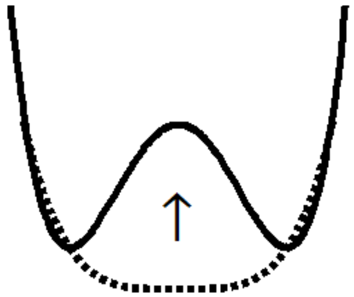
# Computational Perturbation Theory:

## A Tool to be used in Combination with Numerical ODE Solvers



A Thesis Submitted in Partial Satisfaction
Of the Requirements for the Degree of
Bachelor of Science in Physics
At the
University of California, Santa Cruz

By
Samuel Rasmussen Ellis
June 1st, 2010

Michael Dine
Advisor

David P. Belanger
Senior Theses Coordinator

David P. Belanger
Chair, Department of Physics

TABLE OF CONTENTS:

## ABSTRACT

This paper investigates the use of perturbation theory in combination with numerical methods to solve a Schrödinger equation with nonlinear potential. The Numerov method, a fourth order ordinary differential equation solver specialized for the Schrödinger equation, is used to demonstrate how a difficult potential can be split into a piece which is easier to handle and a perturbation. This allows for approximation of eigenvalues and eigenfunctions via perturbation theory, thus negating the necessity of solving the problem with the original potential. It is described how the Numerov method is implemented in C++. A short discussion is given on the energy finding and step size control methods used. This is followed by an exposition of the methods employed in the Mathematica program. This covers the application of perturbation theory to the data output from the Numerov method. The data is organized and perturbation theory is then applied to compute first and second order corrections to the eigenvalues and eigenfunctions as well as the third order correction to the eigenvalues. The corrected eigenvalues are compared to a more accurate numerical method to show the validity of this approach. An overview of perturbation theory is included for completeness. Additionally, the programs used to create and process the data discussed throughout this paper are given at the end as appendices.

## INTRODUCTION

Numerical ordinary differential equation, ODE, solvers can in theory solve a one dimensional Schrödinger equation with any potential to arbitrary precision. In practice, however, this can also take an arbitrarily long computation time. In certain cases it may prove worthwhile to split the original potential into an unperturbed portion and a perturbation allowing for the initial solution, by a numerical method, of only the less complicated unperturbed potential. After solutions are determined to the unperturbed potential, perturbation theory can be applied numerically to correct them for the presence of the perturbation.

In this paper, a scheme for the application of a general perturbation to the eigenfunctions and eigenvalues of a general one dimensional Schrödinger equation with $\hbar$ = m = 1 is worked out under some restrictions. The Mathematica program developed can be applied to any set of eigenfunctions defined as lists of points output by any ODE solver. To demonstrate the flexibility of the program, output for two other perturbations is presented in the *Other Potentials* section. The ODE solver used is the Numerov method.

The application of perturbation theory developed here is analyzed and shown to be consistent with expectations. The analysis is carried out for the quantum double well potential, also known as a quantum quartic anharmonic oscillator. For this test potential the eigenstates and eigenvalues from the perturbative method are quantitatively compared to those from a numerical solver and shown to deviate in a manner consistent with perturbation theory.

## DISCUSSION OF METHODS

### AN OVERVIEW OF PERTURBATION THEORY

Perturbation theory provides a way of constructing approximate solutions to problems which are similar to other problems for which exact solutions are already known. More specifically, approximate eigenfunctions and eigenvalues of an unsolvable differential operator are constructed as linear combinations of the exact eigenfunctions and eigenvalues of a similar differential operator which is solvable.

For a perturbation scaled by $\lambda$, the approximate eigenfunctions, $\psi_n$, are constructed in the form of a perturbative expansion, $\psi_n = \psi^0{}_n + \lambda\psi^1{}_n + \lambda^2\psi^2{}_n + \ldots$ where $\lambda$ is a small parameter. Small means that the solutions of the perturbed problem can be well approximated by linear combinations of solutions of the unperturbed problem. $\psi^0{}_n$ is the $n^{th}$ eigensolution of the unperturbed problem. All of the $\psi^\beta{}_n$ in the perturbative expansion are constructed as linear combinations of eigensolutions $\psi^0{}_m$ for $m \neq n$.

Preliminaries:

Consider an operator H, the infinite set of eigensolutions, $\psi_n$, and their eigenvalues $E_n$. The eigensolutions form an orthonormal basis for the inner product space of H. The eigensolutions and their eigenvalues satisfy the following equation:

$$H\psi_n = E_n\psi_n \tag{1}$$

Any solution to H is a vector, v, in the inner product space of H and can be constructed as a linear combination of the basis functions, $\psi_n$.

$$v = \sum_{n=1}^{\infty} c_n \psi_n \tag{2}$$

The inner product of two such vectors $v_n$, $v_m$ is denoted as follows:

$$< v_n | v_m > = \int_{all\ space} v_n v_m \tag{3}$$

Note that since the basis functions, $f_n$, are orthonormal they satisfy the following property:

$$< \psi_n | \psi_m > = \delta_{nm} = \begin{matrix} 1, & n = m \\ 0, & n \neq m \end{matrix} \tag{4}$$

In this formalism of perturbation theory the unperturbed, original, operator is called $H^0$ and the perturbation is called $H^1$. The new operator, for which approximate solutions are constructed, is H = $H^0$ + $H^1$. Equation (1), combined with the assumption that the solutions to the unperturbed operator are known implies:

$$H^0\psi_n^0 = E_n^0\psi_n^0 \tag{5}$$

The perturbed problem then takes the form of (1):

$$H\psi_n = E_n\psi_n \tag{6}$$

Now, however, the eigensolutions and eigenvalues are unknown. The goal of perturbation theory is not to find these solutions but rather to construct approximations to them. The perturbed problem can be expanded using the idea of the perturbative expansion, $\psi_n = \psi^0{}_n + \lambda\, \psi^1{}_n + \lambda^2\, \psi^2{}_n + \ldots$ From this point forward the perturbation constants, $\lambda^n$, are absorbed into the eigenfunctions to condense notation:

$$(H^0 + H^1)(\psi_n^0 + \psi_n^1 + \psi_n^2 + \cdots) = (E_n^0 + E_n^1 + E_n^2 + \cdots)(\psi_n^0 + \psi_n^1 + \psi_n^2 + \cdots) \tag{7}$$

Note that equation (5) is the collection of 0th order terms from the above equation (7). For this reason the eigensolutions, $\psi^0{}_n$, of the unperturbed operator are called the 0th order approximation of the new eigensolutions, $\psi_n$. Selecting the 1st order terms on both sides of the expansion (7) yields the 1st order equation of the perturbative method:

$$H^0\psi_n^1 + H^1\psi_n^0 = E_n^0\psi_n^1 + E_n^1\psi_n^0 \tag{8}$$

The First Order Correction to the Eigenvalues:

From equation (8) one can construct the 1st order corrections to the $\psi^0{}_n$ and the $E^0{}_n$. Deriving the first order correction to the eigenvalues involves less computation and is shown first. One starts by taking the inner product of both sides of equation (8) with the eigensolutions, $\psi^0{}_n$, this yields:

$$< \psi_n^0|H^0|\psi_n^1 > + < \psi_n^0|H^1|\psi_n^0 > = \; < \psi_n^0|E_n^0|\psi_n^1 > + < \psi_n^0|E_n^1|\psi_n^0 > \tag{9}$$

The first term on the left needs to be simplified; however, the operator cannot be applied to $\psi^1_n$ because the form of $\psi^1_n$ is unknown. One needs to act $H^0$ on $\psi^0_n$ but first an identity is required, as usually operators act only on the function to their right. For an eigenfunction $\psi^0_n$ and operator $H^0$ the Hermitian conjugate, h, has the following property:

$$(\psi_n^0 H^0)^h = (H^0)^h(\psi_n^0)^h \tag{10}$$

To simplify things further, assume that $H^0$ is a Hermitian operator. It follows that $H^0$ is equal to its Hermitian conjugate $(H^0)^h$, because the operation of Hermitian conjugation is simply a transpose operation combined with complex conjugation. Identity (10) combined with the assumption that $H^0$ is Hermitian leads to the commutability of $H^0$ and one of its corresponding eigensolutions. That is:

$$\psi_n^0 H^0 = H^0 \psi_n^0 \tag{11}$$

Working for a moment on just the first term in equation (9), using the above identity (11) one sees that:

$$< \psi_n^0 | H^0 | \psi_n^1 > = < H^0 \psi_n^0 | \psi_n^1 > \tag{12}$$

Which, using equation 5, becomes:

$$< H^0 \psi_n^0 | \psi_n^1 > = < E_n^0 \psi_n^0 | \psi_n^1 > \tag{13}$$

Now the constants $E^0_i$, $E^1_i$ can be pulled out from this term and the third term in equation (9), giving:

$$E_n^0 < \psi_n^0 | \psi_n^1 > + < \psi_n^0 | H^1 | \psi_n^0 > = \ E_n^0 < \psi_n^0 | \psi_n^1 > + E_n^1 < \psi_n^0 | \psi_n^0 > \tag{14}$$

The first term on each side of equation (14) is the same and so they cancel one another, leaving:

$$E_n^1 < \psi_n^0 | \psi_n^0 > = < \psi_n^0 | H^1 | \psi_n^0 > \tag{15}$$

By property (4), orthonormality, this becomes:

$$E_n^1 = < \psi_n^0 | H^1 | \psi_n^0 > \tag{16}$$

This is the first order correction to the eigenvalue. Another way of saying this is that up to first order the eigenvalues are:

$$E_n \approx E_n^0 + E_n^1 = E_n^0 + < \psi_n^0 | H^1 | \psi_n^0 > \tag{17}$$

Here the eigenvalues have been expressed using only quantities from the unperturbed problem, quantities which are known by assumption. Therefore this formula can be used to compute the eigenvalues to first order for a given problem.

The First Order Correction to the Eigenfunctions:

The process of constructing the approximate eigensolutions is slightly more tedious than the method for constructing the approximate eigenvalues. One might have guessed that this would be the case since it is usually harder to find the eigenfunctions than the eigenvalues to an operator which is solvable analytically. The eigenfunctions of H to first order are:

$$\psi_n \approx \psi_n^0 + \psi_n^1 \tag{18}$$

Where the $\psi^0_n$ are known by assumption and the goal is to construct the $\psi^1_n$ as linear combinations of the unperturbed eigenfunctions $\psi^0_m$ for m ≠ n. If m = n then it is possible to absorb $\psi^0_m$ into the $\psi^0_n$. It is assumed that this absorption is done and only eigensolutions with m ≠ n are considered. The first order corrections $\psi^1_n$ are constructed in the form:

$$\psi_n^1 = \sum_{m \neq n} c_{mn} \psi_m^0 \tag{19}$$

The $c_{mn}$ represent how much of $\psi^0_m$ lies in the direction of $\psi^1_n$. As such the $c_{mn}$ are constructed as projections of the $\psi^0_m$ onto $\psi^1_n$. That is:

$$c_{mn} = < \psi_m^0 | \psi_n^1 > \tag{20}$$

This expression is not immediately useful as it contains in itself the function to be solved for, $\psi^1_n$, and thus cannot be computed directly. This representation of $c_{mn}$ does, however, give the chance to find an equivalent expression which can be computed. The equivalent expression is not immediately obvious as it comes out of dotting both sides of the first order equation (8) with the original eigensolutions $\psi^0_m$ and then manipulating the resulting equation to isolate the term $< \psi^0_m| \psi^1_n>$. Taking the inner product of $\psi^0_m$ with both sides of equation (8) yields:

$$< \psi^0_m|H^0|\psi^1_n > + < \psi^0_m|H^1|\psi^0_n > = < \psi^0_m|E^0_n|\psi^1_n > + < \psi^0_m|E^1_n|\psi^0_n > \quad (21)$$

Using properties (12) and (13) and pulling constants out of the inner products, equation (21) becomes:

$$E^0_m < \psi^0_m|\psi^1_n > + < \psi^0_m|H^1|\psi^0_n > = E^0_n < \psi^0_m|\psi^1_n > + E^1_n < \psi^0_m|\psi^0_n > \quad (22)$$

Notice that the first term on both sides contains the piece of interest $< \psi^0_m| \psi^1_n>$. The goal is to write this piece in terms of known quantities so that it can be calculated. This can be done by some rearranging of equation (22). First, the second term on the right hand side of equation (22) is zero by property (4), orthogonality. The following lines show the manipulation:

$$E^0_m < \psi^0_m|\psi^1_n > + < \psi^0_m|H^1|\psi^0_n > = E^0_n < \psi^0_m|\psi^1_n > \quad (23)$$

$$(E^0_n - E^0_m) < \psi^0_m|\psi^1_n > = < \psi^0_m|H^1|\psi^0_n > \quad (24)$$

$$c_{mn} = < \psi^0_m|\psi^1_n > = \frac{<\psi^0_m|H^1|\psi^0_n>}{E^0_n - E^0_m} \quad (25)$$

Having developed a calculable expression for the $c_{mn}$, the eigensolutions of H can now be written down to first order. These are:

$$\psi_n = \psi^0_n + \sum_{m \neq n} \frac{< \psi^0_m|H^1|\psi^0_n >}{E^0_n - E^0_m} \psi^0_m \quad (26)$$

## THE C++ PROGRAM

The C++ program written for this project attempts to solve the Schrödinger equation. This Program uses the Numerov method, a fourth order method of solving ODEs which is particularly adapted for the Schrödinger equation. The steps for this method, which are of width h, are given as:

$$\psi_{i+1} = \frac{2\left(1-\frac{5}{12}h^2k\right)\psi_i-(1+\frac{1}{12}h^2k)\psi_{i-1}}{1+\frac{1}{12}h^2k} \tag{27}$$

Where,

$$k = 2(E-V) \tag{28}$$

At each step k must be evaluated. E is a constant but V = V(x) so its value must be recalculated for each step. If an acceptable set of parameters are set within the program then the attempt will be successful. No perturbation theory takes place here directly, but it is the use of perturbation theory which allows for the potential to be simplified.

The potential is a particularly important parameter because it defines the Schrödinger equation uniquely, thus determining the set of solutions. The program also needs the eigenstate, n, that it is solving for. This is simply a choice but higher n can require smaller step sizes. In general the energy is not known for the potential so it must be guessed. In order to guess the energy one must initially bracket it. Therefore a rough estimate of how energy varies with n must be predetermined so that values both above and below the actual energy value can be entered in the program. The step size is another important parameter which, if lowered, will increase both accuracy and computation time.

The potential solved with this program is the unperturbed potential,

$$V(x) = \tfrac{1}{2}x^4 \tag{29}$$

This potential is determined by using perturbation theory to split the double well into the above piece, (29), and the perturbation, $-\lambda x^2$. The perturbation is not considered further until the Mathematica program is discussed. For this quartic potential the Schrödinger equation is linear and is easily solvable using the Numerov method.

In order to use perturbation theory effectively one must have many of the eigenfunctions and eigenvalues of the unperturbed problem. Each pair of such solutions corresponds to a state number, n. Consequently it makes sense to run the Numerov method over various values of n. In the code given in *Appendix A*, n runs over the ground state and the first 25 excited states.

To find the energy of each eigenstate a simple energy finding algorithm is used which halves the distance twixt bracketing guesses with each iteration. This is accomplished by noticing which way the tail of the eigenfunction, $\psi$, goes. Ideally, if the energy is exact, the tail of $\psi$ will be asymptotic to the x-axis. If the tail appears to bounce off of the x-axis and begin to diverge away from the x-axis then the energy is too high. If the tail instead crosses the x-axis before beginning to diverge away from the x-axis then the energy is too low. In either case, the divergence of the eigenfunction from the x-axis rapidly evolves into a vertical asymptote, yielding solutions which are not valid. This is the bisection method of narrowing in on energies; it is slow but guaranteed to work for a sufficiently small bracketing guess. In this case a bracketing width of about one or two is appropriate. Additionally, if the bracketing guesses are made too far apart it is possible that more than one eigenstate will be bracketed at a time. This can cause quirky behavior in the code and should be avoided.

One needs to make a good initial energy guess for all 25 eigenstates. Guessing the energy can be difficult if one has no idea what it should be; however, with overly large bracketing guesses, and sometimes multiple attempts, one can find the first few energies. Once a few of the energies have been determined more accurately, via an energy finding routine, they can be plotted in terms of n. Fitting a curve to the resulting plot and then extrapolating to further n gives a good estimate for successive energies. A polynomial of order equal to one half the number of points being fitted to works well as a fit function, striking a good compromise of accuracy and reasonable extrapolation behavior. If a polynomial of too high an order is used the known points will be fit to exactly but the fit function will fall off just after the last point, yielding no useful predictions. If a good fit function is used then at least a few new energies will be found. This allows for a refitting to find the next few energies, which are in turn used to find the next few, and so on. The final fit function used is shown in the C++ program, *Appendix A*, where it is offset by +1.2 and -1.2 to set the high and low initial energy guesses. If larger offsets are chosen then some of the early bracketing guesses may contain multiple energies. This should be avoided.

The Numerov method is fourth order in the step size, h. If the step size is decreased the accuracy will increase as the fourth power; however computation time will also increase. In fact, if the step size is lowered sufficiently the Numerov method can be used to solve the quantum double well directly, without the use of perturbation theory. Doing this is very time consuming due to the small step size needed. It is done only as a check and then only for the ground state. A much larger step size is acceptable for the simple quartic potential solved initially. It can be seen in the C++ program, *Appendix A*, that h = 0.001.

The output of the solutions from the C++ program is a series of lists and energies, one such pair for each eigenstate. Each list consists of data points and corresponds to the eigenstate of same number. These lists are printed directly to a Mathematica notebook which is not reproduced in the paper because of its length (each of the 26 lists consisting of 8399 data points).

## THE MATHEMATICA PROGRAM

The Mathematica program, *Appendix B*, implements perturbation theory on the data output from the C++ program. First the data is labeled and categorized as lists. Most calculations throughout the code are done as lists rather than functions because lists are evaluated once and then called after that from storage, where as functions are re-evaluated with each calling.

The novel challenge of this program is the need to either convert the data points for the eigenfunctions to continuous functions which can be used with the usual perturbation theory or modify standard perturbation theory to take functions discretized into data points. While some progress can be made using the first method, there is the need to fit the lists of data points to functions. In a general case, however, one cannot expect to have knowledge of the functions which fit the forms of the eigenfunctions described by the data lists. Also, fitting becomes progressively harder for higher eigenstates and in order to get good perturbative corrections, it is important to fit many of them. For these reasons the second method is chosen. While the Mathematica code attached as an appendix is annotated so as to be navigable on its own, certain aspects warrant greater discussion.

In order for perturbation theory to be applied to the data lists one must rework the inner product between eigenstates which normally is taken as an integral. An equivalent sum function is now presented.

$$< \psi_1(x)|\psi_2(x) > \equiv \int_{-\infty}^{\infty} \psi_1(x)\,\psi_2(x)dx \;\; \rightarrow \;\; \sum_i \psi_{1i}\psi_{2i}\,\Delta x_i \tag{30}$$

The integrand notation is used here so that one can see the correlation between the infinitesimal, dx, and the bin width, Δx. A bin is a section of an eigenfunction represented by a single data point. In this program there are 8399 bins for each eigenfunction. The bin width is the distance in x between two neighboring points. It is the same for each bin because it was set as the step size, h, in the C++ program. The units of dx and Δx are the same and so the units on each side of the relation balance.

The second and third order corrections to the energies and eigenfunctions are long and particularly hard to read in coded format. For completeness the second and third corrections to the energy, $E_n^2$ and $E_n^3$, and the second order correction to the eigenstates, $\psi_n^2$, are reproduced here in cleaner fashion.

$$E_n^2 = \sum_{m \neq n} \frac{< \psi_m^0|H^1|\psi_n^0 >^2}{E_n^0 - E_m^0} \tag{31}$$

$$\begin{aligned} E_n^3 = \sum_{m \neq n} \sum_{l \neq n} & \frac{< \psi_n^0|H^1|\psi_l^0 >< \psi_l^0|H^1|\psi_m^0 >< \psi_m^0|H^1|\psi_n^0 >}{(E_m^0 - E_n^0)(E_l^0 - E_n^0)} \\ & - < \psi_n^0|H^1|\psi_n^0 > \sum_{l \neq n} \frac{< \psi_n^0|H^1|\psi_l^0 >^2}{(E_l^0 - E_m^0)^2} \end{aligned} \tag{32}$$

$$\psi_n^2 = \sum_{m \neq n} \sum_{l \neq n} \frac{< \psi_m^0|H^1|\psi_l^0 >< \psi_l^0|H^1|\psi_n^0 >}{(E_n^0 - E_m^0)(E_n^0 - E_l^0)} \psi_m^0 \tag{33}$$

$$- \sum_{m \neq n} \frac{< \psi_n^0 | H^1 | \psi_n^0 >< \psi_m^0 | H^1 | \psi_n^0 >}{(E_n^0 - E_m^0)^2} \psi_m^0$$
$$- \frac{1}{2} \sum_{m \neq n} \frac{< \psi_n^0 | H^1 | \psi_m^0 >< \psi_m^0 | H^1 | \psi_n^0 >}{(E_m^0 - E_n^0)^2} \psi_n^0$$

Again it should be mentioned that these inner products are carried out in the code as sum functions. To make the process of converting the above equations to their sum function equivalents clear, $E_n^2$ is taken as an example and expanded.

$$E_n^2 = \sum_{m \neq n} \frac{< \psi_m^0 | H^1 | \psi_n^0 >^2}{E_n^0 - E_m^0} \rightarrow \sum_{m \neq n} \frac{[\sum_i \psi_{mi}^0 H_i^1 \psi_{ni}^0 \, \Delta x_i]^2}{E_n^0 - E_m^0} \tag{34}$$

While changing the perturbative correction formulas requires some initial work, the result is powerful. The sum form of the perturbative corrections can be applied to any collection of eigenfunctions known as lists of points, the standard output form from a numerical ODE solver. The program is completely general in that it can take in any eigenfunctions and apply any perturbation to them. To use different eigenfunctions simply redefine the lists for different points. In practice this would mean rerunning the Numerov method. To consider a different perturbation just change the definition of the perturbation in one place in the Mathematica program and rerun it.

## RESULTS

This section presents the final output from the Mathematica program for the example of the quantum double well, the main example discussed throughout the document. Some additional results are shown in the *Other Potentials* section and the *Appendices*.

The Mathematica program will give different graphs of the corrected eigenstates for different values of the perturbative constant, λ. The energy corrections, on the other hand, are computed as functions of λ. This allows for the program to be executed just a single time, showing graphs which illuminate the behavior of each eigenstate while giving energies for multiple λ. If the point data for an eigenstate is needed for a particular value of λ then that value must be set and the program re-run.

The eigenfunctions for the quantum double well can be plotted along with their first and or second order corrections for any λ. Some examples of possible output are given below along with the parameters used to generate them:

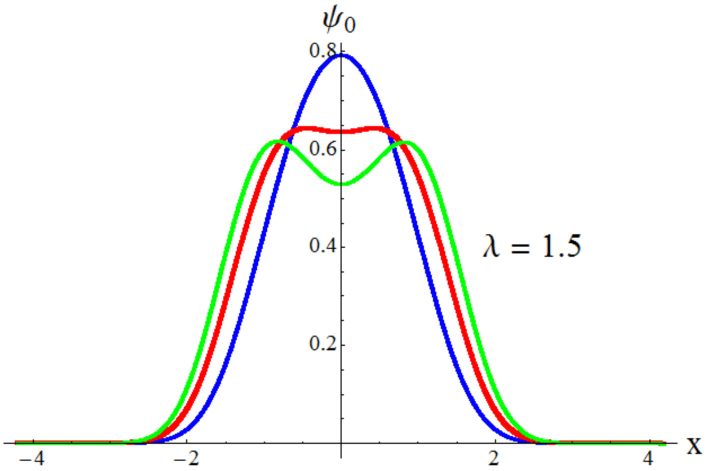


Figure 1: Ground state for the double well. The blue, red, and green lines correspond to the unperturbed, first, and second corrected states respectively.

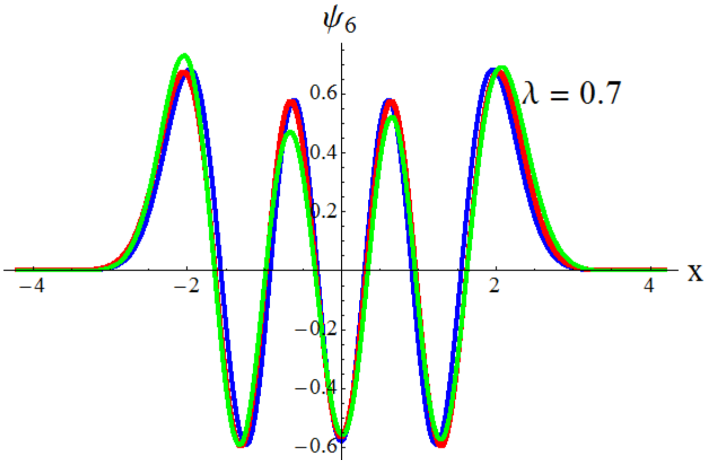
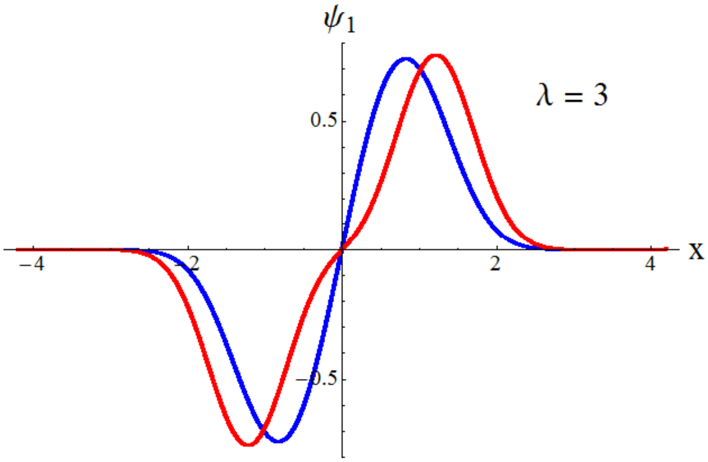


Figure 2: Sixth excited state for the double well. The blue, red, and green lines correspond to the unperturbed, first, and second corrected states respectively.



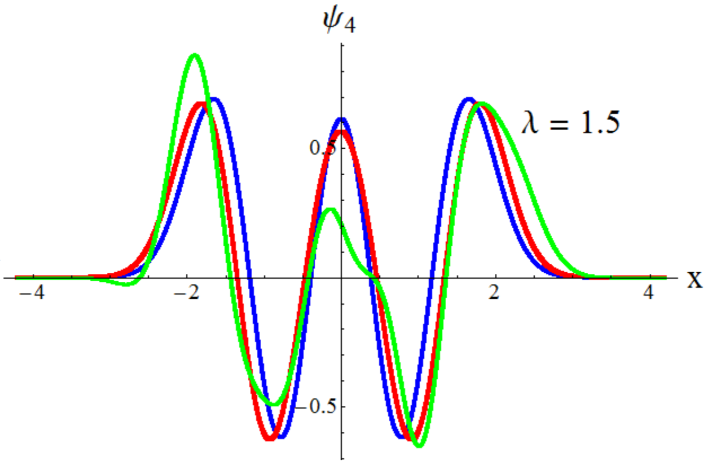


Figure 3: First excited state for the double well. The blue and red lines correspond to the unperturbed and first corrected states respectively.

Figure 4: Fourth excited state for the double well. The blue, red, and green lines correspond to the unperturbed, first, and second corrected states respectively.

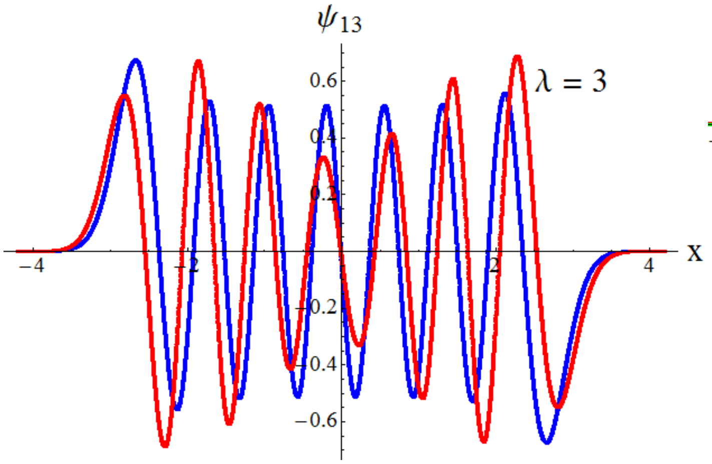


Figure 5: Thirteenth excited state for the double well. The blue and red lines correspond to the unperturbed and first corrected states respectively.

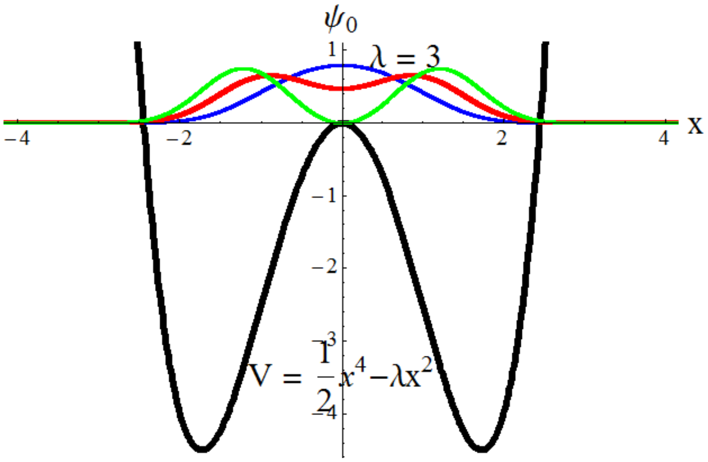


Figure 6: Ground state for the double well. The blue, red, and green lines correspond to the unperturbed, first, and second corrected states respectively. The black line is the double well.

A noteworthy trait in all of the above figures is the aversion of the corrected eigenstates from the center of the well. This makes good qualitative sense as, after being perturbed, the center of the potential is no longer a stable equilibrium. To minimize energy the eigenfunctions clump to the new areas of lowest potential. Greater inspection of Figs. 1-6 is carried out in the *Analysis* section.

The energies are given as functions of λ so computation of them is necessary only once. Before these eigenvalues are listed it is important to note that for larger λ and/or higher eigenstates the energy approximations become less accurate. A more detailed look at this divergence is given in the *Analysis* section. Tabulated in Table 1 in Appendix C are the computed energies along with their first, second, and third order corrections for the ground state and the first 25 excited states.

The entries of Table 1 are not all valid. For eigenvalues above the middle of those considered more corrections are made using lower eigenfunctions than higher eigenfunctions, skewing the corrections. Accordingly, in the fourth and fifth columns of Table 1 the trend of the coefficients is seen to reverse, switching from decreasing to increasing.

**ANALYSIS**

One can understand a great deal about the effect of the perturbation from looking at the corrected eigenstates. In the *Results* section six figures are shown, each demonstrating an interesting property of eigenstate behavior due to the perturbation. Each is covered briefly below.

Figure 1 demonstrates the basic behavior of the eigenfunctions. The plots of the corrected eigenstates are seen to be avoiding $x = 0$. This is because the perturbation causes the center to become a place of high potential energy. As a rough analogy, imagine a rubber bowl with a mound of sand in the middle. As a bump is slowly pushed into the bottom of the bowl the sand begins to slide to either side, creating a dimple in the top of the mound.

Figure 2 shows the sixth excited state with both first and second order corrections. The value of $\lambda$ is set high enough that the effects of the perturbation are visible but not so high as to cause perturbation theory to breakdown. Choosing a higher eigenstate or larger value for $\lambda$ will cause the plot to become much worse. One indication of closeness to breaking down is the visible asymmetry, the leftmost peak of the state corrected to second order is slightly higher than the rightmost.

Figure 3 shows the effect of the perturbation on a state of odd parity, the first excited state. The response to the perturbation is harder to intuit for the odd states than it is for the even ones. One expects the two humps to move away from the center but an additional wiggle is also picked up. This allows for quicker aversion of the center while not forcing the bulk of each hump too close to the sides of the well.

Figure 4 demonstrates how the perturbation theory begins to break down for too high of λ. The corrections are proportional to successive powers of λ so it is expected that the second order correction will become unreasonable before the first does. One can see the unperturbed state, the well-behaved state with first order correction, and the clearly invalid state with second order correction. The violation of symmetry is obvious.

Figure 5 shows that the eigenstates corrected to first order behave well for considerably higher values of λ than do those corrected to second order. Here the thirteenth excited state is shown for a λ double that of the breaking point for the second order perturbation, yet the eigenstate with first order correction is still symmetric. The thirteenth excited state is also important because it is the middle odd state considered. Any higher eigenstates will have fewer functions above them to use in corrections than they will below them. This is because only 26 states were considered.

Figure 6 demonstrates the behavior of the ground state for large λ. The eigenfunctions are shown with a depiction of the double well they occupy. The eigenstate to second order has fully separated; the dip in the middle has reached zero and there are now two separate humps. This is to be expected because as the perturbation becomes large it effectively divides the unperturbed single well into two separate wells.

In all of Figs. 1-6 the corrected eigenfunctions are seen to be avoiding the center of the well even in the figures for which the corrections have become invalid. The above descriptions constitute a qualitative analysis of eigenstate behavior. Below, a quantitative analysis of the corrected eigenstate accuracy is carried out. To accomplish this the ground states corrected to both first and second order are compared to a solution from a numerical method for various values of λ. The Numerov method improved with smaller step size is used for this purpose. The eigenfunctions for some of the values of λ examined are shown below in Figs. 7-9.

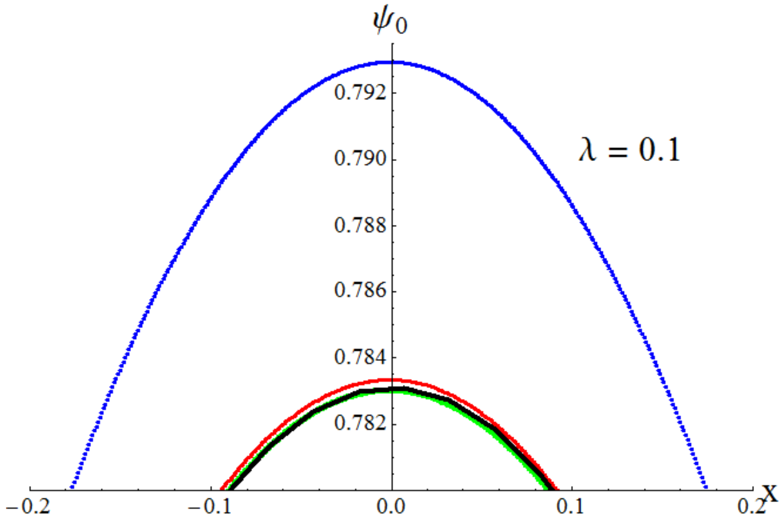


Figure 7: The states from perturbation theory compared to the state from the Numerov method. The blue, red, and green lines correspond to the unperturbed, first, and second corrected states respectively. The Black line is the result from the Numerov method. For such a low value of λ the difference between the corrected states is small. A small viewing range is required in order to see the distinction between the ground state corrected to first and second orders.

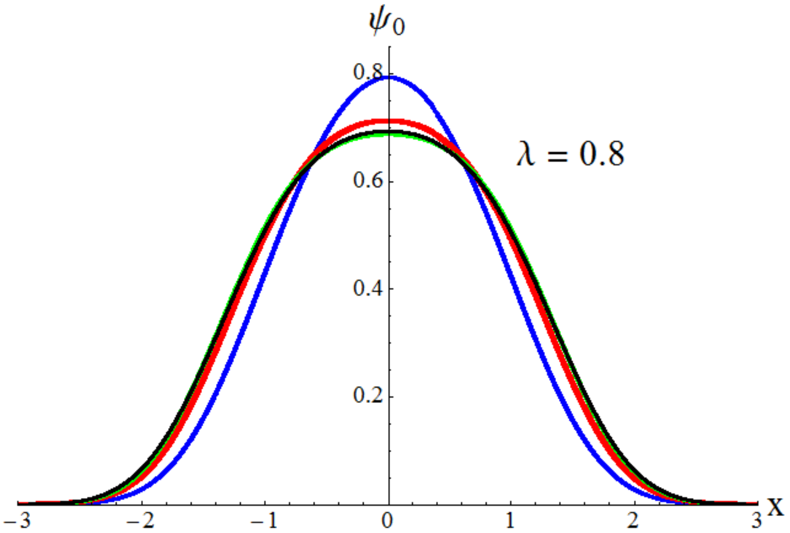


Figure 8: The states from perturbation theory compared to the state from the Numerov method. The blue, red, and green lines correspond to the unperturbed, first, and second corrected states respectively. The Black line is the result from the Numerov method. In this figure the effect of the perturbative corrections is apparent and one can see visually that the first and second order corrections alter the ground state in the correct direction. The result from the Numerov method is overlapping the state corrected to second order to such a degree so as to make them hard to distinguish from one another.

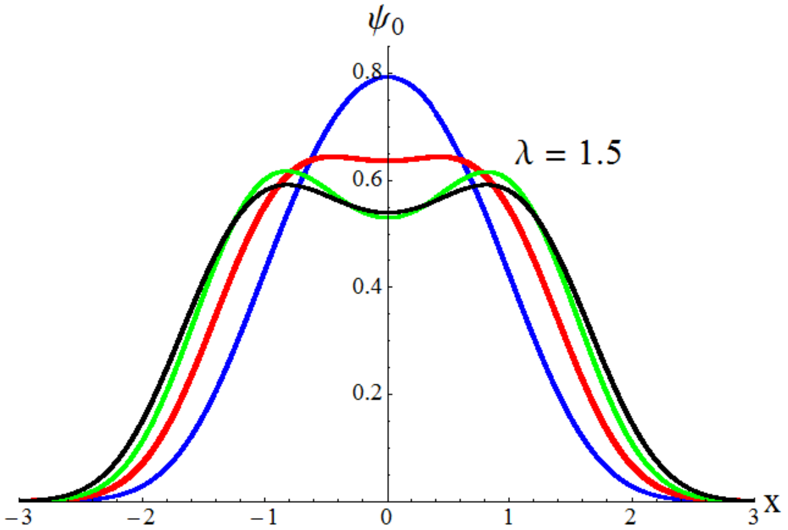


Figure 9: The states from perturbation theory compared to the state from the Numerov method. The blue, red, and green lines correspond to the unperturbed, first, and second corrected states respectively. The Black line is the result from the Numerov method. The eigenstates corrected to first and second order approach the shape of the result from the Numerov method but it can be seen that the corrections are further off from this shape here than they are in Figs. 7 and 8. This is because λ is larger here.

One needs an effective tool to compare the results of the perturbation theory carried out in the Mathematica program to the results of the improved Numerov method. Such a tool is provided by comparing the deviation from unity of the inner products of the functions from the different methods. The inner product definition to use is the one covered in the *Discussion of Methods* subsection The Mathematica Program. The sum definition is used rather than an integral one. Inner products are taken between the ground states resulting from the improved Numerov method and those generated by perturbation theory: the unperturbed, first order, and second order eigenstates. These are done only for the ground state. The values of λ analyzed include those depicted in Figs. 7-9. Plots for the other values are not shown. The results are tabulated in the following table.

| λ | 1 - <Numerov\|Unperturbed> | 1 - <Numerov\|1$^{st}$Corrected> | 1 - <Numerov\|2$^{nd}$Corrected> |
|---|---|---|---|
| 0.1 | $1.14603\cdot10^{-4}$ | $5.08075\cdot10^{-7}$ | $4.09935\cdot10^{-7}$ |
| 0.4 | $2.21315\cdot10^{-3}$ | $3.60469\cdot10^{-5}$ | $2.55092\cdot10^{-6}$ |
| 0.8 | $1.17864\cdot10^{-2}$ | $7.73862\cdot10^{-4}$ | $2.76103\cdot10^{-5}$ |
| 1.1 | $2.82829\cdot10^{-2}$ | $3.51078\cdot10^{-3}$ | $1.54428\cdot10^{-4}$ |
| 1.5 | $7.37790\cdot10^{-2}$ | $1.65842\cdot10^{-2}$ | $1.47229\cdot10^{-3}$ |

Table 2: Differences between inner products and unity. The first column gives the value of λ used to generate the other three columns. The second column shows the deviations from unity of the inner product of the ground state from the Numerov method and the unperturbed ground state of perturbation theory. The third and fourth columns are like the second column but with the unperturbed ground state replaced by the ground state corrected to first and second orders respectively.

In the latter three columns of Table 2, deviation of the inner products from unity can be seen to increase with λ as expected by perturbation theory. The second, third, and fourth columns should scale as $\lambda$, $\lambda^2$, and $\lambda^3$ respectively. This is because error is of one order greater than accuracy. For example, the entries of the third column, which are for the first order correction, should scale as $\lambda^2$. Thus, for the third column it is expected that division by $\lambda^2$ will result in each element being of the same order. The scaling will not be exact as each has different coefficients which also scale them. The results for the fourth column should likewise be of the same order. The following table shows the result of the above described operations on the second, third, and fourth columns of Table 2.

| $\lambda$ | (1 - <Numerov\|Unperturbed>) / $\lambda$ | (1 - <Numerov\|$1^{st}$ Corrected>) / $\lambda^2$ | (1 - <Numerov\|$2^{nd}$ Corrected>) / $\lambda^3$ |
|---|---|---|---|
| 0.1 | $1.14603 \cdot 10^{-3}$ | $5.08075 \cdot 10^{-5}$ | $4.09935 \cdot 10^{-4}$ |
| 0.4 | $5.53288 \cdot 10^{-3}$ | $2.25293 \cdot 10^{-4}$ | $3.98582 \cdot 10^{-5}$ |
| 0.8 | $1.47339 \cdot 10^{-2}$ | $1.20916 \cdot 10^{-3}$ | $5.39264 \cdot 10^{-5}$ |
| 1.1 | $2.57117 \cdot 10^{-2}$ | $2.90147 \cdot 10^{-3}$ | $1.16024 \cdot 10^{-4}$ |
| 1.5 | $4.91869 \cdot 10^{-2}$ | $7.37073 \cdot 10^{-3}$ | $4.36234 \cdot 10^{-4}$ |

Table 3: Deviations of inner products from unity scaled by $\lambda$, $\lambda^2$, and $\lambda^3$. This table shows in the second column the result of dividing the second column of Table 2 by the values of $\lambda$ found in each row. It also shows the same in the third and fourth columns but for the third and fourth columns of Table 2 and division by the values of $\lambda^2$ and $\lambda^3$ respectively.

The quantitative analysis done on the eigenstates from perturbation theory yields mixed results. In Table 3 the entries in the second and third columns are not all of the same order. The elements of the third column span two full orders. This is because other coefficients are present which are larger than the respective $\lambda$. Nevertheless, the inner product deviations from unity are still smaller for smaller $\lambda$. Additionally, as seen in Figs. 8-9, the corrected eigenstates display more similar behavior to the states from the Numerov method than do their unperturbed counter parts. The ground state corrected to second order has been shown visually to behave as expected; in Table 3 this is solidified with a quantitative analysis. The entries in the last column of Table 3 are all close to being of the same order, the ratio of the largest to smallest being just under eleven. Even more significantly, the elements do not grow monotonically with $\lambda$ but instead level off such that the value of the first element is nearly the same as the last.

One can also analyze the eigenvalues corrected by perturbation theory. It is easier to carry out a quantitative comparison of the eigenvalues than it is for the eigenfunctions because only individual numbers must be compared from the numerical and perturbative methods and no inner products need to be computed. Also, the perturbation energies are known as functions of λ unlike the eigenfunctions which are known only for a discrete set of values. In Figs. 10-11 the eigenvalues of the ground state generated by the Mathmatica program via perturbation theory are compared to those of the ground state from an improved Numerov method. In the interest of efficiency, and because values of the eigenfunction at specific values of x are not necessary for computing energies, the Numerov method has been given a step size control which allows it to take steps as small as one hundredth the size used in generating the unperturbed eigenfunctions. The same can be accomplished more simply, although with more computation time, by permanently reducing the step size by a factor of a hundred. A comparison of eigenvalues is depicted visually in the following two figures for the ground state and first excited state.

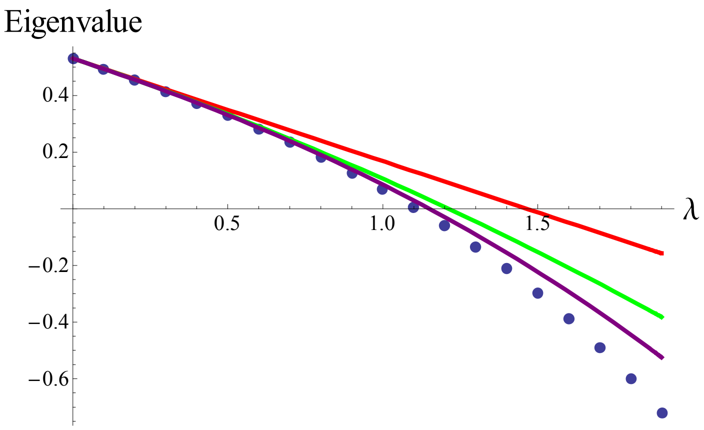


Figure 10: Comparison of ground state eigenvalues from perturbation theory against those from the Numerov method. The red, green, and purple lines represent the first, second and third order eigenvalues while the points represent the output from the improved Numerov method.

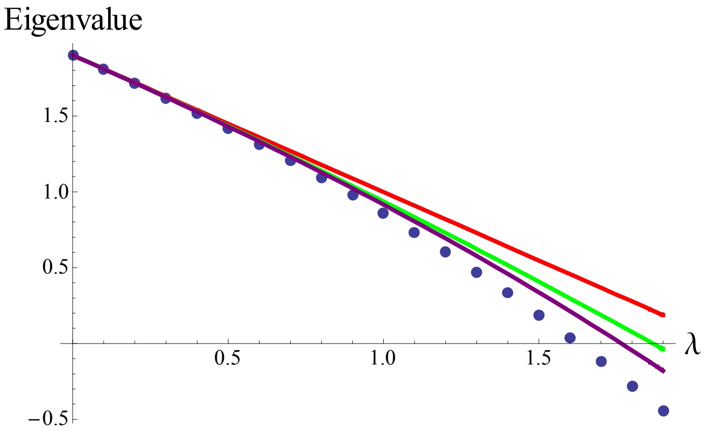


Figure 11: Comparison of first excited state eigenvalues from perturbation theory against those from the Numerov method. The red, green, and purple lines represent the first, second and third order eigenvalues while the points represent the output from the improved Numerov method.

As expected, all of the perturbative approximations in Figs. 10 and 11 diverge from the exact values for larger λ. These figures also show that the eigenvalues corrected to higher order more closely follow the eigenvalues from the Numerov method than do those corrected only to a lower order. The third order line touches more of the points generated by the Numerov method for the ground state than it does for the first excited state. This indicates a trend that higher order eigenstates will be less accurately fit using perturbation theory. To perform a quantitative analysis on Figs. 10 and 11 the distances between the lines and points are inspected. The distances should scale as powers of λ. The distance between the first order line and each point should scale as $\lambda^2$. For the second order line this distance should scale as $\lambda^3$, and for the third order line as $\lambda^4$. The right quantities to analyze, then, are these distances divided by the appropriate powers of λ. As with the analysis of the eigenstates one hopes that these scaled distances will all be of the same order. The results of the above described operations are Tabulated in Tables 4 and 5, *Appendix C*, for Figs. 10 and 11 respectively.

Looking at the last three columns of Tables 4 and 5 it can be seen that each column consists of elements of a single order, except in some cases where the first few elements are larger. These larger low λ entries are due to the difference between the lines and points being known only to a certain number of decimal places. The limited precision causes these very small differences to be overestimated. When they are subsequently divided by $\lambda^3$ or $\lambda^4$ the small value of λ causes the overestimated distances to blow up. The entries in the third column are closer together than those in the second. Likewise, the entries in the fourth column are closer than those in the third. This makes sense because higher order corrections should be smaller.

The preceding analysis of the eigenfunctions and eigenvalues shows proper correspondence of the approximate perturbation theory results to the arbitrary precision results of the numerical Numerov method. It also demonstrates that the perturbation theory is being implemented correctly and effectively. The Mathematica program, *Appendix B*, can be used to gain qualitative as well as quantitative information about changes to eigenstates and eigenvalues due to a perturbation.

## OTHER POTENTIALS

The Mathematica program developed is easily applied to problems other than the quantum double well. To consider other even perturbations, the definition of the perturbation only needs to be altered in a single place. Changing to an odd perturbation requires that the modulus functions in the Mathematica program be offset by one. The necessity of redoing the modulus functions could be removed, at a cost of computation time, by summing over all states rather than just those of the same parity, as can be done for even potentials. In this section some sample output for two other perturbations is shown but an in-depth analysis is not given.

As a first example, consider a perturbative bump of shape other than a negative quadratic in the bottom of the potential well. A differently shaped bump should affect the eigenfunctions differently. Consider a sawtooth perturbation, $H^1$, of the form:

$$H^1 = -\lambda|x| \tag{35}$$

Which when implemented in the Mathematica code, *Appendix B*, takes the form:

$$\text{perturbationE[x\_]:} = -\lambda * \text{Abs}\left[\text{psinorm}\left[[1, x, 1]\right]\right] \tag{36}$$

Below are a few graphs for various values of λ.

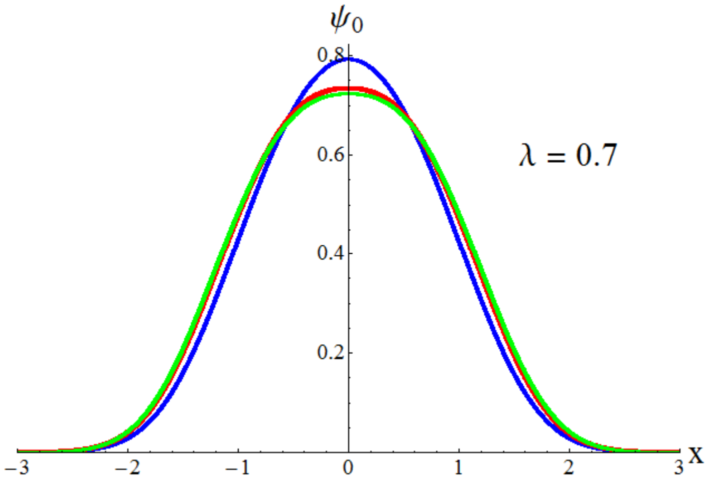


Figure 12: Corrected eigenstates for V = 0.5x$^4$ - 0.7|x|. The blue, red, and green lines correspond to the unperturbed, first, and second corrected states respectively.

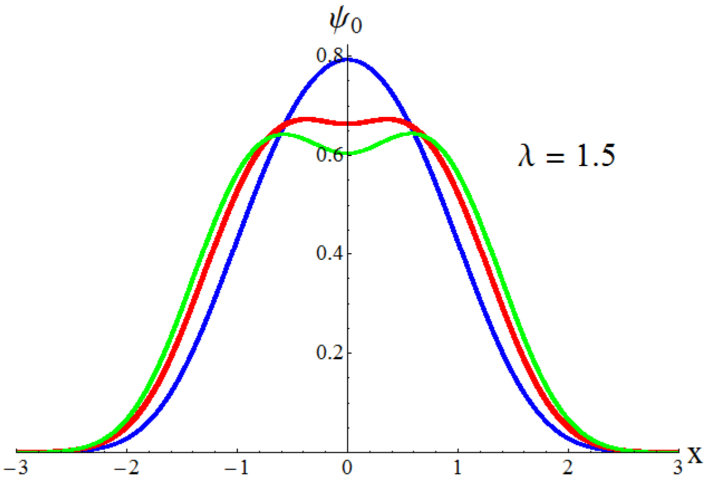


Figure 13: Corrected eigenstates for V = 0.5x$^4$ – 1.5|x|. The blue, red, and green lines correspond to the unperturbed, first, and second corrected states respectively.

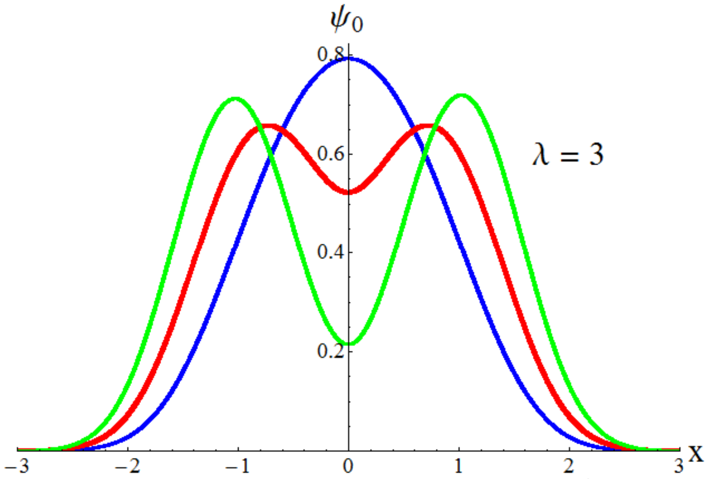


Figure 14: Corrected eigenstates for V = 0.5x$^4$ - 3|x|. The blue, red, and green lines correspond to the unperturbed, first, and second corrected states respectively.

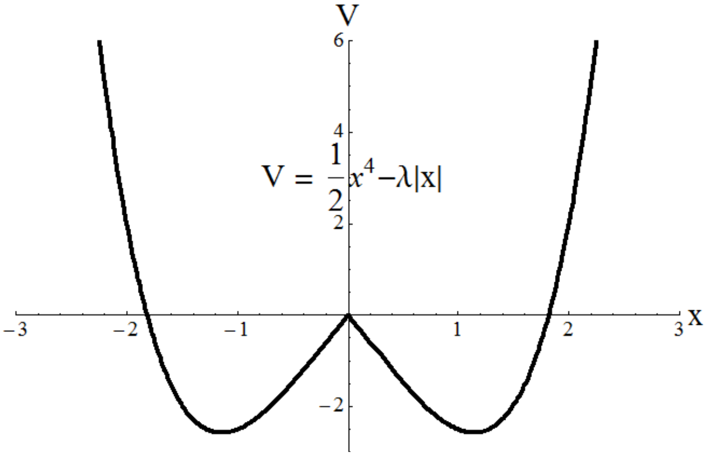


Figure 15: The potential with the sawtooth wave perturbation. This potential is similar to the one for the double well shown in Fig. 6.

Figures 12-14 show that the eigenfunctions for the sawtooth wave perturbation are similar to those for the double well. This is expected because the potential for this case, shown in Fig. 15, is similar to that of the double well, shown in Fig. 6. However, the two potentials are different and the consequences of this can be seen by looking at the eigenvalue corrections. The third order eigenvalue formula for the sawtooth perturbation is:

$$E = 0.530181 - 0.487578\,\lambda - 0.055036\,\lambda^2 - 0.004780\,\lambda^3 \tag{37}$$

This can be compared to the first row of Table 1, and is different.

Now consider a second alternative perturbation of the form:

$$H^1 = -\lambda\, Cos(x) \tag{38}$$

Which when implemented in the Mathematica code, *Appendix B*, takes the form:

$$\text{perturbationE[x\_]} := -\lambda * \text{Cos}\left[8 * \text{psinorm}\left[[1, x, 1]\right]\right] \tag{39}$$

This perturbation effectively puts a series of little bumps in the bottom of the well, giving it a wavy appearance.

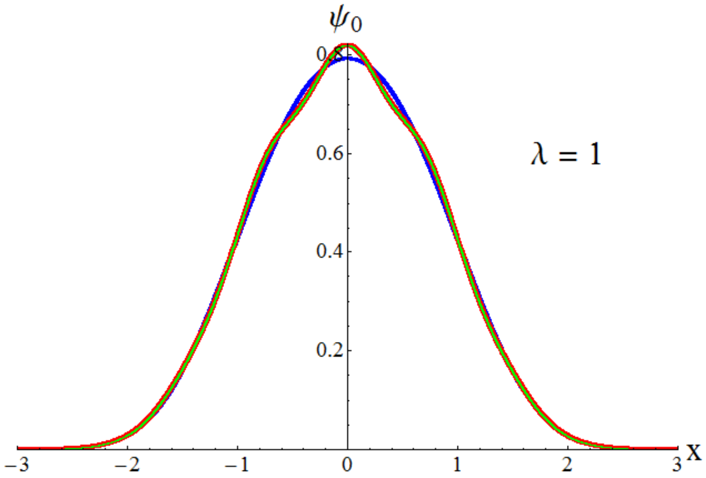


Figure 16: Corrected eigenstates for $V = 0.5x^4 - Cos(8x)$. The blue, red, and green lines correspond to the unperturbed, first, and second corrected states respectively.

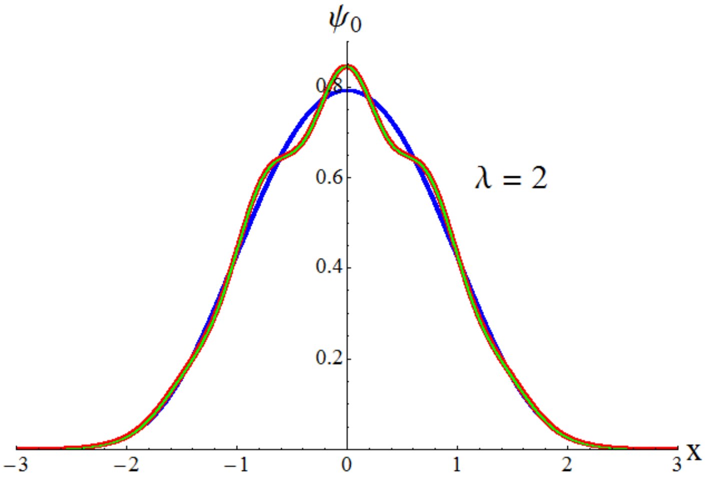


Figure 17: Corrected eigenstates for $V = 0.5x^4 - 2Cos(8x)$. The blue, red, and green lines correspond to the unperturbed, first, and second corrected states respectively.

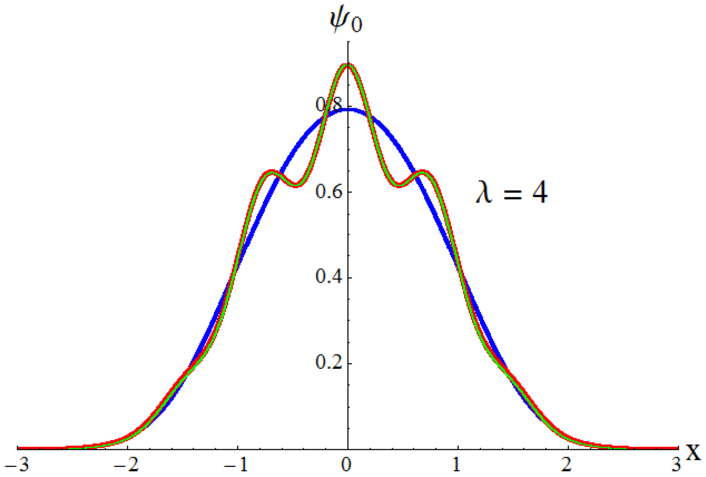


Figure 18: Corrected eigenstates for $V = 0.5x^4 - 4\text{Cos}(8x)$. The blue, red, and green lines correspond to the unperturbed, first, and second corrected states respectively.

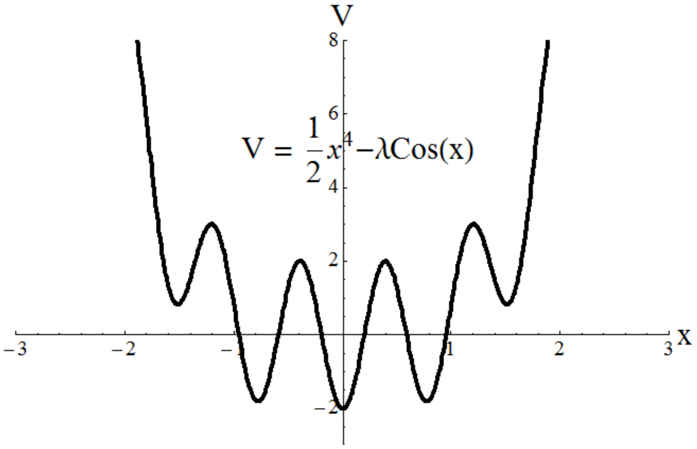


Figure 19: The potential with the cosine perturbation. This potential differs significantly from the one for the double well shown in Fig. 6. This potential models a chain of oscillators inside a larger well.

Figures 16-18 show that the eigenfunctions for the cosine perturbation exhibit novel behavior which differs significantly from the behavior of the eigenfunctions of the unperturbed double well. Without a central bump in the potential well the ground state eigenfunction is not pushed from the middle. Instead, multiple smaller dips are seen to form in the eigenfunction corresponding to the multiple smaller bumps in the potential. The eigenfunction still avoids the steep side walls and so the inner bumps have more influence. The third order energy function for the ground state is found to be:

$$E = 0.530181 - 0.000177\,\lambda - 0.074440\,\lambda^2 + 2.34821 \cdot 10^{-7}\,\lambda^3 \qquad (40)$$

**CONCLUSIONS**

Figures 1-6 and the qualitative analysis of them show how the perturbative corrections to the eigenstates can be visualized using the programs written to gain an intuitive grasp of their behavior. Table 1 shows the eigenvalue information output by the Mathematica program. These approximating functions may in many applications provide enough accuracy to be used in lieu of values obtained from a numerical ODE solver. The validity of the approximations of both the eigenfunctions and eigenvalues was verified in the *Analysis* section. Deviations between the eigenvalues and eigenfunctions from perturbation theory and those from an arbitrary precision numerical method were found, as expected by perturbation theory, to vary as λ to a power equal to one greater than the order of correction. No problems have been found other than the inherent breakdown of perturbation theory for large λ.

The investigation of the quantum double well showed how perturbative corrections can completely alter the unperturbed eigenfunction behavior. It was also shown that as the perturbation constant, λ, grows the eigenfunctions increasingly avoid the center of the well. As seen in Figs. 6-9 the eigenfunction, which is originally a Gaussian-like single hump, begins to split with increasing λ. In Fig. 6, where the value of λ is three, this single hump has fully separated into two humps which individually look similar to the first. This process represents a smooth transition from single oscillator to coupled oscillator to two distinct oscillators.

The end result is a program which is customizable to allow study of various one dimensional potentials. A generalization of this program to higher dimension should be achievable. This application of the perturbative method provides an alternative to the use of more powerful numerical methods as well as a possible addition to them. The Mathematica program developed can be used to apply perturbation theory to the solutions from any numerical ODE solver such as a $k^{th}$ order Runge-Kutta method. In circumstances where a number of different small changes to a single base potential need analysis, the computation time saved using the approximation method developed here would become significant.

## REFERENCES

1. L. D. Landau and E. M. Lifshitz, Quantum Mechanics: Non-Relativistic Theory, translated by J. B. Sykes and J. S. Bell, (Pergamon, London, 1958), pp. 47-49, 64-70, 133-136.
2. W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, Numerical Recipes, 3rd ed. (Cambridge University, Cambridge, 2007), pp. 899-915.
3. K. Yamazaki, "Quantum Theory of Anharmonic Oscillators," J. Prog. of Theo. Phys. 70 (3), (1983)
4. A. V. Turbiner (2009). "Double Well Potential: Perturbation Theory, Tunneling, WKB". arXiv:0907.4485v1 [math-ph].

## APPENDIX A: THE C++ PROGRAM

```
#include <iostream>
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
using namespace std;

int main ()
{
    //Define the necessary variables
    double h=.001,psi3,psi2,psi1,V,x,k,E,E1,E2,n;
    int u=0;
    //Targeted eigenstate, n=0,1,2,3,...
    n=6;
    //Bracketing guess values for n-th energy(starts with average):
    E1=-1.2+0.509372249365938 + 1.223220857041364*n + 0.2241123718386007*pow(n,2) -
    0.0170574212698937*pow(n,3) + 0.000959729860884027*pow(n,4) -
    0.0000292010859019508*pow(n,5) + 3.59836103774546*pow(10.0,-7.0)*pow(n,6);
    E2=1.2+0.509372249365938 + 1.223220857041364*n + 0.2241123718386007*pow(n,2) -
    0.0170574212698937*pow(n,3) + 0.000959729860884027*pow(n,4) -
    0.0000292010859019508*pow(n,5) + 3.59836103774546*pow(10.0,-7.0)*pow(n,6);
    E=(E1+E2)/2.;
    //Setup the print file:
    FILE * aFile;
    aFile = fopen ("x4energy_finder.nb","w");
    fprintf (aFile,"data={");
//The energy Iterations:
for(int j=1;j<=1000;j++)
{
       //Set initial conditions:
       psi1=0;
       psi2=.0001*h;
       u=u+1;
//Steps forwards Schrodinger equation with m=hbar=1:
for(int i=0;i<=10000+200*n;i++)
 {
      x=h*(i-4598-100*n);
      V=.5*x*x*x*x;
      k=2*(E-V);
      //One step forward needs 2 initial conditions:
      psi3=(2.-((10./12.)*h*h*k)*psi2-(1.+(1./12.)*h*h*k)*psi1)/(1.+(1./12.)*h*h*k);

      psi1=psi2;
      psi2=psi3;
 }
 //Energy finder:
 if(fabs(psi3)>.01)
 {
      if(pow(-1,n)*psi3>0.) //Need to raise energy
      {
      //New low bound
      E1=E;
      //New guess
      E=(E+E2)/2.;
      }
      else //Need to lower energy
      {
      //New high bound
      E2=E;
      //New guess
      E=(E+E1)/2.;
      }
 }
 else
 {break;}
}
//Calculate and print trajectory for determined E
for(int i=0;i<=10000+200*n;i++)
 {
      x=h*(i-4998-100*n);
      V=.5*x*x*x*x;
      k=2*(E-V);

psi3=(2*(1.-((5./12.)*h*h*k))*psi2-(1.+(1./12.)*h*h*k)*psi1)/(1.+(1./12.)*h*h*k);
//one step forward needs 2 innitial conditions
      psi1=psi2;
      psi2=psi3;
      if(i%25==8 && i<(8800+200*n) && i>1200)
      {fprintf (aFile,"{%.5f,-(1/10^36)*%.5f},",x,psi3);}
 }
  fprintf (aFile,"{0,0}};\n E=%.50f, j-iterations=%i \n",E,u);
return (0);
}


      //One step forward needs 2 initial conditions:

      psi3=(2.-((10./12.)*h*h*k)*psi2-(1.+(1./12.)*h*h*k)*psi1)/(1.+(1./12.)*h*h*k);

      psi1=psi2;
      psi2=psi3;
 }
 //Energy finder:

 if(fabs(psi3)>.01)
 {
      if(pow(-1,n)*psi3>0.) //Need to raise energy
      {
      //New low bound

      E1=E;

      //New guess

      E=(E+E2)/2.;
      }
      else //Need to lower energy
      {

      //New high bound
      E2=E;

      //New guess

      E=(E+E1)/2.;
      }
 }
 else
 {break;}
}
//Calculate and print trajectory for determined E

for(int i=0;i<=10000+200*n;i++)
 {
      x=h*(i-4998-100*n);
      V=.5*x*x*x*x;
      k=2*(E-V);

psi3=(2*(1.-((5./12.)*h*h*k))*psi2-(1.+(1./12.)*h*h*k)*psi1)/(1.+(1./12.)*h*h*k);
      psi1=psi2;
      psi2=psi3;
      if(i%25==8 && i<(8800+200*n) && i>1200)
      {fprintf (aFile,"{%.5f,-(1/10^36)*%.5f},",x,psi3);}
 }
  fprintf (aFile,"{0,0}};\n E=%.50f, j-iterations=%i \n",E,u);
return (0);
}
```

## APPENDIX B: THE MATHEMATICA PROGRAM

This notebook requires that the notebook output from *Appendix A: The C++ Program* be run first.

First define the perturbative constant lambda. This definition is used only for the eigenstates. Eigenvalues are computed for a general unspecified $\lambda$.

```
In[53]:= lambda = 2.0;
```

The next lists concisely label the data output from *Appendix A*. The *x4list* contains information for the first 26 eigenfunctions. The *energylist* contains the first 26 eigenvalues. The *areasumlist* contains information to be used in normalization of *x4list*.

```
In[54]:= x4list = {datax40, datax41, datax42, datax43, datax44, datax45, datax46, datax47, datax48,
            datax49, datax410, datax411, datax412, datax413, datax414, datax415, datax416, datax417,
            datax418, datax419, datax420, datax421, datax422, datax423, datax424, datax425};
```

```
In[55]:= areasumlist = {areasum0, areasum1, areasum2, areasum3, areasum4,
            areasum5, areasum6, areasum7, areasum8, areasum9, areasum10, areasum11,
            areasum12, areasum13, areasum14, areasum15, areasum16, areasum17, areasum18,
            areasum19, areasum20, areasum21, areasum22, areasum23, areasum24, areasum25};
```

```
In[56]:= energylist = {energyx40, energyx41, energyx42, energyx43, energyx44, energyx45,
            energyx46, energyx47, energyx48, energyx49, energyx410, energyx411, energyx412,
            energyx413, energyx414, energyx415, energyx416, energyx417, energyx418, energyx419,
            energyx420, energyx421, energyx422, energyx423, energyx424, energyx425};
```

It can be seen that each of the 26 elements of *x4list* contain 8399 elements. These elements are points which consist of two numbers each.

```
In[57]:= Length[x4list[[24]]]

Out[57]= 8399
```

Use the normalization constants, *areasumlist*, to rescale *x4list* into a lists of bins with normalized area sums.

```
In[58]:= psinorm = Table[{x4list[[n, x, 1]], Sqrt[1 / areasumlist[[n]]] * x4list[[n, x, 2]]},
            {n, 1, 26}, {x, 1, Length[x4list[[n]]]}];
```

Note that the area of the bin x for the n$^{th}$ eigenstate, represented by *x4norm*[[n,i]], has an area equal to the height, *x4norm*[[n,i,2]], multiplied by the bin width, h=0.001.

```
In[59]:= ListPlot[psinorm]
```

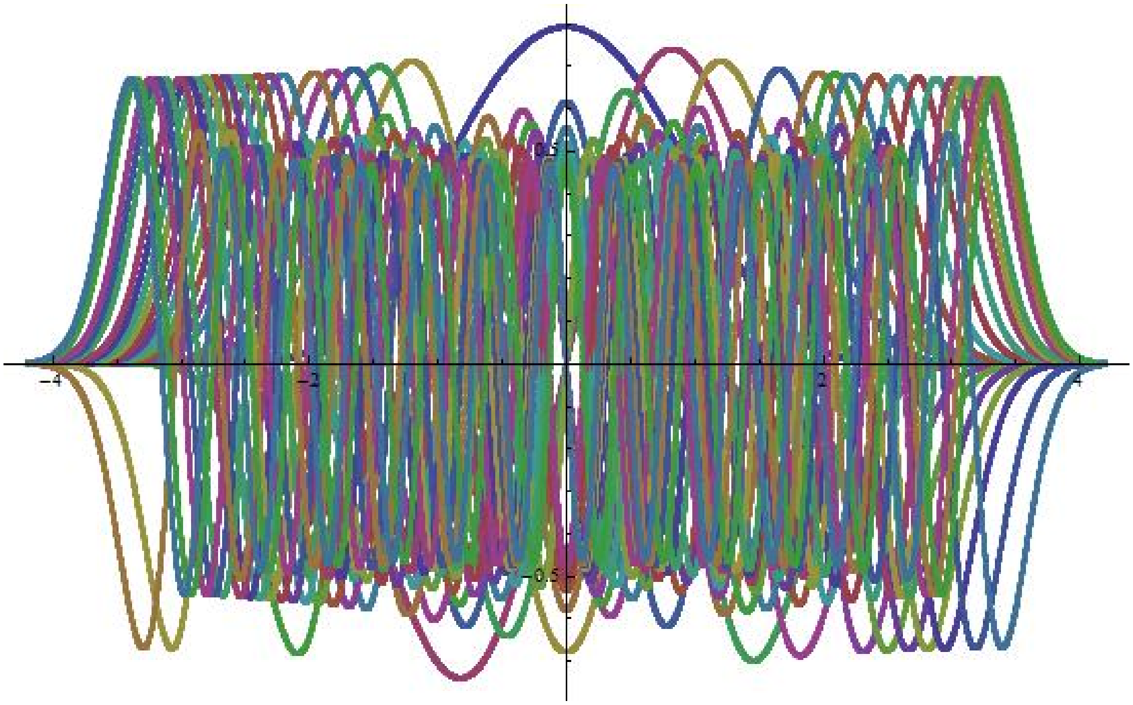


Figure 20: All 26 normalized eigenstates of the unperturbed potential, the quartic well. Visually one can sense a filling or completeness in function space, all though certainly not in any rigorous way. Comparison to Fig. 22 is interesting shows the effects of the first order perturbation.

Define a sum function which will take the place of an integral when taking the inner product between two of the binned eigenstates.

```
In[60]:= innerproduct[n_, m_] :=
          Sum[.001 * psinorm[[n, x, 2]] * psinorm[[m, x, 2]], {x, 1, Length[psinorm[[n]]]}]
```

Define the perturbation. Note that *psinorm*[[1,x,1]] takes the place of x.

```
In[61]:= perturbation[x_] := -lambda * psinorm[[1, x, 1]]^2;

In[62]:= perturbationE[x_] := -λ * psinorm[[1, x, 1]]^2;
```

Check that the inner product of each eigenstate with itself is unity.

```
In[63]:= Table[innerproduct[w, w], {w, 1, 26}]

Out[63]= {1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.,
          1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.}
```

The inner product of any two eigenstates n≠m should yield zero. To verify this it is sufficient to find the maximal distance from zero of the inner products for which n≠m. First a list of all the inner sum values is created.

```
In[64]:= innerproductlist = Flatten[Table[innerproduct[n, m], {n, 1, 26}, {m, 1, 26}]];
```

The maximum should be unity, the inner product value when n=m.

```
In[65]:= Max[innerproductlist]

Out[65]= 1.
```

Now drop the diagonal entries, i.e. those for which n=m, and take the maximum of the resulting list. The returned value should be close to zero, relative say, unity.

```
In[66]:= Max[Drop[innerproductlist, {1, 676, 27}]]

Out[66]= 6.26837 × 10^-6
```

Now that the normalized data has been organized and checked, perturbation theory can be implemented. First determine a single correction, the correction on the ground state due to the second excited state.

```
In[67]:= c13 = Sum[.001 * psinorm[[1, x, 2]] * psinorm[[3, x, 2]] * perturbation[x],
            {x, 1, Length[psinorm[[1]]]}] / (energylist[[1]] - energylist[[3]])

Out[67]= 0.293019

In[68]:= firstwithcorrection = Table[{psinorm[[1, x, 1]], psinorm[[1, x, 2]] + c13 * psinorm[[3, x, 2]]},
            {x, 1, Length[psinorm[[1]]]}];

In[69]:= ListPlot[firstwithcorrection]
```

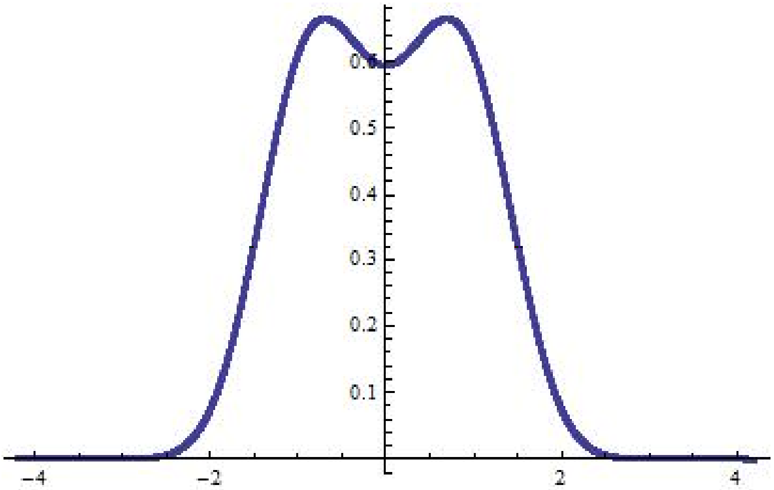


Figure 21: Ground state with first order correction from second excited state.

The first order correction consists of a sum of corrections like the one above, one for each eigenstate of the same parity, less itself. Only the first 26 eigenstates are considered here so each correction sum should have $\frac{26}{2} - 1 = 12$ terms. To find the first order correction to the eigenstates the constants, $c_{\mathrm{nm}}$, must be determined, where $c_{\mathrm{nm}}$ denotes the strength of the correction on the n$^{\mathrm{th}}$ eigenstate due to the m$^{\mathrm{th}}$ eigenstate. Note that this next line of code generates an error, the error however is only for the elements where n=m and division by zero takes place. If none of the elements with n=m are called this will not be a problem. The error output has been suppressed.

```
In[70]:= cnm = Table[Sum[.001 * psinorm[[n, x, 2]] * psinorm[[m, x, 2]] * perturbation[x], {x, 1,
                Length[psinorm[[1]]]}] / (energylist[[n]] - energylist[[m]]), {n, 1, 26}, {m, 1, 26}];
```

A list of the eigenstates corrected to first order can now be created.

```
In[71]:= psiwithcorrection = Table[Table[{psinorm[[1, x, 1]], psinorm[[n, x, 2]] +
                Sum[cnm[[n, Mod[n + 2 * j, 26, 1]]] * psinorm[[Mod[n + 2 * j, 26, 1], x, 2]], {j, 1, 12}]},
             {x, 1, Length[psinorm[[1]]]}], {n, 1, 26}];
```

These are not normalized anymore, they must be normalized again:

```
In[72]:= areasumlist2 = Table[Sum[.001 * psiwithcorrection[[n, x, 2]]^2,
             {x, 1, Length[psiwithcorrection[[1]]]}], {n, 1, 26}];

In[73]:= psifirstorder = Table[{psiwithcorrection[[n, x, 1]],
             (Sqrt[1 / areasumlist2[[n]]]) * psiwithcorrection[[n, x, 2]]},
            {n, 1, 26}, {x, Length[psiwithcorrection[[n]]]}];
```

Plotting the corrected eigenstates shows that they all are pushed away from x=0. Only the 24th and 25th excited states violate this trend and instead clump towards x=0, however, these are respectively the highest even and odd eigenstates considered and thus have no higher order eigenstates to correct them, they can be considered  anomalies and shouldn’t be worried about, also, they shouldn’t be used as they are clearly invalid, they are not shown in the below plot.

```
In[74]:= ListPlot[Drop[psifirstorder, -2]]
```

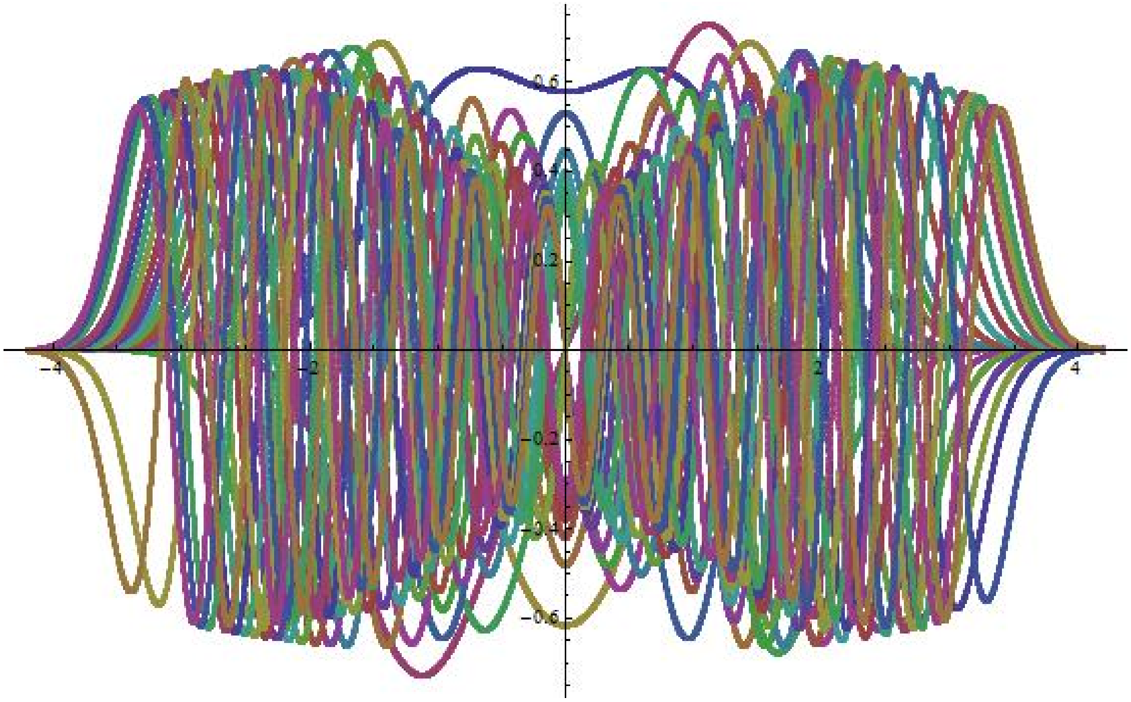


Figure 22: The first 24 eigenstates corrected to first order. In comparison to Fig. 20 it is seen that the corrected eigenstates here avoid the center of the well.

The corresponding energies corrected to first order are comparatively easy to find. The energies to first order are now calculated. The output of *Eofλfirstorder* is suppressed. A truncated version of the output is given in the second and third columns of Table 1, *Appendix C.*

```
In[75]:= Eofλfirstorder =
           Table[energylist[[n]] + Sum[.001 * psinorm[[n, x, 2]] * psinorm[[n, x, 2]] * perturbationE[x],
               {x, 1, Length[psinorm[[1]]]}], {n, 1, Length[energylist]}] // TableForm;
```

The eigenstates to first order can be plotted against their unperturbed counterparts. The output from the next line of code is suppressed but the line was used to create Figs. 3 and 5.

```
In[76]:= Show[ListPlot[psinorm[[1]], PlotStyle → {Blue, PointSize[.007]}],
           ListPlot[psifirstorder[[1]], PlotStyle → {Red, PointSize[.007]}],
           Graphics [ Text[ Style["λ = 1", 30], {3, 0.6}]],
           AxesLabel → {Style["x", 30], Style["ψ13", 30]}, TicksStyle → Directive[20]];
```

The second order corrections are also easily computable but the summation notation becomes cumbersome. First the correction on the n$^{th}$ state due to the m$^{th}$ state is computed. Then these corrections are added up along with the corresponding energies to first order to arrive at the energies to second order. Note that this next line of code generates an error, the error however is only for the elements where n=m and division by zero takes place. If none of the elements with n=m are called this will not be a problem. The error output has been suppressed.

```
In[77]:= Ecorrectionnm =
           Table[Sum[.001 * (psinorm[[n, x, 2]] * psinorm[[m, x, 2]] * perturbationE[x]) ^2, {x, 1,
               Length[psinorm[[1]]]}] / (energylist[[n]] - energylist[[m]]), {n, 1, 26}, {m, 1, 26}];
```

The energies to second order are now calculated. The output of *Eofλsecondorder* is suppressed. A truncated version of the output is given in the second, third, and fourth columns of Table 1, *Appendix C*.

```
In[78]:= Eofλsecondorder =
           Table[Eofλfirstorder[[n]] + Sum[Ecorrectionnm[[n, Mod[n + 2 * j, 26, 1]]], {j, 1, 12}],
              {n, 1, 26}][[1]] // TableForm;
```

Thus it makes sense to define some more user friendly commands. Mathematica has no conditional sum statement built in where one can say sum over all j≠i but modulus statements can suffice. Functions are cumbersome because they are re-evaluated each time they are called, lists need only be evaluated once. An example of a function that could be used is given below, followed by its list equivalents. One list is made for each definition of the perturbation.

```
In[79]:= innerproductfunc[psinorm_, n_, m_, power_] :=
          Sum[.001 * (psinorm[[n, x, 2]] * perturbation[x] * psinorm[[m, x, 2]]) ^power,
           {x, 1, Length[psinorm[[1]]]}]

In[80]:= innerproductH = Table[Sum[.001 * (psinorm[[n, x, 2]] * perturbation[x] * psinorm[[m, x, 2]]),
             {x, 1, Length[psinorm[[1]]]}], {n, 1, 26}, {m, 1, 26}];

In[81]:= innerproductHE = Table[Sum[.001 * (psinorm[[n, x, 2]] * perturbationE[x] * psinorm[[m, x, 2]]),
             {x, 1, Length[psinorm[[1]]]}], {n, 1, 26}, {m, 1, 26}];
```

Define the differences between the n$^{th}$ and m$^{th}$ energies.

```
In[82]:= Edifference = Table[energylist[[n]] - energylist[[m]], {n, 1, 26}, {m, 1, 26}];
```

The next line of code sets all of the energy difference for which n = m to infinity. Since these elements enter in the denominator of the perturbative corrections they will make any term in which they appear equal to zero. This is done because perturbative corrections for which n = m should not be used.

```
In[83]:= Do[Edifference[[n, n]] = ∞, {n, 26}]
```

Next the eigenfunctions to second order are computed.

```
In[84]:= c1nml = Table[(innerproductH[[m, l]] * innerproductH[[l, n]]) /
             (Edifference[[n, m]] * Edifference[[n, l]]), {n, 1, 26}, {m, 1, 26}, {l, 1, 26}];

In[85]:= c2nm = Table[(innerproductH[[n, n]] * innerproductH[[m, n]]) / (Edifference[[n, m]]^2),
           {n, 1, 26}, {m, 1, 26}];

In[86]:= c3nm = Table[(innerproductH[[n, m]] * innerproductH[[m, n]]) / (Edifference[[m, n]]^2),
           {n, 1, 26}, {m, 1, 26}];

In[87]:= psiwithcorrection2 = Table[{psinorm[[1, x, 1]], psifirstorder[[n, x, 2]]
              + Sum[Sum[c1nml[[n, Mod[n + 2 * m, 26, 1], Mod[n + 2 * l, 26, 1]]], {l, 1, 12}] *
                 psinorm[[Mod[n + 2 * m, 26, 1], x, 2]], {m, 1, 12}]
              - Sum[c2nm[[n, Mod[n + 2 * m, 26, 1]]] * psinorm[[m, x, 2]], {m, 1, 12}]
              - (1 / 2) * Sum[c3nm[[n, Mod[n + 2 * m, 26, 1]]] * psinorm[[n, x, 2]], {m, 1, 12}]
           }, {n, 1, 26}, {x, 1, Length[psinorm[[1]]]}];
```

These eigenfunctions must be normalized.

```
In[88]:= areasumlist3 = Table[Sum[.001 * psiwithcorrection2[[n, x, 2]]^2,
             {x, 1, Length[psiwithcorrection2[[1]]]}], {n, 1, 26}];
```

```
In[89]:= psisecondorder = Table[{psiwithcorrection2[[n, x, 1]],
             (Sqrt[1 / areasumlist3[[n]]]) * psiwithcorrection2[[n, x, 2]]},
            {n, 1, 26}, {x, Length[psiwithcorrection2[[n]]]}];
```

The eigenstates to second order can be plotted against their unperturbed and first order counterparts. The output from the next line of code is suppressed but the line was used to create Figs. 1,2,4,6-9,12-14, and 16-18.

```
In[90]:= Show[ListPlot[psinorm[[1]], PlotStyle → {Blue, PointSize[.007]}],
           ListPlot[psifirstorder[[1]], PlotStyle → {Red, PointSize[.008]}],
           ListPlot[psisecondorder[[1]], PlotStyle → {Green, PointSize[.007]}],
           Graphics [ Text[ Style["λ = 0.1", 30], {.133, 0.7903}]],
           AxesLabel → {Style["x", 30], Style["ψ0", 30]},
           TicksStyle → Directive[20], PlotRange → {{-3, 3}, {0, 0.85}}];
```

The energies to third order are now calculated. The output of *Eofλthirdorder* is suppressed. A truncated version of the output is given in the second, third, fourth, and fifth columns of Table 1, *Appendix C*.

```
In[91]:= Ecorrection1nml = Table[
            (innerproductHE[[n, l]] * innerproductHE[[l, m]] * innerproductHE[[m, n]]) /
             (Edifference[[l, n]] * Edifference[[m, n]]), {n, 1, 26}, {m, 1, 26}, {l, 1, 26}];

In[92]:= Ecorrection2nm =
           Table[(innerproductHE[[n, m]]^2) / (Edifference[[m, n]]^2), {n, 1, 26}, {m, 1, 26}];

In[93]:= Ethirdorder = Table[Eofλsecondorder[[n]]
                 + Sum[Sum[Ecorrection1nml[[n,
                       Mod[n + 2 * m, 26, 1], Mod[n + 2 * l, 26, 1]]], {l, 1, 12}], {m, 1, 12}]
                 - innerproductHE[[n, n]] * Sum[Ecorrection2nm[[n, Mod[n + 2 * m, 26, 1]]], {m, 1, 12}],
               {n, 1, 26}][[1]] // TableForm;
```

## APPENDIX C: TABLES

Large tables of data are given here rather than in the text. Table numbers correspond to where they are first referenced in the paper.

| Eigenstate, n | Unperturbed Eigenvalue | 1st Correction to Eigenvalue | 2nd Correction to Eigenvalue | 3rd Correction to Eigenvalue |
|---|---|---|---|---|
| 0 | 0.53018 | $-0.36202\ \lambda$ | $-0.06218\ \lambda^2$ | $-0.020781\ \lambda^3$ |
| 1 | 1.89983 | $-0.90161\ \lambda$ | $-0.17419\ \lambda^2$ | $-0.021970\ \lambda^3$ |
| 2 | 3.72784 | $-1.24471\ \lambda$ | $-0.26128\ \lambda^2$ | $-0.003835\ \lambda^3$ |
| 3 | 5.82237 | $-1.55791\ \lambda$ | $-0.33261\ \lambda^2$ | $-0.004737\ \lambda^3$ |
| 4 | 8.13091 | $-1.84161\ \lambda$ | $-0.38335\ \lambda^2$ | $-0.003915\ \lambda^3$ |
| 5 | 10.61918 | $-2.10501\ \lambda$ | $-0.43077\ \lambda^2$ | $-0.003441\ \lambda^3$ |
| 6 | 13.26423 | $-2.35286\ \lambda$ | $-0.45752\ \lambda^2$ | $-0.003082\ \lambda^3$ |
| 7 | 16.04929 | $-2.58830\ \lambda$ | $-0.48969\ \lambda^2$ | $-0.002803\ \lambda^3$ |
| 8 | 18.96150 | $-2.81347\ \lambda$ | $-0.49548\ \lambda^2$ | $-0.002579\ \lambda^3$ |
| 9 | 21.99057 | $-3.02998\ \lambda$ | $-0.51589\ \lambda^2$ | $-0.002396\ \lambda^3$ |
| 10 | 25.12812 | $-3.23900\ \lambda$ | $-0.50021\ \lambda^2$ | $-0.002241\ \lambda^3$ |
| 11 | 28.36710 | $-3.44149\ \lambda$ | $-0.51050\ \lambda^2$ | $-0.002110\ \lambda^3$ |
| 12 | 31.70152 | $-3.63819\ \lambda$ | $-0.47020\ \lambda^2$ | $-0.001996\ \lambda^3$ |
| 13 | 35.12619 | $-3.82971\ \lambda$ | $-0.47105\ \lambda^2$ | $-0.001896\ \lambda^3$ |
| 14 | 38.63660 | $-4.01655\ \lambda$ | $-0.39979\ \lambda^2$ | $-0.001808\ \lambda^3$ |
| 15 | 42.22873 | $-4.19914\ \lambda$ | $-0.39119\ \lambda^2$ | $-0.001729\ \lambda^3$ |
| 16 | 45.89903 | $-4.37785\ \lambda$ | $-0.27734\ \lambda^2$ | $-0.001659\ \lambda^3$ |
| 17 | 49.64430 | $-4.55298\ \lambda$ | $-0.25857\ \lambda^2$ | $-0.001595\ \lambda^3$ |
| 18 | 53.46165 | $-4.72481\ \lambda$ | $-0.07989\ \lambda^2$ | $-0.001536\ \lambda^3$ |
| 19 | 57.34845 | $-4.89356\ \lambda$ | $-0.04930\ \lambda^2$ | $-0.001483\ \lambda^3$ |
| 20 | 61.30231 | $-5.05946\ \lambda$ | $0.24239\ \lambda^2$ | $-0.001243\ \lambda^3$ |
| 21 | 65.32103 | $-5.22267\ \lambda$ | $0.28825\ \lambda^2$ | $-0.001187\ \lambda^3$ |
| 22 | 69.40257 | $-5.38338\ \lambda$ | $0.82568\ \lambda^2$ | $0.035154\ \lambda^3$ |
| 23 | 73.54506 | $-5.54172\ \lambda$ | $0.89485\ \lambda^2$ | $0.037169\ \lambda^3$ |
| 24 | 77.74675 | $-5.69783\ \lambda$ | $2.31919\ \lambda^2$ | $0.009524\ \lambda^3$ |
| 25 | 82.00602 | $-5.85183\ \lambda$ | $2.44070\ \lambda^2$ | $0.008182\ \lambda^3$ |

Table 1: Unperturbed and corrected eigenvalues for each eigenstate as functions of $\lambda$. Adding up each row gives an equation for each eigenvalue to third order in $\lambda$. These values are output from the Mathmatica program with more accuracy but have been truncated to fit here.

| $\lambda$ | (1st Correction – Numerov) / $\lambda$ | (2nd Correction – Numerov) / $\lambda^2$ | (3rd Correction – Numerov) / $\lambda^3$ |
|---|---|---|---|
| 0.1 | 0.071136 | 0.089522 | 0.687404 |
| 0.2 | 0.073085 | 0.054501 | 0.168596 |
| 0.3 | 0.075904 | 0.045732 | 0.083169 |
| 0.4 | 0.078161 | 0.039942 | 0.047902 |
| 0.5 | 0.081367 | 0.038365 | 0.035168 |
| 0.6 | 0.084400 | 0.037027 | 0.027076 |
| 0.7 | 0.087697 | 0.036447 | 0.022380 |
| 0.8 | 0.091248 | 0.036330 | 0.019436 |
| 0.9 | 0.095093 | 0.036565 | 0.017538 |
| 1 | 0.099262 | 0.037078 | 0.016297 |
| 1.1 | 0.103790 | 0.037824 | 0.015493 |
| 1.2 | 0.108716 | 0.038776 | 0.014996 |
| 1.3 | 0.114060 | 0.039904 | 0.014710 |
| 1.4 | 0.119871 | 0.041205 | 0.014588 |
| 1.5 | 0.126173 | 0.042659 | 0.014585 |
| 1.6 | 0.132989 | 0.044253 | 0.014670 |
| 1.7 | 0.140329 | 0.045967 | 0.014815 |
| 1.8 | 0.148186 | 0.047779 | 0.014999 |
| 1.9 | 0.156533 | 0.049657 | 0.015198 |

Table 4: Scaled distances between the ground state eigenvalues from perturbation theory and those from the Numerov method. Each column should consist of elements of a single order. The formulas used to compute these values are shown at the top of each column. This is a point-wise comparison.

| $\lambda$ | (1st Correction – Numerov) / $\lambda$ | (2nd Correction – Numerov) / $\lambda^2$ | (3rd Correction – Numerov) / $\lambda^3$ |
|---|---|---|---|
| 0.1 | 0.117155 | 0.549711 | 0.687404 |
| 0.2 | 0.119505 | 0.286605 | 0.168596 |
| 0.3 | 0.121928 | 0.199147 | 0.083169 |
| 0.4 | 0.124442 | 0.155644 | 0.047902 |
| 0.5 | 0.126895 | 0.129422 | 0.035168 |
| 0.6 | 0.129764 | 0.112632 | 0.027076 |
| 0.7 | 0.132581 | 0.100567 | 0.022380 |
| 0.8 | 0.135509 | 0.091656 | 0.019436 |
| 0.9 | 0.138551 | 0.084853 | 0.017538 |
| 1 | 0.141708 | 0.079524 | 0.016297 |
| 1.1 | 0.144926 | 0.075220 | 0.015493 |
| 1.2 | 0.148398 | 0.071845 | 0.014996 |
| 1.3 | 0.151899 | 0.069012 | 0.014710 |
| 1.4 | 0.155587 | 0.066716 | 0.014588 |
| 1.5 | 0.159372 | 0.064792 | 0.014585 |
| 1.6 | 0.163283 | 0.063186 | 0.014670 |
| 1.7 | 0.167316 | 0.061842 | 0.014815 |
| 1.8 | 0.171481 | 0.060720 | 0.014999 |
| 1.9 | 0.175753 | 0.059773 | 0.015198 |

Table 5: Scaled distances between the first excited state eigenvalues from perturbation theory and those from the Numerov method. Each column should consist of elements of a single order. The formulas used to compute these values are shown at the top of each column. This is a point-wise comparison.