATH and FOLDERNAME commands

To facilitate writing macros that load in and add together several diffraction images, the program introduces two new macro commands. The first command is “FOLDERNAME” and will always be replaced by the name of the folder containing the current diffraction file (or files). Since the macro insists that all files are loaded from the same Folder, this command is unambiguous. Finally, you can use the command “FOLDERPATH”. It will always be replaced by the path leading up to the folder containing the file. Therefore, we can now specify where the current file is by using the macro command

```
FILENAME/PATHNAME.mar3450
```

or the command

```
FOLDERPATH/FOLDERNAME/FILENAME.mar3450
```

Basically, this is useful because if you are adding together multiple files, you can put them all in a folder with an interesting name and then name the output files something like “FOLDERNAME_int.dat” so that they are all given useful names. This is nice because it can, without loss of generality, be incorporated into a loop. Finally, the “FOLDERPATH” command is useful because you can use it to output files one directory up from where all the diffraction data is stored.

15.8 Setting Colors in a Macro

There are several places in the program where you can pick the color of something using a color selector. It is a little trickier to do. When you issue a macro command that wants to know the color of something, you have to tell it what that color is. By far the easiest way to figure out exactly what the macro line should look like is to simply record a macro where select the color that you want and then copy the macro lines into your file.

But if you are curious exactly what the format for colors looks like, you can see that picking a color will generally look like this:

Listing 15.13: ‘Use the Folder Syntax’

```
1 Polygon Mask Color?  
2     red
```

But it is a bit tricky trying to figure out exactly what colors will work. Technically, this program will accept any color which tk will accept. The colors that tk will accept by name are all described here: <http://wiki.tcl.tk/16166>. But tk can also accept colors based upon their RGB value. To specify a color by its RGB value, the color must be preceded by a # and followed by the RGB values in hexadecimal. Each of the RGB values range from 0 to 255 in decimal or (00 to ff in hexadecimal). For example, pure red would be specified by the color #ff0000. So we could replace the macro command above with the identical

Listing 15.14: 'Use the Folder Syntax'

```
1 Polygon Mask Color?
2      #ff0000
```

15.9 Little Tidbits

- Any of the macro commands themselves are case insensitive. The command “GeT fRoM hEaDeR” is just as valid as the command “gET fRoM hEaDeR” and “Get From Header”. You don’t have to sweat it.
- White spaces at the beginning and end of the line are ignored. In the preceding examples, the spaces separating macro commands from input values such as file names are there only to increase readability. You don’t need them if you don’t want.
- Any new lines in a macro file are ignored.
- comment lines of the form “# A comment” are ignored.
- You don’t have to worry about explicitly moving from tab to tab in the computer program. The computer program will move to the right automatically before performs the action.
- When you issue the macro command “E:” or “E Fixed”, the computer program will automatically set the GUI to “Work in eV”. If you issue the command or “lambda:” or “lambda fixed:” then the computer program will set the GUI to “Work in Lambda”. You can also explicitly set the GUI to either mode using the command “Work in eV” or “Work in Lambda”.

15.10 Macro Commands

Below is a table describing all of the macro command and exactly what they do.

Table 15.1: Macro Commands

Command	Followed By	Effect
Program State Macro Commands		
Work In eV	None	Change the state of the program so that the energy calibration parameter is inputted in units of electron volts. This is called the <i>eV</i> mode of the program.
Work in Lambda	None	Change the state of the program so that the energy calibration parameter is inputted instead as a wavelength in units of angstroms. The conversion is done using the formula $E = hc/\lambda$. This is called the λ mode of the program.
Work in 2theta	None	Change the state of the program so that caking and intensity integration are done of the variable 2θ .
Work in Q	None	Change the state of the program so that caking and intensity integration are done of the variable Q .
Calibration Values		
Data File:	Files & Directories	Loops over loading in each file.
Multiple Data Files"	Files & Directories	Loops over loading several diffraction files and adding them together.
Dark Current:	Filename	Loads in the Dark Current.
Q Data:	Filename	Load in the Q data.
Standard Q	q data	Loads in one of the standard Q files. This command should be followed by the name of the standard Q file as it is displayed by the program in the menu bar.
Get From Header:	None	Sets the calibration data to the value stored in the image header.
Load From File:	Filename	Loads a calibration data file.
Previous Values	None	Loads the previously stored calibration values.
Save To File	Filename	Saves the calibration data to a file.
xc:	Number	Sets the x center.
xc Fixed:	Select or Deselect	Sets whether or not to fix the x center while doing the fit.
yc:	Number	Set the y center.
Continued on next page...		

Table 15.1 – continued from previous page

Command	Followed By	Effect
yc Fixed:	Select or Deselect	Sets whether or not to fix the y center while doing the fit.
d:	Number	Set the distance from the sample to the detector.
d Fixed:	Select or Deselect	Sets whether or not to fix the distance while doing the fit.
E:	Number	Sets the energy. If this command is run while the program is in λ mode, the program will switch to eV mode.
E Fixed:	Select or Deselect	Sets whether or not to fix the energy while doing the fit. If this command is run while the program is in λ mode, the program will switch to eV mode.
lambda:	Number	Sets the wavelength. If this command is run while the program is in eV mode, the program will switch to λ mode.
lambda Fixed:	Select or Deselect	Sets whether or not to fix the wavelength while doing the fit. If this command is run while the program is in eV mode, the program will switch to λ mode.
alpha:	Number	Sets the α angle.
alpha Fixed:	Select or Deselect	Sets whether or not to fix the α angle while doing the fit.
beta:	Number	Sets the β angle.
beta Fixed:	Select or Deselect	Sets whether or not to fix the β angle while doing the fit.
R:	Number	Sets the rotation angle.
R Fixed:	Select or Deselect	Sets whether or not to fix the rotation angle while doing the fit.
pl	Number	The pixel length of the image. This is the width of one pixel (in microns).
ph	Number	The pixel height of the image. This is the height of one pixel (in microns).
Draw Q Lines?	Select or Deselect	Sets whether or not to draw constant Q lines on the screen.
Draw Q Lines Color?	color	Sets the color of the constant Q lines that are displayed on top of the diffraction data and the caked data.
Continued on next page...		

Table 15.1 – continued from previous page

Command	Followed By	Effect
Draw dQ Lines?	Select or Deselect	Draw the delta Q lines on the diffraction image.
Draw dQ Lines Color?	color	Change the color of the delta Q lines that are displayed on top of the diffraction data and the caked data.
Draw Peaks?	Select or Deselect	Display the fit peaks on the diffraction and cake image.
Draw Peaks Color?	color	Change the color of the peaks that are displayed on top of the diffraction data and the caked data.
Update	None	Update the diffraction image.
Save Calibration	Filename	Saves the current calibration values in the GUI as plaintext ASCII to a file.
Do Fit	None	Fit the calibration values to a loaded diffraction image.
Make/Save Peak List	Filename	Creates a peak list just as happens when doing the fit, but instead of actually doing the fit it saves the peaks as an ASCII file for later use.
Use Old Peak List (if possible)?	Select or Deselect	Uses the previously found peak list again when doing the fit.
Fit Number of Chi?	Number	The number of χ slices around the diffraction image to pick and use when doing the calibration.
Stddev	Number	The σ threshold for allowing a peak.
Diffraction Display Options		
Diffraction Data Colormaps	A color map name	Select the color map to use for the diffraction image.
Diffraction Data Invert?	Select or Deselect	Invert the color map that is being used to display the diffraction data.
Diffraction Data Log Scale?	Select or Deselect	Take the log of all the data points before displaying them.
Diffraction Data Low?	Number from 0 to 1	The normalized intensity value which will be scaled to %0 of the image brightness when displaying the diffraction image.
Diffraction Data Hi?	Number from 0 to 1	The normalized intensity value which will be scaled to %100 of the image brightness when displaying the diffraction image.
Continued on next page...		

Table 15.1 – continued from previous page

Command	Followed By	Effect
Save Diffraction Image	Filename	Save the diffraction image to a file (possibly including Q lines and peaks.
Masking Macro Commands		
Do Less Than Mask?	Select or Deselect	Sets whether or not to apply a less than mask to the diffraction data.
(Pixels Can't Be)	Number	Sets the less than mask.
Less Than Mask:		
Less Than Mask	color	Sets the color that all the less than masked pixels are displayed as on the diffraction image and caked image.
Color?		
Do Greater Than Mask?	Select or Deselect	Sets whether or not to apply a greater than mask to the diffraction data.
(Pixels Can't Be)	Number	Sets the greater than mask.
Greater Than Mask:		
Greater Than Mask	color	Sets the color that all the greater than masked pixels are displayed as on the diffraction image and caked image.
Color?		
Do Polygon Mask?	Select or Deselect	Sets whether or not to apply polygon masks to the diffraction data.
Polygon Mask	color	Sets the color that all polygon masked pixels should be displayed as on the diffraction image and the cake image.
Color?		
Save Mask	Filename	Saves all currently loaded or drawn polygons as plain text ASCII to a file.
Load Mask	Filename	Loads into the program from some file one or more polygons.
Clear Mask	None	Removes any polygon masks that are in the program.
Cake Macro Commands		
AutoCake	None	Make the computer pick a nice Q and χ range and Cake the data.
Cake Q Lower?	Number	The lower Q value in the range of Q and χ to use when caking. If this command is run while the program is in 2θ mode, the program will switch to Q mode.
Continued on next page...		

Table 15.1 – continued from previous page

Command	Followed By	Effect
Cake Q Upper?	Number	The upper Q value in the range of Q and χ to use when caking. of the caked data. If this command is run while the program is in 2θ mode, the program will switch to Q mode.
Cake Number Of Q?	Number	The number of Q bins to use while caking the data. If this command is run while the program is in 2θ mode, the program will switch to Q mode.
Cake 2theta Lower?	Number	The lower 2θ value in the range of 2θ and χ to use when caking. If this command is run while the program is in Q mode, the program will switch to 2θ mode.
Cake 2theta Upper?	Number	The upper 2θ value in the range of 2θ and χ to use when caking. If this command is run while the program is in Q mode, the program will switch to 2θ mode.
Cake Number Of 2theta?	Number	The number of 2θ bins to use while caking the data. If this command is run while the program is in Q mode, the program will switch to 2θ mode.
Cake Chi Lower?	Number	The lower χ value of the caked data.
Cake Chi Upper?	Number	The upper χ value of the caked data.
Cake Number Of Chi?	Number	The number of χ bins to use while caking the data.
Do Cake	None	Performs a cake of the data and displays that caked data in the cake window.
Last Cake	None	Go back to the previous cake values.
Save Caked Image	Filename	Saves the cake as a popular image format. The image will be saved as the input filename and the extension of the filename should tell the program what format to save the image as.
Save Caked Data	Filename	Saves the cake as ASCII data with a verbose header.
Continued on next page...		

Table 15.1 – continued from previous page

Command	Followed By	Effect
Cake Do Polarization Correction?	Select or Deselect	Sets whether or not to use a polarization correction when caking the data.
Cake P?	Number from 0 to 1	Sets the value of the polarization correction to use when caking the data.
Cake Display Options		
Cake Data Colormaps:	Color map	Sets the color map to use when displaying the caked data.
Cake Data Invert?	Select or Deselect	Sets whether or not to invert the color map when displaying the caked data.
Cake Data Log Scale?	Select or Deselect	Sets whether or not to use a log scale when applying the color map to the caked data.
Cake Data Low?	Number from 0 to 1	The normalized intensity value which will be scaled to %0 of the image brightness when displaying the caked data.
Cake Data Hi?	Number from 0 to 1	The normalized intensity value which will be scaled to %100 of the image brightness when displaying the caked data.
Intensity Integration Macro Commands		
Integrate Q Lower?	Number	The lower Q value to use when performing an intensity integration. If the command is run when program is in 2θ mode, the program will switch to Q mode.
Integrate Q Upper?	Number	The upper Q value to use when performing an intensity integration. If the command is run when program is in 2θ mode, the program will switch to Q mode.
Integrate Number Of Q?	Number	The number of Q bins to use when performing an intensity integration. If the command is run when program is in 2θ mode, the program will switch to Q mode.
Continued on next page...		

Table 15.1 – continued from previous page

Command	Followed By	Effect
Integrate 2theta Lower?	Number	The lower 2θ value to use when performing an intensity integration. If the command is run when program is in Q mode, the program will switch to 2θ mode.
Integrate 2theta Upper?	Number	The upper 2θ value to use when performing an intensity integration. If the command is run when program is in Q mode, the program will switch to 2θ mode.
Integrate Number Of 2theta?	Number	The number of 2θ bins to use when performing an intensity integration. If the command is run when program is in Q mode, the program will switch to 2θ mode.
Integrate Chi Lower?	Number	The lower χ value to use when performing an intensity integration.
Integrate Chi Upper?	Number	The upper χ value to use when performing an intensity integration.
Integrate Number Of Chi?	Number	The number of χ bins to use when performing an intensity integration.
Integrate Q-I	None	Performs a $Q - I$ integration of the diffraction data. If the command is run when the program is in 2θ mode, the program will switch to Q mode.
AutoIntegrate Q-I	None	Picks a good range of Q values and then does the same thing as the Integrate Q-I command.
Integrate 2theta-I	None	Performs a $2\theta - I$ integration of the diffraction data. If the command is run when the program is in Q mode, the program will switch to 2θ mode.
AutoIntegrate 2theta-I	None	Picks a good range of 2θ values and then does the same thing as the Integrate 2theta-I command.
Integrate chi-I	None	Performs a $\chi - I$ integration of the diffraction data.
AutoIntegrate chi-I	None	Picks a good range of χ values and then does the same thing as the Integrate chi-I command.
Continued on next page...		

Table 15.1 – continued from previous page

Command	Followed By	Effect
Save Integration Data	Filename	Saves out the intensity integrated data as two column plain text ASCII with the given filename.
Constrain With Range On Right?	Select or Deselect	Sets whether or not to apply a constraint to the Q or 2θ vs. I integration so that the integration is only done of pixels who's χ value is within the χ integration range.
Constrain With Range On Left?	Select or Deselect	Sets whether or not to apply a constraint to the χ vs I integration so that the integration is only done of pixels who's Q (or 2θ) value is within the Q (or 2θ) integration range.
Integrate Do Polarization Correction?	Select or Deselect	Sets whether or not to use a polarization correction when performing an intensity integration.
Integrate P?	Number form 0 to 1	sets the value of the polarization correction to use when performing an intensity integration.
Integration Data Log Scale?	Select or Deselect	Sets whether or not to use a log scale when displaying the diffraction data.

15.11 What You Can't Do With Macros

Just to be clear:

- There is no way with a macro to zoom into the diffraction data, the cake data, or the intensity integrated data
- You can't draw individual polygon masks and you can't remove individual polygon masks. All you can do is load in polygon's from file and save all the current polygons to a file.
- When you load in multiple images at once by giving a file name, it will only load in images from the file with known extensions (ie .mar2300, .mar3450, .mccd, .tiff). So give your files proper extensions before running macros.

Chapter 16

Software Licensing

This program is released under the GNU General Public License (GPL) version 2. The license can be found at <http://www.gnu.org/licenses/old-licenses/gpl-2.0.html>. For the most part, you are free to use and distribute this software. You are free to make any modifications to the code under the condition that any modifications are clearly stated and that the modifications are released under the GPL version 2.

This software manual is also licensed under the GPL. This is a bit unconventional. I decided to do so after reading several discussions online. Following Nathanael Nerode's article *Why You Shouldn't Use the GNU FDL*, I include in this paper the clause "for the purpose of applying the GPL to this document, I consider 'source code' to refer to the texinfo source and 'object code' to refer to the generated info, tex, dvi, [pdf] and postscript files." [9]

This program uses the software package levmar for performing Levenberg-Marquardt nonlinear least squares minimization. It is released under the GPL. That package can be found at <http://www.ics.forth.gr/~lourakis/levmar/>. [6]

This program uses the function `get_pck()` from the CCP4 package `DiffractionImage` to uncompress Mar data. It is written by Dr. Claudio Klein. [4] This program also uses the file `marccd_header.h` from the `DiffractionImage` package. It is released under the GPL and can be found at <http://www.ccp4.ac.uk/ccp4bin/viewcvs/ccp4/lib/DiffractionImage/>. [2]

This program uses the `EdfFile` library (`EdfFile.py`) for reading and writing files of the ESRF Data Format. It is part of the PyMCA library and is licensed under the GNU GPL version 2. [12]

This program also uses W. Randolph Franklin's `pnpoly()` function for performing a point inclusion in polygon test. This code can be found at http://www.ecse.rpi.edu/Homepages/wrf/Research/Short_Notes/pnpoly.html. We are in compliance with his software license which is reproduced below [3]:

*Copyright (c) 1970-2003, Wm. Randolph Franklin
Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in the
Software without restriction, including without limitation the rights to use, copy,
modify, merge, publish, distribute, sublicense, and/or sell copies of the Software,*

and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimers. Redistributions in binary form must reproduce the above copyright notice in the documentation and/or other materials provided with the distribution. The name of W. Randolph Franklin may not be used to endorse or promote products derived from this Software without specific prior written permission. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Part V

Plan Exams

Chapter 17

Comprehensive Exam, part 1 - Covers: basic physics, special relativity, classical mechanics

17.1 Problem

“A solid spherical ball of uniform mass density (e.g., a pool ball) rolls without slipping down a ramp which makes angle θ with the horizontal. (a) What is its translational acceleration down the ramp? (b) If the coefficient of friction between the ball and the surface is $\mu = 0.1$, for what value of θ will the ball slip rather than roll without slipping?” – Travis Norsen

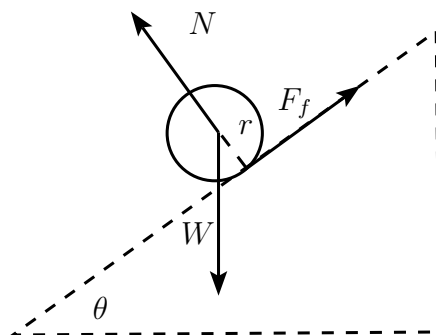


Figure 17.1: The Ramp.

Figure 17.1 shows a free body diagram of this situation. Since we are dealing with a pool ball, $I_{CM} = \frac{2}{5}mr^2$. I will calculate the torque about the contact point between the ball and the ground.

$$\boldsymbol{\tau} = \sum \mathbf{r} \times \mathbf{F}. \quad (17.1)$$

About this point, the only force applying a torque is the weight force:

$$\tau = rm g \sin \theta. \quad (17.2)$$

We can use the parallel axis theorem to calculate the moment of inertia of the ball about the contact point:

$$I = \frac{2}{5}mr^2 + mr^2 = \frac{7}{5}mr^2 \quad (17.3)$$

The linear acceleration is related to the angular acceleration of the ball about the contact point by $a = r\alpha$. We can use $\tau = I\alpha$ to calculate a :

$$I\alpha = \frac{7}{5}mr^2 \times \frac{a}{r} = rm g \sin \theta. \quad (17.4)$$

We get

$$a = \frac{5}{7}g \sin \theta \quad (17.5)$$

To figure out part (b), we can apply $\mathbf{F} = m\mathbf{a}$ to our free body diagram. We first do the direction parallel to the ramp:

$$ma = mg \sin \theta - F_f \quad (17.6)$$

We know that the ball will just have slipped (or be about to slip) when the force of friction just counteracts the downward pull of gravity and $a = 0$:

$$F_f = mg \sin \theta. \quad (17.7)$$

Doing the direction perpendicular to the ramp gets us:

$$N = mg \cos \theta \quad (17.8)$$

Since

$$F_f \leq \mu N, \quad (17.9)$$

we know that

$$mg \sin \theta \leq \mu mg \cos \theta. \quad (17.10)$$

Or,

$$\tan \theta \leq \mu. \quad (17.11)$$

The largest value of θ is

$$\tan \theta = \mu. \quad (17.12)$$

17.2 Problem

“A planet orbits the sun under the influence of the gravitational force

$$F = \frac{GMm}{r^2} \quad (17.13)$$

Suppose the planet has orbital angular momentum L . Write down an expression for the effective potential energy (i.e., the gravitational potential energy plus the term from the kinetic energy associated with the angular motion), and find the value of r (call it R) for which the effective potential is minimized. What does it mean physically if $r(t) = R$? What is the period of the orbit? Now consider small oscillations of r about R . Approximate the effective potential near $r = R$ as a parabola, and find the resulting period of small oscillations. Sketch the shape of a not-quite-circular orbit. Is it what you expect?" – Travis Norsen

The potential of the system is

$$U(r) = -\frac{k}{r} \quad (17.14)$$

With $k = GMm$. We can write the total energy (or Hamiltonian) of the system as

$$H = \frac{1}{2}m|\mathbf{r}_m|^2 + \frac{1}{2}M|\mathbf{r}_M|^2 + U(|\mathbf{r}_m - \mathbf{r}_M|) \quad (17.15)$$

If we let $\mathbf{r} = \mathbf{r}_m - \mathbf{r}_M$ and change to a coordinate system where the center of the system is the center of mass, then

$$m\mathbf{r}_m + M\mathbf{r}_M = 0. \quad (17.16)$$

From this, it follows that

$$\mathbf{r}_m = \frac{M}{m+M}\mathbf{r} \quad (17.17)$$

$$\mathbf{r}_M = -\frac{m}{m+M}\mathbf{r}. \quad (17.18)$$

Plug these back into equation 17.15 gets us

$$H = \frac{1}{2}\mu|\dot{\mathbf{r}}|^2 + U(r). \quad (17.19)$$

With $\mu = mM/(m+M)$. This is the equation of just one particle with mass μ in the same potential. We can treat this two body system as though it were a one body system where the one particle has the reduced mass. Writing $\dot{\mathbf{r}}$ in spherical coordinates gets us

$$\dot{\mathbf{r}} = \dot{r}\hat{r} + r\dot{\theta}\hat{\theta} + r\sin\theta\dot{\phi}\hat{\phi}. \quad (17.20)$$

Since we are dealing with a central force, we can without loss of generality assume that all motion takes place in a plane such that $\phi = 0$ and so that there is no $\hat{\phi}$ component to the velocity. When we do so, we get

$$\dot{\mathbf{r}} = \dot{r}\hat{r} + r\dot{\theta}\hat{\theta} \quad (17.21)$$

Our Hamiltonian becomes

$$H = \frac{1}{2}\mu\mathbf{v}^2 + U(r) = \frac{1}{2}\mu(\dot{r}^2 + r^2\dot{\theta}^2) + U(r). \quad (17.22)$$

Since angular momentum is conserved:

$$\mathbf{L} = \mu \mathbf{r} \times \mathbf{v} = \mu r^2 \dot{\hat{\phi}} = L \hat{\phi} \quad (17.23)$$

where in the last step we used equation 17.21. Since $\phi = 0$, we have $\hat{\phi} = \text{constant}$ as is necessary for angular momentum to be conserved. Thus,

$$L = \mu r^2 \dot{\theta}. \quad (17.24)$$

Plug this into equation 17.22 get us

$$H = \frac{1}{2} \mu \dot{r}^2 + \left(\frac{L^2}{2\mu r^2} + U(r) \right) \quad (17.25)$$

This is the equation of a particle in one dimensions in the effective potential

$$V^{\text{eff}} = \frac{L^2}{2\mu r^2} + U(r) = -\frac{k}{r_0} + \frac{L^2}{2\mu r^2} \quad (17.26)$$

The potential is minimized at r_0 when

$$\left. \frac{dV^{\text{eff}}}{dr} \right|_{r_0} = 0 \quad (17.27)$$

Or,

$$\frac{k}{r_0^2} - \frac{L^2}{\mu r_0^3} = 0 \quad (17.28)$$

so

$$r_0 = \frac{L^2}{k\mu}. \quad (17.29)$$

When the potential is at a minimum, the particle experience no radial force. This means that the particle undergoes uniform circular motion.

Next, we can calculate the velocity using equation 17.24. Since $v = \dot{\theta} r$, we have

$$L = \mu r_0^2 \dot{\theta} = \mu r_0 v \quad (17.30)$$

Or,

$$v = \frac{L}{\mu r_0} = \frac{k}{L} \quad (17.31)$$

The period is

$$T = \frac{2\pi r_0}{v} = \frac{2\pi L^3}{k^2 \mu} \quad (17.32)$$

We can Taylor expand $V(r)$ around r_0 . This will be a good approximation for radii near r_0 . First:

$$V(r_0) = -k \left(\frac{\mu k}{L^2} \right) + \frac{L^2}{2\mu} \left(\frac{\mu k}{L^2} \right)^2 = -\frac{1}{2} \frac{\mu k^2}{L^2} \quad (17.33)$$

Next:

$$\frac{d^2V(r_0)}{dr^2} = -\frac{2k}{r_0^3} + \frac{3L^2}{\mu r_0^4} = \frac{k^4\mu^3}{L^6} \quad (17.34)$$

Thus:

$$V^{\text{eff}}(r) \approx \frac{\mu k^2}{2L^2} + \frac{k^4\mu^3}{2L^6}\Delta r^2 \quad (17.35)$$

with $\Delta r = r - r_0$. We know that

$$F = \mu\ddot{r} = \mu\Delta\ddot{r} = -\frac{dV^{\text{eff}}}{dr} = -\frac{dV^{\text{eff}}}{d(\Delta r)} = -\frac{k^4\mu^2}{L^6}\Delta r \quad (17.36)$$

We can solve for Δr gets

$$\Delta r = A \sin\left(\frac{k^2\mu}{L^3}t\right) \quad (17.37)$$

This has a period of

$$T = 2\pi \frac{L^3}{k^2\mu}. \quad (17.38)$$

This is exactly the same as equation 17.32, the period of orbit. When we plot $r = r_0 + \Delta r$, the perturbed potential, the trajectory looks something like my diagram in figure 17.2. This trajectory looks like an ellipse, just as expected.

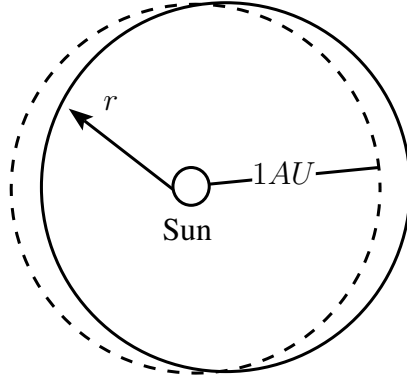


Figure 17.2: A diagram of the path of the perturbed orbit. Note that it roughly looks like a very circular ellipse with the sun at one focii.

17.3 Problem

“Alice and Bob are at opposite ends of a spaceship whose rest length $L = 20cs$ (i.e., twenty light seconds). They have previously synchronized their watches. When her watch reads noon, Alice rolls a ball to the right, toward Bob, at speed $u = 4/5c$. The ball moves at constant speed until Bob catches it. The question is: what does Bobs watch read at the moment he catches the ball? The trick is: you have to answer by working it out entirely from the frame of reference of Charlie, for whom the spaceship is moving to the right at speed $v = 3/5c$. (Hint: what does Bobs watch read when Alice rolls the ball? How long does it take the ball to get to him?)” – Travis Norsen

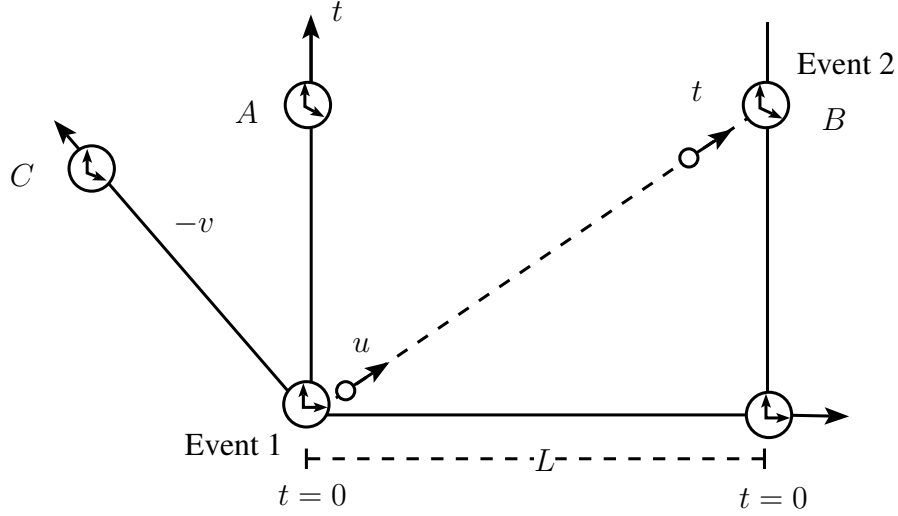


Figure 17.3: A space time diagram of the reference frame where the rocket is at rest

Figure 17.3 shows a space time diagram of this situation as viewed from the rocket rest frame. We will call this the unprimed reference frame. The problem is easy to solve in this frame. Alice's and Bob's clock are synchronized in this frame. If the ball leaves Alice at $t = 0$, it will travel a distance L at velocity u at arrive at Bob at a time

$$t = l/u \quad (17.39)$$

Using your numbers, we see that $t = 25s$ from which it follows that Bob's clock will read 25s after 12 o'clock (which I am calling $t = 0$. We can analyze this whole situation instead

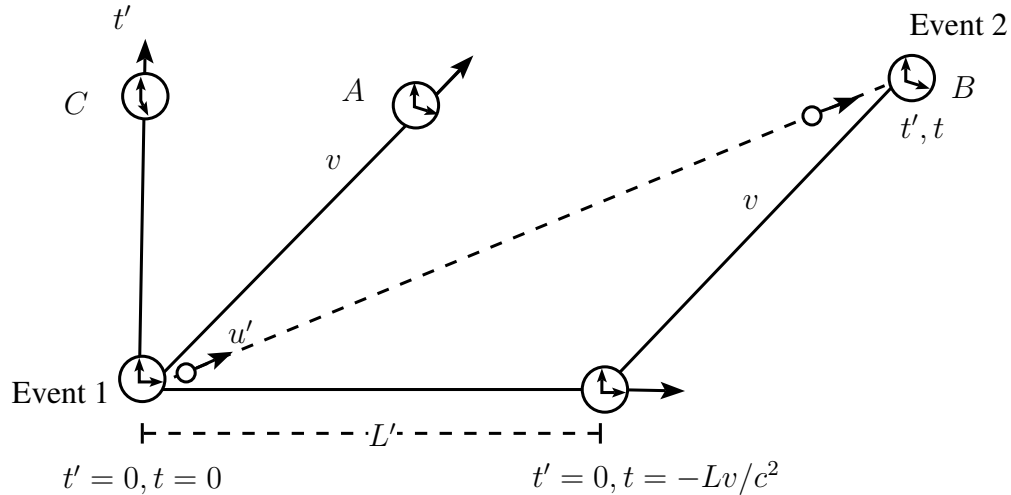


Figure 17.4: A space time diagram of the reference frame where the rocket is moving to the right with velocity v

from a reference frame where Charlie is at rest. A space time diagram for this situation is

shown in figure 17.4. This will be the primed reference frame. The distance between Alice and Bob will be contracted

$$L' = L\sqrt{1 - v^2/c^2}. \quad (17.40)$$

Remember that *moving meter sticks shrink*. Also, Bob's clock will read a time earlier than Alice's clock. I think the saying is *leading clocks lag*. According to Charlie, When Alice's clock reads $t = 0$ Bob's clock will say

$$t = -\frac{Lv}{c^2}. \quad (17.41)$$

Finally, the velocity of the ball, as measured from Charlie's frame has to be corrected. The relationship between the velocities is calculated using the relativistic velocity addition formula

$$u' = \frac{u + v}{1 + uv/c^2}. \quad (17.42)$$

We will call the time interval between Bob's clock when the ball leaves Alice and when it arriving at Bob as measured in Charlie's reference frame as t' . We can calculate it by setting the position of the ball equal to the position of Bob and solving for time:

$$u't' = L' + vt' \quad (17.43)$$

$$\left(\frac{u + v}{1 + uv/c^2}\right)t' = L\sqrt{1 - v^2/c^2} + vt' \quad (17.44)$$

$$t' = \frac{L(1 + uv/c^2)}{u\sqrt{1 - v^2/c^2}} \quad (17.45)$$

We will call this same time interval as measured by Bob's clock t . Bob's clock is moving slow because *Moving clocks run slow*. The reason why we know that Bob's clock is the one that is moving in this situation is because what t is a time interval between is Bob's clock when the ball leaves Alice (as viewed in Charlie's frame) and Bob's clock when the ball gets to Bob. Since Bob is present at both these events, he measures the proper time interval between these two events. We can thus use the formula

$$\Delta(\text{proper}) = \Delta(\text{improper})\sqrt{1 - v^2/c^2} \quad (17.46)$$

to show that

$$t = t'\sqrt{1 - v^2/c^2} \quad (17.47)$$

$$t = \frac{L}{u} \left(1 + \frac{uv}{c^2}\right) \quad (17.48)$$

If Bob's clock initially reads $-Lv/c$ and the time interval is t , the time that Bob's clock will reads when the ball gets to him is

$$-\frac{Lv}{c^2} + \frac{L}{u} \left(1 + \frac{uv}{c^2}\right) = \frac{L}{u} \quad (17.49)$$

This is just what we need!

17.4 Problem

“The point of suspension of a pendulum (mass m , length L) is allowed to move in the horizontal direction. It (the point of suspension) is connected to a spring which exerts a restoring force $F = kx$. (a) Use the coordinates x (the displacement of the point of support) and θ (the angular displacement of the pendulum bob from vertical) to write the Lagrangian and the equations of motion. (b) Linearize the equations of motion by assuming small oscillations; what length would an ordinary simple pendulum need to have in order to oscillate at the same frequency as the one here?” – Travis Norsen

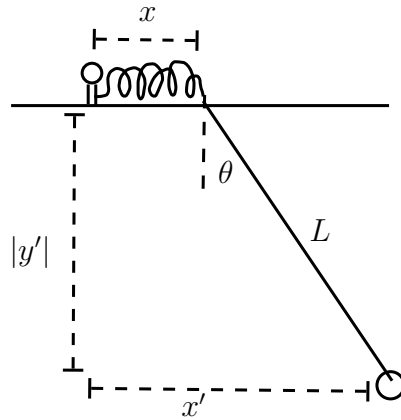


Figure 17.5: A diagram of the physical setup.

If we call (x', y') the spacial coordinate of pendulum bob, we can write these coordinates as

$$x' = x + L \sin \theta \quad (17.50)$$

$$y' = -L \cos \theta \quad (17.51)$$

where x is the displacement of the spring and θ is the angle swept out by the spring (see figure 17.5). We can write the Lagrangian as:

$$L = T - U \quad (17.52)$$

$$= \frac{1}{2}m(\dot{x}'^2 + \dot{y}'^2) - mgy' - \frac{1}{2}kx^2 \quad (17.53)$$

$$= \frac{1}{2}m(\dot{x}^2 + 2L\dot{x}\dot{\theta}\cos\theta + L^2\dot{\theta}^2) + mgL\cos\theta - \frac{1}{2}kx^2 \quad (17.54)$$

The equations of motion are

$$\frac{\partial L}{\partial \theta} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}} \right) = 0 \quad (17.55)$$

$$\frac{\partial L}{\partial x} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{x}} \right) = 0 \quad (17.56)$$

Plugging into the first equation gets

$$-mL\dot{x}\dot{\theta}\sin\theta - mgL\sin\theta - \frac{d}{dt}(mL\dot{x}\cos\theta + mL^2\dot{\theta}) = 0. \quad (17.57)$$

This simplifies to

$$-\frac{g}{L}\sin\theta - \frac{\ddot{x}}{L}\cos\theta - \ddot{\theta} = 0. \quad (17.58)$$

Note that this reduces when $x = 0$ to the equation of state for a regular pendulum. Plugging into the second equation gets

$$-kx - \frac{d}{dt}(m\dot{x} + mL\dot{\theta}\cos\theta) = 0. \quad (17.59)$$

This becomes

$$-kx - m\ddot{x} - mL\ddot{\theta}\cos\theta + mL\dot{\theta}^2\sin\theta = 0. \quad (17.60)$$

This equation reduces when $\theta = 0$ to the equation of state for a regular spring. We can take a small angle limit by letting $\cos\theta \rightarrow 1$ and $\sin\theta \rightarrow \theta$. When we do this, our equations become

$$-\frac{g}{L}\theta - \frac{\ddot{x}}{L} - \ddot{\theta} = 0 \quad (17.61)$$

$$-\frac{k}{m}x - L\ddot{\theta} + L\dot{\theta}^2\theta - \ddot{x} = 0 \quad (17.62)$$

It is not at all clear to me what I am supposed to do to these equations to make the angle dependence look like a simple harmonic oscillator, so I am not sure now to figure out the frequency of oscillations...

17.5 Problem

“A bucket full of water rotates at uniform angular velocity ω . It is near the surface of the earth, so there’s a uniform downward field g . What shape does the surface of the water make? Be as specific as you can” – Travis Norsen

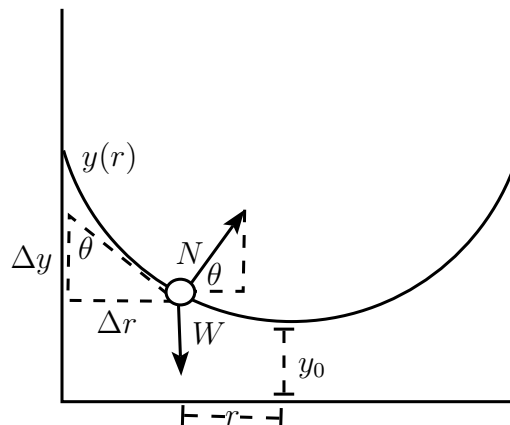
We can assume that the bucket has been rotating for a sufficiently long time that the water will all be rotating along with the bucket at constant angular velocity. We can also assume that the system has been rotating long enough that it has come to equilibrium. Figure 17.6 shows a free body diagram for a small chunk of water on the very surface of the bucket. It experiences two forces: gravity and a normal force from the rest of the water. We know that the particle is not moving up or down so

$$\sum F_y = 0. \quad (17.63)$$

We know that it is experiencing uniform circular motion, so

$$\sum F_x = mr\omega^2. \quad (17.64)$$

Figure 17.6: A free body diagram of a bucket rotating with angular velocity ω . The free body diagram is of some small chunk of water a distance r from the axis of rotation. There are only two forces acting on the chunk of water.



From equation 17.63, we see that

$$N \sin \theta = W. \quad (17.65)$$

From equation 17.64, we see that

$$N \cos \theta = mr\omega^2. \quad (17.66)$$

Dividing gets

$$\tan \theta = \frac{g}{r\omega^2} \quad (17.67)$$

We can read from the diagram that

$$\tan \theta = \frac{\Delta r}{\Delta y} = \frac{dr}{dy} \quad (17.68)$$

We can combined these equations to get

$$\frac{dr}{dy} = \frac{g}{r\omega^2} \quad (17.69)$$

$$dy = \frac{\omega^2}{g} r dr. \quad (17.70)$$

Integrating gets us

$$y = y_0 + \frac{1}{2} \frac{\omega^2}{g} r^2. \quad (17.71)$$

where y_0 is the height of the water at the axis of rotation. We see that this is the equation of a parabola. I think that this means the surface area will be a paraboloid.

Chapter 18

Comprehensive Exam, part 2 - Covers: E&M, Electrodynamics, Circuits and Optics

18.1 Problem

“A charge $+Q$ is distributed uniformly along the z axis from $z = -a$ to $z = +a$. Find an exact expression for the electrostatic potential for points along the z axis (with $z > a$). Then use this to write an approximate expression for the potential at a point (r, θ) not (necessarily) on the z axis. The approximation should be a power series expansion in a/r and should be accurate to fourth order in a/r .” 🧠

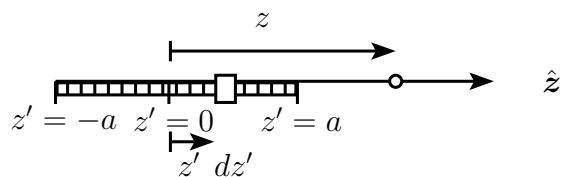


Figure 18.1: A rod.

We can calculate the potential due to the small bit of charge shown in figure 18.1.

$$dV = \frac{1}{4\pi\epsilon_0} \frac{\lambda dz'}{z - z'} \quad (18.1)$$

Integrating over all of the rod gets us the total potential

$$V(z) = \frac{1}{4\pi\epsilon_0} \int_{-a}^a \frac{\lambda dz'}{z - z'} \quad (18.2)$$

$$= \frac{\lambda}{4\pi\epsilon_0} (-\log(z - z'))|_{-a}^a \quad (18.3)$$

$$= \frac{\lambda}{4\pi\epsilon_0} \log\left(\frac{z + a}{z - a}\right) \quad (18.4)$$

This expression is valid for $z > a$. Of course, for $z < a$, we have charge immediately at our point so our potential function would be infinite. If we let $u = a/z$, we have

$$V = \frac{\lambda}{4\pi\epsilon_0} \log \left(\frac{1+u}{1-u} \right) \quad (18.5)$$

We can expand V in powers of u around $u = 0$. To do so, we have to calculate a whole bunch of derivatives. Let

$$f(u) = \log \left(\frac{1+u}{1-u} \right) \quad (18.6)$$

Then,

$$f(0) = 0 \quad (18.7)$$

Next, we calculate

$$f'(u) = \frac{1}{\left(\frac{1+u}{1-u}\right)} \left(\frac{(1-u) - (1+u)(-1)}{(1-u)^2} \right) \quad (18.8)$$

$$= \frac{2}{(1+u)(1-u)} \quad (18.9)$$

$$f'(0) = 2. \quad (18.10)$$

$$(18.11)$$

Then we calculate

$$f''(u) = \frac{2(-1)(-(1+u) + (1-u))}{(1+u)^2(1-u)^2} \quad (18.12)$$

$$= \frac{4u}{(1+u)^2(1-u)^2} \quad (18.13)$$

$$f''(0) = 2. \quad (18.14)$$

$$(18.15)$$

Then we calculate

$$f'''(u) = \frac{4}{(1+u)^2(1-u)^2} + \frac{4u(-2)(-(1+u) + (1-u))}{(1+u)^3(1-u)^3} \quad (18.16)$$

$$= \frac{12u^2 + 4}{(1+u)^3(1-u)^3} \quad (18.17)$$

$$f'''(0) = 4. \quad (18.18)$$

$$(18.19)$$

Then we calculate

$$f'''(u) = \frac{(1+u)^3(1-u)^3 24u - (12u^2 + 4)3(1+u)^2(1-u)^2(-2u)}{(1+u)^6(1-u)^6} \quad (18.20)$$

$$= \frac{48u(1+u^2)}{(1+u)^4(1-u)^4} \quad (18.21)$$

$$f'''(0) = 0. \quad (18.22)$$

$$(18.23)$$

Then we calculate

$$f''''(u) = 48 \left(\frac{(1+u)^4(1-u)^4(1+u^2+u(2u)) - u(1+u^2)4(1+u)^3(1-u)^3(-2u)}{(1+u)^8(1-u)^8} \right) \quad (18.24)$$

$$= 48 \frac{5u^4 + 10u^2 + 1}{(1+u)^5(1-u)^5} \quad (18.25)$$

$$f''''(0) = 48. \quad (18.26)$$

We can now do the expansion

$$V(z) = \frac{\lambda}{4\pi\epsilon_0} f(u) \quad (18.27)$$

$$= \frac{\lambda}{4\pi\epsilon_0} \left(f(0) + f'(0)u + \frac{f''(0)}{2!}u^2 + \frac{f'''(0)}{3!}u^3 + \frac{f''''(0)}{4!}u^4 + \frac{f'''''(0)}{5!}u^5 + \dots \right) \quad (18.28)$$

$$= \frac{\lambda}{4\pi\epsilon_0} \left(2u + \frac{4}{3!}u^2 + \frac{48}{5!}u^4 + \dots \right) \quad (18.29)$$

$$= \frac{Q}{4\pi\epsilon_0} \frac{1}{z} \left(1 + \frac{1}{3} \left(\frac{a}{z} \right)^2 \frac{1}{5} \left(\frac{a}{z} \right)^4 \right) \quad (18.30)$$

Where in the last step we use the fact that $Q = 2\lambda a$. Finally, we can write the potential due to this charge distribution at any arbitrary point as the expansion of Legendre polynomials:

$$V(r, \theta) = \sum_{l=0}^{\infty} \left(A_l r^l + \frac{B_l}{r^{l+1}} \right) P_l(\cos \theta) \quad (18.31)$$

Note that this expansion was derived with the assumption of azimuthal symmetry, which is valid in this situation. Since $V \rightarrow 0$ as $r \rightarrow \infty$, we have $A_l = 0$:

$$V(r, \theta) = \sum_{l=0}^{\infty} \left(\frac{B_l}{r^{l+1}} \right) P_l(\cos \theta) \quad (18.32)$$

We can consider the special case of $\theta = 0$. We exploit the special property that $P_l(\cos 0) = P_l(1) = 1$. Since $z = r \cos \theta$, $r = z$ when $\theta = 0$. Thus,

$$V(r = z, 0) = \frac{B_0}{r} + \frac{B_1}{r^2} + \frac{B_2}{r^3} + \frac{B_3}{r^4} + \frac{B_4}{r^5} + \dots \quad (18.33)$$

By examining our Taylor expansion for the potential on the z axis, we recognize that

$$B_0 = \frac{Q}{4\pi\epsilon_0} \quad (18.34)$$

$$B_1 = 0 \quad (18.35)$$

$$B_2 = \frac{Q}{4\pi\epsilon_0} \frac{a^2}{3} \quad (18.36)$$

$$B_3 = 0 \quad (18.37)$$


$$B_4 = \frac{Q}{4\pi\epsilon_0} \frac{a^4}{5} \quad (18.38)$$

$$(18.39)$$

So,

$$V(r, \theta) = \frac{Q}{4\pi\epsilon_0} \frac{1}{r} \left(P_0(\cos \theta) + \frac{1}{3} \left(\frac{a}{z} \right)^2 P_2(\cos \theta) + \frac{1}{5} \left(\frac{a}{z} \right)^4 P_4(\cos \theta) + \dots \right) \quad (18.40)$$

18.2 Problem

“A long coaxial cable is made from two conducting cylindrical shells of radius a and b . (The space between them is empty.) At one end of the cable, the inner conductor is attached to the positive terminal of a battery (potential $+V$); the outer conductor is attached to the negative terminal (potential zero). At the other end of the cable, the two conductors are connected through a resistor (resistance R). Note that the inner conductor will have some charge per unit length, and will also have some current flowing through it. Find the electric field in the cable for $a < r < b$. Then find the magnetic field in the cable for $a < r < b$. Then find the Poynting vector for $a < r < b$. Then integrate to find the total rate at which electromagnetic field energy flows along the cable. Finally, say something about where this energy comes from and where its headed.” 

The current can be calculated as $I = V/R$. The voltage across the resistor falls off linearly. The reason why we know this is because resistors are made up out of linear materials so that the amount of resistance is proportional to the length of the resistor. We could think of dividing up the resistor into two parts, as is shown in figure 18.3. The resistances shown on the two parts of the resistor in the figure are linear as desired. Since the voltage at a is V , the voltage at s will then have decreased by the voltage times the resistance:

$$V(s) = V - I \left(\frac{s-a}{b-a} R \right) \quad (18.41)$$

$$= \frac{b-s}{b-a} V \quad (18.42)$$

If you notice, this is the linear decrease that I stated above. Now, what we are interested in is the voltage everywhere inside the cylinders constrained by the boundary conditions:

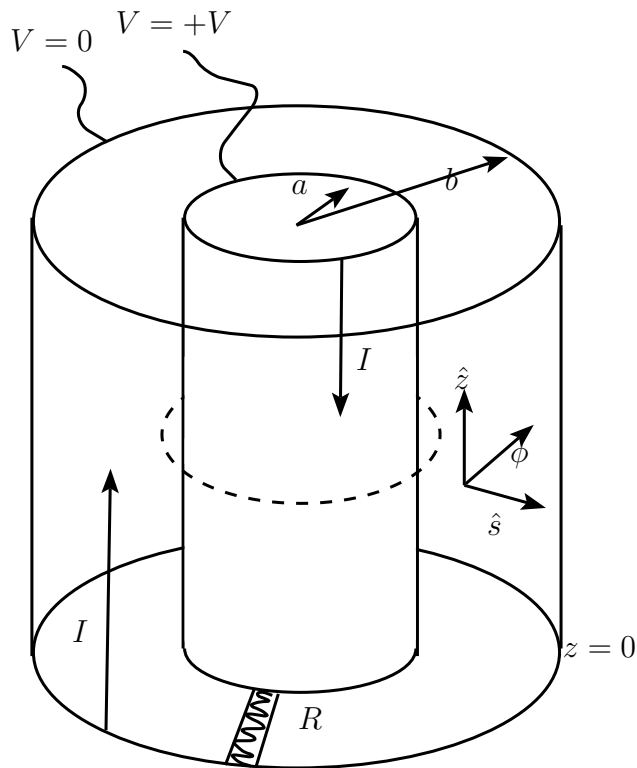


Figure 18.2: Two cylinders connected by a resistor. The sketchy dashed line is an Amperian surface that we will use later.

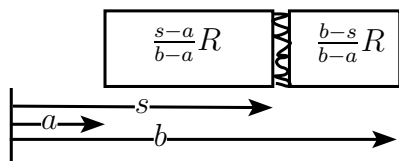


Figure 18.3: We can imagine dividing the resistor into two smaller resistors. This is used to calculate the voltage at some point in the middle of the resistor.

- $V = +V$ for $x = a$.
- $V = 0$ for $x = b$.
- $V(s) = \frac{b-s}{b-a}V$ across the resistor for $a \leq s \leq b$.

Of course, one obvious solution to Laplace's equation for these boundary conditions is

$$V(s) = \frac{b-s}{b-a}V \quad (18.43)$$

for $a \leq s \leq b$. By the 1st uniqueness theorem from Griffith's book, we know that this must be the only unique solution for the potential.

We can calculate the electric field as

$$\mathbf{E} = -\nabla V = -\frac{\partial V}{\partial s}\hat{s} = \frac{V}{b-s}\hat{s}. \quad (18.44)$$

To calculate the magnetic field, we can draw an Amperian loop with radius $a \leq s \leq b$ around the first cylinder. This is the silly dashed line in figure 18.2. We use Ampère's law:

$$\oint \mathbf{B} \cdot d\mathbf{l} = \mu_0 I_{\text{end}}. \quad (18.45)$$

This gets us

$$B = \frac{\mu_0 I}{2\pi s}. \quad (18.46)$$

Using the rule where you put your thumb in the direction of the current and your fingers curl in the direction of the magnetic field, we get

$$\mathbf{B} = -\frac{\mu_0 I}{2\pi s} \hat{\phi} \quad (18.47)$$

We can now calculate the Poynting vector, which is the energy per unit time per unit area that flows in some direction.

$$\mathbf{S} = \frac{1}{\mu_0} (\mathbf{E} \times \mathbf{B}) \quad (18.48)$$

$$= \frac{1}{\mu_0} \left(\frac{V}{b-a} \right) \left(\frac{\mu_0 I}{2\pi s} \right) \hat{s} \times (-\hat{\phi}) \quad (18.49)$$


$$= -\frac{V}{b-a} \frac{I}{2\pi s} \hat{z} \quad (18.50)$$

We can calculate the total energy flowing down the area between the cylinder per unit time:

$$\frac{dE}{dt} = \int_S \mathbf{S} \cdot d\mathbf{a} = \int_0^{2\pi} \int_a^b \frac{IV}{2\pi s(b-a)} s ds d\phi = IV = I^2 R \quad (18.51)$$

This means that energy $I^2 R$ per unit time is carried by the field down the cylinder. This energy must be taken to the resistor and dissipated as Joule eating. *The energy is used to heat up the resistor.*

18.3 Problem

“An AC voltage source (amplitude V_0 , angular frequency ω) drives a circuit consisting of a resistor (R) in series with a capacitor (C). What is the amplitude of the voltage across the resistor? Describe qualitatively what will happen (i.e., what it will sound like) if the voltage source is replaced by a radio and the resistor is replaced by a speaker.” 

kirchoff's Law gives us the equation we need:

$$V_0 e^{i\omega t} - \frac{Q}{C} - IR = 0 \quad (18.52)$$

$$CV_0 e^{i\omega t} - Q - Q'RC = 0 \quad (18.53)$$

$$(18.54)$$

We guess a solution of the form

$$Q = ke^{i\omega t} \quad (18.55)$$

with k some constant. Plugging in gets us we get

$$CV_0e^{i\omega t} - ke^{i\omega t} - i\omega ke^{i\omega t}RC = 0 \quad (18.56)$$

$$CV_0 - k(1 + i\omega RC) = 0 \quad = \frac{CV_0}{1 + i\omega RC} \quad (18.57)$$

Thus,

$$I = \frac{i\omega CV_0}{1 + i\omega RC}e^{i\omega t} = \frac{V_0}{\frac{1}{i\omega C} + R}e^{i\omega t} \quad (18.58)$$

The voltage drop across the resistor is:

$$V = IR = \frac{V_0 R}{\frac{1}{i\omega C} + R}e^{i\omega t} \quad (18.59)$$

The formula shows that there is a larger voltage drop across the resistor for larger angular frequency.

Suppose the voltage source was replaced by a radio and the resistor by a speaker. The radio would output a signal with many different frequencies. The radio would make a sound of each particular frequency proportional to the voltage drop across it for that particular frequency. Since our circuit is set up so that the voltage drop across the speaker is bigger for larger frequencies, our speaker would preferentially play high frequency sound. We would hear the radio with a bias towards the high frequencies. I think this circuit is called a high pass filter.

By the way, the derivation would have been much easier if I used impedance, but I couldn't remember exactly how to calculate it and I didn't have my Optics book.

18.4 Problem


“Finn has a toy magnifying glass designed to look at bugs. Its a cylinder whose bottom is a platform where you can put a bug, and whose top is a converging lens. With the cylinder sitting on a table, you can then look down from above and see a magnified image of the bug through the lens. Suppose the focal length of the lens is 10cm, and the platform where the bugs sit is 5cm behind the lens. Draw a diagram showing the primary rays and indicating the size and location of the image that is seen. If Finn puts his eye a distance d above the lens, what is the magnification? What range of magnifications is possible? How could the device be modified to achieve greater magnification?” 

Figure 18.4 shows an optics diagram of the toy with the primary rays.¹ Note that f labeled in the diagram is the focal length. We have $d = 5$ cm and $f = 10$ cm Using similar

¹Travis, I think figured out the hint that you were getting at. I figured out the diagram on my own. Afterwards, I went to the Wikipedia page on convex lenses and saw the same diagram. But I came up with all of this work before going there

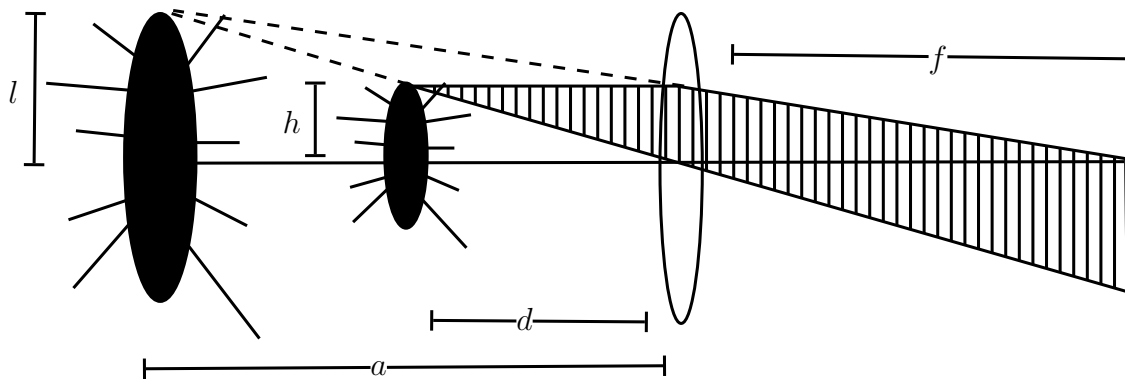


Figure 18.4: The bug, the lens, and the virtual image. *Eeew, bugs!*

triangles

$$\frac{l}{a} = \frac{h}{d} \quad (18.60)$$

and

$$\frac{l}{a+f} = \frac{h}{f}. \quad (18.61)$$

So,

$$a = \frac{d}{h} \times l \quad (18.62)$$

and

$$\frac{l}{\frac{d}{h}l + f} = \frac{h}{f} \quad (18.63)$$

$$l = \frac{d}{f} \times l + h \quad (18.64)$$

$$l \left(1 - \frac{d}{f} \right) = h. \quad (18.65)$$

I believe that we define the magnification M as

$$M \equiv \frac{l}{h}. \quad (18.66)$$

Finally,

$$M = \frac{1}{1 - \frac{d}{f}}. \quad (18.67)$$

Since $d/f = 1/2$, we have $M = 2$. I don't think there is any range to the magnification (is this a trick question?) We could increase the magnification by increasing d/f either by increasing the distance between the bug and the lens or by decreasing the focal length.

18.5 Problem

“Find the transmission coefficient for light waves passing through a pane of glass, of thickness d , at normal incidence. Hints: To the left, there is an incident wave and a reflected wave; to the right there is a transmitted wave only; inside the glass there is a wave going to the right and a wave going to the left. Express each of these waves in terms of its complex amplitude, and relate the amplitudes by imposing suitable boundary conditions at the two edges. Neglect dispersion and assume $\mu = \mu_0$. It is simplest to characterize the light by its wave number in the glass. (This is problem 8.39 in Griffiths *E&M*, 2nd ed.) Note the similarity to a certain standard modern physics type of problem (reflection/transmission from a rectangular potential barrier with “height” V_0). Does the example here with light correspond to $E > V_0$ or $E < V_0$? Is there an analog with light for the other case, too?” 🌐

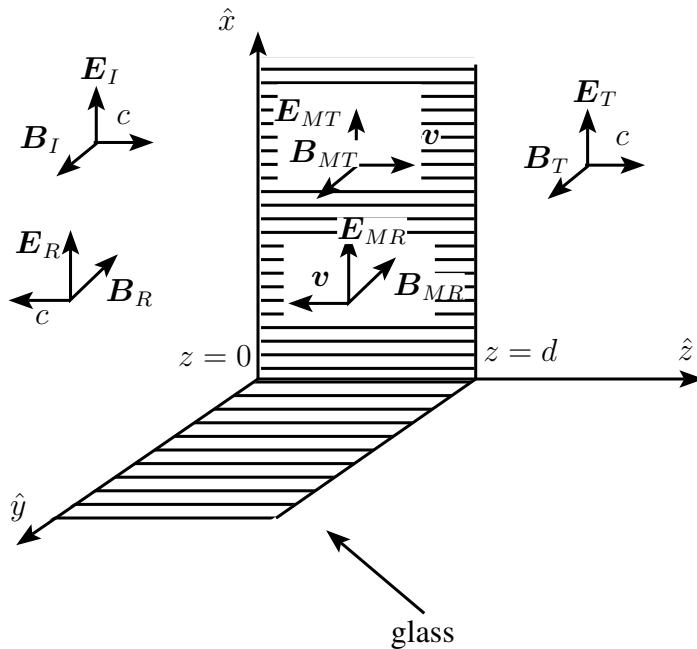


Figure 18.5: This figure shows the glass barrier and the orientation of the waves in the three sections. This whole setup is very similar to the example in Griffith’s book called *Reflection and Transmission at Normal Incidence*.

There is an incident and reflected wave to the left of the glass. There is also an incident and reflected wave in the glass. There is only a transmitted wave to the right of the glass. This is shown in figure 18.5. Outside the glass, the waves travel with velocity c and wave vector k . Inside the glass, the waves travel with velocity $v = c/n$ and wave vector $\kappa = nk$ (since the angular frequency ω is the same inside and outside so $kc = \omega = \kappa v$). Thus, the

incoming and reflected waves are:

$$\mathbf{E}_I = \tilde{E}_{0I} e^{i(kz - \omega t)} \hat{x} \quad (18.68)$$

$$\mathbf{B}_I = \frac{1}{c} \tilde{E}_{0I} e^{i(kz - \omega t)} \hat{y} \quad (18.69)$$

$$\mathbf{E}_R = \tilde{E}_{0R} e^{i(-kz - \omega t)} \hat{x} \quad (18.70)$$

$$\mathbf{B}_R = -\frac{1}{c} \tilde{E}_{0R} e^{i(-kz - \omega t)} \hat{y}. \quad (18.71)$$

Inside the glass, we have

$$\mathbf{E}_{MT} = \tilde{E}_{0MT} e^{i(\kappa z - \omega t)} \hat{x} \quad (18.72)$$

$$\mathbf{B}_{MT} = \frac{n}{c} \tilde{E}_{0MT} e^{i(\kappa z - \omega t)} \hat{y} \quad (18.73)$$

$$\mathbf{E}_{MR} = \tilde{E}_{0MR} e^{i(-\kappa z - \omega t)} \hat{x} \quad (18.74)$$

$$\mathbf{B}_{MR} = -\frac{n}{c} \tilde{E}_{0MR} e^{i(-\kappa z - \omega t)} \hat{y}. \quad (18.75)$$

The transmitted wave is

$$\mathbf{E}_T = \tilde{E}_{0T} e^{i(kz - \omega t)} \hat{x} \quad (18.76)$$

$$\mathbf{B}_T = \frac{1}{c} \tilde{E}_{0T} e^{i(kz - \omega t)} \hat{y}. \quad (18.77)$$

Since we are assuming that $\mu_{\text{air}} = \mu_{\text{glass}}$, our boundary conditions are:

$$E_{\text{left}}^{//}|_{z=0} = E_{\text{middle}}^{//}|_{z=0} \quad (18.78)$$

$$B_{\text{left}}^{//}|_{z=0} = B_{\text{middle}}^{//}|_{z=0} \quad (18.79)$$

$$E_{\text{middle}}^{//}|_{z=d} = E_{\text{right}}^{//}|_{z=d} \quad (18.80)$$

$$B_{\text{middle}}^{//}|_{z=d} = B_{\text{right}}^{//}|_{z=d} \quad (18.81)$$

with $\mathbf{E}_{\text{left}} = \mathbf{E}_I + \mathbf{E}_R$ and $\mathbf{E}_{\text{middle}} = \mathbf{E}_{MR} + \mathbf{E}_{MT}$ and $\mathbf{E}_{\text{right}} = \mathbf{E}_T$. Pluggin' in:

$$\tilde{E}_{OI} + \tilde{E}_{OR} = \tilde{E}_{0MT} + \tilde{E}_{0MR} \quad (18.82)$$

$$\frac{1}{c} \tilde{E}_{OI} - \frac{1}{c} \tilde{E}_{OR} = \frac{n}{c} \tilde{E}_{0MT} - \frac{n}{c} \tilde{E}_{0MR} \quad (18.83)$$

And:

$$\tilde{E}_{0MT} e^{i\kappa d} + \tilde{E}_{0MR} e^{-i\kappa d} = \tilde{E}_{0T} e^{ikd} \quad (18.84)$$

$$\frac{n}{c} \tilde{E}_{0MT} e^{i\kappa d} - \frac{n}{c} \tilde{E}_{0MR} e^{-i\kappa d} = \frac{1}{c} \tilde{E}_{0T} e^{ikd} \quad (18.85)$$

We want to calculate

$$R = \left| \frac{E_{0R}}{E_{0I}} \right|^2 \quad (18.86)$$

and

$$T = \left| \frac{E_{0T}}{E_{0I}} \right|^2 \quad (18.87)$$

Remember that since the medium that E_{0T} is in is the same as the medium that E_{0I} is in, there are no constants in the front of the T term. Using equation 18.82 and 18.83, we get

$$\tilde{E}_{OI} = \frac{1+n}{2} \tilde{E}_{0MT} + \frac{1-n}{2} \tilde{E}_{0MR} \quad (18.88)$$

$$\tilde{E}_{OR} = \frac{1-n}{2} \tilde{E}_{0MT} + \frac{1+n}{2} \tilde{E}_{0MR} \quad (18.89)$$

$$(18.90)$$

Using equation 18.84 and 18.85, we get

$$\tilde{E}_{OMT} = \frac{1}{2} \times \frac{1+n}{n} \tilde{E}_{0T} \frac{e^{ikd}}{e^{i\kappa d}} \quad (18.91)$$

$$\tilde{E}_{OMR} = \frac{1}{2} \times \frac{n-1}{n} \tilde{E}_{0T} \frac{e^{ikd}}{e^{-i\kappa d}} \quad (18.92)$$

From this, we get

$$\tilde{E}_{OI} = \left(\frac{(n+1)^2}{4n} \frac{e^{ikd}}{e^{i\kappa d}} - \frac{(n-1)^2}{4n} \frac{e^{ikd}}{e^{-i\kappa d}} \right) \tilde{E}_{0T} \quad (18.93)$$

After doing some math, we get

$$T = \frac{16n^2}{(n+1)^4 + (n-1)^4 - 2(n+1)^2(n-1)^2 \cos(2\kappa d)} \quad (18.94)$$

$$= \frac{16\kappa^2 k^2}{(\kappa+k)^4 + (\kappa-k)^4 - 2(\kappa+k)^2(\kappa-k)^2 \cos(2\kappa d)}. \quad (18.95)$$

I calculated that

$$\lim_{d \rightarrow 0} T = 1, \quad (18.96)$$

as is needed. We can now calculate R in just the same way:

$$\tilde{E}_{0R} = \frac{1-n}{2} \left(\frac{1}{2} \times \frac{1+n}{n} \tilde{E}_{0T} \frac{e^{ikd}}{e^{i\kappa d}} \right) + \frac{1+n}{2} \left(\frac{1}{2} \times \frac{n-1}{n} \tilde{E}_{0T} \frac{e^{ikd}}{e^{-i\kappa d}} \right) \quad (18.97)$$

Simplifying a bit, this becomes:

$$\tilde{E}_{0R} = \frac{n^2-1}{4n} (-2i) \sin(\kappa d) \tilde{E}_{0T} \quad (18.98)$$

From which we calculate that

$$R = \left| \frac{E_{0R}}{E_{0I}} \right|^2 = \left| \frac{E_{0R}}{E_{0T}} \right|^2 \left| \frac{E_{0T}}{E_{0I}} \right|^2 \quad (18.99)$$

$$= \frac{4(n^2 - 1)^2 \sin^2(\kappa d)}{(n + 1)^4 + (n - 1)^4 - 2(n + 1)^2(n - 1)^2 \cos(2\kappa d)} \quad (18.100)$$

$$= \frac{2(1 - \cos(2\kappa d))}{\left(\frac{\kappa+k}{\kappa-k}\right)^2 + \left(\frac{\kappa-k}{\kappa+k}\right)^2 - 2 \cos(2\kappa d)} \quad (18.101)$$

By working it out, I showed that

$$\lim_{d \rightarrow 0} R = 0 \quad (18.102)$$

I will save you the tedious algebra, but suffice it to note that I worked through all the math and $T + R = 1$, as is required.

I believe that this example corresponds to $E > V$ scattering since most of the light goes through the glass and is transmitted (just as how for $E > V$ most of the light is transmitted. I don't think there is an optical analog for an $E < V$ barrier.

18.6 Problem

“A circular coil of wire (radius R) carries current I and lies in the $x - y$ plane with its center at the origin. (So the z -axis is its symmetry axis.) Find an exact expression for the strength of the magnetic field along the z axis. Now: a second identical coil (parallel to the first) is to be placed with its center at $z = d$. It is desired to make the magnetic field in the region near the center of the coils (i.e., near $z = d/2$) as uniform as possible. Find the value of d which accomplishes this.” ☞

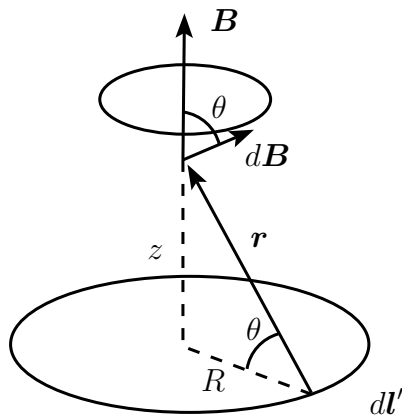


Figure 18.6

Figure 18.6 shows the physical setup. Biot-Savart's Law tells us that

$$\mathbf{B}(\mathbf{r}) = \frac{\mu_0 I}{4\pi} \int \frac{d\mathbf{l}' \times \hat{r}}{r^2} \quad (18.103)$$

Since the different parts of the coil have their x-y components of the magnetic field cancel out, we have

$$\mathbf{B}(\mathbf{r}) = B(z)\hat{z} \quad (18.104)$$

We can solve Biot-Savart's equation:

$$B(z) = \frac{\mu_0 I}{4\pi} \int \frac{dl' \cos \theta}{r^2} \quad (18.105)$$

$$= \frac{\mu_0 I \cos \theta}{4\pi r^2} \times 2\pi R \quad (18.106)$$

$$= \frac{\mu_0 I R^2}{2} \frac{1}{(R^2 + z^2)^{3/2}} \quad (18.107)$$

Notice that even for $z < 0$, the field points up! If we put another coil at a height d , the magnetic field for $0 \leq z \leq d$ is

$$B(z) = \frac{\mu_0 I R^2}{2} \left(\frac{1}{(R^2 + z^2)^{3/2}} + \frac{1}{(R^2 + (d - z)^2)^{3/2}} \right) \quad (18.108)$$

We calculate

$$\frac{\partial B}{\partial z} = \frac{\mu_0 I R^2}{2} \left(-\frac{3z}{(R^2 + z^2)^{5/2}} + \frac{3(d - z)}{(R^2 + (d - z)^2)^{5/2}} \right) \quad (18.109)$$

And

$$\begin{aligned} \frac{\partial^2 B}{\partial z^2} = \frac{\mu_0 I R^2}{2} \left(-\frac{3}{(R^2 + z^2)^{5/2}} + \frac{15z^2}{(R^2 + z^2)^{7/2}} \right. \\ \left. - \frac{3}{(R^2 + (d - z)^2)^{5/2}} + \frac{15(d - z)^2}{(R^2 + (d - z)^2)^{7/2}} \right) \end{aligned} \quad (18.110)$$

Note that

$$\left. \frac{\partial B}{\partial z} \right|_{z=\frac{d}{2}} = 0 \quad (18.111)$$

So we will set

$$\left. \frac{\partial^2 B}{\partial z^2} \right|_{z=\frac{d}{2}} = 0 \quad (18.112)$$

and see what condition arises for d . This value will correspond to the most constant magnetic field. When we do so we get

$$-(R^2 + (\frac{d}{2})^2) + f(\frac{d}{2})^2 = 0 \quad (18.113)$$

which simplifies to $d = R$. So, we can make B vary least at $z = \frac{d}{2}$ if we set $d = R$.

Chapter 19

Comprehensive Exam, part 3 - Covers: Modern Physics, Quantum Mechanics, Particle Physics

19.1 Problem

What is Compton scattering? What role did it play in the early days of quantum theory? You probably recall an argument that treats the photon and electron as classical particles, and derives the correct shift in wavelength for the scattered photon. What can you say (or guess or speculate or vaguely sketch) about how this process can be treated in a fully quantum mechanical way? (You shouldn't really have to do any calculations here at all. The last question amounts to: in broad, qualitative strokes, how might you apply some of the more advanced stuff you've learned recently to analyze Compton scattering?)

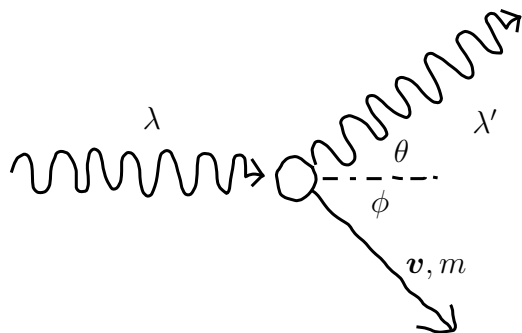


Figure 19.1: The Compton scattering diagram. Here, we have a particle of wavelength λ coming in from the left. It scatters off a particle of mass m and scatters at angle θ . The other particle recoils with velocity v and scatters at an angle ϕ .

Figure 19.1 shows the diagram that we need. A photon treated as a particle with wavelength λ comes in from the left at the speed of light c , collides with a particle at rest, and scatters off at some angle θ . After the collision, the photon has wavelength λ' . We can apply conservation of relativistic energy and momentum (where we use the quantum formulas for

the photon: $p = h/\lambda$ and $E = hc/\lambda$). Conservation of energy gets us

$$\frac{hc}{\lambda} + mc^2 = \frac{hc}{\lambda'} + \gamma mc^2. \quad (19.1)$$

Conservation of momentum gets us

$$\frac{h}{\lambda} = \frac{h}{\lambda'} \cos \theta + \gamma mv \cos \phi \quad (19.2)$$

$$\frac{h}{\lambda'} \sin \theta = \gamma mv \sin \phi. \quad (19.3)$$

Squaring and adding the equations:

$$\gamma^2 m^2 v^2 = \left(\frac{h}{\lambda} - \frac{h \cos \theta}{\lambda'} \right)^2 + \left(\frac{h \sin \theta}{\lambda'} \right)^2 \quad (19.4)$$

$$= \frac{h^2}{\lambda^2} - \frac{2h^2}{\lambda\lambda'} \cos \theta + \frac{h^2}{\lambda'^2}. \quad (19.5)$$

We can solve the energy conservation equation for v and use this to write $\gamma^2 m^2 v^2$ in terms of λ and λ' .

$$\gamma^2 m^2 c^4 = \left(\frac{hc}{\lambda} + mc^2 - \frac{hc}{\lambda'} \right)^2 \quad (19.6)$$

$$1 - \frac{v^2}{c^2} = \frac{m^2 c^4}{\left(\frac{hc}{\lambda} + mc^2 - \frac{hc}{\lambda'} \right)^2} \quad (19.7)$$

$$\frac{v^2}{c^4} = \frac{1}{c^2} - \frac{m^2 c^2}{\left(\frac{hc}{\lambda} + mc^2 - \frac{hc}{\lambda'} \right)^2} \quad (19.8)$$

So,

$$\gamma^2 m^2 v^2 = (\gamma^2 m^2 c^4) \left(\frac{v^2}{c^4} \right) \quad (19.9)$$

$$= \left(\frac{hc}{\lambda} + mc^2 - \frac{hc}{\lambda'} \right)^2 \left(\frac{1}{c^2} - \frac{m^2 c^2}{\left(\frac{hc}{\lambda} + mc^2 - \frac{hc}{\lambda'} \right)^2} \right) \quad (19.10)$$

$$= \frac{h^2}{\lambda^2} + \frac{h^2}{\lambda'^2} + 2 \frac{hmc}{\lambda} + 2 \frac{hmc}{\lambda'} - 2 \frac{h^2}{\lambda\lambda'} \quad (19.11)$$

Thus,

$$\frac{h^2}{\lambda^2} + \frac{h^2}{\lambda'^2} + 2 \frac{hmc}{\lambda} - 2 \frac{hmc}{\lambda'} - 2 \frac{h^2}{\lambda\lambda'} = \frac{h^2}{\lambda^2} - 2 \frac{h^2}{\lambda\lambda'} \cos \theta + \frac{h^2}{\lambda'^2} \quad (19.12)$$

$$(\lambda' - \lambda)mc = h(1 - \cos \theta) \quad (19.13)$$

$$\lambda' = \lambda + (h/mc)(1 - \cos \theta) \quad (19.14)$$

According to Griffith's Particle textbook:

What finally settled the issue [of whether the electric field is quantized as particles called photons] was an experiment conducted by A. H. Compton in 1923. Compton found that the light scattered from a particle at rest is shifted in wavelength, according the equation

$$\lambda' = \lambda + \lambda_c(1 - \cos \theta) \quad (19.15)$$

where λ is the incident wavelength, λ' is the scattered wavelength, θ is the scattering angle, and

$$\lambda_c = h/mc \quad (19.16)$$

is the so-called Compton wavelength of the target particle (mass m). Now this is precisely the formula you get (Problem 3.24) if you treat light as a particle of zero rest mass with energy given by Planck's equation, and apply the laws of conservation of (relativistic) energy and momentum—just what you would for an ordinary elastic collision. That clinched it; here was direct and incontrovertible experimental evidence that light behaves as a particle, on the subatomic scale.

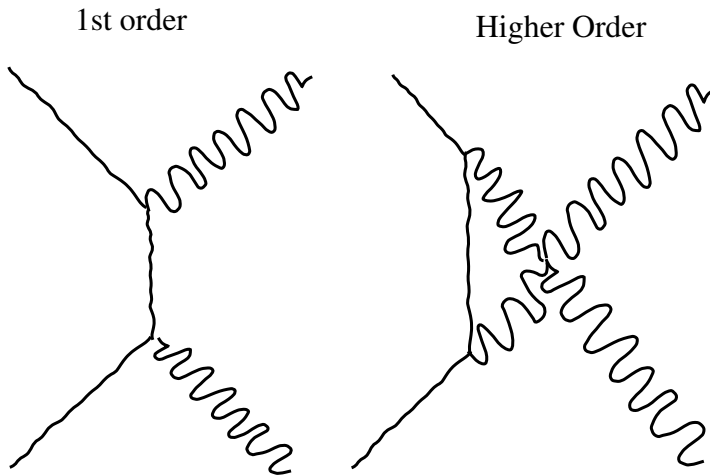


Figure 19.2: The Compton scattering Feynman diagrams. The first diagram is the 1st order Feynman diagram. The second is one of the many higher order diagrams.

In order to treat Compton scattering properly, you have to draw Feynman diagrams for the process and calculate the numbers associated with them. The sum of all the numbers would be the amplitude for Compton scattering to happen. Figure 19.2 shows the first order Feynman diagram for this process and one of the higher order terms. I am not exactly sure how you would use the Feynman calculus to figure out the wavelength shift for a particular angle since things like direction don't matter in Feynman diagrams...

19.2 Problem

Calculate the lifetime (in seconds) for each of the four $n = 2$ states of hydrogen. Hint: You'll need to evaluate matrix elements of the form $\langle \psi_{100} | y | \psi_{211} \rangle$, and so on. Remember that $x = r \sin \theta \cos \phi$, $y = r \sin \theta \sin \phi$, and $z = r \cos \theta$. Most of the integrals are zero, so think before you calculate.

Suppose we have a Hamiltonian $\hat{H} = \hat{H}_0 + \hat{H}_1$. We can write an arbitrary solution as a sum of the eigenstates of H_0 :

$$|\psi(t)\rangle = \sum_n c_n(t) e^{iE_n^{(0)}t/\hbar} |E_n^{(0)}\rangle. \quad (19.17)$$

According to time dependent perturbation theory, we can write the time dependent constants out front as

$$c_f(t) = \delta_{fi} - \frac{i}{\hbar} \int_0^t dt' e^{i(E_f^{(0)} - E_i^{(0)})t'/\hbar} \langle E_f^{(0)} | \hat{H}_1(t') | E_i^{(0)} \rangle. \quad (19.18)$$

We have

$$\left| \langle E_f^{(0)} | \hat{U}_I(t) | E_i^{(0)} \rangle \right|^2 = |c_f(t)|^2. \quad (19.19)$$

This is the probability of making a transition from the initial to the final state in time t .

We will apply this formalism to the case of the hydrogen atom in an electromagnetic field. We will treat the EM field as the perturbing Hamiltonian \hat{H}_1 and use the hydrogen eigenstates as the unperturbed basis.

We will deal with some initial state $|i\rangle$ that is an energy eigenstate of the hydrogen atom and $|f\rangle$ as some other energy eigenstate. Our goal will be to calculate the probability per unit time of making a transition from this state to a lower energy state by emitting a photon in any direction. Eventually, we will want to calculate the lifetime of the excited state.

The perturbing Hamiltonian is

$$\hat{H}_1 \rightarrow \frac{e}{m_e c} \hat{\mathbf{A}} \cdot \frac{\hbar}{i} \nabla + \frac{e^2}{2m_e c^2} \hat{\mathbf{A}}^2 \quad (19.20)$$

With

$$\hat{\mathbf{A}} = \sum_{\mathbf{k}, \lambda} c \sqrt{\frac{2\pi\hbar}{\omega}} \left(\hat{a}_{\mathbf{k}, \lambda} \boldsymbol{\epsilon}(\mathbf{k}, \lambda) \frac{e^{i(\mathbf{k} \cdot \mathbf{r})}}{\sqrt{V}} + \hat{a}_{\mathbf{k}, \lambda}^\dagger \boldsymbol{\epsilon}(\mathbf{k}, \lambda) \frac{e^{-i(\mathbf{k} \cdot \mathbf{r})}}{\sqrt{V}} \right) \quad (19.21)$$

Since \hat{H}_1 is independent of time, we have

$$|\langle f | \hat{U}_I | i \rangle|^2 = \frac{1}{\hbar^2} \left| \int_0^t dt' e^{i(E_f^{(0)} - E_i^{(0)})t'/\hbar} \right|^2 |\langle f | \hat{\mathbf{H}}_1 | i \rangle|^2 \quad (19.22)$$

$$= \frac{1}{\hbar^2} \frac{\sin^2((E_f^{(0)} - E_i^{(0)})t/2\hbar)}{(E_f^{(0)} - E_i^{(0)})/2\hbar} |\langle f | \hat{\mathbf{H}}_1 | i \rangle|^2 \quad (19.23)$$

This is the probability of having a transition from i to f with the emission of some particular photon of wave vector k . The time t imposed by experimental set-ups allows us to without loss of generality take the large time limit. In that case, we can use the representation of the Dirac delta function

$$\lim_{t \rightarrow \infty} \frac{1}{\pi} \frac{\sin^2(ta)}{ta^2} = \delta(a) \quad (19.24)$$

to write the transition probability as

$$\lim_{t \rightarrow \infty} |\langle f | \hat{U}_I | i \rangle|^2 = \frac{\pi t \delta((E_f^{(0)} - E_i^{(0)})/2\hbar)}{\hbar^2} |\langle f | \hat{\mathbf{H}}_1 | i \rangle|^2. \quad (19.25)$$

The total probability of having a photon emission is equal to the sum over all photons of the probability of that photon being emitted.

$$P = \sum_{\lambda} \sum_{\mathbf{k}} |\langle f | \hat{U}_I | i \rangle|^2 \quad (19.26)$$

The sum over λ accounts for the two polarizations of the photons. Following the discussion in the Townsend book, we are supposed to apply periodic boundary conditions to space. This quantized the photons so that their wave vector is always of the form

$$k_x L = 2\pi n_x \quad k_y L = 2\pi n_y \quad k_z L = 2\pi n_z \quad (19.27)$$

with $n_x, n_y, n_z = 0, \pm 1, \pm 2, \dots$. When we do this, we find that the number of states with wave vector between \mathbf{k} and $\mathbf{k} + d\mathbf{k}$ in the solid angle $d\Omega$ is

$$\left(\frac{L}{2\pi}\right)^3 k^2 dk d\Omega \quad (19.28)$$

with L the size of our box. Using $E = \hbar\omega$, we can write the number of states with energy between E and $E + dE$ as:

$$\frac{V}{(2\pi)^3} \frac{\omega^2}{\hbar c^3} dE d\Omega \quad (19.29)$$

Thus, our total probability of having a transition is

$$P = \sum_{\lambda} \sum_{\mathbf{k}} |\langle f | \hat{U}_I(t) | i \rangle|^2 = \sum_{\lambda} \int dE \int d\Omega \frac{\pi t \delta((E_f^{(0)} - E_i^{(0)})/2\hbar)}{\hbar^2} |\langle f | \hat{\mathbf{H}}_1(t) | i \rangle|^2 \frac{V \omega^2}{(2\pi)^3 \hbar c^3} \quad (19.30)$$

Of course, the total final energy is just the energy of the photon plus the energy of the electron: $E_f^{(0)} = E_{n_f} + E$. So we can write our delta function as

$$\delta((E_f^{(0)} - E_i^{(0)})/2\hbar) = \delta\left(\frac{E_{n_f} + E - E_{n_i}}{2\hbar}\right) = 2\hbar \delta(E - (E_{n_i} - E_{n_f})) \quad (19.31)$$

This delta function kills one of our integrals:

$$P = \sum_{\lambda} \int \frac{2\pi t}{\hbar} |\langle f | \hat{\mathbf{H}}_1 | i \rangle|^2 \frac{V \omega^2}{(2\pi)^3 \hbar c^3} \quad (19.32)$$

Thus, the probability of making the transition per unit time is

$$R = \sum_{\lambda} \int \frac{2\pi}{\hbar} |\langle f | \hat{\mathbf{H}}_1 | i \rangle|^2 \frac{V \omega^2}{(2\pi)^3 \hbar c^3}. \quad (19.33)$$

Since R is independent of time, we know that if we have N atoms an amount dN will decay in a time dt such that

$$dN = -NRdt \quad (19.34)$$

$$N(t) = N(0)e^{-Rt} = N(0)e^{-t/\tau} \quad (19.35)$$

Thus, the lifetime of our atom is $\tau = 1/R$. Now, we calculate

$$\langle f | \hat{H}_1 | i \rangle = \langle 1_{\mathbf{k},\lambda} | \langle n_f, l_f, m_f | \hat{H}_1 | n_i, l_i, m_i \rangle | 0 \rangle \quad (19.36)$$

Note that the second term in the perturbing Hamiltonian

$$\hat{H}_1 \rightarrow \frac{e}{m_e c} \hat{\mathbf{A}} \cdot \frac{\hbar}{i} \nabla + \frac{e^2}{2m_e c^2} \hat{\mathbf{A}}^2 \quad (19.37)$$

will contribute nothing to our integral because the terms in it are all of the form $\hat{a}_{\mathbf{k},\lambda} \hat{a}_{\mathbf{k},\lambda}$, $\hat{a}_{\mathbf{k},\lambda}^\dagger \hat{a}_{\mathbf{k},\lambda}^\dagger$, and $\hat{a}_{\mathbf{k},\lambda}^\dagger \hat{a}_{\mathbf{k},\lambda}$. None of these take us from state with no photons to a state with 1 photon. Thus, our inner product becomes

$$\begin{aligned} \langle 1_{\mathbf{k},\lambda} | \langle n_f, l_f, m_f | \hat{H}_1 | n_i, l_i, m_i \rangle | 0 \rangle = \\ \frac{e}{m_e c} e \sqrt{\frac{2\pi\hbar}{\omega V}} \int d^3r \psi_{n_f, l_f, m_f}^* e^{-i\mathbf{k} \cdot \mathbf{r}} \boldsymbol{\epsilon}(\mathbf{k}, \lambda) \cdot \frac{\hbar}{i} \nabla \psi_{n_i, l_i, m_i} (\langle 1_{\mathbf{k},\lambda} | \hat{a}_{\mathbf{k},\lambda}^\dagger | 0 \rangle) \end{aligned} \quad (19.38)$$

with

$$\langle 1_{\mathbf{k},\lambda} | \hat{a}_{\mathbf{k},\lambda}^\dagger | 0 \rangle = \langle 1_{\mathbf{k},\lambda} | 1_{\mathbf{k},\lambda} \rangle. \quad (19.39)$$

We make the approximation

$$e^{-i\mathbf{k} \cdot \mathbf{r}} \rightarrow 1. \quad (19.40)$$

This is called the electric dipole transition. Now, we can use a trick to simplify equation 19.38. The trick is:

$$[\hat{H}_0, \hat{x}_i] = \left[\frac{\mathbf{p}^2}{2\mu}, \hat{x}_i \right] \quad (19.41)$$

$$= \sum_j \left[\frac{\hat{p}_j \hat{p}_j}{2\mu}, \hat{x}_i \right] \quad (19.42)$$

$$= \frac{1}{2\mu} \sum_j (\hat{p}_j [\hat{p}_j, \hat{x}_i] + [\hat{p}_j, \hat{x}_i] \hat{p}_j) \quad (19.43)$$

$$(19.44)$$

$$= -\frac{1}{\mu} \sum_j \hat{p}_j i\hbar \delta_{i,j} = -\frac{i\hbar}{\mu} \hat{p}_i \quad (19.45)$$

Therefore,

$$\langle n_f, l_f, m_f | \hat{p}_i | n_i, l_i, m_i \rangle = \frac{i\mu}{\hbar} \langle n_f, l_f, m_f | [\hat{\mathbf{H}}_0, \hat{x}_i] | n_i, l_i, m_i \rangle \quad (19.46)$$

$$= \frac{i\mu}{\hbar} (E_{n_f} - E_{n_i}) \langle n_f, l_f, m_f | \hat{x}_i | n_i, l_i, m_i \rangle \quad (19.47)$$

$$= -i\mu\omega \langle n_f, l_f, m_f | \hat{x}_i | n_i, l_i, m_i \rangle \quad (19.48)$$

Thus,

$$\langle n_f, l_f, m_f | \hat{\mathbf{p}} | n_i, l_i, m_i \rangle = -i\mu\omega \langle n_f, l_f, m_f | \hat{\mathbf{r}} | n_i, l_i, m_i \rangle \quad (19.49)$$

Therefore, we can replace $\frac{\hbar}{i}\nabla = \hat{\mathbf{p}}$ in equation 19.38 with $-i\mu\omega\hat{\mathbf{r}}$. When we do this, we get

$$R = \sum_{\lambda} \int \frac{2\pi}{\hbar} \left(c \sqrt{\frac{2\pi\hbar}{\omega V}} \right)^2 e^2 \frac{\omega^2}{c^2} \left| \int d^3r \psi_{n_f, l_f, m_f}^* \mathbf{r} \cdot \boldsymbol{\epsilon}(\mathbf{k}, \lambda) \psi_{n_i, l_i, m_i} \right|^2 \frac{V\omega^2 d\Omega}{(2\pi)^3 c^3 \hbar} \quad (19.50)$$

$$= \frac{\alpha\omega^3}{2\pi c^2} \sum_{\lambda} \int \left| \int d^3r \psi_{n_f, l_f, m_f}^* \mathbf{r} \cdot \boldsymbol{\epsilon}(\mathbf{k}, \lambda) \psi_{n_i, l_i, m_i} \right|^2 d\Omega \quad (19.51)$$

We can write $\mathbf{r} \cdot \boldsymbol{\epsilon}$ as

$$\mathbf{r} \cdot \boldsymbol{\epsilon} = \left(\frac{\epsilon_x + i\epsilon_y}{\sqrt{2}} \right) \left(\frac{x - iy}{\sqrt{2}} \right) + \left(\frac{\epsilon_x - i\epsilon_y}{\sqrt{2}} \right) \left(\frac{x + iy}{\sqrt{2}} \right) + \epsilon_z z \quad (19.52)$$

$$= r \sqrt{\frac{4\pi}{3}} \left(\frac{\epsilon_x + i\epsilon_y}{\sqrt{2}} Y_{1,-1} - \frac{\epsilon_x - i\epsilon_y}{\sqrt{2}} Y_{1,1} + \epsilon_z Y_{1,0} \right) \quad (19.53)$$

If we are dealing with the 2s to 1s transition, equation 19.51 contain an integral of the form

$$\int d^3r R_{1,0}^* Y_{0,0}^* r \sqrt{\frac{4\pi}{3}} \left(\frac{\epsilon_x + i\epsilon_y}{\sqrt{2}} Y_{1,-1} - \frac{\epsilon_x - i\epsilon_y}{\sqrt{2}} Y_{1,1} + \epsilon_z Y_{1,0} \right) R_{2,0} Y_{0,0} \quad (19.54)$$

Inside of this are three summed integrals of the form

$$\int d\Omega Y_{1,m}^* Y_{0,0} \quad (19.55)$$

This integrates to 0 because of the orthogonality of the Y 's. Thus, $R \rightarrow 0$ and $\tau \rightarrow \infty$. Or, the 2s \rightarrow 1s transition is forbidden, at least to first order. Next, we can calculate the 2p \rightarrow 1s transition. It contains the integral

$$\int d^3r R_{1,0}^* Y_{0,0}^* r \sqrt{\frac{4\pi}{3}} \left(\frac{\epsilon_x + i\epsilon_y}{\sqrt{2}} Y_{1,-1} - \frac{\epsilon_x - i\epsilon_y}{\sqrt{2}} Y_{1,1} + \epsilon_z Y_{1,0} \right) R_{2,1} Y_{1,m_i} \quad (19.56)$$

We can again exploit the orthogonality of the Y 's

$$\int d\Omega Y_{1,m}^* Y_{1,m_i} = \delta_{m,m_i} \quad (19.57)$$

to rewrite this integral as

$$\sqrt{\frac{1}{3}} \left(\frac{\epsilon_x + i\epsilon_y}{\sqrt{2}} \delta_{m_i,1} - \frac{\epsilon_x - i\epsilon_y}{\sqrt{2}} \delta_{m_i,-1} + \epsilon_z \delta_{m_i,0} \right) \int_0^\infty dr r^3 R_{1,0}^* R_{2,1} \quad (19.58)$$

The R dependence integrates to

$$\int_0^\infty dr r^3 R_{1,0}^* R_{2,1} = \sqrt{\frac{3}{2}} \frac{2^8}{3^5} a_0 \quad (19.59)$$

so the absolute value squared of the large integral is

$$\frac{1}{3} \left(\frac{\epsilon_x^2 + \epsilon_y^2}{2} \delta_{m_i,1} \frac{\epsilon_x^2 + \epsilon_y^2}{2} \delta_{m_i,-1} \epsilon_z^2 \delta_{m_i,0} \right) \frac{2^{15}}{3^9} a_0^2. \quad (19.60)$$

Now, we can assume that a photon is equally likely to be emitted with any polarization, from which we deduce that

$$\langle \epsilon_x^2 \rangle = \langle \epsilon_y^2 \rangle = \langle \epsilon_z^2 \rangle. \quad (19.61)$$

But we also know that ϵ is a unit vector. So,

$$\langle \epsilon_x^2 \rangle + \langle \epsilon_y^2 \rangle + \langle \epsilon_z^2 \rangle = 1 \quad (19.62)$$

Thus

$$\langle \epsilon_x^2 \rangle = \langle \epsilon_y^2 \rangle = \langle \epsilon_z^2 \rangle = \frac{1}{3} \quad (19.63)$$

Since all we are interested in is the average life time of the state, we can take the average of our integral and it become, for all the m_i :

$$\left| \int d^3r \psi_{n_f, l_f, m_f}^* \mathbf{r} \cdot \epsilon(\mathbf{k}, \lambda) \psi_{n_i, l_i, m_i} \right|^2 = \frac{1}{3} \left(\frac{1}{3} \right) \frac{2^{15}}{3^9} a_0^2 = \frac{2^{15}}{3^{11}} a_0^2. \quad (19.64)$$

Thus, we have

$$R_{2p \rightarrow 1s} = \frac{\alpha \omega^3}{2\pi c^2} \sum_\lambda \int \left(\frac{2^{15}}{3^{11}} a_0^2 \right) d\Omega \quad (19.65)$$

There are 2 polarization states. We are interested in decays into all of space so the infinitesimal solid angle integrates to 4π . Thus, our integral becomes

$$R_{2p \rightarrow 1s} = \frac{\alpha \omega^3}{c^2} \frac{2^{17}}{3^{11}} a_0^2. \quad (19.66)$$

Note that $\hbar\omega = E_{2p} - E_{1s} = \frac{1}{2} m_e c^2 \alpha^2 (1 - 1/2^2)$ so

$$R_{2p \rightarrow 1s} = \left(\frac{2}{3} \right)^8 \alpha^5 \frac{m_e c^2}{\hbar} \quad (19.67)$$

and

$$\tau_{2p \rightarrow 1s} = \frac{1}{R_{2p \rightarrow 1s}}. \quad (19.68)$$

19.3 Problem

Estimate (or really: put a bound on) the ground state energy of Hydrogen using the variational principle, using a trial wave function of the form: $\psi(\mathbf{r}) = Ae^{-br^2}$.

I can't crack this problem but I will write down what I did. The whole trick to this problem is to calculate $\langle E \rangle$. We know is at least as large as the ground state. Therefore, it is an upper bound on ground the state energy. Once we calculate $\langle E \rangle$, we can find the b that minimizes this function to get the lowest upper bound. First, we need ψ to be properly normalized:

$$1 = \int \psi^2 d^3r = \int_0^\infty A^2 e^{-2br^2} 4\pi r^2 dr \quad (19.69)$$

$$1 = 4\pi A^2 \frac{1}{4} \sqrt{\frac{\pi}{(2b)^3}} \quad (19.70)$$

$$A^2 = \left(\frac{2b}{\pi}\right)^{3/2} \quad (19.71)$$

Now, I will calculate $\langle E \rangle$. We know that the Hamiltonian is

$$\hat{H} \rightarrow -\frac{\hbar^2 \nabla^2}{2\mu} - \frac{e^2}{r} \quad (19.72)$$

$$\langle E \rangle = \langle \psi | \hat{H} | \psi \rangle \quad (19.73)$$

$$= \int_0^\infty Ae^{-br^2} \left(-\frac{\hbar^2}{2\mu} \frac{1}{r^2} \frac{\partial}{\partial r} \left(r^2 \frac{\partial}{\partial r} \right) - \frac{e^2}{r} \right) Ae^{-br^2} 4\pi r^2 dr \quad (19.74)$$

$$= 4\pi A^2 \int_0^\infty dr e^{-2br^2} \left(-\frac{\hbar^2}{2\mu} (6 - 14br^2 + 4b^2r^4) - re^2 \right) \quad (19.75)$$

Now, using the Gaussian integral equations from the Townsend book¹, we get

$$\langle E \rangle = 4\pi \left(\frac{2b}{\pi}\right)^{3/2} \left(-\frac{\hbar^2}{2\mu} \left(6\frac{1}{2} \sqrt{\frac{\pi}{2b}} - 14b\frac{1}{4} \sqrt{\frac{\pi}{(2b)^3}} + 4b^2\frac{3}{8} \sqrt{\frac{\pi}{(2b)^5}} \right) - \frac{e^2}{4b} \right) \quad (19.76)$$

$$= 8b\sqrt{\frac{2b}{\pi}} \left(-\frac{\hbar^2}{2\mu} \frac{13}{8} \sqrt{\frac{\pi}{2b}} - \frac{e^2}{4b} \right) \quad (19.77)$$

$$= -\frac{13}{2} \frac{\hbar^2}{\mu} b - 2e^2 \sqrt{\frac{2b}{\pi}}. \quad (19.78)$$

Unfortunately, this equation is wrong since it grows indefinitely negative for sufficiently large b . This is unphysical. If I could have gotten the true equation, I would have set

$$\frac{\partial}{\partial b} \langle E \rangle = 0. \quad (19.79)$$

I would have solve for this for b and then plug b 's value into $\langle E \rangle$ to find my upper bound for the ground state energy. Do you have any idea what I am doing wrong in this problem?

¹Travis told me how to do the last Gaussian integral over the phone – thanks for the tip!

19.4 Problem

You've learned about the Born approximation in the context of 3D scattering problems. It is possible to define a Born approximation also for 1D problems, where instead of wanting to calculate the differential cross section, the relevant predictions are just the reflection and transmission coefficients R and T . Explain why the Born approximation expression for R is:

$$R \approx \left(\frac{m}{\hbar^2 k} \right)^2 \left| \int e^{2ikx} V(x) dx \right|^2. \quad (19.80)$$

Then use it to calculate the reflection probability when the scattering center is a rectangular barrier of "height" V_0 and width a . In what limit(s) do you expect the answer to be accurate?²

Ah, problem 13.4 from Townsend's book. A solution to the one-dimensional Schrödinger equation is

$$\psi(x) = Ae^{ikx} + \int dx' G(x, x') \frac{2m}{\hbar^2} V(x') \psi(x') \quad (19.81)$$

where

$$\frac{\partial^2}{\partial x^2} G(x, x') + k^2 G(x, x') = \delta(x - x'). \quad (19.82)$$

To prove this, all I have show that it is a solution to

$$\hat{H}\psi(x) = E\psi(x) \quad (19.83)$$

where $E = \hbar^2 k^2 / 2m$.

$$\hat{H}\psi(x) = \left(-\frac{\hbar^2}{2m} \frac{\partial^2}{\partial x^2} + V(x) \right) \psi(x) \quad (19.84)$$

$$= -\frac{\hbar^2}{2m} \frac{\partial^2}{\partial x^2} \left(Ae^{ikx} + \int dx' G(x, x') \frac{2m}{\hbar^2} V(x') \psi(x') \right) + V(x) \psi(x) \quad (19.85)$$

$$= \frac{\hbar^2 k^2}{2m} Ae^{ikx} - \frac{\hbar^2}{2m} \int dx' (\delta(x - x') - k^2 G(x, x')) \frac{2m}{\hbar^2} V(x') \psi(x') + V(x) \psi(x) \quad (19.86)$$

$$= \frac{\hbar^2 k^2}{2m} Ae^{ikx} - \cancel{V(x)\psi(x)} + \frac{\hbar^2 k^2}{2m} \int dx' G(x, x') \frac{2m}{\hbar^2} V(x') \psi(x') + \cancel{V(x)\psi(x)} \quad (19.87)$$

$$= \left(\frac{\hbar^2 k^2}{2m} \right) \left(Ae^{ikx} + \int dx' G(x, x') \frac{2m}{\hbar^2} V(x') \psi(x') \right) \quad (19.88)$$

$$= E\psi(x). \quad (19.89)$$

²By the way, there is an error in your formula. The fraction in front is $m/\hbar^2 k$ but you wrote $m/\hbar k$. Oops

We can integrate $G(x, x')$ from just below to just above $x = x'$. Note that the $k^2 G(x, x')$ term integrates to 0 in this infinitesimal range and the delta function integrates to 1:

$$\int_{x=x'_-}^{x=x'_+} \frac{\partial^2}{\partial x^2} G(x, x') + \int_{x=x'_-}^{x=x'_+} k^2 G(x, x') = \int_{x=x'_-}^{x=x'_+} \delta(x - x') \quad (19.90)$$

$$\left(\frac{\partial G}{\partial x} \right)_{x=x'_+} - \left(\frac{\partial G}{\partial x} \right)_{x=x'_-} = 1. \quad (19.91)$$

We can now show that a solution for $G(x, x')$ is

$$G = \begin{cases} \frac{1}{2ik} e^{ik(x-x')} & x > x' \\ \frac{1}{2ik} e^{-ik(x-x')} & x < x' \end{cases} \quad (19.92)$$

To do so, we show that this is a solution for $x < x'$, $x = x'$, and $x > x'$. First, observe that for $x < x'$ we have

$$\frac{\partial^2}{\partial x^2} G(x, x') + k^2 G(x, x') = \frac{-k^2}{2ik} e^{ik(x-x')} + \frac{k^2}{2ik} e^{ik(x-x')} = 0 \quad (19.93)$$

as expected. For $x > x'$, we have

$$\frac{\partial^2}{\partial x^2} G(x, x') + k^2 G(x, x') = \frac{-k^2}{2ik} e^{-ik(x-x')} + \frac{k^2}{2ik} e^{-ik(x-x')} = 0 \quad (19.94)$$

as expected. Finally, we calculate

$$\left(\frac{\partial G}{\partial x} \right)_{x=x'_+} - \left(\frac{\partial G}{\partial x} \right)_{x=x'_-} = \frac{ik}{2ik} e^{ik(x-x')} \Big|_{x=x'} - \frac{-ik}{2ik} e^{-ik(x-x')} \Big|_{x=x'} = 1 \quad (19.95)$$

as is needed for our function to work at $x = x'$. So this is a valid solution for G .

We can now apply the Born approximation:

$$\psi(x) \approx A e^{ikx}. \quad (19.96)$$

We use this approximate solution on the right side of equation 19.81. We will then take the $x \rightarrow -\infty$ limit of the equation to find the reflection term. In this limit, we have: $G(x, x') = \frac{1}{2ik} \exp(-ik(x - x'))$:

$$\psi(x) \xrightarrow{x \rightarrow -\infty} A e^{ikx} + \int dx' \frac{1}{2ik} e^{-ik(x-x')} \frac{2m}{\hbar^2} V(x') A e^{ikx} \quad (19.97)$$

$$\xrightarrow{x \rightarrow -\infty} A e^{ikx} + A e^{-ikx} \int_{-\infty}^{\infty} dx' \frac{e^{2ikx'}}{2ik} \frac{2m}{\hbar^2} V(x') \quad (19.98)$$

Thus

$$R \approx \left(\frac{m}{\hbar^2 k} \right)^2 \left| \int e^{2ikx} V(x) dx \right|^2. \quad (19.99)$$

We will now work with the potential barrier

$$V(x) = \begin{cases} V_0 & 0 < x < a \\ 0 & \text{elsewhere} \end{cases}. \quad (19.100)$$

Solving for R gets

$$R = \left(\frac{m}{\hbar^2 k} \right)^2 \left| \int_0^a e^{2ikx} V_0 dx \right|^2 \quad (19.101)$$

$$= \left(\frac{mV_0}{\hbar^2 k} \right)^2 \left| \frac{1}{2ik} e^{2ikx} \right|_0^a^2 \quad (19.102)$$

$$= \left(\frac{mV_0}{\hbar^2 k^2} \right)^2 \left| \frac{e^{2ika} - 1}{2i} \right|^2 \quad (19.103)$$

$$= \left(\frac{mV_0}{\hbar^2 k^2} \right)^2 \left| \frac{e^{ika} - e^{-ika}}{2i} \right|^2 \quad (19.104)$$

$$= \left(\frac{mV_0}{\hbar^2 k^2} \right)^2 \sin^2(ka) \quad (19.105)$$

$$= \left(\frac{V_0}{2E} \right)^2 \sin^2(ka) \quad (19.106)$$

This formula is valid in the large energy limit $E \gg V_0$. In fact, the Townsend book gives the exact expression for $T = 1 - R$ as

$$T = \frac{1}{1 + (V_0^2/4E(E - V_0)) \sin^2(\sqrt{(2m/\hbar^2)(E - V_0)}a)} \quad (19.107)$$

Or,

$$R = \frac{(V_0^2/4E(E - V_0)) \sin^2(\sqrt{(2m/\hbar^2)(E - V_0)}a)}{1 + (V_0^2/4E(E - V_0)) \sin^2(\sqrt{(2m/\hbar^2)(E - V_0)}a)} \quad (19.108)$$

In the large energy limit, this equation reduces to

$$R \xrightarrow{E \gg V_0} \frac{(V_0/2E)^2 \sin^2(\sqrt{2mE/\hbar^2}a)}{1 + O(V_0/E)^2} = \left(\frac{V_0}{2E} \right)^2 \sin^2(ka) \quad (19.109)$$

just as we predicted.

19.5 Problem

Let

$$\begin{pmatrix} E_0 & 0 & A \\ 0 & E_1 & 0 \\ A & 0 & E_0 \end{pmatrix} \quad (19.110)$$

be the matrix representation of the Hamiltonian for a three-state system using basis states $|1\rangle$, $|2\rangle$, and $|3\rangle$. If the state of the system at $t = 0$ is $|\psi(0)\rangle = |2\rangle$, what is $|\psi(t)\rangle$? How about if $|\psi(0)\rangle = |3\rangle$?

The first part is easy. Note that $|2\rangle$ is an energy eigenstate of the Hamiltonian with eigenvalue (energy) E_1 :

$$\begin{pmatrix} E_0 & 0 & A \\ 0 & E_1 & 0 \\ A & 0 & E_0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = E_1 \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \quad (19.111)$$

Since it is an eigenstate, we know that $|\psi(t)\rangle = |2\rangle$. $|3\rangle$ is not an eigenstate so we will have to write it as a linear combination of energy eigenstates. We begin by finding the energy eigenstates of the Hamiltonian:

$$\begin{vmatrix} E_0 - \lambda & 0 & A \\ 0 & E_1 - \lambda & 0 \\ A & 0 & E_0 - \lambda \end{vmatrix} = 0. \quad (19.112)$$

Or,

$$(E_0 - \lambda)^2(E_1 - \lambda) - A^2(E_1 - \lambda) = 0 \quad (19.113)$$

So $\lambda = E_1$, and $\lambda = E_0 \pm A$. The $\lambda = E_0 + A$ solution leads to:

$$\begin{pmatrix} E_0 & 0 & A \\ 0 & E_1 & 0 \\ A & 0 & E_0 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \end{pmatrix} = (E_0 + A) \begin{pmatrix} a \\ b \\ c \end{pmatrix} \quad (19.114)$$

or

$$\cancel{E_0}a + Ac = (\cancel{E_0} + A)a \quad (19.115)$$

$$a = c \quad (19.116)$$

The middle equation says that $b = 0$. An eigenstate is

$$|E_0 + A\rangle = \frac{1}{\sqrt{2}}(|1\rangle + |3\rangle) \quad (19.117)$$

The $\lambda = E_0 - A$ solution leads to

$$\begin{pmatrix} E_0 & 0 & A \\ 0 & E_1 & 0 \\ A & 0 & E_0 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \end{pmatrix} = (E_0 - A) \begin{pmatrix} a \\ b \\ c \end{pmatrix} \quad (19.118)$$

or

$$\cancel{E_0}a + Ac = (\cancel{E_0} - A)a \quad (19.119)$$

$$a = -c. \quad (19.120)$$

Another eigenstate is

$$|E_0 - A\rangle = \frac{1}{\sqrt{2}}(|1\rangle - |3\rangle). \quad (19.121)$$

The final eigenstate (which we already saw) is

$$|E_1\rangle = |2\rangle. \quad (19.122)$$

We can write $|2\rangle$ as a linear combination of the eigenstates:

$$|2\rangle = \frac{1}{\sqrt{2}}(|E_0 + A\rangle - |E_0 - A\rangle). \quad (19.123)$$

We can easily do the time evolution to the eigenstates

$$|\psi(t)\rangle = e^{i\hat{H}t/\hbar} |\psi(0)\rangle \quad (19.124)$$

$$= e^{i\hat{H}t/\hbar} |2\rangle \quad (19.125)$$

$$= e^{i\hat{H}t/\hbar} \frac{1}{\sqrt{2}}(|E_0 + A\rangle - |E_0 - A\rangle) \quad (19.126)$$

$$= \frac{1}{\sqrt{2}}(e^{i(E_0+A)t/\hbar} |E_0 + A\rangle - e^{i(E_0-A)t/\hbar} |E_0 - A\rangle) \quad (19.127)$$

$$= \frac{1}{2}e^{iE_0t/\hbar} ((e^{iAt/\hbar} - e^{-iAt/\hbar}) |1\rangle + (e^{iAt/\hbar} + e^{-iAt/\hbar}) |3\rangle) \quad (19.128)$$

$$= e^{iE_0t/\hbar} (i \sin(At/\hbar) |1\rangle + \cos(At/\hbar) |3\rangle). \quad (19.129)$$

We find that

$$|\langle 1|\psi(t)\rangle|^2 = \sin^2(At/\hbar) \quad (19.130)$$

$$|\langle 3|\psi(t)\rangle|^2 = \cos^2(At/\hbar) \quad (19.131)$$

19.6 Problem

What is Bell's Theorem and what does it prove? (No need to recapitulate the mathematical derivation, which is a standard thing in several texts. Just summarize the structure of the argument, and then explain its implications.)

The general setup of Bell's inequality is a physical process where two spin 1/2 particles are at the same time. Because of conservation laws, we know that the total spin in any direction must be 0. If one particle is measured to be spin up in some direction, the other particle must be spin down. For momentum to be conserved, the two particles must leave in opposite directions. We can set up two Stern-Gerlach machines to measure the spin components of the two particles along any axis that we wish.

Bell's theorem requires assuming that there are hidden variables (the spin of the particles) and that there is locality. When you assume these two features, you can assign to the particles

a definite value of spin for three different spacial components. We might not know what the value is before measuring it, but it is there. The particle must already know going into the SG machine what its spin is. Because we set the SG machines far apart, we know from locality that the measurement of one particle's spin cannot affect the measurement of the other particle's spin. If you do the book keeping, you can show that these two assumptions leads to predictions about the probability of certain spin measurements. We can use or SG machine to measure spin along three separate axis. We will call these three axis \mathbf{a} , \mathbf{b} , and \mathbf{c} . We can set up the first SG machine to measure first particle's spin along one of these axis and the second SG machine to measure the second particle's spin along the other axis. Bell's inequality says that

$$P(+\mathbf{a}; +\mathbf{b}) \leq P(+\mathbf{a}; +\mathbf{c})P(+\mathbf{c}; +\mathbf{b}) \quad (19.132)$$

where $P(+\mathbf{a}; +\mathbf{b})$ is the probability of the first SG machine measuring spin $+\mathbf{a}$ and the second SG machine measuring spin $+\mathbf{b}$, etc. The important point is that standard quantum mechanics predicts probabilities in certain situations that violate this inequality. And this inequality has been put up to experimental tests. The Experiments are in agreement with quantum mechanics but in violation of Bell's inequality.

From these experimental results, we must conclude that some of the assumptions that Bell uses in deriving his inequality are wrong. Since Bell assumes that there are hidden variables and that the world works in a local way, we must conclude that it is impossible for the world to both have hidden variables and be local.

Chapter 20

Comprehensive Exam, part 4 - Covers: Statistical Mechanics, Thermodynamics, Astrophysics

20.1 Problem

“Here is a very simplified model of the unwinding of two-stranded DNA molecules: a zipper has N links; each link has a state in which it is closed with energy 0 and a state in which it is open with energy ϵ . We require that the zipper can only unzip from the left end, and that the link number s can only open if all links to the left ($1, 2, \dots, s-1$) are already open. Show that the partition function is given by

$$X = \frac{1 - \exp[-(N+1)\epsilon/kT]}{1 - \exp(-\epsilon/kT)} \quad (20.1)$$

and then find the average number of open links in the limit $\epsilon \gg kT$.” – Travis Norsen

The partition function is

$$Z \equiv \sum_r e^{-E_r/kT}. \quad (20.2)$$

The sum is over all possible states. For this example, we can make a list of all possible states and their particular energy. There is only one state where all the links are closed and it has energy 0. There is only one state where one link is open and it has energy ϵ . There is only one state where two links are open and it has energy 2ϵ . The argument continues until we get to the final state where all links are open and the energy is $N\epsilon/kT$. Using this The

partition function is

$$Z = e^0 + e^{-\epsilon/kT} + e^{-2\epsilon/kT} + \dots + e^{-(N-1)\epsilon/kT} + e^{-N\epsilon/kT} \quad (20.3)$$

$$Z = \sum_{n=0}^N (e^{-\epsilon/kT})^n. \quad (20.4)$$

It is a mathematical fact that

$$S = \sum_{n=0}^N R^n = \frac{1 - R^{N+1}}{1 - R}. \quad (20.5)$$

From this:

$$Z = \frac{1 - e^{-(N+1)\epsilon/kT}}{1 - e^{-\epsilon/kT}}. \quad (20.6)$$

The probability for each state is

$$P_r = \frac{e^{-E_r/kT}}{\sum e^{-E_r/kT}}. \quad (20.7)$$

The average number of open links is equal to the weighted average of the number of open links:

$$\bar{N} = \sum_{n=0}^N \frac{ne^{-n\epsilon/kT}}{\sum e^{-E_r/kT}}. \quad (20.8)$$

In the large ϵ limit, all the terms are 0. The average number of open links is 0. This makes sense. When the system is very cold, the protein will stay mostly intact.

20.2 Problem

“A cold white dwarf is held up against gravitational collapse by the pressure of degenerate electrons. What is the total energy of a gas of N non-interacting, non-relativistic degenerate electrons confined to a sphere of radius R ? Assuming the white dwarf contains equal numbers of neutrons and protons, rewrite this energy in terms of the mass M of the star. The other main contribution to the energy is the gravitational binding energy (which is of course negative). Write down an expression for the total energy of the star as a function of R (and other relevant parameters). Show that $E(R)$ has a minimum for some particular value of R , and solve for this to find the mass-radius relation for a white dwarf. (Check your answer by confirming that, unlike chocolate cakes, white dwarfs shrink when you add mass to them.)” – Travis Norsen

Stars are big so boundary effects are negligible in comparison to the effects due to the matter well inside the star. Therefore, the particular boundary conditions we pick won't matter. We will therefore solve the time independent Schrödinger equation for the electrons in the

star as though our star is a cube of length L and volume $V = L^3$. We will use periodic boundary conditions:

$$\psi(x + L, y, z) = \psi(x, y + L, z) = \psi(x, y, z + L) = \psi(x, y, z). \quad (20.9)$$

We can approximate the electrons as a free gas. The reason why this is a reasonable assumption is because the positive charge from the protons will be fairly evenly distributed so their net effect will be weak. Approximately, the only net effect due to the protons will be to confine the electrons within the star. The solution to the Schrödinger equation is therefore

$$\psi \propto e^{k_x x + k_y y + k_z z}. \quad (20.10)$$

The energy of the electron is

$$\epsilon = \frac{\hbar^2 k^2}{2m_e} \quad (20.11)$$

where m_e is the mass of an electron. To satisfy the boundary conditions, we have

$$k_x = \frac{2\pi}{L} n_x \quad k_y = \frac{2\pi}{L} n_y \quad k_z = \frac{2\pi}{L} n_z \quad (20.12)$$

Electrons obey the Fermi exclusion principle so only one electron can occupy each quantum state. The number of possible integers n_x for which k_x lies in the range between k_x and $k_x + dk_x$ is

$$\Delta n_x = \frac{L}{2\pi} dk_x. \quad (20.13)$$

The total number of states with wave vector between k_x and $k_x + \Delta k_x$, k_y and $k_y + \Delta k_y$, and k_z and $k_z + \Delta k_z$ is the product of the number of possible integers in the three ranges. We have to add an extra factor of 2 since each of these state can be filled up with two electrons each of different spin:

$$\rho_k d^3 k = 2 \left(\frac{L}{2\pi} dk_x \right) \left(\frac{L}{2\pi} dk_y \right) \left(\frac{L}{2\pi} dk_z \right) = 2 \frac{V}{(2\pi)^3} dk_x dk_y dk_z = \frac{V}{\pi^2} k^2 dk. \quad (20.14)$$

The total number of states in this same range with energy between ϵ and $\epsilon + d\epsilon$ is

$$|\rho_\epsilon d\epsilon| = |\rho_k dk| = \rho_k \left| \frac{dk}{d\epsilon} \right| d\epsilon = \rho_k \left| \frac{d\epsilon}{dk} \right|^{-1} d\epsilon = \frac{V}{\pi^2} \frac{(2m_e)^{3/2}}{\hbar^3} \epsilon^{1/2} d\epsilon. \quad (20.15)$$

We calculate the electron with highest energy as

$$N = \int_0^{\epsilon_f} \rho_\epsilon d\epsilon = \frac{V}{\pi^2} \frac{(2m_e)^{3/2}}{\hbar^3} \frac{2}{3} \epsilon_f^{3/2}. \quad (20.16)$$

It follows that

$$\epsilon_f = \left(\frac{3}{2} \pi^2 \frac{\hbar^3}{(2m)^{3/2}} \frac{N}{V} \right)^{2/3}. \quad (20.17)$$

We can calculate the total energy due to electron degeneracy as

$$E_d = \int_0^{\epsilon_f} \epsilon \rho_\epsilon d\epsilon \quad (20.18)$$

We get

$$E_d = \int_0^{\epsilon_f} \frac{V}{\pi^2} \frac{(2m_e)^{3/2}}{\hbar^3} \epsilon^{3/2} d\epsilon = \frac{1}{5} \frac{3^{5/3}}{2^{5/3}} \frac{\pi^{4/3} \hbar^2 N^{5/3}}{m_e V^{2/3}}. \quad (20.19)$$

For every electron, there is a proton and a neutron (of roughly the same mass m_p). Therefore, we have $M = 2m_p N$. Also, $V = \frac{4}{3}\pi R^3$.

$$E_d = \frac{1}{5} \frac{3^{7/3}}{2^{14/3} m_p^{5/3} m_e} \frac{M^{5/3}}{R^2}. \quad (20.20)$$

We can write E_d as

$$E_d \equiv \frac{KM^{5/3}}{R^2} \quad (20.21)$$

where K is a constant of proportionality.

Next, we consider the gravitational potential energy. We can imagine building up the star from thin shells one at a time. When we add the shell of radius r and thickness dr , the gravitational potential energy due to this shell is

$$dU(r) = dM \times V(r) = -\frac{GM(r)dM}{r}. \quad (20.22)$$

M is the mass of the already assembled star of radius r and dM is the mass of the thin shell. The mass of the interior sphere is equal to the volume times the density so $M = \frac{4}{3}\pi r^3 \frac{M}{V}$. The mass of the shell is $dM = 4\pi r^2 dr \frac{M}{V}$. The energy associated with the shell is

$$dU(r) = -\frac{G\frac{4}{3}\pi r^3 \frac{M}{V} 4\pi r^2 dr \frac{M}{V}}{r} = -\frac{16\pi^2 GM^2}{3V^2} r^4 dr. \quad (20.23)$$

The total potential energy is the integral over all the shells

$$U = \int_0^R dU = -\frac{16\pi^2 GM^2}{3V^2} \frac{1}{5} R^5 = -\frac{3}{5} \frac{GM^2}{R}. \quad (20.24)$$

The total energy is thus

$$E = E_d + U = \frac{KM^{5/3}}{R^2} - \frac{3}{5} \frac{GM^2}{R}. \quad (20.25)$$

We set $\partial E / \partial R = 0$ and solve for R to minimize the energy. We find that

$$R = \frac{10}{3} \frac{K}{GM^{1/3}}. \quad (20.26)$$

Plugging in for K , we find that

$$R = \frac{2}{3} \frac{3^{7/3}}{2^{14/3} m_p^{5/3} m_e} \frac{1}{GM^{1/3}}. \quad (20.27)$$

20.3 Problem

“Suppose a star were made of an ideal gas composed of molecules of mass m at a uniform temperature T . By considering hydro-static equilibrium, develop a differential equation that should be satisfied by $\rho(r)$, the mass density as a function of radius. (You shouldn’t bother solving the equation – the work here is just setting up a well-defined DE with only the one dependent variable, $\rho(r)$.)” – Travis Norsen

We will consider a small rectangle chunk of gas a radius r from the center of the star. The rectangle has a width dA and a height dr . The mass density in the chunk is $\rho(r)$. Gauss’ Law says the gravitational force on the chunk is

$$F = \frac{GM\rho dA dr}{r^2} \quad (20.28)$$

where M is the mass inside of the chunk. The mass is

$$M = \int_0^r \rho(r) 4\pi r^2 dr. \quad (20.29)$$

The pressure difference between the top and bottom of the chunk due to the surrounding gas must exactly cancel out the gravitational force in order for there to be equilibrium:

$$[P(r + dr) - P(r)]dA = \frac{GM\rho(r)dA dr}{r^2}. \quad (20.30)$$

The idea gas law says that

$$P(r) = n(r)kT \quad (20.31)$$

where $n(r)$ is the number density of the gas. we know there is only a radial dependence to the pressure because of the radial symmetry of the star. The mass density is related to the number density by $\rho(r) = mn(r)$. Our equilibrium equation becomes

$$[\rho(r + dr) - \rho(r)] \frac{kT}{m} dA = \frac{GM\rho(r)dA dr}{r^2}. \quad (20.32)$$

We note that

$$\rho(r + dr) - \rho(r) = \frac{d\rho}{dr} dr \quad (20.33)$$

so our equation becomes

$$\frac{d\rho(r)}{dr} = \frac{Gm}{kTr^2} \left(\int_0^r \rho(r) 4\pi r^2 dr \right) \rho(r). \quad (20.34)$$

Rearranging and the differentiating both sides of the equation with respect to r gets us

$$\frac{kT}{Gm} r^2 \rho \frac{d\rho}{dr} = \int_0^r \rho 4\pi r^2 dr \quad (20.35)$$

$$\frac{kT}{Gm} \left(2r\rho \frac{d\rho}{dr} + r^2 \left(\frac{d\rho}{dr} \right)^2 + r^2 \rho \frac{d^2\rho}{dr^2} \right) = \rho 4\pi r^2. \quad (20.36)$$

20.4 Problem

“The latent heat (or ‘heat of fusion’) for the ice-water phase transition is 80 calories/gram. What is the probability that a bucket of pure water (no ice) at zero degrees Celsius spontaneously forms a one gram ice cube?” – Travis Norsen

The heat of fusion is the negative of the energy per unit mass required to convert water to ice. Thus, to create a gram of ice requires adding to the system $Q = -80$ calories. Google says that $Q = 330$ J. We know that the change in entropy for this transition is

$$\Delta S = \int \frac{dQ}{T} = \frac{Q}{T} \quad (20.37)$$

where we have used the fact that the temperature does not change during the freezing. The definition of entropy is $S = k \log(\Omega)$. We know that

$$P \propto \Omega. \quad (20.38)$$

Formally, the probability of this transition happening is equal to the number of micro states for which this is possible divided by the total number of micro states:

$$P = \frac{\Omega_{\text{freeze}}}{\Omega_{\text{total}}} \quad (20.39)$$

For our example we have

$$S_0 + \Delta S = k \log(\Omega_{\text{freeze}}) \quad (20.40)$$

Where S_0 is the entropy of the liquid with no frozen ice. Thus,

$$P = \frac{e^{(S_0 + \Delta S)/k}}{\Omega_{\text{total}}} = \frac{e^{S_0/k} e^{\Delta S/k}}{\Omega_{\text{total}}}. \quad (20.41)$$

Now, we know that an overwhelming number of the possible micro states will involve the whole system in its equilibrium situation where there is just water. In this situation, the entropy is S_0 . In equation form, we have

$$S_0 \approx k \log(\Omega_{\text{total}}) \quad (20.42)$$

So

$$\Omega_{\text{total}} \approx e^{S_0/k}. \quad (20.43)$$

Therefore, to good approximation the probability for our initial state to spontaneously form an ice cube is approximately

$$P \approx e^{\Delta S/k} \approx e^{Q/kT} \quad (20.44)$$

With $Q = 330$ J, $k = 1.38 \times 10^{-23}$ J/K, and $T = 273.15$ K, we have

$$P \approx e^{-10^{23}}. \quad (20.45)$$

This is very small!

20.5 Problem

“In The Physical Universe, Shu discusses the ‘Missing-Mass Problem’ on pages 259-60. Summarize what you know about this topic that goes beyond what’s in Shu’s (rather outdated) book.” – Travis Norsen

There have been two major developments that I am familiar with. First was the discovery of gravitational lensing. Studying dark matter with gravitational lensing is kind of like studying the shape of a piece of glass by looking through it and seeing how the background is distorted. General Relativity says that that the trajectory of light is bent by the presence of gravity. Actually, I think we are supposed to say that light still goes in straight lines and it is instead the structure of space time that gets bent by the presence of matter. Anyway, I will wave my hands here because I don’t know much General Relativity. By it is fair to think of light as being deflected by gravity.

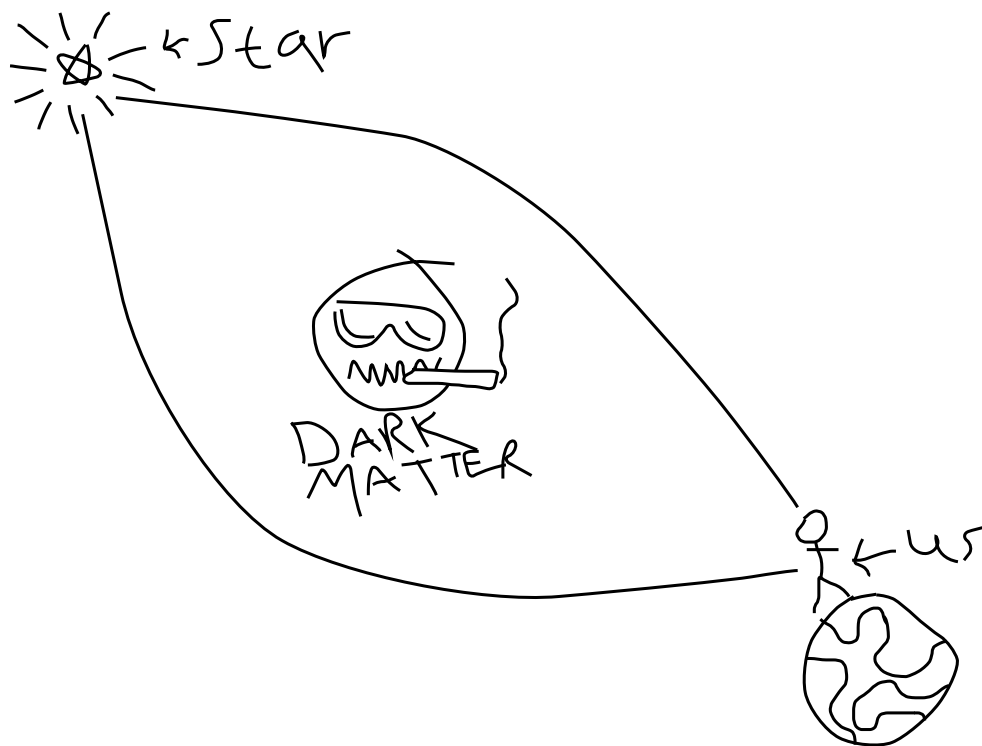


Figure 20.1: The shady looking man in the figure represents dark matter and is supposed to remind the reader that the jury is still out on what exactly dark matter is made up out of.

The classic example of this is shown in figure 20.1. It shows a diagram of the most famous example of gravitational lensing. A star sits behind a bunch of dark matter. The light from the star leaves isotropically but only particular directions of light will be bent enough to come back to the earth. What the person on earth sees when he looks into the sky is a ring of light from the star. These rings are called Einstein rings and have actually been observed in the sky.

Of course, Einstein rings are seen only when the alignment is very close to perfect so they are rare. In real life, we usually only see multiple images of the object. This is enough information to learn quite a bit about the dark matter lens. This technique where multiple lenses are involved is called strong gravitational lensing and has been used to learn much about the structure of heavy clumps of dark matter.

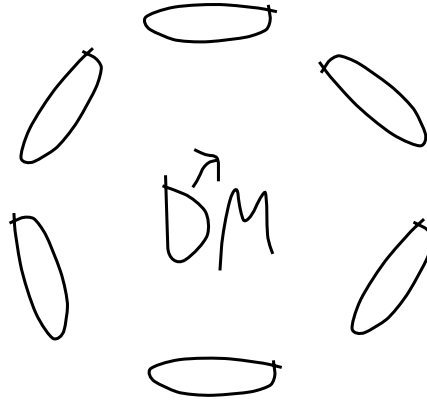


Figure 20.2: The shear correlation of galaxies caused by weak gravitational lensing. Each of these galaxies are supposed to come from a unique source who's shape is distorted only slightly.

After strong lensing came weak lensing. Not always will a gravitational lens be so strong that multiple images are seen. Often, the dark matter will just distort the shape and orientation of a galaxy. The dark matter adds a radial shear correlation to the ellipticity of the galaxies. Where there are big spots of dark matter in the sky, the galaxies will preferentially be distributed around the dark matter as in figure 20.2. By measuring shear correlation, we can learn about the structure of the dark matter.

This is a statistical technique. Were it not for dark matter, one would have no reason to believe that there was any correlation between the shapes of galaxies. After all, the galaxies in a particular spot of the sky come from vastly different depths so how could their galactic evolution have any relation to one another. When we found this correlation, it was very strong evidence for the existence of dark matter. And now it can be used as a method for mapping out the dark matter in the universe.

It is actually also being used as a way to get better statistics about regular galaxies in the universe. The fear with gathering statistics about galaxies by spotting them in the electromagnetic spectrum is that there will be systematic errors associated with what types of galaxies we will see. When we instead gather statistics about galaxies that we find using gravitational lensing, the thought is that there are fewer systematics involved.

The second major development in the missing mass problem came from an observation of what is called the bullet cluster. A large challenge facing the theory of dark matter is the empirical fact that whenever we see dark matter in the sky (either from the velocity distribution of stars in a galaxy or from lensing), we also see regular matter in the same place. This led some to hypothesize that instead of there being mass that we can't see, that

instead our knowledge of how gravity works is wrong on the large astronomical scales. Theorists have proposed alternate gravity theories which try to explain the dark matter effects without the need for invisible matter.

Very convincing evidence came out against the viability of modified gravity when scientists found in the sky two galaxies that had collided with each other. What is exciting about the collision is that it vastly changed the trajectory and the shape of the visible matter. This can be seen in the electromagnetic spectrum. But the dark matter was relatively unaffected by the collision and kept going in its normal trajectory. This can be seen by weak lensing. The dark matter and the regular matter are separated in the sky. The simulations of what this should look like if dark matter exists look perfectly like the observational data. But it is very hard to explain this observation with a modified gravity theory.

Another thing. Apparently physicists have figured out what percentage of the universe is dark matter. I think it is like 30%. I have no idea how this is done.

20.6 Problem

“The heat capacity of non-metallic solids at sufficiently low temperatures is proportional to T^3 . Explain why. Also explain why metals behave differently.” – Travis Norsen

The specific heat caused by lattice vibrations is of the form $c_v^{(L)} = AT^3$. The specific heat due to the electron gas is $c_v^{(e)} = \gamma T$. The specific heat for a non-metallic solid is entirely due to lattice vibrations and is equal to AT^3 . The specific heat for a metal is due to both the electron gas and the lattice vibrations and is therefore equal to $\gamma T + AT^3$.

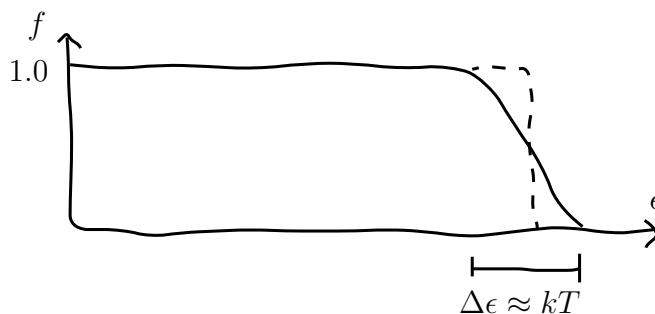


Figure 20.3: The Fermi function for a free electron gas.

We can understand the lattice vibration's T^3 dependence as follows. Rief write down the Hamiltonian for all of the positive charges in a metal. He then describes how we can do a change of variables to make the Hamiltonian have the form of $3N$ independent simple harmonic oscillators. Each of these is a phonon—a matter wave in the metal. Each oscillator has angular frequency ω_r and energy $\epsilon_r = (n_r + \frac{1}{2})\hbar\omega_r$. This is equation 10 · 1 · 11 in Rief.

At temperature T , most of our phonons will have energy less than kT . As an equation, we have $\hbar\omega \approx kT$. The total number of phonons \bar{N} will be proportional to the volume in ω

space containing all the phonons (with energy less than kT). The volume is proportional to ω^3 which is proportional to T^3 . Thus, the average energy is proportional to the number of states times their energy. This is proportional to T^4 . Since the energy is proportional to T^4 , the specific heat (the change in energy versus change in temperature) must be proportional to T^3 .

The only thing that could cause a wrinkle in my argument is a fact brought up in Rief that there is a certain cutoff angular frequency that phonons can not exceed. Since our upper bound angular frequency is $\hbar\omega \approx kT$, we can safely assume that all of our phonons will have angular frequency smaller than the cutoff.

We can understand the electron gas' T dependence as follows. We can approximate the behavior of the electrons in a metal as a free electron gas. In its lowest energy state, all the electrons are in the lowest allowed energy states. As the temperature increases, more and more electrons begin to occupy higher and higher energy states. Reif proves that for an electron gas, the probability of a state of energy ϵ being occupied is equal to the Fermi function f with

$$F(\epsilon) = \frac{1}{e^{\beta(\epsilon-\mu)} + 1}. \quad (20.46)$$

This is equation 9 · 16 · 4. For relatively small T , we sketch this function in figure 20.3. We see that Electrons with energy roughly within kT of the edge energy ϵ_F will move to a higher energy state. There are roughly $g(\epsilon_F)kT$ of these electrons where $G(\epsilon)$ is the density of levels per unit volume with energy ϵ . Therefore, the total energy change will be approximately equal to the number of states times their change in energy:

$$E \approx G(\epsilon_F)kT \times kT \propto T^2. \quad (20.47)$$

We know that the density of levels is a property of the metal in question and is not a function of temperature. The specific heat must be proportional to temperature.

20.7 Problem

“Considering the earth as a thermodynamic system, it’s clear that over geological timescales the total energy is (roughly) constant: on average, the earth radiates heat out into space at the same rate it absorbs heat from the sun. But what about the second law of thermodynamics? Why doesn’t the entropy of the earth increase steadily over geologic timescales? And hence: what fact or facts about the earth or the universe as a whole is/are ultimately responsible for the viability of life on earth?” – Travis Norsen

We know that entropy always goes up. Since the entropy on the earth remains roughly constant, it must be the case that entropy somewhere else goes up. The increase in entropy comes from radiation leaving the earth. The light that the sun gives to us is highly ordered and has a low entropy. The light and heat that the earth radiates is fairly disordered and has a high entropy. Therefore, we conclude that the only way that life on earth can stay nice and orderly is by creating a mess somewhere else (outer space).

Part VI

Appendix

Appendix A

How and Why to Think About Scattering in Terms of Wave-Packets Instead of Plane-Waves

Travis Norsen, Joshua Lande
Marlboro College, Marlboro, VT 05344

S. B. McKagan
JILA and NIST, University of Boulder, CO, 80309

Abstract

We discuss “the plane wave approximation” to quantum scattering and tunneling using simple one-dimensional examples. The central point of the paper is that the calculations of reflection and transmission probabilities in standard textbook presentations involve an approximation which is almost never discussed. We argue that it should be discussed explicitly, and that doing so provides a simple alternative way to derive certain formulas that are used in the standard calculations. We also calculate, for a simple standard example, expressions for the R and T probabilities for an incident Gaussian wave-packet of arbitrary width. These expressions can be written as a power series expansion in the inverse packet width. We calculate the first non-vanishing corrections explicitly.

A.1 Introduction

Scattering is arguably the most important topic in quantum physics. Virtually everything we know about the micro-structure of matter, we know from scattering experiments. And so the theoretical techniques involved in predicting and explaining the results of these experiments play a justifiably central role in quantum physics courses at all levels in the physics curriculum, from Modern Physics for sophomores through Quantum Field Theory for graduate

students.

Given the importance and centrality of this topic, we should be particularly careful about clarifying its physical and conceptual foundations – both for ourselves and for our students. It is the main contention of this paper that these foundations are not typically as clear as they could be. The specific problem we address is the fact that the scattering particle is almost always described as a (suitably modified) plane-wave, rather than a physical, normalizable, finite-width wave-packet. As we will explain in the following section, this standard plane-wave account is fraught with conceptual problems which have been documented to cause confusion and errors among students, and which may also cause confusion among experts.

In the following section, we present a simple alternative way of deriving certain formulas (which play a central role in the calculation of various scattering probabilities and which are usually justified in a complicated and confusing way in the context of the plane-wave account of scattering) from a straightforward analysis of the kinematics of wave-packets. We thus demonstrate that many of the conceptual problems associated with the plane-wave analysis (and some pointless mathematical complications, to boot) can be quite simply avoided – all while preserving the mathematical simplicity and accessibility of the standard plane-wave calculation.

In subsequent sections, we present what we believe (surprisingly) to be some novel calculations of the scattering probabilities when the incident particle is represented by a Gaussian wave-packet. The novelty consists in exact expressions for the reflection and transmission (R and T) probabilities: these can be expanded in powers of the inverse packet width, and the individual terms can be calculated analytically. We thus show explicitly that the usual plane-wave expressions for R and T emerge in the limit of an infinitely-wide packet.

That, of course, is no surprise. But often (in that small minority of texts which even discuss it) the wave-packet analysis is presented as an afterthought – e.g., a more physically and conceptually realistic way of re-deriving the plane-wave expressions for R and T . This conveys the impression that the wave-packet analysis is only a sort of heuristic, with the “really correct” plane-wave results emerging when one takes the packet width to infinity. But this impression is both false and dangerous. The really correct probabilities are the ones based on the actual properties of incident particles, and these will always be properly represented as finite-width wave-packets. It is the plane-wave expressions which are an approximation to the wave-packet probabilities, not vice versa. There is thus harmony between the mathematical and the conceptual: the thing that is properly regarded as fundamental (both conceptually and in terms of providing the rigorously exact predictions for experiments) is wave-packets. Hence our conclusion: it is in terms of wave-packets that we should think about scattering ourselves, and introduce scattering to students.

A.2 The plane-wave account and its problems

Most students first encounter the quantum mechanical treatment of scattering with the simple example of a 1-D particle incident on a potential step:

$$V(x) = V_0 \theta(x) = \begin{cases} 0 & \text{for } x < 0 \\ V_0 & \text{for } x > 0 \end{cases} . \quad (\text{A.1})$$

We will base most of our discussion on this example, though, as will be obvious, most of what we have to say applies to scattering problems in general.

The familiar calculation of R and T probabilities for the potential step proceeds as follows. One finds solutions to the time-independent Schrödinger equation

$$-\frac{\hbar^2}{2m} \psi''(x) + V(x) \psi(x) = E \psi(x) \quad (\text{A.2})$$

valid on the two sides of the origin:

$$\psi_k(x) = \begin{cases} Ae^{ikx} + Be^{-ikx} & \text{for } x < 0 \\ Ce^{i\kappa x} + De^{-i\kappa x} & \text{for } x > 0 \end{cases} \quad (\text{A.3})$$

where

$$\kappa^2 = k^2 - \frac{2mV_0}{\hbar^2} = k^2 - p^2. \quad (\text{A.4})$$

Then, citing as an initial condition that particles be incident from (let us say) the left, one argues on physical grounds that the coefficient D (describing particle flux incident from the right) should vanish, leaving

$$\psi_k(x) = \begin{cases} Ae^{ikx} + Be^{-ikx} & \text{for } x < 0 \\ Ce^{i\kappa x} & \text{for } x > 0 \end{cases} . \quad (\text{A.5})$$

where one interprets the A term as representing incident flux, B as the reflected flux, and C as the transmitted flux.

A number of conceptual problems associated with the plane-wave analysis are already manifest. Several features of the argument (such as writing the most general solution of the Schrödinger equation in terms of complex exponentials rather than sines and cosines, and the elimination of the D term) are based on a certain intuitive physical picture of the scattering process: particles propagate in from the left, reflect or transmit at $x = 0$, and subsequently propagate out to the left or right. The fact that the particles *propagate* suggests the complex exponentials, and the fact that particles can never be propagating to the left in the $x > 0$ region warrants setting $D = 0$.

But rigorously speaking, Equation A.5 and the intuitive physical picture we partially based it on, are in conflict. For example, according to Equation A.5, there is never a time when the particle was definitely incident from the left (and hence no real argument that it shouldn't be in the $x > 0$ region moving to the left). Another way to say this is that

the physically realistic *initial condition* we had in mind (that once upon a time there was a particle approaching the step with some definite width, position, and speed) is inconsistent with the wave function we actually write down: the latter represents a particle which is infinitely spread out through all of space and which as been forever, timelessly reflecting and transmitting from the barrier. The standard argument thus muddles together two distinct steps – setting up the initial conditions and finding a solution. This may seem efficient, since it is difficult to write down a general solution without already having in mind the idea of an initially-incident propagating plane-wave. But what experts perceive as efficient, students find confusing.

Experts are probably also used to thinking of the timeless, steady-state wave function as some kind of limit for an infinitely-wide incident packet. But how exactly this limit works is unclear, even to most experts. We will show in the subsequent section that it is actually quite straightforward to understand – so simple in fact that we advocate introducing it to students from the very beginning and thus avoiding completely the kinds of issues being raised here.

Let us continue now sketching and critiquing the standard plane-wave analysis of this problem.

Equation (A.5) actually solves Equation (A.2) at $x = 0$ only if ψ and ψ' are continuous at there. Imposing these conditions gives the following familiar expressions relating the amplitudes of the incident, reflected, and scattered waves:

$$\frac{B}{A} = \frac{k - \kappa}{k + \kappa} \quad (\text{A.6})$$

and

$$\frac{C}{A} = \frac{2k}{k + \kappa}. \quad (\text{A.7})$$

Note that even writing down equations A.6 and A.7 requires recognizing that the value of A is an arbitrary initial condition which then sets the values of B and C . In working through this derivation with students, we have observed that while students have no trouble verifying that these equations are true, they are often baffled by why we choose to write them down in the first place. Writing these particular equations also anticipates an ultimate goal of deriving the reflection and transmission probabilities R and T , a goal which is often not obvious a priori to students.

Further, even when it is clarified that the goal is to derive R and T , it is not entirely clear how to proceed, unless one is already familiar with the derivation. According to the standard probability interpretation of the wave function, the reflection and transmission probabilities should be given by the area under the reflected and transmitted part of $|\psi|^2$, respectively, divided by the area under the incident part of $|\psi|^2$. Since all these areas are infinite, one can't calculate the reflection and transmission probabilities as one would naively expect. However, it is quite tempting (and quite wrong) to assume that the infinite widths simply cancel and that the reflection and transmission coefficients are given by:

$$R = |B|^2/|A|^2 \quad (\text{A.8})$$

and

$$T = |C|^2/|A|^2 \quad (\text{A.9})$$

We have observed that this is a common mistake for students to make, but most textbooks do not address it. The actual R and T values are proportional not to the ratio of probability densities (associated with the appropriate outgoing and incident part of the wave), but of the probability densities times the group velocities, or equivalently, times the wave numbers:

$$R = \frac{v_g(k)|B|^2}{v_g(k)|A|^2} = \frac{k|B|^2}{k|A|^2} = \frac{|B|^2}{|A|^2} = \left(\frac{k - \kappa}{k + \kappa}\right)^2 \quad (\text{A.10})$$

and

$$T = \frac{v_g(\kappa)|C|^2}{v_g(k)|A|^2} = \frac{\kappa|C|^2}{k|A|^2} = \frac{4k\kappa}{(k + \kappa)^2}. \quad (\text{A.11})$$

where

$$v_g(k) = \frac{d\omega(k)}{dk} = \frac{\hbar k}{m}. \quad (\text{A.12})$$

Here $\omega(k) = E(k)/\hbar = \hbar k^2/2m$. One can verify that $R + T = 1$.

Many textbooks simply write down Equations A.10-A.11 without explanation, or worse, avoid them altogether by skipping the step potential and going straight to tunneling through a square barrier, using Equations A.8-A.9 without mentioning that they happen to be correct only for the special case where the wave numbers are equal on both sides. At least one textbook even writes down Equation A.8 as the obvious expression for R , and then “derives” the correct expression for T by stating that it follows from the convention that $R + T = 1$! This is bad because it deliberately hides an important issue that should be confronted explicitly. But, one might think, at least it’s mathematically valid. But even that is questionable: with the plane-wave scattering state (which is timelessly, simultaneously incident, reflected, and transmitted) why should it be true that $R + T = 1$? At any particular moment (no matter how far in the future) it seems quite possible that the particle is neither reflected nor transmitted but it rather still incident. And so practically every mathematical step is clouded by physical assumptions which are at odds with the actual mathematics.

The vast majority of QM textbooks justify Equations A.10-A.11 by introducing the *probability current*

$$j = \frac{-i\hbar}{2m} \left(\psi^* \frac{\partial \psi}{\partial x} - \psi \frac{\partial \psi^*}{\partial x} \right) \quad (\text{A.13})$$

which describes the flow of quantum mechanical probability, as proved by the fact that the time-dependent Schrödinger equation entails the continuity equation

$$\frac{\partial \rho}{\partial t} + \frac{\partial j}{\partial x} = 0 \quad (\text{A.14})$$

with $\rho = |\psi|^2$ the standard expression for probability density in the theory.

For a plane wave with $\psi = Ae^{ikx}$, Equation A.13 gives the probability current:

$$j = \frac{\hbar k}{m} |A|^2 \quad (\text{A.15})$$

which equals the probability density $|A|^2$ times the group velocity defined previously.

These textbooks state, usually with little explanation, that the reflection and transmission coefficients are given by the ratios of the individual probability currents for the reflected and transmitted terms to the incident current:

$$R = \frac{|j_R|}{j_I} \quad (\text{A.16})$$

and

$$T = \frac{j_T}{j_I}. \quad (\text{A.17})$$

where $j_I \sim k|A|^2$ is the probability current for the incident wave function $\psi_I = Ae^{ikx}$, and analogously $j_R \sim -k|B|^2$ and $j_T \sim \kappa|C|^2$.

Equations A.16-A.17 and the resulting Equations A.10-A.11 can be understood somewhat intuitively by arguing that if the incoming and transmitted waves are traveling at different speeds, then it makes sense that the amount transmitted should be proportional to the ratio of the speeds. However, it is difficult to make a rigorous, rather than hand-waving, argument for why, a priori, Equations A.16-A.17 are the correct expressions for the reflection and transmission coefficients.

It is also difficult to intuitively relate the probability current approach to the interpretation of the probability as the area under the curve. Furthermore, it is not intuitively clear why the relevant speed to use is the group velocity, $d\omega/dk$, rather than the phase velocity, ω/k . In fact, if students investigate an animation of plane wave motion by writing a computer program or using a simulation ¹, only the phase velocity will be apparent to the eye. Furthermore, while the the group velocity of the transmitted wave is smaller than that of the incident wave, the phase velocity will actually be larger, so it is quite easy to develop incorrect intuitions based on the behavior of plane waves. It is quite difficult to get an intuitive sense of the group velocity of a plane wave at all, unless one thinks of it as an infinitely wide wave packet, in which case one can imagine the group velocity as the speed with which this entire packet moves through space. In fact, thinking of very large wave packets seems to be the only way to gain an intuitive sense of plane waves at all – as we will argue in more detail subsequently.

Thus, although the ratio of probability currents does give the correct answer it is far from clear to students (and no doubt many experts) why this should be. Moreover, probability current is a sophisticated concept, which is typically introduced solely for the purpose of deriving the formulas for R and T . Introducing such a concept in the middle of a derivation places extra cognitive load on students, increasing the likelihood that they will give up on understanding and just accept the results “on faith,” as magic formulas to be memorized and used without comprehension.

Further, the same fact that makes this detour into probability currents necessary – that we are dealing with unphysical plane-wave states – can cause further conceptual difficulties,

¹See, for example, the PhET simulation *Quantum Tunneling and Wave Packets*: <http://phet.colorado.edu/new/simulations/sims.php?sim=quantumtunneling>

as shown by physics education research on this topic [7]. Plane waves are mathematically simple. But they completely fail to capture the inherently time-dependent processes that they are being used to describe. For example, we say that a particle approaches a barrier from the left, and then part of it is transmitted and part of it is reflected. The language we use to talk about scattering processes matches the physical processes themselves (e.g., in a real experiment, particles are shot toward a target at a certain time and emerge in some direction or other at some later time) – but there is a deep disconnect between, on the one hand, the language and the physical processes and, on the other hand, the quantum mechanical description in terms of plane-waves.

In summary, the analysis of 1-D scattering in terms of plane wave states, although mathematically simple, requires enough overhead and raises enough conceptual difficulties that the central physical lessons are significantly obscured. Wouldn't it be nice if there were some way of treating this topic that (a) didn't require the overhead of probability current and (b) forced students to think, from the very beginning, that we are really dealing with physical, normalizable *wave packets* to which the plane waves are merely a convenient *approximation*?

Such an approach will be outlined in the following section. In later sections we present also techniques for calculating and approximating R and T probabilities when the incident particle is represented by a gaussian wave packet. These techniques are probably too advanced for students in an introductory course. But our hope (and reason for including them here) is that they may help teachers of quantum physics to realize, fully and explicitly, that the plane wave formulas – e.g., Equations (A.10) and (A.11) – are *approximations*, which are “good” (only) in a certain, intuitively sensible range of physical situations (having to do with the width of the incident packet relative to other length scales in the problem). This perspective is clarifying, and may help repair and prevent the sorts of difficulties mentioned above.

A.3 Scattering probabilities and packet widths

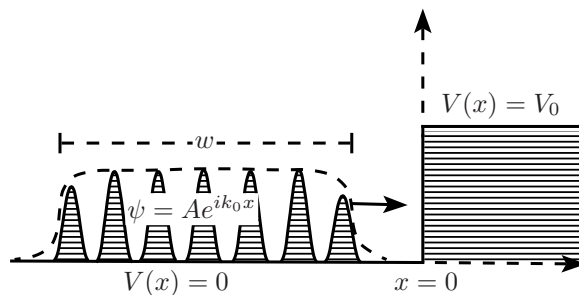


Figure A.1: A Caption Needs to Go Here...

Consider a wave packet approaching the “scattering target” at $x = 0$ for the potential defined in Equation (A.1). Figure A.1 is a diagram of this setup. Assume the packet has

an almost-exactly constant amplitude (A) and wavelength ($\lambda_0 = 2\pi/k_0$) in the region (of width w_I) where the amplitude is non-vanishing, as shown in the Figure. Thus, where the amplitude is non-zero, the packet will be well-approximated by a plane wave:

$$\psi = A e^{ik_0 x}. \quad (\text{A.18})$$

We may assume this incident packet is normalized, so that $w_I |A|^2 \approx 1$.

What happens as the packet approaches and then interacts with the potential step at $x = 0$? To begin with, the packet retains its overall shape as it approaches the scattering center (that is, we assume that the inevitable spreading of the wave packet is negligible on the relevant timescales). It simply moves at the group velocity corresponding to the central wave number for the region $x < 0$:

$$v_g^< = \frac{\hbar k_0}{m}. \quad (\text{A.19})$$

We then divide the process into the following three stages:

- The leading edge of the packet arrives at $x = 0$
- The constant-amplitude “middle” of the packet is arriving at $x = 0$
- The trailing edge of the packet arrives at $x = 0$

Suppose the leading edge arrives at time t_1 . Then the trailing edge will arrive at t_2 satisfying

$$t_2 - t_1 = w_I / v_g^< = w_I m / \hbar k_0. \quad (\text{A.20})$$

And for intermediate times, $t_1 < t < t_2$, we will have – in some (initially small, then bigger, then small again) region surrounding $x = 0$ – essentially the situation described in Equation (A.5), namely: a superposition of rightward- and leftward-directed plane waves (just to the left of $x = 0$) and a rightward-directed plane wave with a different wave number (to the right). And the same relations derived in the previous section for the relative amplitudes of these three pieces will still apply.

While crashing into the scattering center, the incident packet “spools out” waves – with amplitudes B and C given in Equations (A.6) and (A.7) – which propagate back to the left and onward to the right, respectively. These scattered waves will also be wave packets, with the leading edges of the reflected and transmitted packets formed at time t_1 and the trailing edges of the reflected and transmitted packets formed at time t_2 .

This gives a very simple and illuminating way to *derive* Equations (A.10) and (A.11). Consider first the reflected packet. The probability of reflection, R , is by definition just its total integrated probability density – which here will be its intensity $|B|^2$ times its width w_R . But the width of the reflected packet will be the same as the width of the incident packet: because these two packets both propagate in the same region, they have the same group velocity, so the leading edge of the reflected packet will be a distance w_I to the left of $x = 0$ when the trailing edge of the reflected packet is formed. Thus, we have

$$R = w_R |B|^2 = w_I |B|^2 \approx \frac{|B|^2}{|A|^2} \quad (\text{A.21})$$

where we have used the normalization condition for the incident packet $w_I|A|^2 \approx 1$.

Similarly, the total probability associated with the transmitted wave will be its intensity $|C|^2$ times its width w_T . But w_T will be *smaller* than w_I because the group velocity on the right is slower than on the left. In particular: the leading edge of the transmitted packet is created at t_1 ; the trailing edge is created at t_2 ; and between these two times the leading edge will be moving to the right at speed

$$v_g^> = \frac{\hbar\kappa_0}{m} \quad (\text{A.22})$$

where $\kappa_0^2 = k_0^2 - p^2$ is the (central) wave number associated with the transmitted packet. Thus, the width of the transmitted packet – the distance between its leading and trailing edges – is

$$w_T = v_g^> (t_2 - t_1) = \frac{\kappa_0}{k_0} w \quad (\text{A.23})$$

and so the transmission probability is

$$T = w_T|C|^2 \approx \frac{\kappa_0|C|^2}{k_0|A|^2} \quad (\text{A.24})$$

in agreement with Equation (A.11).

To summarize, one can derive the correct general expressions for R and T merely by considering the kinematics of wave packets, without ever mentioning probability current. In particular, the perhaps-puzzling factor of κ_0/k_0 in the expression for T has an intuitive and physically clear origin in the differing *widths* of the incident and transmitted packets, which in turn originates from the differing group velocities on the two sides.

This route to the important formulas is actually simpler than the one traditionally taken in introductory quantum texts: there is a clearly defined initial condition and a definite process occurring in time; probability only enters in the standard way (as an integral of the probability density $|\psi|^2$); and the two quantities needed to define the probabilities (the packet widths and amplitudes) are arrived at separately and cleanly. This approach thus has several virtues in addition to simplicity. First, with proper guidance, focusing on wave packets and a dynamical process in which something (namely scattering) actually happens in time can help students think about the physical process *physically* and/or to connect the mathematics up with real examples. Second, thinking in terms of wave packets can help students recognize that the formulas developed above for reflection and transmission probabilities (and this point applies equally well to three-dimensional scattering situations) are *approximations* and to understand when those approximations do and do not apply.

In particular, the argument presented here suggests that the precise mathematical expressions for R and T above will apply only in the limit of very wide incident packets. This has several aspects. First, we are justified in neglecting the dynamical spreading of the wave packet (and hence, e.g., treating the reflected packet as having the same width as the incident packet) only if the speed of spreading is less than the group velocity, that is, if $\Delta k \ll k_0$, where $\Delta k \sim 1/\Delta x \sim 1/w_I$ is the width of the incident packet in k-space. This

implies that $w_I \gg \lambda_0$, in other words, that the width of the wave packet is much larger than the characteristic wavelength in the region where the amplitude is non-vanishing.

Further, the plane-wave style derivation of the amplitudes assumes that, for some time interval (roughly, $t_1 < t < t_2$), the wave function's structure in some (variable) spatial region around $x = 0$ is indeed given by Equation A.5. But these conditions will simply fail to apply if the actual wave function is (in the appropriate space and time regions) insufficiently plane-wave-like, e.g., if the amplitude of the wave varies appreciably over a length scale $\lambda_0 = 2\pi/k_0$. Thus (assuming a smooth spatial envelope for the packet) the formulas will be valid in the limit $w_I \lambda_0$, which is mathematically equivalent to the limit noted previously.

A.4 Gaussian wave packet scattering from a step potential

It is possible to work out the *exact* R and T probabilities for a Gaussian wave packet incident on the potential step of Equation (A.1). Most of the derivation is worked out in several texts [1, 11], though invariably these texts fail to write down the exact expressions for R and T and instead make last-minute approximations which result in the plane wave results developed previously. But it is worth pushing through the calculation to the end, if only to illustrate that there *is* an exact result to which the plane wave formulas are approximations. Having the exact result in hand also allows one to analytically pick off explicit expressions for first non-vanishing corrections to the plane wave result. That the corrections are small in precisely the limits discussed at the end of the previous section, is a nice confirmation of that discussion.

We begin with an incident Gaussian wave packet, with central wave number k_0 and width σ and centered, at $t = 0$, at $x = -a$:

$$\psi(x, 0) = (\pi\sigma^2)^{-1/4} e^{ik_0(x+a)} e^{-(x+a)^2/2\sigma^2} \quad (\text{A.25})$$

We then follow Shankar's text and proceed in four steps.

Step 1 is to find appropriately normalized energy eigenfunctions for the step potential. These may be parametrized by k and are (up to normalization) just the plane wave states given previously:

$$\psi_k(x) = \frac{1}{\sqrt{2\pi}} \left[\left(e^{ikx} + \frac{B}{A} e^{-ikx} \right) \theta(-x) + \frac{C}{A} e^{i\kappa x} \theta(x) \right] \quad (\text{A.26})$$

where, as before, $\kappa^2 = k^2 - 2mV_0/\hbar^2$ and B/A and C/A are to be interpreted as the *functions of k* given by Equations (A.6) and (A.7). The overall factor of $1/\sqrt{2\pi}$ out front is chosen so that

$$\int \psi_{k'}^*(x) \psi_k(x) dx = \delta(k - k'). \quad (\text{A.27})$$

We are here assuming that only eigenstates with energy eigenvalues $E = \hbar^2 k^2/2m > V_0$ will be present in the Fourier decomposition of the incident packet (and hence we fail to make

explicit special provision for those ψ_k for which κ is imaginary). Note also that there are two linearly independent states for each E only one of which is included here. The orthogonal states will have incoming, rather than outgoing, plane waves for $x > 0$; such states will never enter given our initial conditions.

Step 2 is to write the incident packet as a linear combination of the ψ_k s:

$$\psi(x, 0) = \int \psi_k(x) \phi(k, 0) dk \quad (\text{A.28})$$

where (assuming $\sigma \ll a$ so the amplitude of the incident packet vanishes for $x > 0$)

$$\phi(k, 0) = \left(\frac{\sigma^2}{\pi} \right)^{1/4} e^{-(k-k_0)^2 \sigma^2 / 2} e^{ika} \quad (\text{A.29})$$

turns out to be the ordinary Fourier Transform of $\psi(x, 0)$.

Step 3 is to write $\psi(x, t)$ by appending the time-dependent phase factor to each of the energy eigenstate components of $\psi(x, 0)$:

$$\begin{aligned} \psi(x, t) &= \int \psi_k(x) \phi(k, t) dk \\ &= \int \psi_k(x) \phi(k, 0) e^{-iE(k)t/\hbar} dk \\ &= \left(\frac{\sigma^2}{4\pi^3} \right)^{1/4} \int e^{\frac{-i\hbar k^2 t}{2m}} e^{\frac{-(k-k_0)^2 \sigma^2}{2}} e^{ika} \times \left[e^{ikx} \theta(-x) + \left(\frac{B}{A} \right) e^{-ikx} \theta(-x) + \left(\frac{C}{A} \right) e^{i\kappa x} \theta(x) \right] dk. \end{aligned}$$

We can then finally – Step 4 – analyze the three terms for physical content. The first term, aside from the $\theta(-x)$, describes the incident Gaussian packet propagating to the right. For sufficiently large times (when the incident packet would have support exclusively in the region $x > 0$) the $\theta(-x)$ kills this term – i.e., the incident packet eventually vanishes.

The second and third terms describe the reflected and transmitted packets, respectively. If the factors (B/A) and (C/A) were constants, we would have Gaussian integrals which we could evaluate explicitly to get exact expressions for the reflected and transmitted packets – which would themselves, in turn, be Gaussian wave packets which could be (squared and) integrated to get exact expressions for the R and T probabilities. However, these factors are functions of k . It is not unreasonable to treat them as roughly constant over the (remember, quite narrow) range of k where $\phi(k, 0)$ has support. This is the approach taken by Shankar (and, at least by implication, several other texts) and the result is precisely the plane wave expressions for R and T we developed earlier.

But another approach (which, surprisingly, we have not found in the literature) is also appealing. Consider the second and third terms of Equation (A.30) – which represent (for late times when these terms are non-vanishing) the reflected and transmitted packets. These can be massaged to have the overall form (again assuming t sufficiently large that the θ factors can be dropped)

$$\psi_{R/T}(x, t) = \int \frac{e^{ikx}}{\sqrt{2\pi}} \phi_{R/T}(k, t) dk. \quad (\text{A.30})$$

Putting the two terms in this form requires a change of variables – from k to $-k$ for the R term, and from k to $\sqrt{k^2 + p^2}$ for the T term. The resulting expressions for the k -space distributions of the reflected and transmitted packets are:

$$\phi_R(k, t) = \left(\frac{\sigma^2}{\pi}\right)^{1/4} e^{\frac{i\hbar k^2 t}{2m}} e^{\frac{-(k+k_0)^2 \sigma^2}{2}} e^{-ika} \left(\frac{k + \kappa}{k - \kappa}\right) \quad (\text{A.31})$$

and

$$\phi_T(k, t) = \left(\frac{\sigma^2}{\pi}\right)^{1/4} e^{\frac{-i\hbar(k^2+p^2)t}{2m}} e^{\frac{-(\sqrt{k^2+p^2}-k_0)^2 \sigma^2}{2}} \times e^{ika} \left(\frac{2\sqrt{k^2+p^2}}{\sqrt{k^2+p^2}+k}\right) \frac{k}{\sqrt{k^2+p^2}}. \quad (\text{A.32})$$

But we can just as well integrate the momentum-space wave functions (to find the total probability associated with a given packet) as the position-space wave functions. Thus,

$$\begin{aligned} R &= \int |\phi_R(k, t)|^2 dk \\ &= \left(\frac{\sigma^2}{\pi}\right)^{1/2} \int e^{-(k+k_0)^2 \sigma^2} \left(\frac{k + \kappa}{k - \kappa}\right)^2 dk \\ &= \left(\frac{\sigma^2}{\pi}\right)^{1/2} \int e^{-(k-k_0)^2 \sigma^2} \left(\frac{B}{A}\right)^2 dk \end{aligned} \quad (\text{A.33})$$

where in the last step we have done another change of variables from k to $-k$. This result can be summarized as follows:

$$R = \int P(k) R_k dk \quad (\text{A.34})$$

where $P(k) = |\phi(k, 0)|^2$ is the probability for a given k associated with the incident packet, and R_k is simply the reflection probability for a particular value of k as expressed in Equation A.10.

The analogous result for the T term emerges after some more convoluted algebra:

$$\begin{aligned} T &= \int |\phi_T(k, t)|^2 dk \\ &= \left(\frac{\sigma^2}{\pi}\right)^{1/2} \int e^{-(\sqrt{k^2+p^2}-k_0)^2 \sigma^2} \times \left(\frac{2\sqrt{k^2+p^2}}{\sqrt{k^2+p^2}+k}\right)^2 \frac{k^2}{k^2+p^2} dk \\ &= \left(\frac{\sigma^2}{\pi}\right)^{1/2} \int e^{-(k-k_0)^2 \sigma^2} \left(\frac{C}{A}\right)^2 \frac{\kappa}{k} dk \\ &= \int P(k) T_k dk. \end{aligned} \quad (\text{A.35})$$

where in the next-to-last step we have made a change of variables (back!) from k to $\sqrt{k^2 + p^2}$.

These expressions are exact (subject to the assumptions noted earlier). Note that, if we treat $(B/A)^2$ and $(C/A)^2(\kappa/k)$ as constants that do not depend on k (i.e., if we approximate

these functions by their values at $k = k_0$, which is a good approximation so long as the functions don't vary appreciably in a region of width $1/\sigma$ around k_0 , i.e., if the width σ of the incident packet is very big) we are left with plain Gaussian integrals that can be done to get back the plane-wave-approximation results we started with: $R = (B/A)^2$ evaluated at $k = k_0$, etc.

Unfortunately, the actual integrals are too messy to do exactly. But we can Taylor expand the $(B/A)^2$ and $(C/A)^2(\kappa/k)$ factors around $k = k_0$ to get a series of integrals that can be done, resulting in a power-series expansion (in inverse powers of the packet width w) of the exact R and T . approximations of R and T .

The first two non-vanishing terms for R and T are as follows:

$$R = \left(\frac{k_0 - \kappa_0}{k_0 + \kappa_0} \right)^2 + \left(\frac{2k_0}{\kappa_0^3} + \frac{8}{\kappa_0^2} \right) \left(\frac{k_0 - \kappa_0}{k_0 + \kappa_0} \right)^2 \frac{1}{\sigma^2} + \dots \quad (\text{A.36})$$

and

$$T = \frac{4k_0\kappa_0}{(k_0 + \kappa_0)^2} - \left(\frac{2k_0}{\kappa_0^3} + \frac{8}{\kappa_0^2} \right) \left(\frac{k_0 - \kappa_0}{k_0 + \kappa_0} \right)^2 \frac{1}{\sigma^2} + \dots \quad (\text{A.37})$$

We propose christening as “the plane wave approximation” the large- σ limit of these exact results.

A.5 Discussion

Things to discuss here:

- Respond to the possible objection that “of course” the real R/T probabilities are $\int P(k)R_k dk$, etc. This objection presupposes that it is meaningful to define R and T for plane wave states, which it is really the fundamental point of our paper to deny. So it's a good opportunity to clarify.
- Discuss the generalization of this $R = \int P(k)R_k dk$ type result. What if $P(k)$ has support for k 's where funny things happen, e.g., the associated κ goes imaginary? Does it apply to square barrier tunneling? Or is there something special about this potential step example that makes this work out so nicely? And does it only apply for Gaussian packets, or is it really really general?
- Discuss “real life” JILA type experiments where the plane-wave approximation is bad, and lobby for talking about these with students in order to help motivate the wave-packet approach.
- Figure out some better way of integrating the two main sections here, so that they both become parts of one coherent argument.

Part VII

Bibliography

Bibliography

- [1] C. Cohen-Tannoudji, B. Diu, and F. Laloë. *Quantum Mechanics*. John Wiley, New York, 1977.
- [2] G. Winter F. Remacle. Diffraction Image: a new CCP4 library.
- [3] W. Randolph Franklin. PNPOLY - Point Inclusion in Polygon Test. [web page] http://www.ecse.rpi.edu/Homepages/wrf/Research/Short_Notes/npoly.html, Oct. 2005. [Accessed on 06 Feb. 2008.].
- [4] Dr. Claudio Klein. pck.c. [web page] <http://www.ccp4.ac.uk/ccp4bin/viewcvs/ccp4/lib/DiffractionImage/>, Oct. 1995. [Accessed on 06 Feb. 2008.].
- [5] Abhik Kumar. Analysis Strategy of Powder Diffraction Data with 2-D Detector. Tech. Rep., Office of Science, SULI Program, Stanford Linear Accelerator Center, Menlo Park, CA, December 2005.
- [6] M.I.A. Lourakis. levmar: Levenberg-Marquardt nonlinear least squares algorithms in C/C++. [web page] <http://www.ics.forth.gr/~lourakis/levmar/>, Jul. 2004. [Accessed on 06 Feb. 2008.].
- [7] S. B. McKagan, K. K. Perkins, and C. E. Wieman. A deeper look at student learning of quantum mechanics: the case of tunneling, submitted to Am. J. Phys. arXiv:physics/0802.3194.
- [8] Neil W. Ashcroft & N. David Mermin. *Solid State Physics*. Brooks/Cole Thomson Learning, United States, 1976.
- [9] Nathanael Nerode. Why You Shouldn't Use the GNU FDL. [web page] <http://home.twny.rr.com/nerode/neroden/fdl.html>, Sept. 2003. [Accessed on 23 Feb. 2008.].
- [10] Pasco Scientific, Roseville, CA. *Instruction Manual and Experiment Guide for the PASCO scientific Model WA-9314B*, 1999.
- [11] R. Shankar. *Principles of Quantum Mechanics*. Springer, 1994.
- [12] M. Cotte Ph. Walter J. Susini V.A. Sol, E. Papillon. A multiplatform code for the analysis of energy-dispersive x-ray fluorescence spectra. *Spectrochim. Acta Part B*, 62:63–68, 2007.