Numerical Solution to the Schrödinger Equation for the Tunneling of Nitrogen in the Ammonia Molecule

Rustem Bilyalov

Abstract

In the ammonia molecule, the nitrogen atom has two equilibrium positions and oscillates back and forth between them. A potential energy approximation looks like a harmonic oscillator potential with a barrier in the middle. This research determined the effects of moving the barrier and changing its width on the frequency of the nitrogen's oscillation. An algorithm to numerically solve the Schrödinger equation was developed based on what the differential equation means conceptually. In order to test the algorithm, it was used to approximate the wave function for two classic potentials and the results matched the known solutions. The potential for ammonia was then solved for various barrier positions and barrier widths. A demonstration of the result was then developed. The frequency of oscillations of a horizontal string with a barrier consisting of two vertical strings that restrict movement on either side was measured for various barrier positions. The frequencies were measured with an accuracy of 90%. The results agreed with the numerical calculations. It was found that moving the barrier away from the middle results in the decrease of the frequency of nitrogen's oscillation and widening the barrier increases the frequency.

Contents

Introduction

On December 14th, 1900, the field of quantum mechanics was introduced to the world when Max Planck presented his paper, "On the Theory of the Energy Distribution Law of the Normal Spectrum," at a meeting of the German Physical Society [2]. In this paper he discussed the differences between experimental results and theoretical expectations of the blackbody thermal radiation problem. He found that the only way to explain those differences was to assume that for particles, energy is quantized. He also found that energy had to be proportional to frequency. This proportionality constant, which is now known as Planck's constant, and the idea that energy is quantized started a revolution in physics which led to the development of what we now call quantum mechanics.

An essential element in the theory of quantum mechanics is the Schrödinger wave equation developed by Erwin Schrödinger in 1926:

$$-\frac{\hbar}{2m}\frac{\partial^2\Psi(x,t)}{\partial x^2}+V(x,t)\Psi(x,t)=i\hbar\frac{\partial\Psi(x,t)}{\partial t} \tag{1}$$

where $\hbar$ is Planck's constant, 6.626×10^{-34}J·s, divided by 2π, m is the mass of the particle, $\Psi(x,t)$ is a wave function that acts as a probability distribution, and $V(x,t)$ is the potential energy of the particle. Solutions to the above equation can be used to calculate the discrete energy levels of the particle. When solving the Schrödinger equation, a technique called separation of variables is applied. We assume that the wave function $\Psi(x,t)$ can be written as $\psi(x)\phi(t)$, a product of a wave function of just position and a wave function of just time. From this method, the time-independent Schrödinger equation is obtained:

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

where E is the total energy of the particle. Although some potentials can be solved analytically, most real life problems require numerical methods to approximate $\psi(x)$.

In quantum mechanics, many quantities are found only in certain minimum amounts, or integer multiples of those elementary amounts. The electron can only exist at certain energy levels. At the lowest energy level, E_0, the wave function, $\psi(x)$, will look like the fundamental of a standing wave. At the next energy level, E_1, it will look like the first overtone, at E_2 it will look like the second overtone, and so on.

A property unique to quantum mechanics is tunneling. Imagine a potential energy similar to the one in Figure 1 on the left. In classical physics, if a particle is placed into the well, it would exist in either Region I or Region II. In quantum mechanics, the particle will exist in both Region I and Region II. In order to move between the two regions, however, the particle will need to penetrate the barrier between them. This phenomenon is called barrier penetration or tunneling.

Barrier penetration occurs during the inversion of the ammonia molecule (NH_3). An ammonia molecule is structured as a pyramid with a triangular base. The hydrogen atoms are at the vertices of the triangle and the nitrogen is at the apex. The nitrogen atom has two equivalent equilibrium positions, each located on either side of the hydrogen plane. A geometric structure is demonstrated in Figure 1 above. The nitrogen oscillates back and

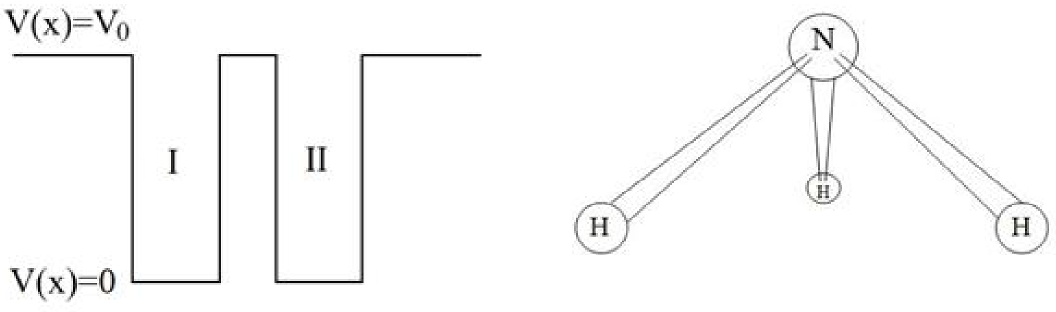

Figure 1: The square well potential with a barrier (left). A geometric structure of the ammonia molecule (right).

forth between the two equilibrium positions passing through the plane of hydrogen atoms, thus inverting the molecule.

Experimentally it was found that the nitrogen oscillates with a frequency of 2.3786×10^{10} Hz when the molecule is in its ground state [2]. This frequency is directly proportional to the ground state energy of the molecule. This means that the frequency of the inversion would change if the energy changed.

The purpose of this research was to determine the effects of changing the position and width of a barrier in the potential energy for the ammonia molecule of the frequency of the molecule's inversion. Changing the position of the barrier is equivalent to making it more plausible for the nitrogen atom to be on one side of the hydrogen plane than the other. Changing the width of the barrier is equivalent to changing the distance between the three hydrogen atoms.

Methods

Algorithm

A numerical method was developed to solve the time-independent Schrödinger equation (Equation 2) based on what the differential equation means conceptually. Consider a coordinate system with the hydrogen plane at $x = 0$. For simplicity we will first consider a case with a symmetric potential energy. This means the wave function $\psi(x)$ will be symmetric and the first derivative at $x = 0$ will be zero. Since the wave function is symmetric we will only approximate it on one side. The values of the wave function will be the same on the other side.

First we will assume that the value of the wave function at $x = 0$ is one. We are able to do that because after we finish approximating the wave function we will normalize it. In order to find the second derivative at $x = 0$ we need to first find the average first derivative from $x = 0$ to some $x = \Delta x$ and the average first derivative from $x = -\Delta x$ to $x = 0$. The value of the wave function at $x = 0$ is ψ_0 and the values of the wave function and $x = \Delta x$ and $x = -\Delta x$ are ψ_1 due to the fact that the wave function is symmetric. The average first derivative from $x = 0$ to $x = \Delta x$ is $\frac{\psi_1 - \psi_0}{\Delta x}$ and the average first derivative from $x = -\Delta x$ to $x = 0$ is $\frac{\psi_0 - \psi_1}{\Delta x}$. The second derivative at $x = 0$ is $\frac{\frac{\psi_1 - \psi_0}{\Delta x} - \frac{\psi_0 - \psi_1}{\Delta x}}{\Delta x}$. From the Schrödinger equation, the second derivative is proportional to the wave function. The proportionality constant is the opposite of kinetic energy times $\frac{2m}{\hbar^2}$. For simplicity we will define $-K$ as $\frac{2m}{\hbar^2}[V(x) - E]$. From this we get $\frac{\frac{\psi_1 - \psi_0}{\Delta x} - \frac{\psi_0 - \psi_1}{\Delta x}}{\Delta x} = -K\psi_0$. The left hand side simplifies to $\frac{2(\psi_1 - \psi_0)}{(\Delta x)^2}$. Solving for ψ_1 yields $\psi_1 = \psi_0 - \frac{K\psi_0(\Delta x)^2}{2}$.

Every successive point is then going to be approximated from the two points before it. Similar to the case of ψ_0, the second derivative at ψ_1 is $\frac{\psi_2 - 2\psi_1 + \psi_0}{(\Delta x)^2}$. The second derivative is equal to $-K\psi_1$. Solving for ψ_2 yields $\psi_2 = 2\psi_1 - \psi_0 - K\psi_1(\Delta x)^2$. As the step size, Δx, is decreased the approximation becomes more and more accurate. The above calculation is demonstrated in Figure 2.

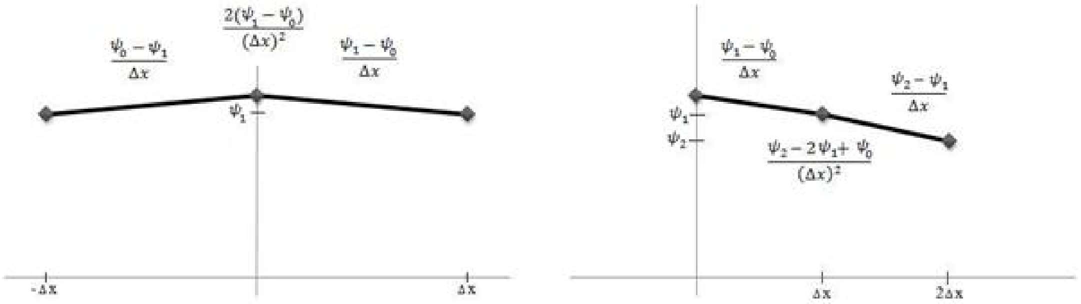

Figure 2: The calculation of the second derivative at $x = 0$ (left). The second derivative at any other point (right).

After all the values have been approximated, the wave function needs to be normalized. The particle has to exist somewhere in the region $(-\infty, \infty)$ so the probability of finding the particle there is one. Consequently $\int_{-\infty}^{\infty} \psi^*(x)\psi(x)\,dx$, which represents the probability, has to equal one. $psi^*(x)$ is the complex conjugate of the wave function. For the potential energies used in this research, the wave function is real so $\psi(x) = \psi^*(x)$. The integral can then be rewritten as $\int_{-\infty}^{\infty} [\psi(x)]^2\,dx$. We can now take the value of the wave function at each individual point, square it, and add them together. We will then divide that total by the step size used. This is the value of the normalization integral, and we want it to equal one. We then have to go back and divide the values of the wave function by the value of the normalization integral we got. This normalizes the function.

It is required for the wave function to converge to zero as x approaches infinity. Otherwise, the integral discussed above will never be equal one. The wave function will converge to zero at certain energy levels. At all other energy values, the wave function will diverge. If the wave function diverges to $-\infty$, the energy must be lowered and if the function diverges to ∞, the energy must be raised.

Testing the Algorithm

The best way to make sure the algorithm works is to use it to solve the equation for a potential energy for which the solution is well known. The first such potential is the square well. The equation for this potential is:

$$V(x) = \begin{cases} V_0 \ \ \text{(a constant)} & x < -\frac{a}{2} \text{ or } x > \frac{a}{2} \\ 0 & -\frac{a}{2} < x < \frac{a}{2} \end{cases} \tag{3}$$

and its graph is demonstrated in Figure 3.

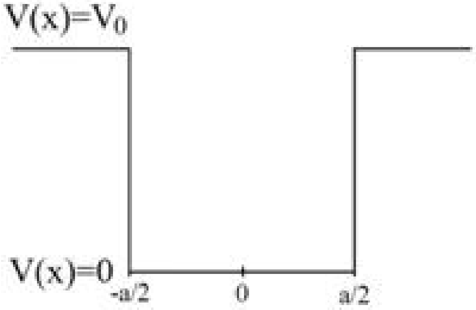

Figure 3: The Square Well Potential.

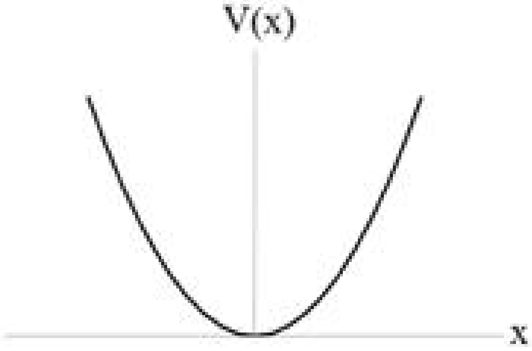

Figure 4: Harmonic Oscillator Potential.

The well known wave function is graphed in Figure 5 along with the approximation obtained using the algorithm. The values of the wave function on the two graphs are extremely similar.

Another common potential is the harmonic oscillator. The potential is:

$$V(x) = \frac{C}{2}x^2 \tag{4}$$

and its graph is shown in Figure 4. The graph of the known wave function and the approximation obtained using the algorithm are both shown in Figure 6.

The approximated wave function $\psi(x)$ obtained using the algorithm for these two potential energy functions was very close to the known solutions. Therefore we conclude that the algorithm used is accurate.

Ammonia Molecule

The potential energy of the ammonia molecule must have the following properties. With no external influence, it must be symmetric about $x = 0$. It must have a barrier centered around $x = 0$ which represents the plane of hydrogen atoms. From the barrier, the energy must decrease on both sides until it reaches a minimum which corresponds to the equilibrium position of the nitrogen atom. From there the potential energy must increase.

In order for this model to realistically represent the ammonia molecule, some approximations must be made. First we must calculate the constant $\frac{2m}{\hbar^2}$. Since both the nitrogen

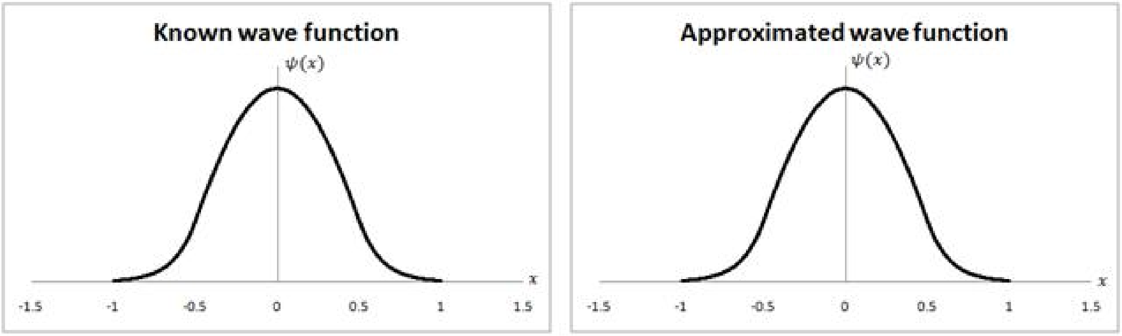

Figure 5: The graph of the known solution to the square well potential (left). The graph of the approximation using the algorithm (right).

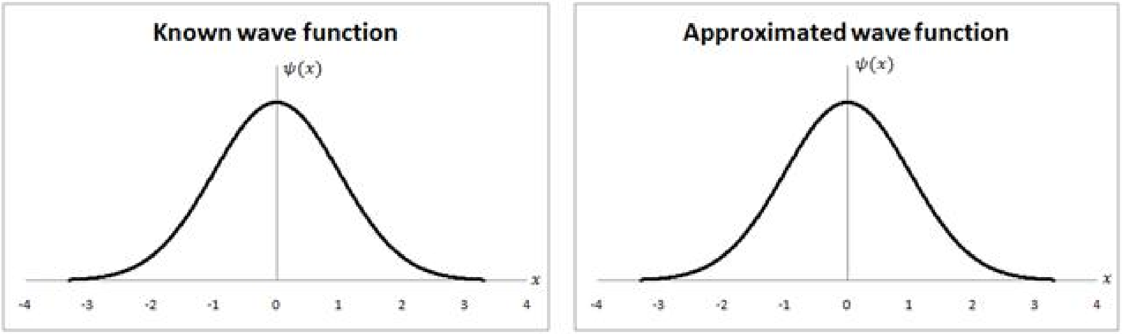

Figure 6: The graph of the known solution to the harmonic oscillator potential (left). The graph of the approximation using the algorithm (right).

and the hydrogen plane move, we must be careful with which mass to use. We will use the "reduced mass" of the system. The "reduced mass" is the mass of the nitrogen when we assume the hydrogen plane does not move. The "reduced mass" $\mu = \frac{m_1 m_2}{m_1+m_2} \times m_p$. We will use the atomic mass of nitrogen, 14.00674, and three times the atomic mass of hydrogen, 1.00794. m_p is the mass of a proton: 1.67262×10^{-27} kg. Thus the "reduced mass" is:

$$\mu = \frac{m_1 m_2}{m_1 + m_2} \times m_p = \frac{(14.00674)(3)(1.00794)}{(14.00674) + (3)(1.00794)} \times 1.67262 \times 10^{-27} \text{ kg}.$$

Thus the constant $\frac{2m}{\hbar^2} = \frac{2(4.15970\times10^{-27})}{\left(6.626\times10^{-34}/2\pi\right)^2} \frac{\text{kg}}{\text{J}^2\text{s}^2} \approx 7.48081 \times 10^{41} \frac{\text{kg}}{\text{J}^2\text{s}^2}$. However, this number is inconveniently large. There is a second derivative with respect to space on the left-hand side of the equation which has a m^{-2} in the units. It is more appropriate in our case to use Angströms (Å). Since 10^{10} Å= 1 m, if we multiply the equation through by 10^{-20} we will convert the meters into Angströms. The constant on the right-hand side is still too large. However, notice that $[V(x) - E]$ on the right-hand side will have the units of Joules. In our case it is more convenient to use electron volts (one electron volt = 1 eV= 1.60218×10^{-19} joules is the energy gained by an electron, of charge 1.60218×10^{-19} coulombs, in falling through a potential drop of 1 V). In order to convert the energy on the right-hand side into electron volts, we must "take" the conversion factor from the constant $\frac{2m}{\hbar^2}$ by multiplying it by 1.60218×10^{-19}. This means the constant becomes $1198.58\frac{\text{kg}}{\text{J}^2\text{s}^2}$. This constant factors into the K in the algorithm above.

Mathematically, we will use a parabola added to a Gaussian to produce a function for the potential energy of ammonia. In order to make our model as accurate as possible, we must calculate several properties of the function. First we need to decide how "wide" the well is. This will depend on the size of the ammonia molecule. The bond length of an ammonia molecule is approximately 101.7 pm or 1.1017 Å, so that's what we will use for the "width" of the well. We must also determine how wide is the barrier in the potential energy. The molecule is a triangular pyramid with the nitrogen at the apex (see Figure 7). The distance from the nitrogen atom to the plane of the hydrogen atoms or to the base of the pyramid is determined as follows.

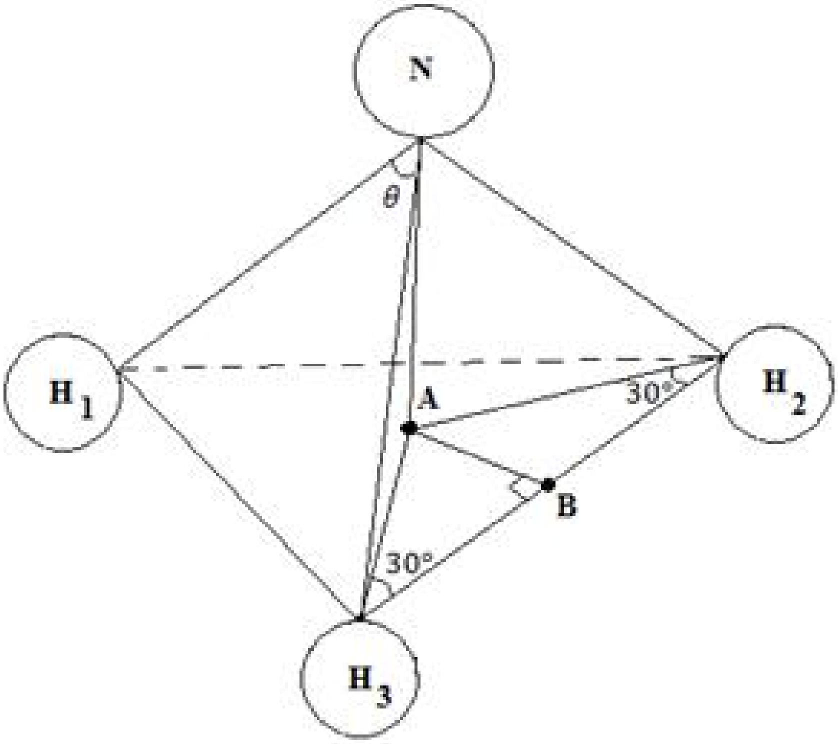

Figure 7: A diagram of the ammonia molecule.

We know the bond length, which is the length $\overline{\text{H}_3\text{N}}$ (and of course $\overline{\text{H}_2\text{N}}$ and $\overline{\text{H}_1\text{N}}$), is 1.017 Å. We also know that the angles $\angle \text{H}_1\text{NH}_3$, $\angle \text{H}_2\text{NH}_3$, and $\angle \text{H}_1\text{NH}_2$ are equal to 107.8°. We want to calculate the length from point A, which is the center of the $\triangle \text{H}_1\text{H}_2\text{H}_3$, and N. First we notice that $\triangle \text{NH}_2\text{H}_3$ is an isosceles triangle, meaning $\angle \text{NH}_2\text{H}_3 = \angle \text{NH}_3\text{H}_2$. From this we calculate $\angle \mathrm{NH_2H_3} = \frac{180^\circ - \angle \mathrm{H_2NH_3}}{2} = \frac{180^\circ - 107.8^\circ}{2} = 36.1^\circ$. Now we know two angles and a side of $\triangle \mathrm{NH_2H_3}$ so it is convenient to use the Law of Sines: $\frac{\sin \angle \mathrm{H_2NH_3}}{\overline{\mathrm{H_3H_2}}} = \frac{\sin \angle \mathrm{NH_2H_3}}{\overline{\mathrm{H_3N}}}$. Solving for $\overline{\mathrm{H_3H_2}}$ yields $\overline{\mathrm{H_3H_2}} = \frac{\overline{\mathrm{H_3N}} \cdot \sin \angle 2\mathrm{NH_3}}{\sin \angle \mathrm{NH_2H_3}} = \frac{1.017 \cdot \sin 107.8^\circ}{\sin 36.1^\circ} \approx 1.643$Å.

Now we will consider $\triangle \mathrm{H_3AH_2}$, which is also isosceles. This means that $\angle \mathrm{H_3AH_2} = 120^\circ$. If we define the midpoint between H_3 and H_2 to be B, then $\overline{\mathrm{H_3}B} = \frac{1.643}{2} \approx 0.822$Å. Now we look at $\Delta \mathrm{H_3AB}$. We know it is a 30-60-90 triangle and we know one of the sides so we can figure out the length $\overline{\mathrm{H_3A}}$. It is $0.822 \cdot \frac{2}{\sqrt{3}} \approx 0.949$Å. Finally we can look at $\triangle \mathrm{H_3AN}$ of which we know two sides and also that it is a right triangle. We will use the Pythagorean Theorem to find $\overline{\mathrm{NA}} = \sqrt{\overline{\mathrm{N_3N}}^2 - \overline{\mathrm{H_3A}}^2} = \sqrt{1.017^2 - 0.949^2} \approx 0.4$Å. This will be the width of the barrier used to model the potential energy of the ammonia molecule.

Now all we have to determine is the "depth" of the well. This directly depends on the energy of the molecule. However, here we face a problem. There are two forces acting on the nitrogen atom. One of them is a repulsion force and the other is an attraction force. We can only calculate one of these forces: the attraction that the nitrogen atom's electron cloud feels from the three hydrogen atoms. Our estimation will be very crude but all we really want is the order of magnitude. We use Coulomb's Law to estimate the potential energy due to this attraction. However, here we face problem: we do not know exactly how many electrons stick closer to nitrogen and how many prefer to stay with the hydrogen atoms.

We make several assumptions and then based on the results, decide which is the most accurate. We will first assume that there are three electrons with nitrogen. Next assume two electrons are with nitrogen. Finally assume one electron is with nitrogen. The first assumption makes the two charges $+3e$ and $-3e$, where e is the charge of an electron. The second assumption makes the two charges $+2e$ and $-2e$ and similarly the last assumption makes the charges $+1e$ and $-1e$.

First Assumption (3 electrons with nitrogen):

$U_e = k\frac{q_1 q_2}{R}$ where k is Coulomb's constant, q_1 and q_2 are the two charges, and R is the distance between the two charges. This means that

$$|U_e| = (8.98755 \times 10^9)\frac{3(1.60218 \times 10^{-19})(3)(1.60218 \times 10^{-19})}{1.017 \times 10^{-10}} \approx 2.11168 \times 10^{-17} \text{ J} \approx 131 \text{ eV}.$$

Second Assumption (2 electrons with nitrogen):

$$|U_e| = (8.98755 \times 10^9)\frac{2(1.60218 \times 10^{-19})(2)(1.60218 \times 10^{-19})}{1.017 \times 10^{-10}} \approx 9.38523 \times 10^{-18} \text{ J} \approx 58.6 \text{ eV}.$$

Third Assumption (1 electrons with nitrogen):

$$|U_e| = (8.98755 \times 10^9)\frac{1(1.60218 \times 10^{-19})(1)(1.60218 \times 10^{-19})}{1.017 \times 10^{-10}} \approx 2.34631 \times 10^{-18} \text{ J} \approx 14.6 \text{ eV}.$$

As was mentioned before, mathematically the model potential energy is a parabola added to a Gaussian. We want it to hit the potential energy, $|U_e|$, at $x = -1.017$, $x =$ barrier position, and $x = 1.017$. We also want to be able to control the barrier's width and position.

The general form of the equation is then $Ax^2+Be^{-\left({}^{x-C}/_{D}\right)^2}$. We must determine the constants A, B, C, and D so that our model follows the requirements mentioned above. We want C to control the position of the barrier and D to control the width of the barrier. At $x = C$, $Ax^2 + Be^{-\left({}^{x-C}/_{D}\right)^2}$ will equal B. Thus $B = |U_e|$. The value of the second term at $x = -1.017$ and $x = 1.017$ will be negligible compared to the first term. Thus at $x = -1.017$ and $x = 1.017$, the first term will be $A(1.017)^2$. We want that to equal $|U_e|$. Therefore $A = \frac{|U_E|}{1.017^2}$. It is important to note that D is not the full width of the barrier but only a half-width. It is also important to note that D is the half-width of the barrier at ${}^1/_e \approx 37\%$ of its height. Since we are not changing the height of the barrier, as long as we are consistent it does not matter where we measure the width. Therefore our equation for the potential energy of ammonia is:

$$\frac{|U_e|}{1.017^2}x^2 + |U_e|e^{-\left({}^{x-C}/D\right)^2} \tag{5}$$

where C is the position of the barrier and D is the barrier's width. The graph of this potential is demonstrated in Figure 8 for the third assumption with $|U_3| = 14.6, A = 14.1, B = 14.6, C = 0$, and $D = 0.2$.

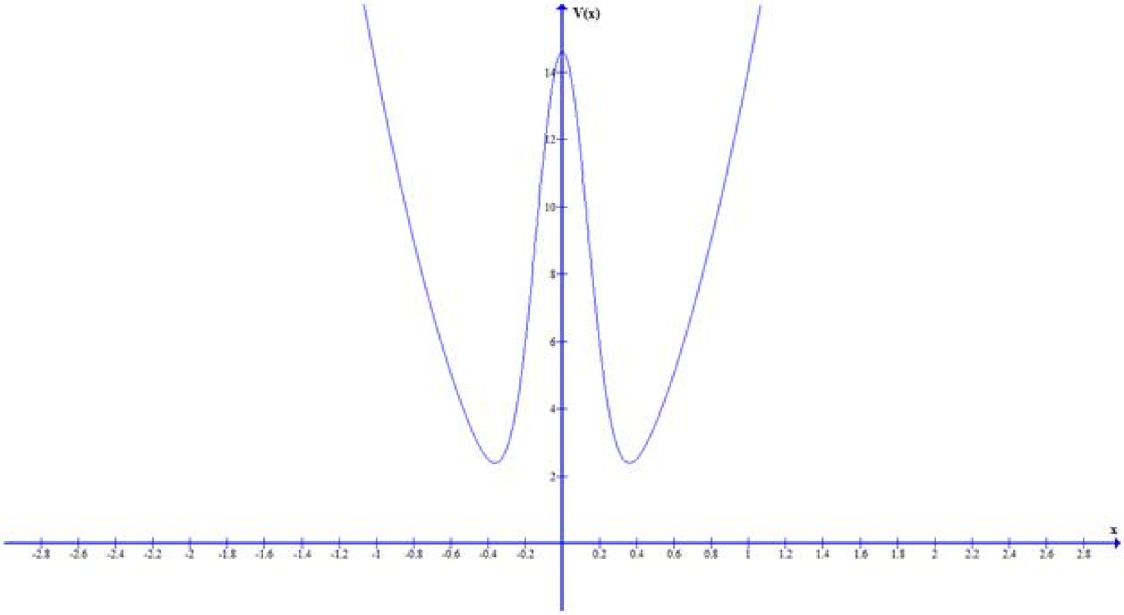

Figure 8: The approximation of the potential energy of the ammonia molecule for $A = 14.1, B = 14.6, C = 0$, and $D = 0.2$.

The algorithm described above now needs a small modification. It will work when the barrier is centered around $x = 0$ but when it is shifted, the first derivative at $x = 0$ is no longer zero. We will still approximate the wave function starting at $x = 0$ and moving first in the positive x direction and then the negative x direction. Varying the first derivative will determine how the two ends of $\psi(x)$ behave relative to each other. This means that first we must vary the first derivative at $x = 0$ until the two ends behave in a similar manner (for example until they are within 25% of each other at $x = -2$ and $x = 2$) and then vary the energy until the two ends get closer and closer to 0. However, as we change the energy, the two ends will get further apart from each other so it is necessary to go back and vary the first derivative. The C++ program implementing the described algorithm is provided in Appendix A.

Demonstration

Since the ammonia molecule is very small and the wave function $\psi(x)$ only tells us the probability of finding the nitrogen atom in a particular location, it is very difficult to imagine the molecule. However, since the wave function $\psi(x)$ is mathematically analogous to a wave in classical mechanics, and the energy of the particle is analogous to the frequency of oscillations in the wave, the results of this research can be demonstrated by observing standing waves on a string.

In order to do that, a set-up similar to the one in Figure 9 is needed. The string of beads is tied between two pieces of wood. Two vertical strings are then tied in such a way as to restrict the movement of the horizontal string, one from each side. The strings are twisted once above the horizontal string and once below it. The length of the horizontal string is 88 cm when it is stretched, the distance between the two hooks to which the horizontal string is attached is 84 cm, and the lengths of the vertical strings are 38 cm. The vertical strings have three main positions: in the middle, 10 cm from the middle, and 20 cm from the middle. Different positions, such as the one 15 cm from the middle (where the strings appear in Figure 9) can be constructed if necessary.

The vertical strings restricted the movement of the beads too much thus imitating a situation where the barrier is very “tall.” To make this demonstration more analogous to the potential energy of the ammonia molecule, the effect of the vertical strings had to be reduced. This could be achieved by two methods: either make the vertical strings thinner or make the horizontal string heavier. The second method was applied. Washers were attached to every bead to increase the mass of the horizontal string.

First it is useful to determine exactly how far the string moves from its rest position during oscillations. Effort must be made to pull the string back exactly by this distance. When the barrier is in the middle, the oscillations of the two halves will be the same but when the barrier is not in the middle, how far the string moves from its rest position will be different for the two regions so adjustment must be made when setting the string in motion.

In order to determine the frequency, two methods were used. First, a metronome was used to approximate the frequency of the standing wave. The second method consists of counting the number of periods in a set amount of time and then dividing the obtained number of periods by the amount of time chosen. The first method relies heavily on the judgement of the person conduction the experiment so the second method is preferred.

Using this set-up we can observe the effects of moving the barrier on the frequency of oscillations of the string. Since the frequency is analogous to the energy of the particle, the behavior of frequency when the barrier is moved should be similar to the behavior of the energy when the barrier in the potential energy of ammonia is moved.

Figure 9: Set-up used to determine the frequency of oscillations of the horizontal string for different positions of the vertical barrier.

Results

Ammonia Molecule

When the program was run with the first assumption that 3 electrons are with nitrogen, the wave function reached 0 too quickly, meaning the potential energy was too high. With the second assumption, the same problem was faced. The third assumption produced the result that would most likely occur in nature. Therefore, the best approximation we can make is that only 1 electron is with nitrogen. In reality, the situation is more complicated but this is a good approximation.

The effect of changing the barrier position on the energy of the particle was determined by calculating the energy for potentials which contained barriers of constant width of 0.4 Å but at different positions. Similarly, the effect of changing the barrier width on the energy of the particle was determined by calculating the energy for potentials which contained barriers at a fixed positions of 0.1 Å but of various widths. Sample wave function and potential energy graphs are shown in Figure 10. The energy as a function of barrier width are shown in Figure 11. Some values of energy at particular barrier positions and for particular barrier widths are shown in Table 1.

As evident from the graphs in Figure 11, as the barrier got further from the middle, the energy decreased and as the barrier got wider, the energy increased.

Barrier Position (Angströms)	Energy (eV)	Barrier Width (Angströms)	Energy (eV)
0.0	7.85188	0.0	3.73176
0.1	7.57448	0.1	4.98444
0.2	6.96712	0.2	6.03856
0.3	6.29844	0.3	6.97880
0.4	5.67940	0.4	7.85188
0.5	5.15234	0.5	8.67240
0.6	4.72602	0.6	9.45496
0.7	4.39752	0.7	10.19956
0.8	4.15954	0.8	10.90620
0.9	3.99310	0.9	11.57488

Table 1: Sample energy values for various barrier positions and barrier widths.

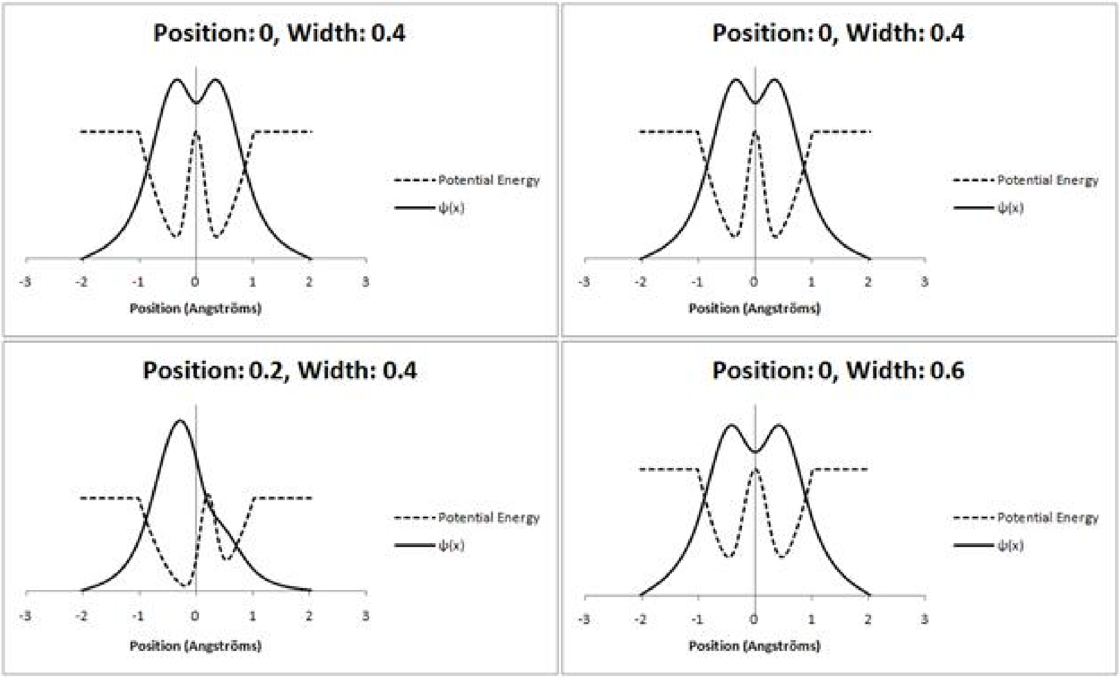

Figure 10: $\psi(x)$ and potential energy for different barrier positions with constant width of 0.4 Å (left). $\psi(x)$ and potential energy for different barrier widths with constant position of 0.1 Å (right).

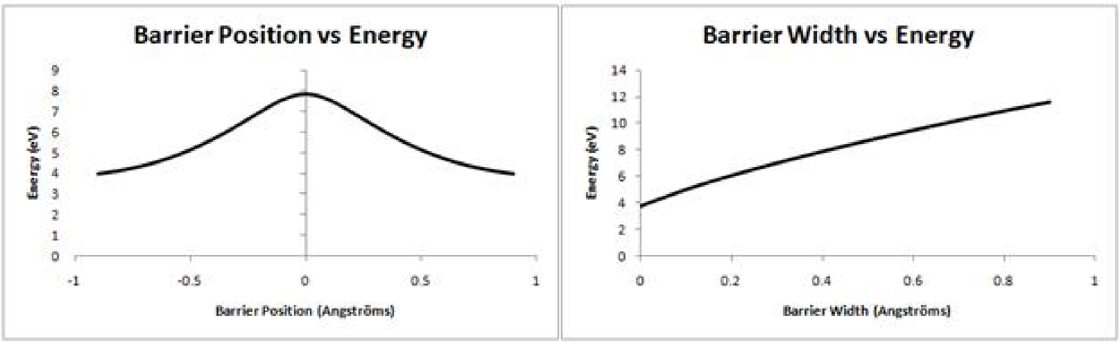

Figure 11: Energy as a function of Barrier Position (left). Energy as a function of Barrier Width (right).

Demonstration

The frequency of the string's oscillation was measured for three different barrier positions. The time interval used was 30 seconds. The results are demonstrated in Table 2 and Figure 12. Before the results can be interpreted, the error must be considered.

For the barrier 0 cm from the middle, the average number of oscillations, $\overline{x_0}$, is $\overline{x_0} = \frac{63+61+62}{3} = 62.0$. The standard deviation, σ_0, is $\sigma_0 = \sqrt{\frac{(63-62)^2+(61-62)^2+(62-62)^2}{3-1}} = 1.00$. From this we know that the frequency for this barrier is ${}^{62.0}/_{30.0} \pm^{\sigma_0} /_{30.0} = 2.07 \pm 0.03$.

The average number of oscillations for the barrier 10 cm from the middle, $\overline{x_{10}}$, is $\overline{x_{10}} = \frac{60+59+60}{3} = 59.7$. The standard deviation, σ_{10}, is $\sigma_{10} = \sqrt{\frac{(60-59.6)^2+(59-59.7)^2+(60-59.6)^2}{3-1}} = 0.58$. From this we know that the frequency for this barrier is ${}^{59.7}/_{30.0} \pm^{\sigma_{10}} /_{30.0} = 1.99 \pm 0.02$.

The average number of oscillations for the barrier 20 cm from the middle, $\overline{x_{20}}$, is $\overline{x_{20}} = \frac{57+56+56}{3} = 56.3$. The standard deviation, σ_{20}, is $\sigma_{20} = \sqrt{\frac{(57-56.3)^2+(56-56.3)^2+(56-56.3)^2}{3-1}} = 0.86$. From this we know that the frequency for this barrier is ${}^{56.3}/_{30.0} \pm^{\sigma_{20}} /_{30.0} = 1.88 \pm 0.03$.

The relative discrepancy between the first and second barrier is $\frac{2.07-1.99}{0.03} = 2.7 = 2.7\sigma_0$. The relative discrepancy between the second and third barriers is $\frac{1.99-1.88}{0.03} = 3.7 = 4.3\sigma_{20}$. The average relative discrepancy is then $\frac{2.7+4.3}{2}\sigma = 3.5\sigma$. A relative discrepancy of greater than 3σ means the measurements are accurate to 90%.

The energy for both ammonia and the demonstration are shown in Figure 13. Even though we cannot accurately determine the width of the barrier in the demonstration, the behavior of the energy is the same for the demonstration and the ammonia molecule: as the barrier became further from the middle, the energy decreased.

Discussion

In the potential for the ammonia molecule, if the barrier is moved, it means that the nitrogen atom prefers to spend more time on one side of the hydrogen plane. The further the barrier is from the middle, the more apparent the nitrogen's preference will be. This also means that when the barrier is moved, the distances between the two equilibrium positions

Barrier Position (cm from middle)	Number of Oscillations				Frequency (Hz)
	Trial 1	Trial 2	Trial 3	Average	
0	63	61	62	62.0	2.07 ± 0.03
10	60	59	60	59.7	1.99 ± 0.02
20	57	56	56	56.3	1.88 ± 0.03

Table 2: Frequency of the string oscillation for three barrier positions determined from measuring the number of oscillations in 30 seconds.

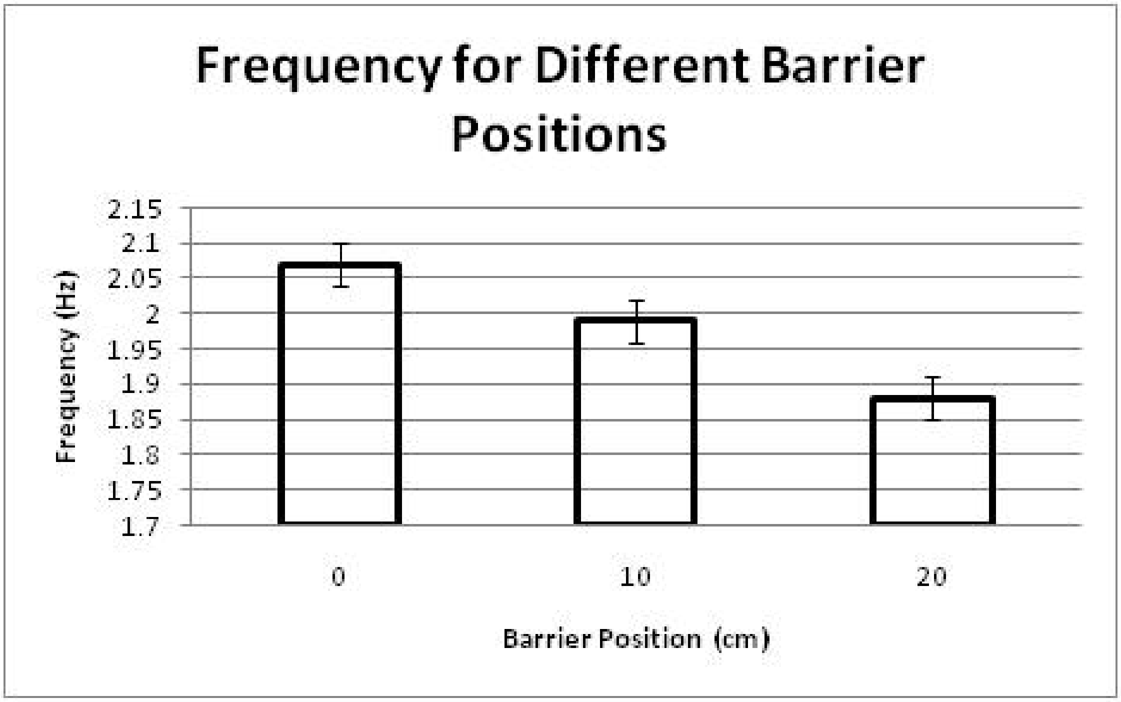

Figure 12: Frequency of the string oscillations for various barrier positions.

and the hydrogen plane will not be equal. When the width of the barrier is changed, it means that the area of the hydrogen plane changes. In other words, the width of the barrier corresponds to how close the three hydrogen atoms are to each other. The wider the barrier, the smaller the area of the triangle and the narrower the barrier, the larger the area of the triangle. The energy of nitrogen is directly proportional to the frequency with which the nitrogen oscillates between its equilibrium positions.

The results demonstrated in Figure 11 show that as the barrier became further from the middle, the energy decreased. This means that if some outside influence, for example an electric field, made it more plausible for the nitrogen atom to exist on one side of the hydrogen plane than the other, the total energy and thus the frequency would decrease. This can be explained conceptually. If it is more likely to find the nitrogen on one side, the nitrogen will spend more time on that side. Consequently it will get farther away from the hydrogen plane when it is at that side and it will also be closer to the hydrogen plane on

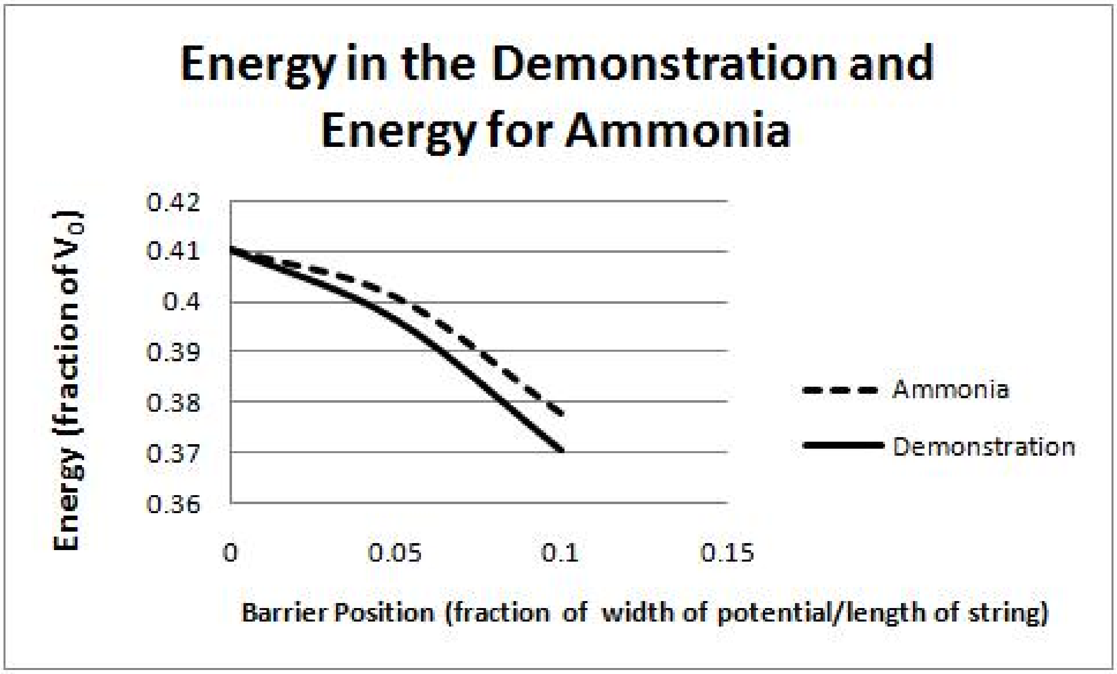

Figure 13: Energy in the demonstration and in the ammonia molecule for various barrier positions.

the other side. The nitrogen has lower energy when it is farther away from the hydrogen plane and it spends more time further away than it would if there was no outside influence. The average energy of the nitrogen will then decrease. Since energy is proportional to the frequency, the frequency will also decrease.

The results also demonstrate that if some outside influence made the hydrogen atoms come closer together, widening the barrier, the total energy and thus the frequency would increase. This is harder to explain. Let us look back at moving the barrier. When the barrier is further from the center, the particle is less likely to encounter it. Also when the barrier is further from the center, the energy of the particle is lower. From this we assume that the action of passing through the barrier requires energy. When the particle has to cross the barrier more often (when the barrier is closer to the middle), its energy is increased. When the barrier is wider, it is harder for the particle to penetrate it so its energy would be higher. The results demonstrated in Figure 11 agree with this explanation.

This suggests that as the width of the barrier approaches the width of the potential energy function, the energy will go to infinity. In other words, eventually there will come a point where there is no energy at which the particle can exist for a barrier of that width. For ammonia, this means that as the hydrogen atoms get closer together, eventually there will come a point where the nitrogen can no longer oscillate.

As the results in Figure 13 show, the behavior of the frequency in the demonstration and the behavior of the energy in ammonia are similar. The demonstration qualitatively shows that moving the barrier away from the middle reduces the frequency of oscillations and thus the energy.

Conclusion and Further Research

This research has demonstrated that moving the barrier in the potential for the ammonia molecule would result in the decrease of the frequency of the nitrogen atom's oscillation. This was shown by numerically solving the Schrödinger equation. The results were then qualitatively confirmed by observing a mathematically analogous case of standing waves on a string.

This research has also demonstrated that increasing the width of the barrier in the potential for the ammonia molecule would result in the increase of the frequency of oscillation of nitrogen. However, the frequency will only increase until the point at which the barrier is too wide to allow the particle to exist in the potential. This was demonstrated by numerically solving the Schrödinger equation.

Further research would include an experiment to confirm the predictions. In a laboratory, an outside force could be used to influence the nitrogen to favor one of its equilibrium points over the other or to increase or decrease the area of the triangular hydrogen plane and observe the effects on the frequency of oscillations of the nitrogen atom.

Appendix A

The C++ program used to implement the numerical method discussed above is presented here.

```
// Schroed.cpp : Defines the entry point for the console
   application.
//

#include "stdafx.h"

#define DIRECTION_POSITIVE 0x0
#define DIRECTION_NEGATIVE 0x1

// global variables
double g_dAlpha = 0.1L; // energy relative to potential V/E
double g_dMinAlpha = 0.0L;
double g_dMaxAlpha = 1.0L;
double g_dTangent = 0.0L; // slope at 0 position
double g_dMinTangent = -50.0L;
double g_dMaxTangent =  50.0L;
double g_dStep = 0.001L; // step for x: u1 - u0
double g_dPotentialFactor = 64.0L; // factor C
double g_dPotentialThresholdPosition = 0.5L; // x point where
   potential goes up
double g_dBarrierPosition = 0.0L;
double g_dBarrierWidth = 0.1L;
```

```
double g_dBarrierMinWidth = 0.0L;
double g_dBarrierMaxWidth = 0.5L;
double g_dStartValue = 1.0L; // Psi at x = 0
double g_dMaxEndValue = 0.0L;
double g_dMinEndValue = 0.0L;
double g_dStartPosition = -1.0L;
double g_dEndPosition = 1.0L;
double *g_dValue = 0;
bool   g_bDoManual = true;
int    g_iNumberIterations = 0;
int g_iIterationsLimit = 100000;
double g_dAccuracy = 0.00001L;

//double *g_pdValue = NULL;
WCHAR g_strDestinationFolder[] = L"c:\\Schroed\\";
WCHAR g_strDestinationFilePrefix[] = L"Ammonia";
WCHAR g_strDestinationFileExtention[] = L".csv";
int g_iSteps;

// declare functions here
void Init(double dBarrierWidth, double dBarrierPosition, double
    dAlpha, double dTangent);
void CalcDistribution(DWORD direction);
double KineticEnergy(double dPosition);
double PotentialEnergy(double dPosition);
HANDLE OpenFileForWriting();
bool WriteToFile();
void Normalize();


int wmain(int argc, WCHAR* argv[])
{
        wprintf(L"Schroedinger␣equation␣numerical␣solution,\n␣
           Ammonia␣molecule␣potential␣energy␣approximation,\n␣
           Rustem␣Bilyalov,␣2010\n");

ENTER_Barrier_WIDTH:
        wprintf(L"Enter␣Barrier␣width:␣");
    double dBarrierWidth;
        scanf_s("%lf", &dBarrierWidth);

        if(dBarrierWidth <= g_dBarrierMinWidth || dBarrierWidth
            >= g_dBarrierMaxWidth)
    {
                wprintf(L"Invalid␣Barrier␣width\n");
```

```
                goto ENTER_Barrier_WIDTH;
    }

ENTER_Barrier:
        wprintf(L"Enter␣Barrier␣position:␣");
    double dBarrierPosition;
        scanf_s("%lf", &dBarrierPosition);

        if(dBarrierPosition <= g_dStartPosition ||
           dBarrierPosition >= g_dEndPosition)
    {
                wprintf(L"Invalid␣Barrier␣position\n");
                goto ENTER_Barrier;
    }

ENTER_ALPHA:
        wprintf(L"Enter␣alpha:␣");
    double dAlpha;
        scanf_s("%lf", &dAlpha);
    if(dAlpha <= g_dMinAlpha || dAlpha >= g_dMaxAlpha)
    {
        wprintf(L"Invalid␣alpha\n");
        goto ENTER_ALPHA;
    }

ENTER_TANGENT:
        wprintf(L"Enter␣tangent:␣");
    double dTangent;
        scanf_s("%lf", &dTangent);
    if(dTangent <= g_dMinTangent || dTangent >= g_dMaxTangent)
    {
        wprintf(L"Invalid␣tangent\n");
        goto ENTER_TANGENT;
    }

        wprintf_s(L"Do␣you␣want␣to␣manage␣iterations␣manualy?␣(
           y/n)␣");
        char yesnoiter[2] = "";
        scanf_s("%s", &yesnoiter, 2);

        if (yesnoiter[0] == ’Y’ || yesnoiter[0] == ’y’)
                g_bDoManual = true;
        else
                g_bDoManual = false;
```

```
    Init(dBarrierWidth, dBarrierPosition, dAlpha, dTangent); //
        initialize values array and other variables

        wprintf_s(L"Alpha:␣%.4f;␣Tangent:␣%+.4f\n", g_dAlpha,
           g_dTangent);

        g_iNumberIterations = 0;
        while(true)
        {
                CalcDistribution(DIRECTION_POSITIVE);
                CalcDistribution(DIRECTION_NEGATIVE);

                WCHAR format[1024] = L"%+.4f␣%+.4f␣%+.4f␣%+.4f␣
                   %+.4f\n";
                wprintf_s(format, g_dValue[0], g_dValue[
                   g_iSteps/4], g_dValue[g_iSteps/2], g_dValue[
                   g_iSteps * 3 / 4], g_dValue[g_iSteps]);

                if(g_bDoManual)
                {
                        wprintf_s(L"Do␣you␣want␣to␣create␣data␣
                           file?␣(y/n)␣");
                        char yesno[2] = "";
                        scanf_s("%s", &yesno, 2);

                        if (yesno[0] == ’Y’ || yesno[0] == ’y’)
                                break;
ENTER_ALPHA2:
                        wprintf(L"Enter␣new␣alpha:␣");
                        scanf_s("%lf", &dAlpha);
                        if(dAlpha <= g_dMinAlpha || dAlpha >=
                           g_dMaxAlpha)
                        {
                                wprintf(L"Invalid␣alpha\n");
                                goto ENTER_ALPHA2;
                        }

ENTER_TANGENT2:
                        wprintf(L"Enter␣new␣tangent:␣");
                        scanf_s("%lf", &dTangent);
                        if(dTangent <= g_dMinTangent ||
                           dTangent >= g_dMaxTangent)
                        {
                                wprintf(L"Invalid␣tangent\n");
                                goto ENTER_TANGENT2;
```

```
    }

    // at this moment we use the same
     Barrier position
    dBarrierWidth = g_dBarrierWidth;
    dBarrierPosition = g_dBarrierPosition;

    Init(dBarrierWidth, dBarrierPosition,
     dAlpha, dTangent);
}
else
{
    if( g_dValue[0] > 0 && g_dValue[
     g_iSteps] < 0)
        g_dTangent += g_dAccuracy;
    else if(g_dValue[0] < 0 && g_dValue[
     g_iSteps] > 0)
        g_dTangent -= g_dAccuracy;
    else if(g_dValue[0] > 0.0L && g_dValue
     [0] > g_dValue[g_iSteps] * 1.25L)
        g_dTangent += g_dAccuracy;
    else if(g_dValue[0] > 0.0L && g_dValue
     [0] * 1.25L < g_dValue[g_iSteps])
        g_dTangent -= g_dAccuracy;
    else if(g_dValue[0] < 0.0L && g_dValue
     [0] > g_dValue[g_iSteps] * 0.80L)
        g_dTangent += g_dAccuracy;
    else if(g_dValue[0] < 0.0L && g_dValue
     [0] * 0.80L < g_dValue[g_iSteps])
        g_dTangent -= g_dAccuracy;
    else if(g_dValue[0] > 0)
        g_dAlpha += g_dAccuracy;
    else if(g_dValue[0] < 0)
        g_dAlpha -= g_dAccuracy;

    wprintf_s(L"Alpha:␣%.5f;␣Tangent:␣%+.5f
     \n", g_dAlpha, g_dTangent);

    g_iNumberIterations ++;

    if(g_iNumberIterations >=
     g_iIterationsLimit)
    {
        goto ENTER_ALPHA;
    }
```

```
                }
        }

        wprintf_s(L"Do␣you␣want␣to␣normalize␣data?␣(y/n)␣");
        char yesno[2] = "";
        scanf_s("%s", &yesno, 2);

        if (yesno[0] == ’Y’ || yesno[0] == ’y’)
        {
                //wprintf_s(L"Normalizing\n");
                Normalize();
        }

        WriteToFile();

        /*
        if(g_dValue != 0)
        {
                delete [] g_dValue;
                g_dValue = 0;
        }
        */

        //wprintf_s(L"Enter any key to continue... ");
        //char anykey[2] = "";
        //scanf_s("%s", &anykey, 2);

        return 0;
}

void Init(double dBarrierWidth, double dBarrierPosition, double
    dAlpha, double dTangent)
{
        g_dBarrierWidth = dBarrierWidth;
        g_dBarrierPosition = dBarrierPosition;
    g_dAlpha = dAlpha;
    g_iSteps = (int) ((g_dEndPosition - g_dStartPosition) /
       g_dStep) + 1;

        if(g_dValue == 0)
                g_dValue = new double[g_iSteps];

        g_dTangent = dTangent;
    //g_pdValue = new double[g_iSteps + 1];
}
```

```
void CalcDistribution(DWORD direction)
{
    double dAlpha = g_dAlpha;
    double dValue0,    dValue1,    dValue2;
    double dPosition0, dPosition1, dPosition2;
        double dPotential0, dPotential1, dPotential2;
    double dKinetic0, dKinetic1, dKinetic2;
    double dx = g_dStep;

        if(direction == DIRECTION_NEGATIVE)
                dx *= -1.0L;

    int iSteps = g_iSteps / 2;
        int iArrayIndex = iSteps;
    int i;

    // calc at step 0
    i = 0;
    dPosition0 = 0.0L;
    dValue0 = g_dStartValue;
        g_dValue[iArrayIndex] = dValue0;
        dPotential0 = PotentialEnergy(dPosition0);
    dKinetic0 = KineticEnergy(dPosition0);
    // output data for step 0

    // calculate at step 1
    i = 1;
    dPosition1 = dPosition0 + dx;
        dValue1 = dValue0 - dKinetic0 * dValue0 * dx * dx / 2.0
           L + g_dTangent * dx ;

        if(direction == DIRECTION_POSITIVE)
                iArrayIndex++;
        else
                iArrayIndex--;

        g_dValue[iArrayIndex] = dValue1;

        dPotential1 = PotentialEnergy(dPosition1);
    dKinetic1 = KineticEnergy(dPosition1);

    // steps 2 to iSteps
    i = 2;
    dPosition2 = dPosition1 + dx;
```

```
    while(i <= iSteps + 1)
    {
        dKinetic2 = KineticEnergy(dPosition2);
                dPotential2 = PotentialEnergy(dPosition2);
        dValue2 = dValue1*2.0L - dValue0 - dKinetic1 * dValue1
           * dx * dx;

                if(direction == DIRECTION_POSITIVE)
                        iArrayIndex++;
                else
                        iArrayIndex--;

                g_dValue[iArrayIndex] = dValue2;

        i++;
        dPosition2 += dx;

        dValue0 = dValue1;
        dValue1 = dValue2;

        dKinetic1 = dKinetic2;
    }
}


double KineticEnergy(double dPosition)
{
    double dKinetic = 0;

        dKinetic = g_dPotentialFactor * g_dAlpha -
           PotentialEnergy(dPosition);

        /*
    if(dPosition < g_dPotentialThresholdPosition)
        dKinetic = g_dPotentialFactor * g_dAlpha; // 64 * Alpha
    else
        dKinetic = - g_dPotentialFactor * (1 - g_dAlpha); // -
           64 * (1 - Alpha)
                */

    return dKinetic;
}

double PotentialEnergy(double dPosition)
{
```

```
    double dP = max(min(dPosition,
      g_dPotentialThresholdPosition), -
      g_dPotentialThresholdPosition);
    double dBP = g_dBarrierPosition;
    double dBHW = g_dBarrierWidth / 2.0L;
    double dF1 = g_dPotentialFactor / pow(
      g_dPotentialThresholdPosition, 2);
    double dF2 = g_dPotentialFactor;

    // dF1 * dP^2 + dF2 * e^(-((dP-dBP)*2/dBW)^2)


      return dF1 * pow(dP,2) + dF2 * exp(-pow(((dP-dBP)/dBHW)
         ,2));
}

bool WriteStepToFile(int iStep, double dPosition, double
   dPotential, double Value)
{
    return true;
}

HANDLE OpenFileForWriting()
{
    int iDestMaxLen = 512;
    WCHAR strDestPath[512];

    swprintf_s( strDestPath, iDestMaxLen, L"%s\\%s_%.2f_%.2f_
      %.5f_%.5f%s",
        g_strDestinationFolder,
        g_strDestinationFilePrefix,
                g_dBarrierWidth,
                g_dBarrierPosition,
                g_dAlpha,
                g_dTangent,
        g_strDestinationFileExtention);

    HANDLE hFile = CreateFile( strDestPath, GENERIC_WRITE,
      FILE_SHARE_WRITE, NULL, CREATE_ALWAYS,
                        FILE_ATTRIBUTE_NORMAL, NULL );
    if( !hFile || hFile == INVALID_HANDLE_VALUE )
    {
        DWORD err = GetLastError();
        wprintf(L"Failed␣to␣create␣\"%s\"\nError␣Code:␣%d",
           strDestPath, err);
```

```
        return 0;
    }

    return hFile;
}

bool WriteToFile()
{
    HANDLE hFile = OpenFileForWriting();
    if(hFile == 0) return false;

    char buffer[1024];
        //HANDLE hFile, double pos, double potential, double
           value
    char format[1024] = "%.4f,%.4f,%.4f\n";
        DWORD dwBytesWritten;
        double dPosition, dPotential;

        for(int i = 0; i < g_iSteps; i++)
        {
                dPosition = g_dStartPosition + g_dStep * (
                   double)i;
                dPotential = PotentialEnergy(dPosition);
                sprintf_s(buffer, 1024, format, dPosition,
                   dPotential, g_dValue[i]);

                if(!WriteFile(hFile, buffer, strlen(buffer), &
                   dwBytesWritten, NULL))
                {
                        DWORD err = GetLastError();
                        CloseHandle( hFile );
                        wprintf(L"Failed␣to␣write␣to␣data␣file\
                           nError␣Code:␣%d", err);
                        return false;
                }
        }

    CloseHandle( hFile );

    return true;
}

void Normalize()
{
        double dArea = 0.0L;
```

```
        for(int i = 0; i < g_iSteps; i++)
                dArea += pow(g_dValue[i], 2) * g_dStep;

        for(int j = 0; j < g_iSteps; j++)
                g_dValue[j] /= sqrt(dArea);
}
```

Bibliography

[1] Cohen-Tannoudji, Claude, Bernard Diu, and Franck Laloë. Quantum Mechanics. S. R. Hemley, N. Ostrowsky, and D. Ostrowsky, Trans., New York: John Wiley & Sons, Inc.; 1977.

[2] Eisberg, Robert and Robert Resnick. Quantum Physics of Atoms, Molecules, Solids, Nuclei, and Particles. New York: John Wiley & Sons, Inc.; 1974.

[3] Liboff, Richard L. Introductory Quantum Mechanics. Oakland, California: Holden-Day Inc.; 1980.

[4] Morrison, Michael A. Undersdanding Quantum Physics. Upper Saddle River, NJ: Prentice Hall; 1990.

[5] Rae, Alastair I. M. Quantum Mechanics. 4th Ed., London: Institute of Physics Publishing; 2002.

[6] Serway, Raymond A. Physics for Scientists & Engineers. 3rd Ed., Philadelphia: Saunders College Publishing; 1992.