Senior Projects Spring 2016

Bard Undergraduate Senior Projects

Spring 2016

Photovoltaics: An Investigation into the Origins of Efficiency on All Scales

Jeremy Alexander Bannister
Bard College

Recommended Citation

Bard

Photovoltaics: An Investigation into the Origins of Efficiency on All Scales

A Senior Project submitted to
The Division of Science, Mathematics, and Computing
of
Bard College

by
Jeremy Bannister

Annandale-on-Hudson, New York
May, 2016

Abstract

This project is comprised of a set of parallel investigations, which share the common motivation of increasing the efficiency of photovoltaics. First, the reader is introduced to core concepts of photovoltaic energy conversion via a semi-classical description of the physical system. Second, a key player in photovoltaic efficiency calculations, the exciton, is discussed in greater quantum mechanical detail. The reader will be taken through a numerical derivation of the low-energy exciton states in various geometries, including a line segment, a circle and a sphere. These numerical calculations are done using Mathematica, a computer program which, due to its powerful symbolic programming language and sophisticated built-in algorithms, is widely used for computational physics. The instructions for replicating the calculations are provided. Finally, the reader will be introduced to the experimentation I performed throughout the year, involving the purchasing, assembling and testing of miniature solar cells. In reading this paper, the reader will begin to gain an understanding of the landscape of photovoltaics and the factors that affect their efficiency, as well as the avenues by which we might hope to achieve an increase in that efficiency in the near future.

Contents

Dedication

I would like to dedicate this project to my loving parents, grandparents, aunts and uncles and to my wonderful girlfriend. Thank you all for supporting me through everything.

Acknowledgments

1
Physics of Photovoltaics

1.1 Doped Materials

The most common type of solar cell is made from crystalline silicon, doped (infused) in different places with phosphorous and boron. Silicon doped with phosphorous is termed "n-type" silicon, while silicon doped with boron is referred to as "p-type". The n and p in these designations stand for "negative" and "positive" respectively, and represent the charge of the dominant mobile charge carrier in the material. I will clarify this below.
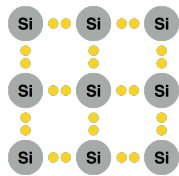
Figure 1.1.1. A pure crystalline silicon lattice.
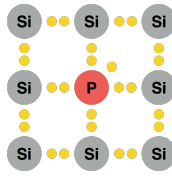
Figure 1.1.2. N-type silicon, doped with phosphorous.
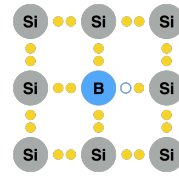
Figure 1.1.3. P-type silicon, doped with boron.

Let's start with pure silicon. In a perfect crystalline silicon lattice each silicon atom is bonded to exactly four neighboring silicon atoms. Since silicon has exactly four valence electrons, and one valence electron is required for each bond, there are no extra electrons left over. Every electron in the lattice is tied up in a bond, as shown in **fig. 1.1.1**. The

nuclei of the silicon atoms are too large to move around very much, and the bound electrons do not move freely, so that there is no moving charge in this lattice. This will be contrasted with the next example, which is n-type silicon.

We refer to silicon that is doped with phosphorous as n-type silicon. Doping is a process in which impurities are intentionally infused into a substance. A doped sample ends up with a nearly perfect silicon lattice, with the exception that there will periodically be a phosphorous atom in place of a silicon atom, as shown in **fig. 1.1.2**. Phosphorous has five valence electrons, four of which will be used to bond with its neighboring silicon atoms and thus be rendered immobile. However, the fifth valence electron of phosphorous (shown in **fig. 1.1.2** drifting to the upper right of the nucleus) is not used in any bond, and is consequently relatively easy to move around the lattice. This fifth electron is therefore a mobile charge carrier, because it is charged (negatively) and can move around the lattice at will. Hopefully this somewhat clarifies the statement I wrote above which said that n-type silicon is so-called because the charge of its dominant charge carrier is negative.
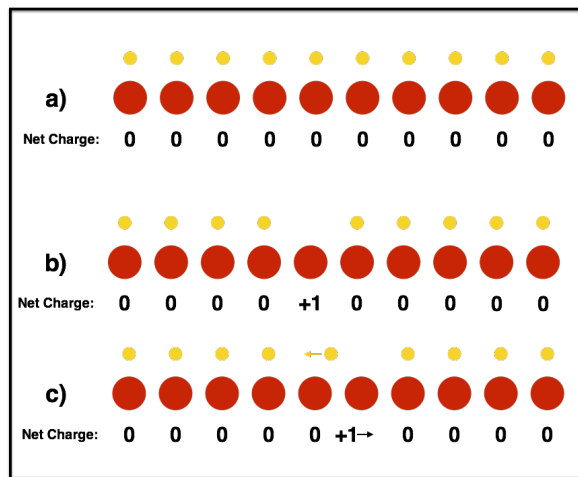
Figure 1.1.4. A string of protons illustrates the behavior of an electron hole.

The question that may come to mind at this point is, "if the positively charged nuclei in the lattice are too large to move, then what are the positive charges that move around in

p-type silicon?" The answer to this question is "holes". A hole is not a particle, but rather, the lack of an electron. To understand how this works, imagine that we have a string of 10 protons, and on top of each one there is an electron, as shown in **fig. 1.1.4 a)**. An electron has exactly the opposite charge of a proton which means that the net charge at every point in **fig. 1.1.4 a)** is 0. Now imagine that we remove the fifth electron, as shown in **fig. 1.1.4 b)**. The fifth proton is no longer cancelled out by an electron, so the net charge at that point is now positive. As you can see, the location of the positive charge corresponds to the place where there is a "hole" in the string of electrons. The tricky thing to understand about holes is that, although they are not physical particles, mathematically they can be treated as such. In other words, they move like physical particles. You can see in **fig. 1.1.4 c)** that, while in reality the sixth electron is sliding over to the left, the effect is that the net positive charge moves to the right in unison with the hole. In this way we can treat the hole as a positive charge carrier.

This brings me to the final example of doped silicon, which is p-type silicon. Generally p-type silicon is doped with boron, but in theory it could be doped with any element that has only three valence electrons, such as aluminum. In **fig. 1.1.3** you can see a boron atom positioned within the silicon lattice. Boron has only three valence electrons, which means that it only bonds to three of its four neighboring silicon atoms. This results in a missing electron between the boron atom and the fourth silicon atom. Again, this hole is mobile and can be treated as a physical object. When we say that the hole "moved" to the right, what we really mean is that the electron next to it moved to the left to fill that hole, leaving a new hole where that electron had been. Mathematically, however, this can be simply treated as a positive charge moving to the right.

1.2 PN Junctions

1.2.1 Introduction

The centerpiece of an operating solar cell is something called a p-n junction. This is simply the interface between a piece of p-type silicon and a piece of n-type silicon that have been pressed together. In order to understand the physics of a p-n junction we need to understand the various ways in which mobile charge carriers (electrons and holes) can be caused to move. There are two major forces that influence charge carriers, and they are called diffusion and drift.

1.2.2 Diffusion

Diffusion is the first "force" that I will discuss, and I put force in quotation marks because it is not a real force which acts on the charge carrier, but rather what you might call a statistical force. I will discuss diffusion of electrons, but everything said here applies to holes as well. We all have an intuitive understanding of the concept of diffusion, but to use it in the context of solar cells we must make it rigorous. Imagine that we have a box full of electrons, in which we have pushed every electron off to one side, as pictured in **fig. 1.2.1**. One can imagine that, if we were to let go of the electrons and allow them to whiz around freely, they would diffuse around the space, eventually being pretty much evenly distributed around the whole box. Another way to say this is that the electrons started on the left, but after some time they take up the whole box (**fig. 1.2.2**), meaning that, on average, the electrons moved to the right. Thus, diffusion acts as a force, in this case pushing the electrons to the right. The direction of this force is given by the negative gradient of the electron density n,

$$\vec{F} \propto -\vec{\nabla} n \ .$$

The reason that it is the negative gradient is because the gradient points in the direction of steepest ascent of the density, meaning that it points in the most dense direction, but as we saw in this example, the electrons actually tended to move in the least dense direction, meaning that the force of diffusion acts in the opposite direction of the gradient.
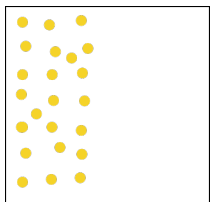
Figure 1.2.1. A box of electrons in which all the electrons are clustered on one side. The clustering means there is a density gradient.
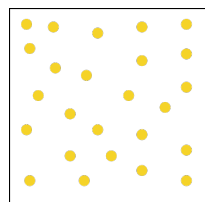
Figure 1.2.2. The same box as in **fig. 1.2.1**, at a later time, illustrating the natural movement of the electrons due to diffusion.

1.2.3   Drift

The other force that can accelerate electrons (and holes) is drift. Drift is simply the electromagnetic force experienced by a charge in an electric field. By convention, we define the direction of the electric field to be the opposite of the direction that an electron in that field would be accelerated. Consequently, electrons tend to drift against the electric field that is present at the p-n junction, and holes tend to drift with the field.

1.3   Going Deeper

While learning about the broader strokes of the operating principles of a solar cell, I felt compelled to first understand the more primary physics, getting down to the quantum mechanical level. This transition in my learning, from the classical description of photovoltaics to the quantum description, was swift and occurred early on in the year. I therefore will stop here in my discussion of solar cells in a classical tone in favor of accurately reflecting the work that I did during this year on the quantum nature of photovoltaics.

2

Excitons in Curved Geometries: A Quantum Description

2.1   Introduction

In this chapter I will explore a variety of quantum mechanical situations that relate to the study of photovoltaics. The central entity that guides this chapter and serves as the link to photovoltaic physics is the exciton. An exciton is an electron and an electron hole coupled together. One is produced when an electron is excited into the conduction band of a material, leaving behind it a positively charged hole. The opposite charge of the electron and the hole causes an attraction which tends to couple the motion of the two. When they are bound in this way they are collectively referred to as an exciton.

The reason that this particular entity is appearing in a project about photovoltaics is that it is actually central to the quantum mechanical operation of a P-N junction. When a photon strikes the junction, it excites an electron out of the ground state. Where the electron used to be is now a positive hole, which tries to pull the electron back into the ground state. If the solar cell is well designed then the hole will fail at this, and the electron will escape around the circuit where it is free of the influence of the hole. However, in the time between the initial excitation of the electron and its final trip around the circuit there

is still some interaction between the electron and the hole. This interaction is one of the processes that must be understood in order to increase the efficiency of the solar cell.

In light of this important role that excitons play in the operation of a solar cell, I will go through a number of quantum mechanical calculations that relate to excitons, increasing the complexity as I go. The way in which I will increase the complexity is by changing the geometry of the space in which the exciton exists. To begin with I will get us warmed up with the simplest geometry of all—a line segment. After this I will move on to a curved version of this geometry, i.e., a circle. Finally, I will move on to the natural extension of a circle into two dimensions: a sphere.

2.1.1   The Reduced Mass

In the following calculations I will be making an important and highly simplifying transformation known as the reduced mass transformation. The way it simplifies the problem is that it allows me to instead solve for the motion of only one fictitous particle with half the mass of an electron, and the result from that will contain all of the information of the real solution I am looking for. In this case, half the mass of the electron is known as the *reduced mass*. In general, the reduced mass of a two body system is $\mu \equiv \frac{m_1 m_2}{m_1 + m_2}$. In the case of the exciton, the reduced mass is equal to $\frac{1}{2}m_{e^-}$ because the hole, even though it is not a real particle, has an effective mass equal to that of an electron. It can be easily verified that $\mu = \frac{1}{2}m$ when $m_1 = m_2$. Before beginning the calculations, I will explore further this reformulation of the system in terms of the reduced mass, because it is a critical component of this paper and merits some deeper discussion.

2.2   Reduced Mass Transformation

2.2.1   Change of Coordinates

In classical mechanics, a two body system can be described in a certain way which makes use of the *reduced mass*. The first step toward describing the system in the new way is to make a change of coordinates. The standard way to describe the system would be to use the position vectors of the two bodies separately:

$$\vec{r}_1 = \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} \qquad \vec{r}_2 = \begin{pmatrix} x_2 \\ y_2 \end{pmatrix}. \tag{2.2.1}$$

To make use of the reduced mass we will instead describe the system in terms of the new coordinates,

$$\vec{r} = \vec{r}_2 - \vec{r}_1 \qquad \vec{R} = \frac{m_1 \vec{r}_1 + m_2 \vec{r}_2}{m_1 + m_2}. \tag{2.2.2}$$

The first coordinate, $\vec{r}$, is what is called the *separation coordinate*. It is clear to see that it is the vector that points from the first body to the second body. The second coordinate, $\vec{R}$, is the *center of mass coordinate*. The reason that it is called this is because it actually gives the position of the collective center of mass of the two bodies. In the case of an electron and a hole, where the masses are equal, the center of mass coordinate reduces to $\vec{R} = \frac{1}{2} \begin{pmatrix} x_1 + x_2 \\ y_1 + y_2 \end{pmatrix}$, which makes perfect sense because it says that for two bodies of equal mass, the center of mass is exactly halfway between them.

2.2.2   Enter the Reduced Mass

Now we are ready to appreciate the advantage of switching to these coordinates. The first insight to note is that, as long as the two bodies are interacting with nothing but each other, the center of mass coordinate will evolve like a free particle, i.e., it will advance linearly in one direction, or not move at all. As always, constant linear motion of the whole

system is basically the same thing as no motion at all, because we could always imagine that we ourselves are moving smoothly alongside the system, causing it to appear to us as if it were standing still. Thus, the center of mass of the two bodies is effectively motionless, meaning that the only interesting physics is encoded in the separation coordinate, $\vec{r}$. The point here is that, if the two bodies are interacting in free space, it doesn't matter *where* they are in the universe, it just matters where one is *relative* to the other. This is a huge breakthrough, because with two of our four coordinates now rendered trivial, the mathematics is greatly simplified.

If we can figure out how $\vec{r}$ evolves in time, then we know everything we want to know about the system. The question then is, "how do we solve for the separation coordinate?" The answer is that we pretend that we're solving a different problem, and it will turn out that the answer to this different problem is the same as the answer for $\vec{r}$. The different problem is this: we fix one of our two original bodies at the origin, and we just solve for the motion of the other body around it. The only thing we have to change is that the body that is moving has mass equal to the reduced mass of the system, $\mu \equiv \frac{m_1 m_2}{m_1 + m_2}$. Whatever solution we get for the position of the moving body as a function of time will be the same as the separation coordinate as a function of time. Solving for the motion of one body in a potential is far easier than solving for the independent motion of two interacting bodies, and thus the reduced mass transformation helps us to crack the problem.

2.3   Reduced Mass in Quantum Mechanics

In quantum mechanics, we can start out by writing down the Schrödinger equation. When the potential is not explicitly dependent on time we can jump straight to what is known as the time-independent Schrödinger equation:

$$\frac{-\hbar^2}{2m}\nabla^2\psi + V(\vec{r})\psi = E\psi, \tag{2.3.1}$$

where ∇^2 is the Laplacian operator in three dimensions, V is the potential, the argument of V, $\vec{r}$, is the position of the particle, and E is the total energy of the particle. In order to generalize this equation to represent two particles, we must instead write,

$$\frac{-\hbar^2}{2m}(\nabla_{\vec{r}_1}^2 + \nabla_{\vec{r}_2}^2)\psi(\vec{r}_1, \vec{r}_2) + V(\vec{r}_1, \vec{r}_2)\psi(\vec{r}_1, \vec{r}_2) = E\psi(\vec{r}_1, \vec{r}_2), \qquad (2.3.2)$$

where $\nabla_{\vec{r}_1}^2$ and $\nabla_{\vec{r}_2}^2$ are Laplacian operators with respect to each of the coordinates of particle 1 and particle 2 respectively, $\vec{r}_1$ and $\vec{r}_2$ are the positions of particle 1 and particle 2 respectively, and E is the total energy of the two particles combined. In writing Eq. (2.3.2) in this way I have assumed that the two particles have the same mass, and I will continue to make this assumption in every calculation moving forward. I do this because, in the case of the exciton, the two particles do indeed have the same mass and it will make all of the equations a good bit cleaner.

2.3.1 Change of Coordinates

The first step to reduce this to a one-particle problem is to change coordinates, and we will of course choose our new favorite coordinates, $\vec{r}$ and $\vec{R}$. Since we haven't yet specified what $V(\vec{r}_1, \vec{r}_2)$ is we will first focus on transforming the first term of Eq. (2.3.2) to our new coordinates. To start, let's write our new coordinates explicitly:

$$\vec{r} = \vec{r}_2 - \vec{r}_1 = \begin{pmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_2 - z_1 \end{pmatrix} \equiv \begin{pmatrix} x \\ y \\ z \end{pmatrix} \qquad \vec{R} = \frac{1}{2}(\vec{r}_1 + \vec{r}_2) = \frac{1}{2}\begin{pmatrix} x_1 + x_2 \\ y_1 + y_2 \\ z_1 + z_2 \end{pmatrix} \equiv \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}.$$

The explicit form of the Laplacian operator is $\nabla_{\vec{r}_1}^2 = \frac{\partial^2}{\partial x_1^2} + \frac{\partial^2}{\partial y_1^2} + \frac{\partial^2}{\partial z_1^2}$, so our task is to transform each of these three terms to our new coordinates. Let's start with $\frac{\partial^2}{\partial x_1^2}$. By the chain rule,

$$\frac{\partial}{\partial x_1} = \frac{\partial x}{\partial x_1}\frac{\partial}{\partial x} + \frac{\partial X}{\partial x_1}\frac{\partial}{\partial X}$$
$$= -\frac{\partial}{\partial x} + \frac{1}{2}\frac{\partial}{\partial X}$$

so,

$$\frac{\partial^2}{\partial x_1^2} = (-\frac{\partial}{\partial x} + \frac{1}{2}\frac{\partial}{\partial X})^2$$
$$= \frac{\partial^2}{\partial x^2} - \frac{\partial^2}{\partial x \partial X} + \frac{1}{4}\frac{\partial^2}{\partial X^2}.$$

The same calculation can be done for $\frac{\partial^2}{\partial x_2^2}$, and will yield the result,

$$\frac{\partial^2}{\partial x_2^2} = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial x \partial X} + \frac{1}{4}\frac{\partial^2}{\partial X^2},$$

the only difference being the sign of the cross term, $\frac{\partial^2}{\partial x \partial X}$. This is important because in Eq. (2.3.2) the two Laplacian operators $\nabla_{\vec{r}_1}^2$ and $\nabla_{\vec{r}_2}^2$ are summed, meaning that the cross terms will cancel leaving $2(\frac{\partial^2}{\partial x^2} + \frac{1}{4}\frac{\partial^2}{\partial X^2})$. Furthermore, it can be easily verified that the second partial derivatives with respect to the y and z variables will have the same structure when transformed to the new coordinates. The result, after the full coordinate transform, is,

$$\nabla_{\vec{r}_1}^2 + \nabla_{\vec{r}_2}^2 = 2\frac{\partial^2}{\partial x^2} + \frac{1}{2}\frac{\partial^2}{\partial X^2} + 2\frac{\partial^2}{\partial y^2} + \frac{1}{2}\frac{\partial^2}{\partial Y^2} + 2\frac{\partial^2}{\partial z^2} + \frac{1}{2}\frac{\partial^2}{\partial Z^2}$$
$$= 2(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}) + \frac{1}{2}(\frac{\partial^2}{\partial X^2} + \frac{\partial^2}{\partial Y^2} + \frac{\partial^2}{\partial Z^2})$$
$$\equiv 2\nabla_{\vec{r}}^2 + \frac{1}{2}\nabla_{\vec{R}}^2.$$

Before plugging this back into Eq. (2.3.2), we will observe one more convenience of these new coordinates. We currently have the potential written as $V(\vec{r}_1, \vec{r}_2)$. In the case of the

exciton (and almost every other physical two body system) the potential is a function only of the distance between the particles, which in our new coordinates is exactly $|\vec{r}|$. This dependence of the potential solely on the separation, and not on the center of mass, is the mathematical reason that the center of mass will always behave like a free particle. Thus, in our new coordinates the potential can be rewritten as simply $V(\vec{r})$. Now we can finally rewrite Eq. (2.3.2) in our new coordinates,

$$-\frac{\hbar^2}{4m}\nabla^2_{\vec{R}}\psi(\vec{r},\vec{R}) - \frac{\hbar^2}{m}\nabla^2_{\vec{r}}\psi(\vec{r},\vec{R}) + V(\vec{r})\psi(\vec{r},\vec{R}) = E\psi(\vec{r},\vec{R}). \tag{2.3.3}$$

Eq. (2.3.3) is the Schrödinger equation for our system, now expressed in the coordinates $\vec{R}$ and $\vec{r}$ (the center of mass and the relative position of the electron with respect to the hole). The first term represents the kinetic energy of the center of mass of the system, but as I mentioned earlier, that kinetic energy is relative and nothing changes if we assume that it is 0, so that is what we will do. Thus, with $\vec{R}$ always equal to 0, the equation becomes,

$$-\frac{\hbar^2}{m}\nabla^2_{\vec{r}}\psi(\vec{r}) + V(\vec{r})\psi(\vec{r}) = E\psi(\vec{r}). \tag{2.3.4}$$

You will notice that this is identical to the Schrödinger equation for a single particle in the potential $V(\vec{r})$, except that where we would normally have $2m$ we now only have m. This is the punch line of the reduced mass: we can solve the much easier problem of a single particle in a potential and as long as we use the reduced mass, the result that we get will be the answer to how the particles will move relative to each other.

2.3.2 *Interpreting the Results*

One of the trickier aspects of this strategy is interpreting the answer that you get. You might ask, "which particle is described by the resulting wavefunction?" The answer is both. Essentially, the way the wavefunction works is that you give it a value of the coordinate

and it gives you a corresponding probability. Usually we interpret the input value as the location of the particle, and we interpret the output as the probability of finding that particle at that location. In the case of the reduced mass wavefunction, the input is indeed a value of the coordinate, but rather than that coordinate corresponding to the position of a single particle, it corresponds to the separation vector pointing from one particle to the other. Since we have decided that the center of mass will remain motionless at the origin, the vector value of $\vec{r}$ that we input fully specifies the positions of each particle. In this way, we are able to obtain the probability of finding the exciton in any given configuration. Remember that we can't actually input an exact value of $\vec{r}$, because the answer will always be 0. There is 0 chance of finding the exciton in *exactly* one configuration. Instead we use a small range of values of $\vec{r}$ and obtain a probability of the system being somewhere in that range.

2.3.3 Summary

To summarize, here are the steps to using the reduced mass transformation for an exciton:

1. Determine the geometry and physical constraints you are working in.

2. Imagine that you fix the positively charged hole at the origin.

3. Determine the exact form of the potential in the given geometry, imagining that the hole is fixed at the origin. This basically boils down to getting an expression for distance in the given geometry which you can plug into the standard form of the coulomb potential: $\frac{+q}{4\pi\epsilon_0|\vec{r}|}$. We shall see, however, that there can be slightly more to this step when working in compact geometries (such as a sphere).

4. Plug the expression you obtain for the coulomb potential into the reduced mass Schrödinger equation: $-\frac{\hbar^2}{m}\nabla_{\vec{r}}^2\psi(\vec{r}) + V(\vec{r})\psi(\vec{r}) = E\psi(\vec{r})$

5. Obtain solutions (the wavefunctions) from this equation.

2.4   Line Segment

2.4.1   Set Up

The absolute simplest case that I will discuss is an exciton on a line segment. For the reduced mass approximation I will fix the hole at one end of the line segment $(x = 0)$. At the other end $(x = L)$ I will impose that the potential is infinite, in order to confine the particle and obtain normalizable solutions. A very important caveat is that we are not interested in recombination, which is what happens when the electron is allowed to go to the actual location of the hole. Mathematically we prevent this by saying that the potential is also infinite at some fixed radius from the hole. The reason we need to do this is because the potential has a singularity at the hole (at $x = 0$) where it goes to negative infinity, which would essentially suck the electron in. In reality there is no singularity at the center of a hole, because it is not a real particle, and so it is perfectly reasonable to decide to force it to go away.
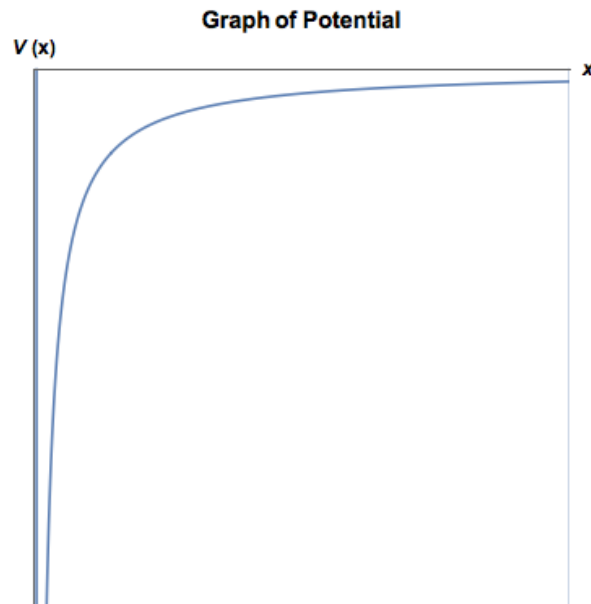
Figure 2.4.1. The graph of the potential.

2.4.2   Analytic Solution and Mathematica

Since the potential is independent of time we can jump right to solving the time-independent reduced-mass Schrödinger equation:

$$\frac{-\hbar^2}{m}\frac{d^2\psi}{dx^2} + V(x)\psi = E\psi \tag{2.4.1}$$

Although the potential is piecewise, there is really only one interesting part, which is the coulomb curve. The other two pieces (the vertical lines) only serve to make the wavefunction go to 0 at those points, which we can enforce using boundary conditions. Thus the potential is,

$$V(x) = \frac{-q^2}{4\pi\epsilon_0 x} \tag{2.4.2}$$

and substituting this into Eq. (2.4.1) gives,

$$\frac{-\hbar^2}{m}\frac{d^2\psi}{dx^2} + \frac{-q^2}{4\pi\epsilon_0 x}\psi = E\psi \tag{2.4.3}$$

In order to obtain the solutions to this problem and to the other problems presented in this project I used Mathematica. In the figure below (**fig. 2.4.2**) the first four inputs in my Mathematica notebook are shown. The first is where I set up the differential equation (2.4.1). The second and third inputs are to define the boundary conditions which will be used. The fourth line obtains the analytic solution to Eq. (2.4.1).

As you can see, the general structure of the analytic solution to this differential equation (without imposing boundary conditions) is,

$$A\cos(kx) + B\sin(kx) + (\cos(kx)\,\text{sinIntegral}(kx) - \sin(kx)\,\text{cosIntegral}(kx)) \tag{2.4.4}$$

where, $k \equiv \frac{\sqrt{2mE}}{\hbar}$. The sine integral function, in case you are unfamiliar with it, is $Si(x) = \int_0^x \frac{sin(x)}{x}dx$. The first boundary condition, $\psi(holeRadius) = 0$, is one of the ones discussed above, which we are imposing in order to prevent problems with the singularity at $x = 0$. The second boundary condition in **fig. 2.4.2**, however, is not the other boundary condition

The Equation

In[1]:= $\text{eqn} = \dfrac{-\hbar^2}{m} \psi''[x] - \dfrac{q^2}{4 \pi \epsilon_0 x} == e_i \psi[x];$

Boundary Conditions

In[2]:= $\text{boundary1} = \psi[\text{holeRadius}] == 0;$
$\text{boundary2} = \psi[2*\text{holeRadius}] == 0.01;$

Symbolic Solution

Without boundary conditions

In[4]:= $\text{symbolicSolutionNoBoundary} = \text{DSolve}[\{\text{eqn}\}, \psi, x]$

Out[4]= $\left\{\left\{\psi \to \text{Function}\left[\{x\}, C[2] \cos\left[\dfrac{\sqrt{m}\, x\, \sqrt{e_i}}{\hbar}\right] + C[1] \sin\left[\dfrac{\sqrt{m}\, x\, \sqrt{e_i}}{\hbar}\right] + \dfrac{1}{4\pi\hbar\sqrt{e_i}\,\epsilon_0}\right.\right.\right.$

$\left.\left.\left.\left(-\sqrt{m}\, q^2\, \text{CosIntegral}\left[\dfrac{\sqrt{m}\, x\, \sqrt{e_i}}{\hbar}\right] \sin\left[\dfrac{\sqrt{m}\, x\, \sqrt{e_i}}{\hbar}\right] + \sqrt{m}\, q^2 \cos\left[\dfrac{\sqrt{m}\, x\, \sqrt{e_i}}{\hbar}\right] \text{SinIntegral}\left[\dfrac{\sqrt{m}\, x\, \sqrt{e_i}}{\hbar}\right]\right)\right]\right\}\right\}$

Figure 2.4.2. The first four inputs of my Mathematica notebook for this problem.

discussed above, namely that at $x = L$, ψ is also equal to 0. Instead there is written a seemingly arbitrary condition, $\psi(2*holeRadius) = 0.01$. This condition is indeed arbitrary, the only constraint on it being that it sets ψ at some arbitrary point in the range to some value other than 0.

The reason for this is that, in most cases, Mathematica is not smart enough to handle quantized solutions to a differential equation. We know that once we apply our boundary conditions (that ψ must go to 0 at the end points) the solutions to the Schrödinger equation are discrete. What I mean is that, for most values of E (the energy), if we impose that ψ is 0 at one end, it will not also go to 0 at the other end. Only specific values of E will cause both boundary conditions to be met simultaneously. So as to not confuse Mathematica, we start out by giving the boundary condition that it can always handle (which is that ψ is 0 at only one of the ends) and then we give a boundary condition that Mathematica can always satisfy by scaling the solution appropriately. More on this below.

With this particular example Mathematica was able to produce an exact analytic solution in terms of the symbols that the equation is written in such as $\hbar$, ϵ_0 and E. Because

this is not always possible (and in fact is very rarely possible) I will develop here a method by which we can numerically derive the solutions.

2.4.3 Numerical Solution

The first thing that is necessary in order to numerically solve Eq. (2.4.3) is to give numerical values to all of the symbols which are not variables. In **fig. 2.4.3** you can see that I set all of the constants equal to 1 for simplicity, except for holeRadius which is defined as a small length relative to the length (L) of the line segment. The real meat of this technique comes in the next line, in which the built-in Mathematica function **ParametricNDSolve** is used.

Here is a summary of three standard differential-equation-solving Mathematica functions and how they differ from each other.

1. The first is **DSolve**, which you saw used in **fig. 2.4.2**. Used in the most rudimentary way, it accepts as inputs, 1) an equation, 2) the function you would like to solve for, and 3) the variable that the function takes as its argument. In the example above, the function is ψ and the variable is x, indicating that I am looking for what ψ, as a function of x, is equal to. **DSolve** attempts to return a closed form symbolic solution to the equation. It can be given partial differential equations, systems of equations, boundary conditions, and multiple independent variables as well, though that is not how I used it here.

2. The second is **NDSolve**, which accepts very similar inputs to **DSolve**, the main difference being that rather than returning a closed form solution, **NDSolve** returns what is called an "interpolating function", which is a bundle of data that can be plotted and numerically queried, but not manipulated algebraically. The only difference in the inputs is that, rather than giving just the independent variable (x in the previous example) you must specify a range for that variable like this: $\{x, 0, 10\}$.

Defining Constants

In[5]:= $\hbar = 1$; $m = 1$; $q = 1$; $\epsilon_0 = 1$; $L = 1$; holeRadius $= \dfrac{L}{100.}$;

Numeric Solution (parameterized by energy)

Solution

In[6]:= `nsol = ParametricNDSolve[{eqn, boundary1, boundary2},` ψ`, {x, holeRadius, L}, {e`$_i$`}]`

Out[6]= $\left\{ \psi \rightarrow \texttt{ParametricFunction}\left[\boxplus \quad \mathbf{M} \quad \begin{array}{l} \text{Expression: } \psi \\ \text{Parameters: } \{e_i\} \end{array} \right] \right\}$

Finding Allowed Energies

In[12]:= `endPoint[`*energy_*`] :=` ψ`[`*energy*`][L] /. nsol`

In[13]:= `allowedEnergies = findAllRoots[endPoint[t], {t, 0, 400}]`

Out[13]= {8.98855, 38.8411, 88.2096, 158.497, 248.274, 358.928}

In[14]:= `Plot[Evaluate[endPoint[a], {a, 0, 400}]]`

Out[14]=
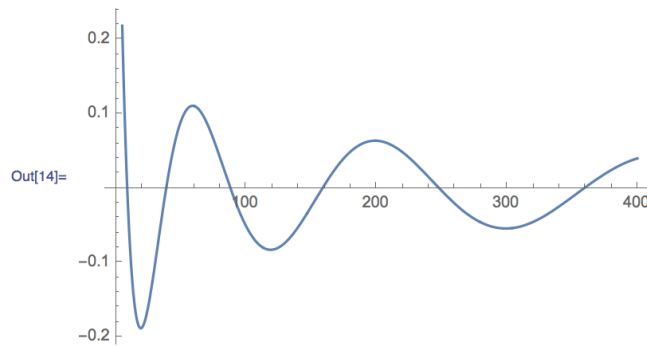
Figure 2.4.3. This code is used to automatically obtain the allowed energies for the exciton within a certain energy range.

3. The third Mathematica function is called **ParametricNDSolve**. It also employs a numeric algorithm, but with one crucial difference: you can leave some of the variables in the equation unspecified, and it will basically solve the differential equation for all values of that parameter. This allows us to obtain an interpolating (plottable) function for any value of the parameter that we want. In the context of quantum mechanics I will use this to leave the energy unspecified, such that I can choose any energy I want and plot the corresponding wavefunction.

As you can see, I used **ParametricNDSolve** on Eq. (2.4.3), including the boundary conditions, solving for ψ as a function of x on the region $(holeRadius, L)$, leaving the energy e_i as an unspecified parameter. What we get out of that input, as described above, is a Mathematica object that will return a different interpolating function for every value of the parameter e_i. The thing to remember is that, while we are now able to give any energy we want and get out the corresponding wavefunction, we actually only want what we call the "allowed" wavefunctions. Only specific values of the parameter e_i will yield wavefunctions which are acceptable. What do we mean by "acceptable"? We mean that it satisfies the boundary conditions—in this case that it is 0 at both end points. This brings me back to the boundary conditions which I discussed above. The *real* boundary conditions are that ψ is 0 at both end points. However, I replaced one of those conditions with a fake one, which is simply that the wavefunction has some arbitrary value at some arbitrary point. This is done because no matter what value of e_i I provide, there is always a corresponding wavefunction which is simultaneously 0 at the left end point and is equal to the arbitrary value at the arbitrary point. It is not true, however, that no matter what value of e_i I provide there will be a wavefunction which goes to 0 at *both* endpoints. That is only true for specific values of e_i, and our first job is to find out what those values are.

The strategy that I employ here is fairly simple. Using the output of **ParametricND-Solve** I am able to define a function (called **endPoint**) that takes a value for energy as input and returns the corresponding wavefunction evaluated at the right endpoint. The reason this is a useful definition is because I can guess an energy and feed it into this function, and if I get back 0 then I know it was an allowed energy. This is because allowed energies are specifically those for which the wavefunction goes to 0 at the right endpoint (due to the first boundary condition they will all go to 0 at the left endpoint).

In order to find the allowed energies then, we only need to find the zeros of our new function called **endPoint**. This can be done manually by plotting the function and then

using the built-in Mathematica function **FindRoot** around the locations where you see a zero-crossing. It can also be done in a slightly more automated way, however, which can save quite a lot of time. I was able to find a very useful bit of Mathematica code on mathematica.stackexchange.com which returns a list of the zeros of a function in a given range. The code is pictured below in **fig. 2.4.4**.

```
Find All Roots Function

In[7]:= Clear[findAllRoots]
       SyntaxInformation[findAllRoots] = {"LocalVariables" → {"Plot", {2, 2}}, "ArgumentsPattern" → {_, _, OptionsPattern[]}};
       SetAttributes[findAllRoots, HoldAll];

       Options[findAllRoots] = Join[{"ShowPlot" → False, PlotRange → All}, FilterRules[Options[Plot], Except[PlotRange]]];

       findAllRoots[fn_, {l_, lmin_, lmax_}, opts : OptionsPattern[]] :=
        Module[{pl, p, x, localFunction, brackets}, localFunction = ReleaseHold[Hold[fn] /. HoldPattern[l] ⧴ x];
         If[lmin ≠ lmax, pl = Plot[localFunction, {x, lmin, lmax}, Evaluate@FilterRules[Join[{opts}, Options[findAllRoots]], Options[Plot]]];
          p = Cases[pl, Line[{x__}] ⧴ x, Infinity];
          If[OptionValue["ShowPlot"], Print[Show[pl, PlotLabel → "Finding roots for this function", ImageSize → 200, BaseStyle → {FontSize → 8}]]], p = {}];
         brackets =
          Map[First,
           Select[(*This Split trick pretends that two points on the curve are "equal" if the function values have _opposite_ sign.Pairs of such sign-
             changes form the brackets for the subsequent FindRoot*)Split[p, Sign[Last[#2]] == -Sign[Last[#1]] &], Length[#1] == 2 &], {2}]];
         x /. Apply[FindRoot[localFunction == 0, {x, ##1}] &, brackets, {1}] /. x → {}]
```

Figure 2.4.4. This borrowed piece of code defines a function called **findAllRoots** which takes as input a function and a range for the variable, and returns all values of the variable in that range for which the function is 0. I have only used this code on smooth, well-behaved functions, so I do not know how robust it is.

Thus, the prescription for finding the allowed energies is:

1. Using **ParametricNDSolve**, obtain a Mathematica "Parametric Function" which takes an energy as input and returns an interpolating function corresponding to the wavefunction for that energy, which can be plotted.

2. Define a function called **endPoint** which gives the value of the wavefunction corresponding to a certain energy at the right endpoint.

3. Use the function **findAllRoots** to find the zeros of **endPoint** in whatever range you provide.

The final step will be to plot the wavefunctions using the allowed energies. Before we do that, however, it is good practice to find the normalization constants. This can be done in

one line of code using the Mathematica "/@" operator to apply the code to each element of a list. The necessary code is pictured below:

Figure 2.4.5. Finding the normalization constants.

Finally, it is time to plot the normalized wavefunctions. The way that I chose to do this was to group the allowed energies with their corresponding normalization constants into a single list titled **energiesAndNormalizationConstants**. This can be achieved using the built-in Mathematica function called **Transpose**. I then plotted the graph for each allowed energy, scaled by the respective normalization constant. Pictured below in **fig. 2.4.6** is the code to do this and the resulting graphs.

Figure 2.4.6. The first six energy eigenstates of an exciton on a line segment.

It is important to remember when looking at these graphs that the x-axis actually represents the separation of the two particles. For example, the first one, which clearly

yields the highest probability at $x \approx 0.5$, actually means that the electron and hole will most probably be separated by half of the length of the line segment, and says nothing about the absolute location of either one. To find out about the absolute location we would need to solve the center of mass equation that we derived earlier. That equation has no potential term, because the interaction potential depends only on the separation of the particles, so the exciton as a whole will behave as a free particle in a box, the solutions for which can be found in any introductory quantum mechanics textbook.

2.4.4   Conclusion

We have now fully solved the problem of an exciton on a line segment. To make sure that you understand the results and how to interpret them I will describe the ground state in words. The wavefunctions in **fig. 2.4.6** represent the separation of the electron and the hole. The first one, which is the ground state, is just one big hump that starts and ends at 0 and peaks in the middle. While normally this would mean that there is a single particle that is most likely to be found in the middle and very unlikely to be found toward the edges, in our reduced mass formulation this graph gets a different interpretation. In our context this graph tells us nothing about where we might expect to find either particle, but it does tell us that once we do find one we know that we are most likely to find the other at a distance of half of the length of the line segment from it, and we are very unlikely to find it extremely close to or far from the first one.

One of the perks of switching to the center of mass and separation coordinates is that there is no potential term in the center of mass equation, which means that it acts as free as a particle can act in the geometry we are working in. This means that, in this instance, the center of mass will behave like a standard particle in a box. As it turns out, the solutions to the particle in a box look just like the wavefunctions in **fig. 2.4.6**, so we

can see that, in the ground state, the most probable location for the center of mass is the halfway point.

Now we have the tools to say what the overall most probable configuration for the exciton on a line segment in the ground state is. The center of mass is most likely to be at 0.5, and the separation is also most likely to be 0.5, meaning that the most probable configuration is that one particle is at 0.25 while the other is at 0.75. (Note that these numerical values come with the assumption that the line segment is of length 1.) With that lengthy introduction to exciton quantum mechanics complete, let's tackle some more complicated geometries!

2.5 Circle

2.5.1 Introduction

Finding the energy eigenstates for an exciton on a circle ends up being an almost identical process to that for the line segment, but there are a few differences that are essential. The most significant difference is that we are now dealing with a compact geometry, meaning that there is no infinity in any direction. As we know very well, a circle wraps around on itself, and this warrants some major changes to the way we think about the problem.

For example, whereas on the line segment we had to impose that the wavefunction goes to 0 at some arbitrary distance L from the hole, we now need no such condition. Let's say that we fix the hole at the top of the circle, which we will call $\phi = 0$. All we impose is that, again, the electron is not allowed to recombine with the hole, but this time that condition applies on both sides of the hole. The resulting boundary conditions are essentially the same, namely just that the wavefunction goes to 0 at the endpoints of the coordinate ϕ, but conceptually this is a much cleaner and more intuitive physical situation.

Another consequence of the compactness of this geometry is that it is curved, which means that the coordinate is no longer equal to the distance. We measure location on the

circle by angle, ϕ, which is not the same as arclength, d (distance around the circle). To convert between the two we need to use the radius of the circle, r, such that $d = r\phi$. This formula for arclength will be used in the coulomb interaction between the two particles, which depends upon the distance between the two.

There is another way we could have chosen to consider the problem, which is that the particles exist in full 2-dimensional space and are merely constrained to the circle by some earthly means, meaning that the distance between the two is just the standard euclidean distance between them in the plane. As an illustration of the difference between these two options, consider the two particles being on opposite sides of the circle. In the setup that I am choosing to use, the distance between them is equal to half of the circumference of the circle, whereas in the other setup the distance is just the diameter of the circle. I prefer to treat the particles as if they really exist only in the 1-dimensional world of the circle rather than as if they are inexplicably bound to travel on one circle in a 2-dimensional plane because the latter implies some kind of external force which is keeping the particles on the circle, and that presents a different set of issues which I do not wish to address here.

The curvature of the geometry also means that we have to adjust the kinetic energy term of the hamiltonian, because we are dealing with angular coordinates. Specifically, it is useful to note the transformation of the laplacian from 2-D cartesian to polar coordinates:

$$\nabla^2 = \frac{\partial^2}{\partial r^2} + \frac{1}{r}\frac{\partial}{\partial r} + \frac{1}{r^2}\frac{\partial^2}{\partial \phi^2} \; , \tag{2.5.1}$$

In the case of the circle, however, we do not allow change in the radial direction, meaning we must only use the final term of Eq. (2.5.1). This leaves the kinetic energy term of the laplacian on the circle as $\frac{-\hbar^2}{m}\frac{1}{r^2}\frac{\partial^2}{\partial \phi^2}$.

The final consequence of the compact geometry is the one that I find to be the most fun to incorporate. In our standard equations for the interaction of two particles, be it by

gravity, electromagnetism, or some other force, we tacitly assume that the particles exist in flat (non-warped) space, and that the force between them acts along the straight line that connects them. The underlying reason for the straight line in this instance is that it is what is called a geodesic, which for our purposes basically just means the shortest path between two points. In flat space there is no ambiguity, because there is exactly one geodesic that connects two points (the straight line). In compact geometries (geometries that wrap back on themselves), however, things are a little more complicated because, on the circle for example, you can get from any point to any other point along two different geodesics: clockwise and counterclockwise. This is in contrast to flat space where there is only one direction (toward the other particle), because if you were to go the other direction (away from the other particle) you would just go out to infinity and never come back. In order to account for these two paths on the circle we will modify the potential term such that it is composed of two terms: the attraction along the clockwise path plus the attraction along the counterclockwise path.

Let's think about the reason for this new potential in qualitative terms. A first guess for how to model the potential might simply be that $V(d) = \frac{-q^2}{4\pi\epsilon_0 d}$, with $d \equiv$ arclength $= r\phi$. Considering the point at $\phi = \frac{\pi}{2}$, which in our set up corresponds to 3 o'clock on a clock, we get that $d = \frac{r\pi}{2}$, and therefore that the potential at that point is $\frac{-q^2}{2r\pi^2\epsilon_0}$. This seems perfectly reasonable until we try to extend this to the point $\phi = \frac{3\pi}{2}$ (9 o'clock on a clock). Here we would find that the potential is $\frac{-q^2}{6r\pi^2\epsilon_0}$, which is 3 times smaller than the potential at $\phi = \frac{\pi}{2}$, but we know that this is nonsense because by common sense laws of symmetry this point should be identical to $\phi = \frac{\pi}{2}$. We could try to define the potential in a piecewise manner, such that $d = r\phi$ if $\phi < \pi$ and $d = -r\phi$ if $\phi \geq \pi$, but piecewise functions are no fun and hard to work with mathematically. The solution that I opted for is to assume that every point is affected by the positive charge at the top of the circle both from the clockwise and counterclockwise directions. Thus, an electron at $\phi = \frac{\pi}{2}$ will actually be pulled less

strongly toward the top because it is simultaneously being pulled in the opposite direction by the same positive charge. One result of this is that at $\phi = \pi$ the positive charge at the top is pulling equally on both sides, and thus an electron at that location would feel no bias in either direction, which nicely matches the symmetry of the situation.

2.5.2 The Potential

Now I will briefly derive a simplified form of this potential.

$$
\begin{aligned}
V(\phi) &= -\frac{q^2}{4\pi\epsilon_0 r(2\pi - \phi)} - \frac{q^2}{4\pi\epsilon_0(r\phi)} \\
&= \frac{q^2}{4\pi\epsilon_0 r}\left(-\frac{1}{2\pi - \phi} - \frac{1}{\phi}\right) \\
&= \frac{q^2}{4\pi\epsilon_0 r}\left(-\frac{\phi}{\phi(2\pi - \phi)} - \frac{2\pi - \phi}{\phi(2\pi - \phi)}\right) \\
&= \frac{q^2}{4\pi\epsilon_0 r}\left(\frac{-2\pi}{\phi(2\pi - \phi)}\right) \\
&= \frac{-q^2}{2\epsilon_0 r\phi(2\pi - \phi)}
\end{aligned}
\tag{2.5.2}
$$

Eq. (2.5.2) is the formula that we will go on to use in order to find the answers we are looking for.

2.5.3 Mathematica

Below is an image analogous to **fig. 2.4.2**:

As you can see, the code has the identical structure to that of the line segment note-book from the previous section, the only differences being that the potential term in the definition of **eqn** is now the potential we just derived for the circle, and the kinetic term has an r^2 in the denominator. Moving right along to the next step, we will define our constants, obtain a numerical solution parameterized by energy, define the function called **endPoint**, and use **findAllRoots** to get precise values for the allowed energies.

The Equation

In[1]:= $\mathbf{eqn} = \dfrac{-\hbar^2}{m} \dfrac{\psi''[\phi]}{r^2} - \dfrac{q^2}{2\,\epsilon_0\,r\,\phi\,(2\,\pi - \phi)} == e_i\,\psi[\phi];$

Boundary Conditions

In[2]:= $\mathbf{boundary1} = \psi[\mathbf{holeRadius}] == 0;$
$\mathbf{boundary2} = \psi[2 * \mathbf{holeRadius}] == 0.01;$

Figure 2.5.1. The first two inputs in my Mathematica notebook for this problem.

Defining Constants

In[4]:= $\hbar = 1;\; m = 1;\; r = 2;\; q = 1;\; \epsilon_0 = 1;\; \mathbf{holeRadius} = \dfrac{2\,\pi}{10\,000.};$

Numeric Solution (parameterized by energy)

Solution

In[5]:= $\mathbf{nsol} = \mathbf{ParametricNDSolve}[\{\mathbf{eqn}, \mathbf{boundary1}, \mathbf{boundary2}\}, \psi, \{\phi, \mathbf{holeRadius}, 2\,\pi - \mathbf{holeRadius}\}, \{e_i\}]$

Out[5]= $\left\{\psi \to \mathtt{ParametricFunction}\left[\;\boxplus\;\overset{\text{Expression: }\psi}{\underset{\text{Parameters: }\{e_i\}}{}}\;\right]\right\}$

Finding Allowed Energies

In[6]:= $\mathbf{endPoint}[\mathit{energy\_}] := \psi[\mathit{energy}][2\,\pi - \mathbf{holeRadius}] /. \mathbf{nsol}$

In[12]:= $\mathbf{allowedEnergies} = \mathbf{findAllRoots}[\mathbf{endPoint}[t], \{t, 0, 20\}]$

Out[12]= {0.060949, 0.2501, 0.558495, 1.0004, 1.5563, 2.2509, 3.05432, 4.0016,
        5.05257, 6.2525, 7.55103, 9.0036, 10.5497, 12.2549, 14.0486, 16.0064, 18.0477}

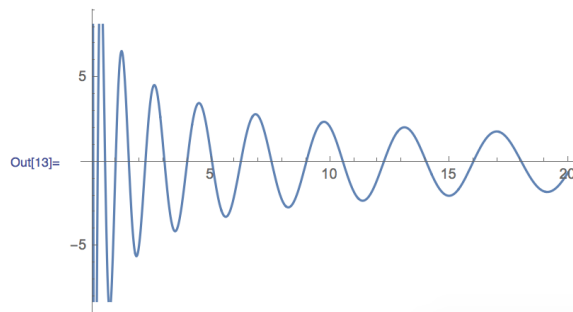In[13]:= $\mathbf{Plot}[\mathbf{endPoint}[a], \{a, 0, 20\}]$

Out[13]=

Figure 2.5.2. Defining the constants, obtaining a numerical solution parameterized by energy, defining the function called **endPoint**, and finding allowed energies.

Note that I have arbitrarily chosen the radius of the circle to be 2. With the allowed energies calculated we can normalize the wavefunctions using exactly the same code as we used in the previous section:

Finding Normalization Constants

```
In[14]:= normalizationConstants = √(1 / NIntegrate[Evaluate[(ψ[#][x] /. nsol)^2], {x, holeRadius, 2 π - holeRadius}]) & /@ allowedEnergies

Out[14]= {0.0189797, 0.0378721, 0.0564813, 0.0752034, 0.0936943, 0.112328, 0.130748,
         0.149321, 0.16769, 0.186218, 0.204546, 0.223038, 0.241333, 0.259794, 0.278062, 0.296496, 0.314741}
```

Figure 2.5.3. Finding the normalization constants for the energy eigenstates of an exciton on a circle.

Finally, using our newly found allowed energies and normalization constants we can plot the wavefunctions from 0 to 2π, and we see that they look essentially identical to the results for the line segment, except for the fact that domain is now 0 to 2π, and the range is correspondingly shrunken to maintain normalization. I have chosen to show the first three energy eigenstates as well as three higher energy states.

Plotting Wavefunctions

```
In[15]:= energiesAndNormalizationConstants = Transpose[{allowedEnergies, normalizationConstants}]

Out[15]= {{0.060949, 0.0189797}, {0.2501, 0.0378721}, {0.558495, 0.0564813}, {1.0004, 0.0752034}, {1.5563, 0.0936943},
         {2.2509, 0.112328}, {3.05432, 0.130748}, {4.0016, 0.149321}, {5.05257, 0.16769}, {6.2525, 0.186218}, {7.55103, 0.204546},
         {9.0036, 0.223038}, {10.5497, 0.241333}, {12.2549, 0.259794}, {14.0486, 0.278062}, {16.0064, 0.296496}, {18.0477, 0.314741}}

In[16]:= GraphicsGrid[Partition[Plot[Evaluate[#[[2]] (ψ[#[[1]]][ϕ] /. nsol)], {ϕ, holeRadius, 2 π - holeRadius}, Filling → Axis] & /@
         energiesAndNormalizationConstants, 3, 9]]
```
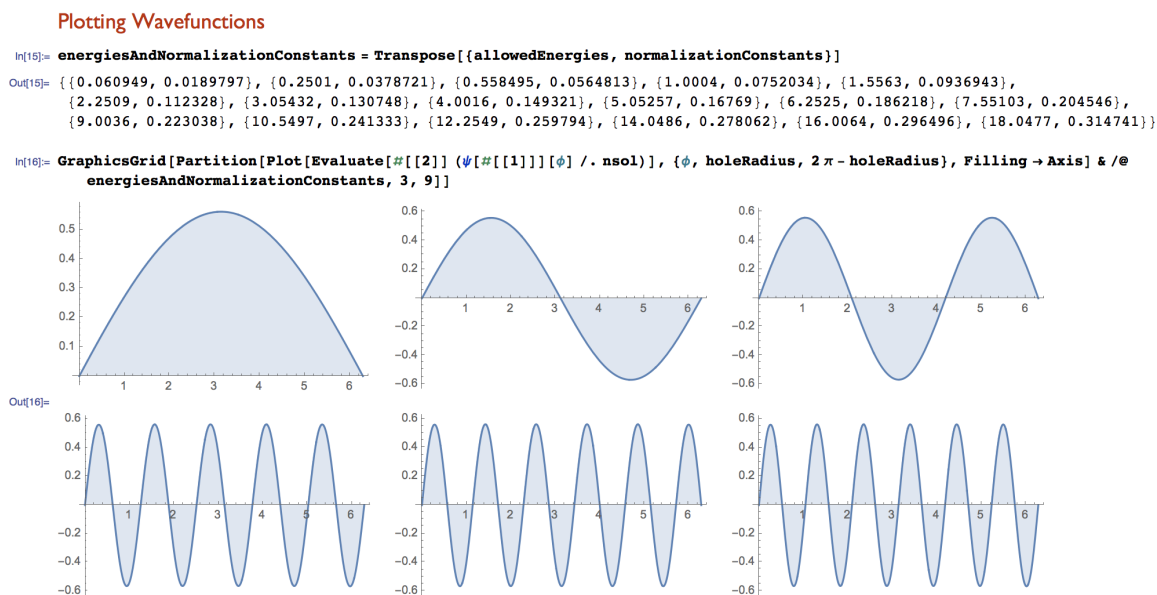

Figure 2.5.4. Plotting the wavefunctions of the first three allowed energies, and also three higher energy states, from 0 to 2π.

While these graphs do not seem all that interesting in light of the fact that they look no different from the results from the previous section, it is fun to realize that these represent wavefunctions which exist on a circle, and thus could be plotted accordingly. The graphs in **fig. 2.5.5** are the same as the ones shown in **fig. 2.5.4**, except that they are now being wrapped around the circle which they were calculated for.

```
In[30]:= GraphicsGrid[
    Partition[
      ParametricPlot[{Sin[u] (r + v #[[2]] (ψ[#[[1]]][u] /. nsol)), Cos[u] (r + v #[[2]] (ψ[#[[1]]][u] /. nsol))}, {u, holeRadius, 2 π - holeRadius},
        {v, 0, 1}, PlotPoints → 60, PlotStyle → Red] & /@ energiesAndNormalizationConstants, 3, 9]]
```
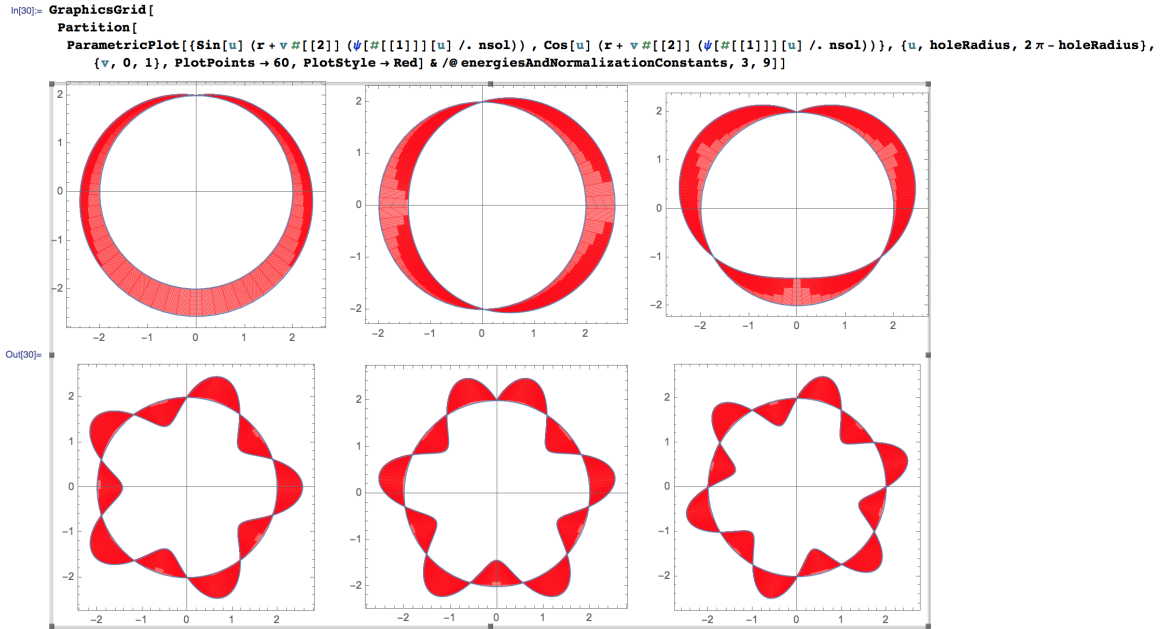
Figure 2.5.5. Superimposing the wavefunctions onto the circle that they were calculated for. The upper row consists of the first three energy eigenstates, and lower row consists of the 10th, 11th and 12th energy eigenstate wavefunctions.

I think it is only fair to point out that the images in **fig. 2.5.5** may be a little bit misleading. Again, these wavefunctions yield the probabilities for finding a certain *separation* between the two particles, so showing them graphed on a circle does not really convey the true meaning that they hold. For example, if a peak of one of these graphs occurs at $\phi = \frac{\pi}{2}$ it does not mean that any particle is likely to be found at the 3 o'clock position, but rather that the two particles are likely to be found with a separation between them of $\frac{\pi}{2}$ radians.

2.6 Sphere

2.6.1 Introduction

The final geometry that I will be exploring in this chapter is a sphere. It nicely showcases various complexities, such as multiple spatial dimensions and curved, compact geometry. The first thing to get done in the Mathematica notebook is to define the correct reduced mass Schrödinger equation. We begin with the most general form,

$$\frac{-\hbar^2}{m}\nabla^2\psi(\vec{r}) + V(\vec{r})\psi(\vec{r}) = E\psi(\vec{r}) \ . \tag{2.6.1}$$

The first thing we will do is to convert Eq. (2.6.1) to spherical coordinates, transforming the laplacian according to $\nabla^2 = \frac{1}{r^2}\frac{\partial}{\partial r}(r^2\frac{\partial}{\partial r}) + \frac{1}{r^2\sin\theta}\frac{\partial}{\partial\theta}(\sin\theta\frac{\partial}{\partial\theta}) + \frac{1}{r^2\sin^2\theta}\frac{\partial^2}{\partial\phi^2}$. Thus, Eq. (2.6.1) becomes,

$$\frac{-\hbar^2}{m}\Big(\frac{1}{r^2}\frac{\partial}{\partial r}(r^2\frac{\partial\psi(r,\theta,\phi)}{\partial r}) + \frac{1}{r^2\sin\theta}\frac{\partial}{\partial\theta}(\sin\theta\frac{\partial\psi(r,\theta,\phi)}{\partial\theta})$$
$$+ \frac{1}{r^2\sin^2\theta}\frac{\partial^2\psi(r,\theta,\phi)}{\partial\phi^2}\Big) + V(r,\theta,\phi)\psi(r,\theta,\phi) = E\psi(r,\theta,\phi) \tag{2.6.2}$$

Before attempting to simplify this rather long expression we will gratefully note that, similar to the circle, this geometry has no radial direction, meaning that we can reduce the complexity of the equation by eliminating the radial part of the laplacian:

$$\frac{-\hbar^2}{m}\Big(\frac{1}{r^2\sin\theta}\frac{\partial}{\partial\theta}(\sin\theta\frac{\partial\psi(\theta,\phi)}{\partial\theta}) + \frac{1}{r^2\sin^2\theta}\frac{\partial^2\psi(\theta,\phi)}{\partial\phi^2}\Big) + V(\theta,\phi)\psi(\theta,\phi) = E\psi(\theta,\phi) \tag{2.6.3}$$

There is one more thing we can do with this equation (2.6.3) as it stands, which is to expand out the first term:

$$\frac{1}{r^2 \sin\theta}\frac{\partial}{\partial\theta}(\sin\theta\frac{\partial\psi(\theta,\phi)}{\partial\theta}) = \frac{1}{r^2 \sin\theta}(\cos\theta\frac{\partial\psi(\theta,\phi)}{\partial\theta} + \sin\theta\frac{\partial^2\psi(\theta,\phi)}{\partial\theta^2})$$
$$= \frac{\cot\theta}{r^2}\frac{\partial\psi(\theta,\phi)}{\partial\theta} + \frac{1}{r^2}\frac{\partial^2\psi(\theta,\phi)}{\partial\theta^2} \tag{2.6.4}$$

Next we will make the standard guess that $\psi(\theta,\phi)$ can be written as a product of two functions each of only one variable: $\psi(\theta,\phi) = \Theta(\theta)\Phi(\phi)$. Using this assumption and substituting Eq. (2.6.4) into Eq. (2.6.3),

$$\frac{-\hbar^2}{m}(\frac{\Phi\cot\theta}{r^2}\frac{d\Theta}{\partial\theta} + \frac{\Phi}{r^2}\frac{d^2\Theta}{d\theta^2} + \frac{\Theta}{r^2\sin^2\theta}\frac{d^2\Phi}{d\phi^2}) + V(\theta,\phi)\Theta\Phi = E\Theta\Phi \tag{2.6.5}$$

Dividing by $\Theta\Phi$,

$$\frac{-\hbar^2}{m}(\frac{\cot\theta}{\Theta r^2}\frac{d\Theta}{d\theta} + \frac{1}{\Theta r^2}\frac{d^2\Theta}{d\theta^2} + \frac{1}{\Phi r^2\sin^2\theta}\frac{d^2\Phi}{d\phi^2}) + V(\theta,\phi) = E \tag{2.6.6}$$

Eq. (2.6.6) is as far as we can go before specifying the potential, $V(\theta,\phi)$.

2.6.2   The Potential

In discussing the potential on the circle, I mentioned the word geodesic, which I summarized as the shortest path between two points in a given geometry. On the circle the concept of a geodesic is somewhat trivial, since there are only ever two paths between two points and they are both geodesics (a geodesic actually has the more general definition of an "extremal" path, according to which both directions around a circle count as geodesics, not just the shorter one). On the sphere, however, a geodesic is a very important and non-trivial entity. As it turns out, the geodesics on a sphere are what we call "great circles", which are defined as the circles on the surface of a sphere whose radii are equal to that of the sphere. On earth, for example, the equator is a geodesic, while the very roughly circular border of the north pole is not.

In keeping with the idea that coulomb attraction between two particles occurs along the geodesic connecting them, we will assume that the hole and the electron are attracted along a great circle. In the reduced mass set up, we will fix the hole at the north pole of the sphere ($\theta = 0$). This location has a great advantage, which is that is causes the potential to depend only on θ. This is because no matter where you are on the sphere, the shortest path to the north pole is straight up, keeping ϕ constant and only varying θ, and thus the distance between any point and the north pole depends only on θ. Since the potential depends only on the distance, it must depend only on θ.

The sphere, similar to the circle, is also a compact geometry, and so we will employ the same logic as we did with the circle, defining the potential at each point as the sum of the potentials from each direction. The result of all of this is that we are able to completely reuse the potential we derived for the circle, except that where we used ϕ as the angle down from the top of the circle, we now are using θ as the angle down from the north pole of the sphere.

$$V(\theta) = \frac{-q^2}{2\epsilon_0 r \theta (2\pi - \theta)} \tag{2.6.7}$$

2.6.3 Separating ϕ Dependence

We can now plug this into Eq. (2.6.6) in order to continue:

$$\frac{-\hbar^2}{m}\left(\frac{\cot\theta}{\Theta r^2}\frac{d\Theta}{d\theta} + \frac{1}{\Theta r^2}\frac{d^2\Theta}{d\theta^2} + \frac{1}{\Phi r^2 \sin^2\theta}\frac{d^2\Phi}{d\phi^2}\right) - \frac{q^2}{2\epsilon_0 r \theta(2\pi - \theta)} = E$$

Multiplying by $-\frac{mr^2}{\hbar^2}\sin^2\theta$,

$$\frac{\sin\theta\cos\theta}{\Theta}\frac{d\Theta}{d\theta} + \frac{\sin^2\theta}{\Theta}\frac{d^2\Theta}{d\theta^2} + \frac{1}{\Phi}\frac{d^2\Phi}{d\phi^2} + \frac{mr^2\sin^2\theta}{\hbar^2}\left(\frac{q^2}{2\epsilon_0 r\theta(2\pi - \theta)} + E\right) = 0$$

and finally separating the θ and ϕ dependent terms,

$$\frac{\sin\theta\cos\theta}{\Theta}\frac{d\Theta}{d\theta} + \frac{\sin^2\theta}{\Theta}\frac{d^2\Theta}{d\theta^2} + \frac{mr^2\sin^2\theta}{\hbar^2}\left(\frac{q^2}{2\epsilon_0 r\theta(2\pi-\theta)} + E\right) = -\frac{1}{\Phi}\frac{d^2\Phi}{d\phi^2}$$

We now have two expressions which depended on different independent variables and yet are always equal to each other, meaning that they must both be equal to some constant. We will call this constant l^2:

$$\frac{\sin\theta\cos\theta}{\Theta}\frac{d\Theta}{d\theta} + \frac{\sin^2\theta}{\Theta}\frac{d^2\Theta}{d\theta^2} + \frac{mr^2\sin^2\theta}{\hbar^2}\left(\frac{q^2}{2\epsilon_0 r\theta(2\pi-\theta)} + E\right) = l^2 \tag{2.6.8}$$

$$-\frac{1}{\Phi}\frac{d^2\Phi}{d\phi^2} = l^2 \tag{2.6.9}$$

Eq. (2.6.9) is a differential equation we are very used to and can solve with ease:

$$\Phi(\phi) = Ae^{il\phi} + Be^{-il\phi} \ . \tag{2.6.10}$$

One of the first things we must do with this solution is to account for the redundancy of points corresponding to values of ϕ, by imposing that $\Phi(\phi) = \Phi(\phi + 2\pi)$. Substituting in our solution we find that,

$$Ae^{il\phi} + Be^{-il\phi} = Ae^{il(\phi+2\pi)} + Be^{-il(\phi+2\pi)}$$

$$Ae^{il\phi} + Be^{-il\phi} = Ae^{il\phi}e^{2il\pi} + Be^{-il\phi}e^{-2il\pi}$$

$$e^{2il\pi} = 1 \qquad \text{and} \qquad e^{-2il\pi} = 1$$

$$l \in \mathbb{Z}$$

We have discovered that our new separation constant, l, cannot just be any old number, but must be an integer.

2.6.4 Mathematica

The θ equation (2.6.8) is the heart of the problem and this is where we will need to turn to Mathematica for help. In order to better avoid potential numerical issues I chose to multiply Eq. (2.6.8) through by $\Theta(\theta)$, yielding,

$$\sin\theta\cos\theta\frac{d\Theta}{d\theta} + \sin^2\theta\frac{d^2\Theta}{d\theta^2} + \frac{mr^2\sin^2\theta}{\hbar^2}\left(\frac{q^2}{2\epsilon_0 r\theta(2\pi-\theta)} + E\right)\Theta = l^2\Theta \qquad (2.6.11)$$

As always we start out by giving Mathematica the equation we are interested in, as well as our standard boundary conditions.

The Equation

```
In[1]:= polarEqn = Sin[θ] Cos[θ] θ'[θ] + Sin[θ]² θ''[θ] + (m r² Sin[θ]²)/ℏ² (q²/(2 ε₀ r θ (2 π - θ)) + energy) θ[θ] == l² θ[θ];
```

Boundary Conditions

```
In[2]:= boundary1 = θ[holeRadius] == 0;
        boundary2 = θ[2 * holeRadius] == 0.01;
```

Figure 2.6.1. The standard first two inputs in my Mathematica notebook.

One very important distinction between this calculation and the calculation on the circle is that on the circle there is no such thing as angular momentum, because whether a particle is moving clockwise or counterclockwise, it is always moving directly toward any other point, and there is no way to rotate around a point if you are forced to move directly toward it. On the sphere, however, there are two dimensions, meaning that the exciton is free to have angular momentum. In our equations, it turns out that the new separation constant is related to the angular momentum, which is why I decided to call it l. **Fig. 2.6.2** shows the usual proceedings in Mathematica. We determined above that l is restricted to take on only integer values, and so for this calculation we will give it a value of 1.

Defining Constants

In[4]:= $\hbar = 1$; m = 1; r = 1; l = 1; q = 1; ϵ_0 = 1; holeRadius = $\dfrac{\pi}{10\,000}$;

Numeric Solution (parameterized by energy)

Solution

In[5]:= nsol = ParametricNDSolve[{polarEqn, boundary1, boundary2}, Θ, {Θ, holeRadius, π - holeRadius}, {energy}]

Out[5]= $\Big\{\Theta \to \text{ParametricFunction}\Big[\boxed{\,\boxplus\,\, \text{\scriptsize Expression: }\Theta \atop \text{\scriptsize Parameters: }\{\text{energy}\}}\,\Big]\Big\}$

Finding Allowed Energies

In[6]:= endPoint[testEnergy_] := Θ[testEnergy][π - holeRadius] /. nsol

In[12]:= allowedEnergies = findAllRoots[endPoint[t], {t, 0, 400}]

Out[12]= {1.92489, 5.91274, 11.9045, 19.8983, 29.8934, 41.8892, 55.8856, 71.8825,
      89.8798, 109.877, 131.875, 155.873, 181.871, 209.869, 239.868, 271.866, 305.865, 341.864, 379.862}

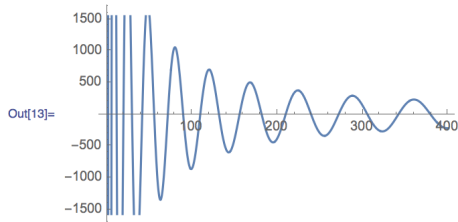In[13]:= Plot[endPoint[a], {a, 0, 400}]

Out[13]=

Figure 2.6.2. Defining constants including choosing l to be equal to 1, obtaining the numerical solution parameterized by energy and determining the allowed energies.

Once we have the allowed energies we are able to calculate the normalization constants (**fig. 2.6.3**).

```
In[14]:= normalizationConstants = Sqrt[ 1 / NIntegrate[Evaluate[(Θ[#][x] /. nsol)^2], {x, holeRadius, π - holeRadius}] ] & /@ allowedEnergies

Out[14]= {0.0385919, 0.0765554, 0.119608, 0.167117, 0.218572, 0.273587, 0.331862, 0.393158,
         0.457276, 0.524053, 0.593347, 0.665036, 0.739012, 0.815183, 0.893462, 0.973774, 1.05605, 1.14023, 1.22625}
```

Figure 2.6.3. Using the same code as in the previous sections to calculate the normalization constants.

We now have everything we need to plot the polar part of the wavefunction. The graphs in **figs. 2.6.4** and **2.6.5** show very distinct behavior of the energy eigenstates of an exciton on a sphere. One trend that becomes easier to see in the higher energy states is the shortening of the peaks in the middle. Since these graphs represent the separation between the two particles, this shortening corresponds to a lowered probability of finding the particles spaced moderately, and a heightened probability of finding them either very close together or nearly at opposite poles of the sphere.
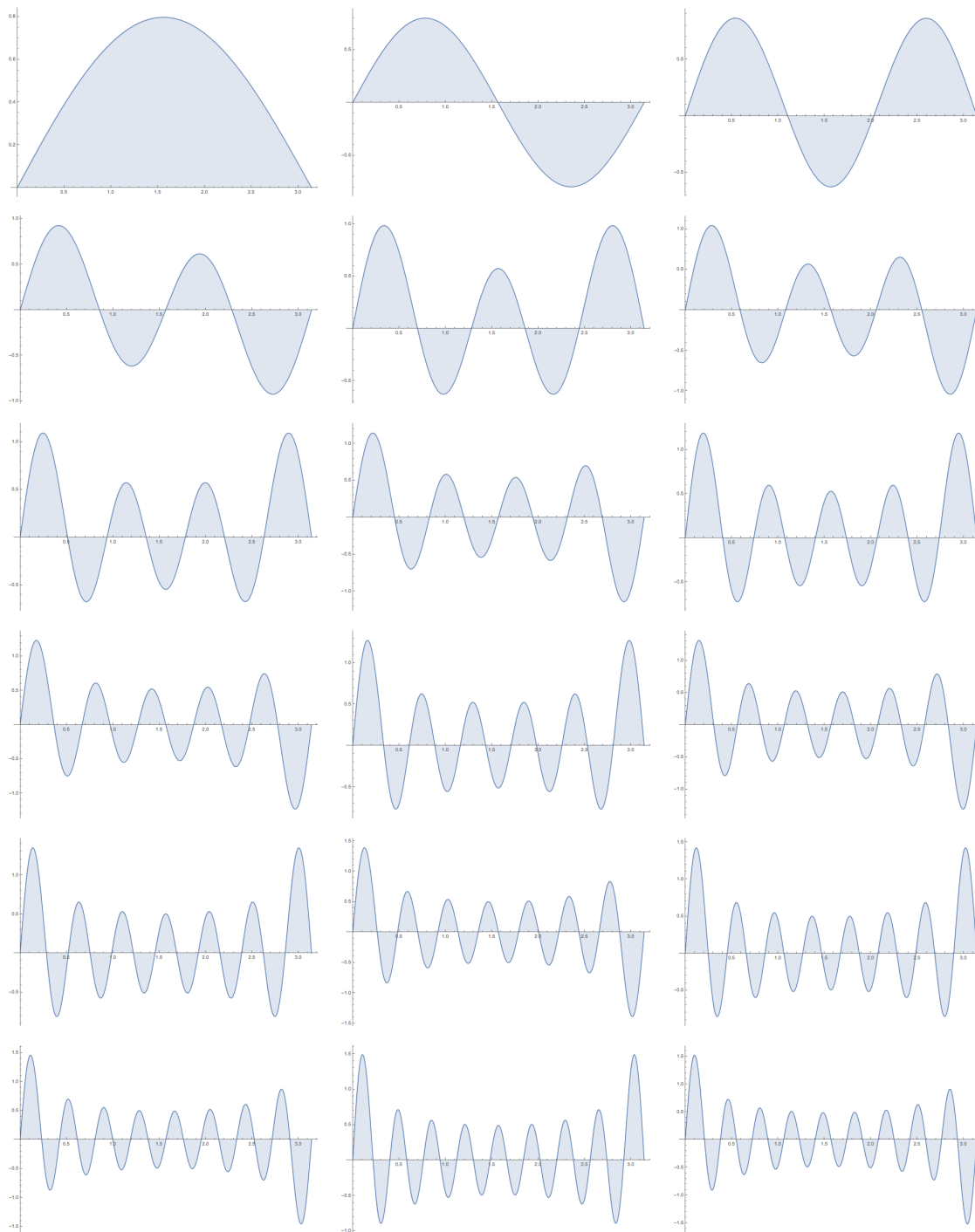
Figure 2.6.4. The polar parts of the first eighteen energy eigenstates of an exciton on a sphere.

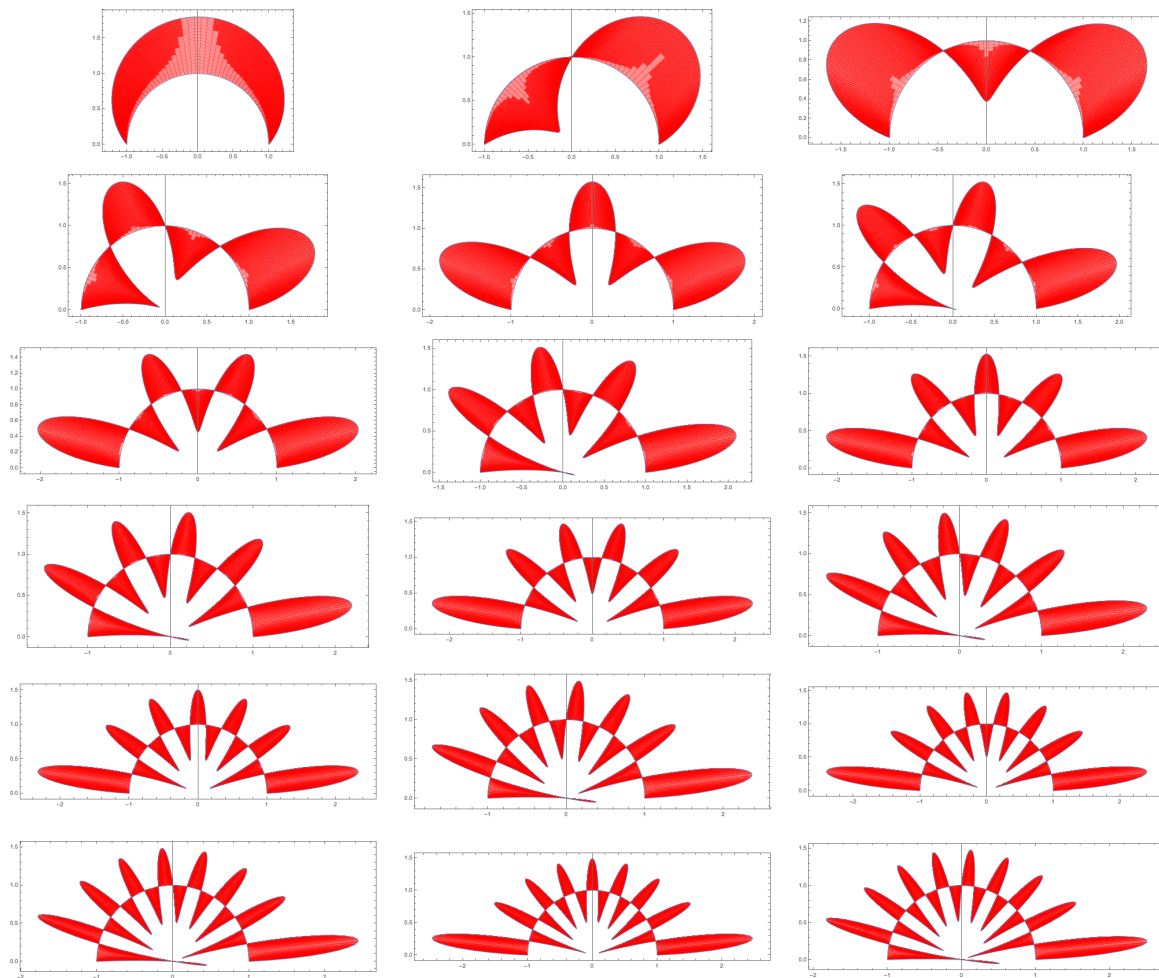Figure 2.6.5. The polar parts of the first eighteen energy eigenstates of an exciton on a sphere, plotted around the border of a semi-circle.
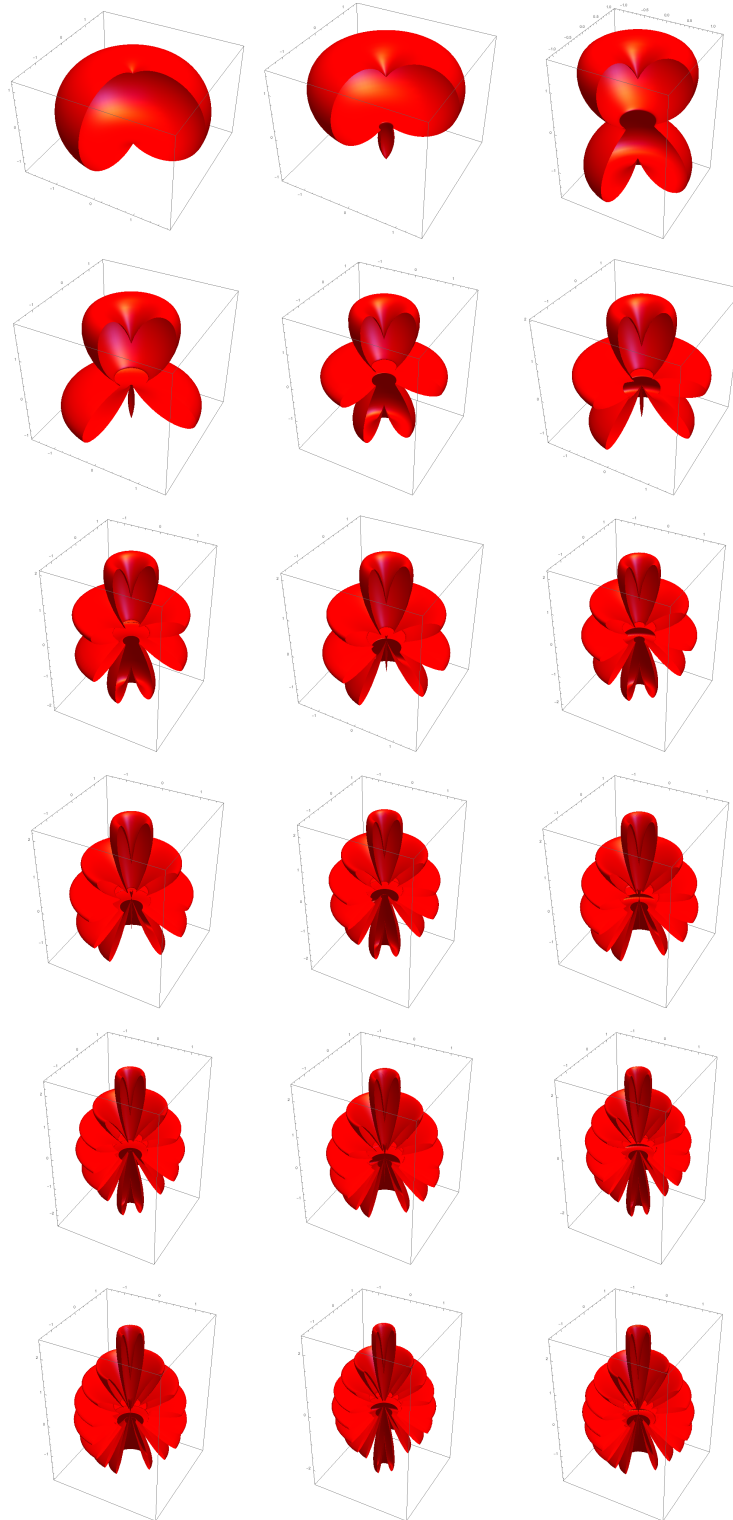
Figure 2.6.6. The ϕ dependence only adds a complex phase to the wavefunction, so the polar plots from **fig. 2.6.5** can just be revolved around the z-axis in order to get a visualization of what the wavefunction would actually look like on the sphere.

3
Experimentation with Solar Panels

3.1 Introduction

The quantum mechanical theory of this project is the matured outgrowth of a year of
engaging with solar panels in every way that I could, which included physically tinkering
with them. This final section will be devoted to describing the pieces of that journey.
It will include descriptions of the problems I encountered, the methods I developed for
solving them, the accomplishments I made as well as the failures, and the future work that
together they inform. I hope that future students will be able to make use of this chapter
for their own work, and be guided to avoid some of the pitfalls which I encountered.

3.2 The First Solar Cells

The first big step I took toward getting engaged with solar panels was simply to go to
P&T Surplus, a store in Kingston, NY that sells everything from springs to circuit boards
to, yes, solar panels. The first time I went to them I found a stash of miniature solar cells,
about 2 inches by 3 inches each, which they were selling for $2.00 a piece. They were very

simply constructed: just a rectangle of photovoltaic material encased in resin with two metal leads on the back for soldering onto (see **figs. 3.2.1** and **3.2.2**).

Figure 3.2.1. A front view of one of the miniature solar cells I first purchased from P&T Surplus. The dark blue strips are the photovoltaic material, and are shielded by resin.

Figure 3.2.2. The back of the cell in **fig 3.2.1**. The two pill shaped metal contacts half way down on the left and right are for electrically installing the cell, most commonly by soldering.

Each cell was made of five thin strips of photovoltaic material connected together in series underneath the resin exterior. Each strip produced a little over 0.5V, meaning that the cell had a total open circuit voltage of about 3V. The cells had a short circuit current in the range of 0.1A. I purchased all of the ones they had, which totaled 42, and I started to combine them in different configurations. This endeavor led me to begin doing a lot of work with the 3D printer, which, while not directly related to photovoltaic physics, was an invaluable experience with regard to gaining competence with the burgeoning craft of 3D printing, which is expected to play a large role in technological manufacturing in the coming decades.

3.3   3D Printing

3.3.1   Basics

The 3D printer that I used was the MakerBot Replicator 2. The maximum volume that it can produce is about 11" x 6" x 6", and it uses PLA filament on a non-heated build plate. In order to generate the 3D models I used a combination of software. I designed

the models themselves using a program called 123D Design, which is made by Autodesk. Then I used the MakerBot Desktop software to convert those models to the correct file format for feeding to the printer.

3.3.2 Software

123D Design has a clean layout and fairly intuitive controls. I was able to design all of my models using simple geometric shapes along with the basic functions of subtracting one shape from another and merging two shapes together. An important thing to note is that the Replicator 2 can only print files with the extension .x3g. Other file types will not even be recognized by the printer if they are present on the SD card. In order to package your 3D model in this format you have to pass through the intermediate file format .stl. The first file format is .123dx, which is the extension for 123D Design project files, and are what you are actively editing when you design your model in 123D Design. When you have finished designing your model, look for the option "Export as 3D..." and then choose STL. This will convert your model to the .stl file format.

Once you have your .stl file you have gotten as much as 123D Design can give you, and it is time to open up the MakerBot Desktop software. When using this software for the first time there are certain steps to configure the MakerBot software for your specific model of 3D printer. Assuming you have done this configuration, you need only to select "Add File" and then choose the .stl file that you just exported from 123D Design. Finally, click "Export Print File" and select the .x3g format and it will produce the correct file for you. Now you can put that .x3g file onto an SD card and stick that card into the slot on the printer and you are ready to print your model.

3.3.3 Printing Tips

- It is crucial to measure everything very carefully while you are modeling, because if after waiting hours for the printer to finish you discover that the object is not quite

the right shape or size you will be very frustrated. I have found that if I am designing a groove or space into which some other physical object must fit, it is necessary to tack on a millimeter or two to any dimension larger than about 25 millimeters.

- Rafts can help preserve the intended shape of your object, but can be very difficult to separate from your object, and so should be used discriminately. A raft is a textured bed which is printed first, onto which the actual model is printed, and is intended to aid with the unavoidable warping that can result as the plastic cools down. The raft is textured in order to make it easier to pry away from the model. I have found that while it is a helpful and largely harmless addition to smaller more detailed objects, it is very unhelpful when printing something wide and flat, such as I was doing. In this case it was sometimes nearly impossible to remove the raft from the model, and many bloodied knuckles were produced in the attempts to do so. The MakerBot Desktop software allows for a lot of customized settings on the .x3g file that it produces, one of which is the option to include a raft.

- It is common practice to cover the acrylic build plate with masking tape, which can make it easier to remove the printed object once it is complete. It is crucial, however, that the build plate remain perfectly flat, and one time I experienced that a large print job got ruined because the tape began to peel, dramatically changing the landscape of the build plate and causing the model to become unusably warped. If you do choose to not use tape though, be warned that going commando with the build plate will expose it severe scratching and indentation over time. The good news is that the plate is very replaceable.

3.3.4   Repair

At one point the 3D printer began to very severely malfunction, rendering it completely incapable of producing even a small model. Every time I tried to print something it

would come out as a pile of plastic spaghetti. It turns out that the cables which convey information to the x-axis motor were damaged and were intermittently failing to deliver the correct signals. The result was that the print head would momentarily stutter in the x direction, leaving the printer with incorrect information about the x position of the nozzle. Thus, whenever a stutter occurred the entire print job would be shifted in the x direction by some amount, and from that point forward every layer would be placed incorrectly.

The cause of this cable malfunction was that the cable comes wrapped around some pegs and is attached to the print head which is always moving, eventually causing internal damage to the section which is being repeatedly bent back and forth. The solution was to cut out the section of cable that was damaged, and then solder the rest of the cable back together. I did this, and the printer has worked perfectly since.

3.3.5   The First Board

The first thing I printed was a long rectangular board which was specially designed to hold six of my miniature solar cells. You can see the final product in **fig. 3.3.1**. The six cells are soldered in series. The reason I decided to create this object in the first place was that it was difficult bordering on impossible to solder six of them together without very quickly breaking the connection at one spot and having to fix it. Putting the cells on a board like this, with specially placed holes for soldering them together, created a very stable electrical system that still supplies power 7 months later.
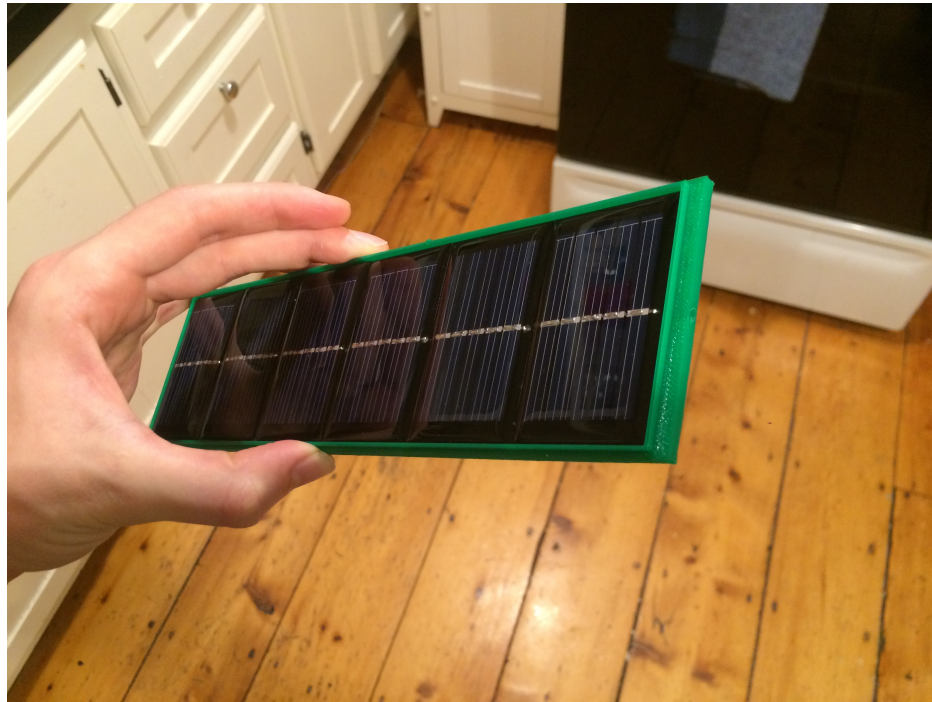
Figure 3.3.1. The first board that I printed to hold solar cells. They are soldered in series, each contributing ∼3V, totaling ∼18V.
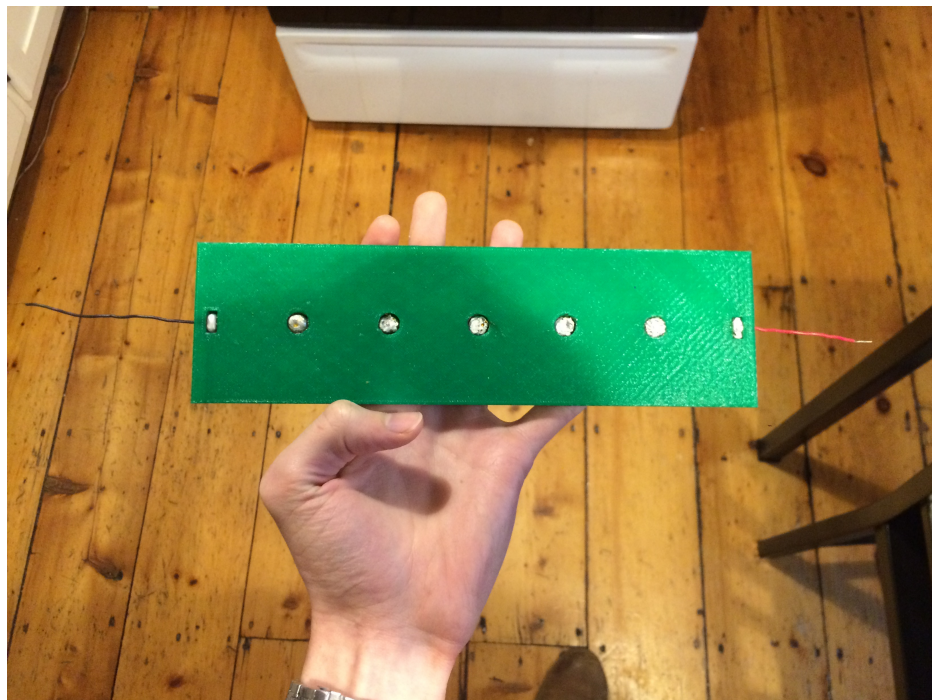

Figure 3.3.2. The back of the board in **fig 3.3.1**. The holes are filled with solder, which is serving to establish the electrical connection between the cells.

3.4   Charging an iPhone

Once I had finished producing the first solar panel board I became inspired to achieve a functional solar powered iPhone charger. To begin with, it was necessary to establish how many cells I would need to make this a reality, as well as exactly how they would need to be wired together. A standard USB plug operates at around 5 volts, meaning that the 18 volts that I was getting out of my 6x1 solar board was far too much. Since each cell is about 3V, only two need to be in series to achieve a high enough voltage for a USB plug, but since the voltage will vary depending on the intensity of the light, I chose to go with strings of three cells in series in order to ensure a high enough voltage. If I were to just put three cells in series though, the voltage would be high enough to charge an iPhone but the current would not be. In order to increase the current it is necessary to connect cells in parallel. I chose to connect twelve three-cell-strings together in parallel. Mostly for aesthetic reasons I decided to go for a slightly trickier wiring job in order to spatially arrange the 36 cells in a 6x6 grid, pictured in **fig. 3.4.1**.

Producing the structure for this 36-cell array with the 3D printer took some time, mostly because the severe malfunction that I mentioned earlier occurred during this time. The build plate of the Replicator 2 is not nearly big enough to print something of the necessary size, so I had to split up the design into four pieces. Each piece is identical, with an extensive network of holes and tunnels allowing for wiring, soldering, and the binding of the four pieces into one coherent panel. Eventually, I was able to produce four copies of the designed piece, as well as the auxiliary connector pieces which snapped into slots on the four boards, serving to hold them together. After doing all of the wiring and soldering, I finally had a solar panel with a sizable power output.

Alas, simply having the correct current and voltage coming out of a solar panel is not enough. An iPhone charging brick plugs into a wall socket, which supplies it with alter-

nating current. Solar panels always put out direct current. As it turns out, the iPhone charging brick converts the AC to DC because the USB plug requires it. Thus, for my purposes, the alternating current is a superfluous step in the process which could theoretically be skipped. Though it took some time for me to figure out, there is luckily a common device which is specifically designed to eliminate the need for AC, converting DC directly to an acceptable USB input. That device is a car charger. This is because the power that is being supplied to the cigarette lighter in a car is coming from the 12V DC car battery. All that I needed to do was to alligator clip the wires from my array of solar cells to the leads on the car charger, and then plug my iPhone into the car charger as I would do in my car.

3.4.1   The Caveat

While I was indeed able to cause my phone to light up and vibrate in announcement of that it is technically charging, I never witnessed the battery percentage actually go up, and in fact it tended to go down even while plugged in. I believe that the solar array was at least causing the battery to drain more slowly than it otherwise would have done, though I have not yet performed any rigorous experiments to test that hypothesis. Given the bulky and somewhat rickety nature of the array, as well as the fact that it does not produce enough power to overcome the inevitable drain that simply comes from the phone being on, I would not say that this is a workable product design yet. It was very fun to design and create though, and could potentially be useful in very specific circumstances.

Figure 3.4.1. The 6x6 solar array that I created for the purpose of charging an iPhone. Each set of three cells is connected in series, and all twelve sets of three are connected in parallel. A complicated design for the tunnels which run throughout the board was required to allow for this wiring to be done correctly. The array comes in four separate pieces, because the whole thing was far too large to be 3D printed all at once.

3.5   Solar Tracking

3.5.1   Introduction

After accomplishing my goal of a solar powered iPhone charger, I decided to switch gears—

I wanted to create a solar tracking system that operated without the need for a computer

chip or an electrical source. By the due date of this project I was not able to realize the

system which I designed, though I did come very close. With a few more weeks I believe

I would be able to present a functioning model.

3.5.2   What is a Solar Tracker?

A solar tracker is a device (usually a solar panel) that in some way changes its orientation

throughout the course of a day to reflect the position of the sun. The reason this is a very

relevant and important idea in photovoltaics is that the power absorbed by a solar panel,

P, is proportional to the power coming in from the sun, S, multiplied by the cosine of the

angle of the sun relative to the panel, θ.

$$P \propto S \cos \theta \qquad (3.5.1)$$

We define θ as the angle between the incoming beams of sunlight and the normal vector

of the panel, such that when the sunlight is coming in exactly perpendicularly to the

panel $\theta = 0$ and $\cos \theta = 1$, such that according to Eq. (3.5.1) the panel is absorbing a

maximal amount of the power coming from the sun. Bear in mind that this says nothing

about the internal efficiency of the solar cell. Just because a solar cell is absorbing 100%

of the energy coming from the sun does not mean that it is successfully converting all of

it to electricity (in fact it usually only converts between 10% and 20% of it to electricity).

However, it is still better to be getting 20% of *all* of the sun's power than to be getting

20% of just some fraction of it. Thus, a solar tracker in this instance might be a solar

panel on a sophisticated computerized gimbal which knows the time of day, the date, and

its own position on earth, and therefore knows the exact position of the sun and rotates the panel accordingly such that it remains exactly perpendicular to the sunlight all day long.

3.5.3 My Idea

The idea that I was pursuing was to create a solar tracker that relies neither on a digital knowledge of the sun's location nor on an external source of electricity to follow the sun. I was inspired by a product called GravityLight, which produces constant low-level light for about half an hour by extracting the gravitational potential energy of a weight which is slowly allowed to fall to the floor. The energy, of course, comes from the weight which had to be lifted, in this case by a person. I wanted to create a similar system, which requires a one-time initial input of gravitational potential energy by a human, after which the mechanism will do all of the work.

3.5.4 My Design

I designed and created an almost functional example of this type of solar tracker. My design relies on the operating principle of an hourglass, in which the gravitational potential energy is stored by sand which seeps at a constant rate through a small hole.

I will describe my mechanism in words here, but to visualize it **fig. 3.5.2** should be consulted. To begin with, a few rows of solar cells are suspended by pegs which rest in ball bearings, thereby allowing the rows of cells to rotate freely. Small round gears controls the rotation of each row independently. Each row of solar cells has its own gear. The round gears of each row sit at the same height as each other, and are caused to rotate by a straight gear passing underneath them. A straight gear is just a rectangular stick with gear teeth along one side. The teeth on the straight gear have the same spacing as the teeth on the round gears which control the rotation of the solar cells, so when the long

straight gear passes underneath the round gears they are all turned in synchrony, and therefore the linear motion of the straight gear has been turned into rotational motion.

The challenge, then, is to cause the straight gear to be pulled incredibly slowly under the round gears. The round gears are supposed to only rotate by about 160° over the course of the whole day, meaning that, with the parts that I was using, the straight gear has to move with a velocity of about 0.002 m/s. Ignoring for a moment how slow this is, it also has to be constant because the sun advances at an approximately constant rate, and this is not the effect that a hanging weight produces. The *acceleration* due to gravity is constant, meaning the velocity due to gravity is linearly increasing. I did, however, come up with what believe would be a solution, were it not for some minor technical difficulties.

One end of the straight gear has a string attached to it, and this string is what needs to be pulled about 1 inch over the course of 12 hours. The way I thought of to enable this is by the use of springs. Springs, in the limit where they are not being compressed very much, obey Hooke's law, which is $F = kx$. The remarkable thing about this is that force and position are linearly related. If I place 1 pound on a spring it will compress by some distance, and if I place 2 pounds on it it will compress twice that distance. Therefore, I can get a platform which is resting on springs to linearly sink lower and lower if I can find a way to linearly increase the force with which I am pushing down on it. Conveniently, force is also linearly related to mass. My solution, therefore, is to have sand falling at just the right rate such that the spring-supported platform on which it is accumulating will sink by about 1 inch over the course of the 12 hour day. The string which is connected at one end to the straight gear, will be affixed at the other end to the surface of this sinking platform, thereby pulling the straight gear just the right amount and causing the rows of solar panels to rotate with the sun.
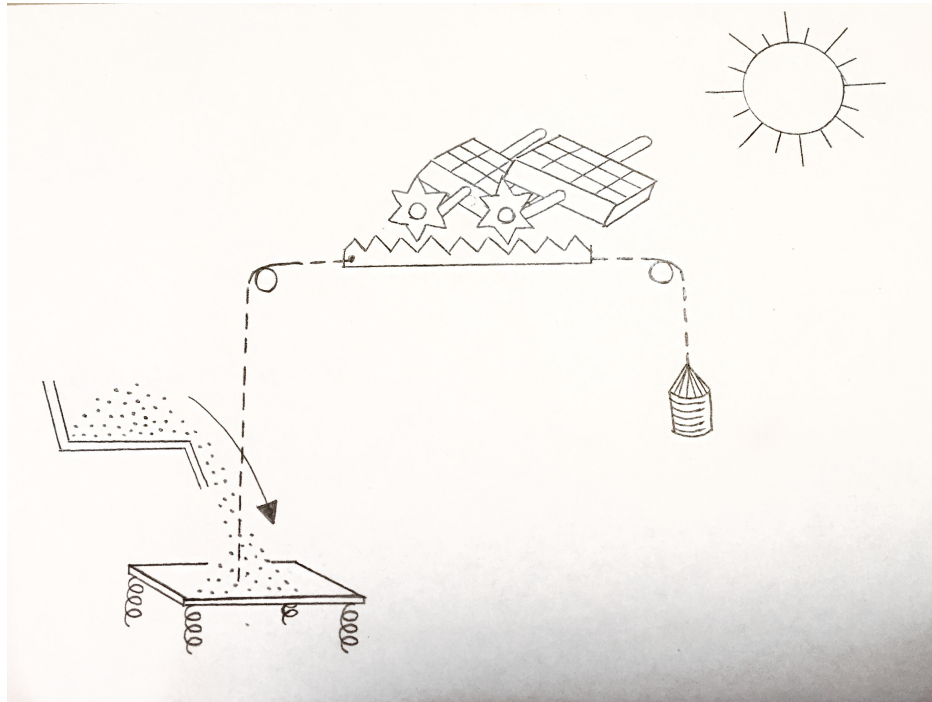
Figure 3.5.1. An illustration of the solar tracking system which I designed. The salt is poured onto the spring-supported platform which depresses at a constant rate, pulling the straight gear underneath the round gears which rotate at that constant rate in order to track the sun.

3.5.5 Caveats

There are some things to consider about this design.

- The springs underneath the platform should be much longer than 1 inch, because Hooke's law breaks down as the compression approaches the maximum possible compression of the spring.

- The sand that I purchased from Home Depot had too much moisture in it to fall through a tiny hole, and instead just clumped together. Salt works much better in this capacity.

- The straight gear has to be resting on a very smooth surface, because when trying to pull it so slowly static friction can completely prevent any motion.

- It is a good idea to hang a counterweight from the other end of the straight gear which is very slightly lighter than the weight of the platform, such that the springs are entirely uncompressed at $t = 0$ with no salt yet on the platform. This will isolate the influence of the weight of the salt on the position of the straight gear.

- This system does not account for variations in the sun's path across the sky at different times of year, and so is only a very primitive implementation of solar tracking.

3.5.6 Data

I constructed the apparatus described above, but due to the various caveats, especially the friction on the straight gear, it did not operate successfully. Nonetheless, I took some manual sample data of the power output of a tracking vs. non-tracking panel to illustrate the effect this type of system can have on the efficiency. **Fig. 3.5.2** shows the advantage of using a solar tracking system over the course of a day. As we would expect, this advantage becomes more dramatic as the sun moves to more oblique angles.
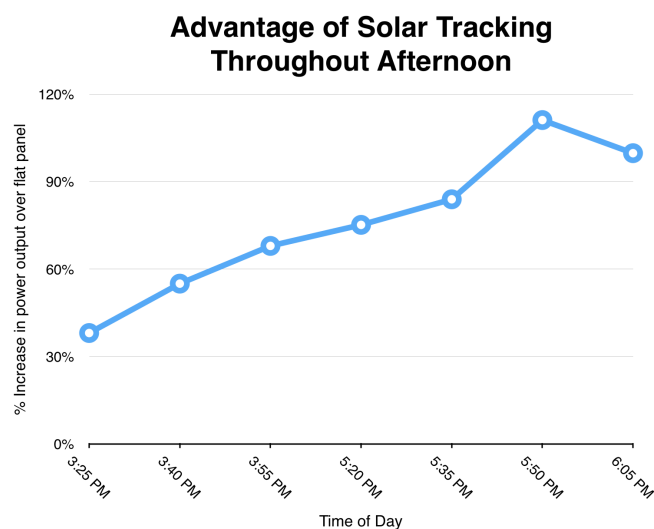

Figure 3.5.2. This graph plots the percent increase in power output of the panel when angled toward the sun over the same panel when flat. This solar tracking advantage increases as the sun gets lower in the sky, as we would expect. The dip at the end can be explained by a shadow being cast on the panel when rotated at that time.

3.6   Future Work

There are many avenues for continued work in the various areas discussed in this chapter. I would very much like to make another panel for charging an iPhone using higher quality solar cells, such that it can be not only smaller but also effective. From a product design point of view, the car charger should be incorporated more seamlessly into the design, eliminating the need for alligator clips. I would also potentially like to make the panel foldable so that it is portable and can be used by the average smartphone user. The societal benefits of this type of sustainable product would lie much more with bringing solar technology into the public eye than with the actual energy savings.

The solar tracking system has a great number of ways it could be improved. Firstly, I would, of course, like to get a prototype that actually works autonomously once it is loaded with the salt. Beyond that, the hole apparatus could be propped up on one side in order to angle it more directly toward the sun when it is being used closer to the poles of earth. Moreover, the angle at which it should be propped will have to change throughout the year, and this could potentially also be automated. If this system were to be actually put into use it would surely be for a home solar installation, in which case a revised version which is more compatible with a roof should be designed. If there were some way for the resident of the house to trigger the tracking from inside the house every morning, rather than having to climb up on the roof, it would of course make it much more viable.

Bibliography

[1] David Griffiths, *Introduction to Quantum Mechanics, 2nd Edition*, Pearson Education, Inc, Upper Saddle River, NJ 07458, 2003.

[2] Jenny Nelson, *The Physics of Solar Cells*, Imperial College Press, 57 Shelton Street, Covent Garden, London, WC2H 9HE, 2005.