Special Functions - Harmonic Oscillator and Hermite Polynomials

Christian B. Molina

March 9, 2021

Abstract

The harmonic oscillator in quantum mechanics is used to describe the oscillatory behavior of a particle within a smooth well of potential energy, and is the quantum-mechanical analog of the classical harmonic oscillator. The one-dimensional harmonic oscillator is a simplified model describing the probability of finding a particle within a potential well based on its excitation levels. A particle at its ground state would likely be found at the bottom of the potential well, but as energy levels increase, probability density begins to peak where the excitation levels coincides with the potential energy.
At higher excitation levels, the quantum harmonic oscillator begins to reflect the behavior of the classical harmonic oscillator. This is considered the correspondence principle, in which "overall picture of probability of finding the oscillator at a given value of x converges for the quantum and classical pictures"[1]

In this report, we will generate solutions to the one dimensional harmonic oscillator using Hermite polynomials, then observe the probability density wave function as excitation levels increase to reflect the correspondence principle

1 Introduction

1.1 The One Dimensional Harmonic Oscillator

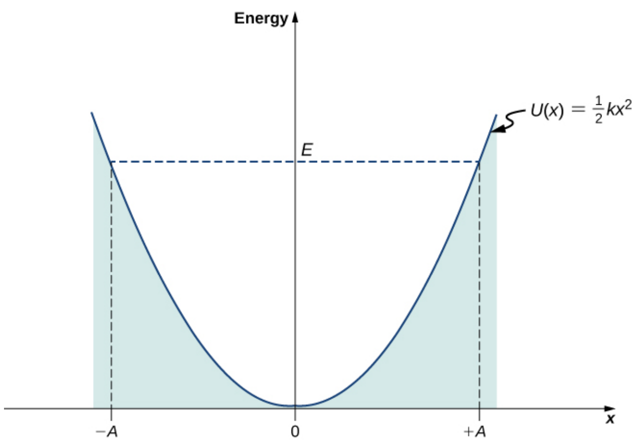

Figure 1: The potential energy well of a classical harmonic oscillator[2]

To illustrate the fundamental aspect of both classical and quantum mechanics, the potential for the one-dimensional harmonic oscillator is in the form of:

$$V_{h.o}(x) = \frac{1}{2}\mu\omega^2x^2 \tag{1}$$

where μ is the mass and ω is the angular frequency of the oscillator. This potential is ideal to model a particle trapped at the bottom of the well. The Hamiltonian of a particle of mass μ moving in the potential well of equation (1) can then be described as:

$$H = \frac{1}{2\mu}p_x^2 + \frac{1}{2}\mu\omega^2 x^2 \tag{2}$$

The Hamiltonian defines the total energy of the particle, the first term describes its kinetic energy, where p_x is its linear momentum, and the second term describes its potential energy. In order to solve the quantum mechanical solution to the problem, the Hamiltonian must be expressed as an operator:

$$H = -\frac{\hbar^2}{2\mu}\frac{d^2}{dx^2} + \frac{1}{2}\mu\omega^2 x^2 \tag{3}$$

where p_x is replaced with $(\hbar/\mathrm{i})(\mathrm{d}/\mathrm{dx})$

With the Hamiltonian operator, we can define the time-independent Schrodinger Equation:

$$H\psi(x) = E\psi(x) \tag{4}$$

that is equivalent to the second order differential equation:

$$-\frac{\hbar^2}{2\mu^2}\frac{d^2\psi}{dx^2} + (\frac{1}{2}\mu\omega^2 x^2 - E)\psi(x) = 0 \tag{5}$$

If we measure energy in units of $\frac{1}{2}(\hbar\omega)$

$$E \longrightarrow \lambda = \frac{E}{\frac{1}{2}\hbar\omega} \tag{6}$$

and express length in terms of the dimensionless quantity:

$$\rho = \sqrt{\frac{\mu\omega}{\hbar}}x \tag{7}$$

the differential equation then becomes

$$\frac{d^2\psi}{d\rho^2} + (\lambda - \rho^2)\psi(\rho) = 0 \tag{8}$$

Dropping the factor of $1/2(\hbar\omega)$

1.2 Hermite Polynomials

Solutions to equation (5) exists only under certain values of E. Those values can be generalized as:

$$E_n = (2n+1)\frac{1}{2}\hbar\omega \qquad\qquad n = 0, 1, ... \tag{9}$$

Those values of E generate the eigenfunction, a solution for the given values of n:

$$\psi_n(\rho) = \frac{1}{\sqrt{2^n n!\sqrt{\pi}}}e^{-\rho^2/2}H_n(\rho) \tag{10}$$

$H_n(\rho)$ is the Hermite polynomial of degree n.

By substituting equation (10) into equation (7), we can see that the Hermite polynomials are the solution of:

$$\frac{d^2}{d\rho^2}H_n(\rho) - 2\rho\frac{d}{d\rho}H_n(\rho) + (\lambda - 1)H_n(\rho) = 0 \tag{11}$$

The Hermite polynomials may be expressed as a power series consisting of either even or odd powers of ρ

$$H_n(\rho) = \sum_k^n a_{n,k}\rho^k \tag{12}$$

We can generate higher order of polynomials with the recursion relation:

$$H_{n+1}(\rho) = 2_n(\rho) - 2nH_{n-1}(\rho) \tag{13}$$

knowing the following lowest order polynomials:

$$\begin{aligned} H_0(\rho) &= 1 & H_1(\rho) &= 2\rho \\ H_2(\rho) &= 4\rho^2 - 2 & H_3(\rho) &= 8\rho^3 - 12\rho \\ H_2(\rho) &= 16^4 - 48\rho^2 + 12 & H_5(\rho) &= 32^5 - 160\rho^3 + 120\rho \end{aligned} \tag{14}$$

For this report, equation (13) will be simplified to:

$$a_{n+1,k} = 2a_{n,k-1} - 2na_{n-1,k} \tag{15}$$

to find the coefficients of (n+1) for a given power of k

2 Methods

2.1 Algorithm: Hermite Coefficients

The following algorithm is based on equation (15). Due to the recursive nature of equation (15), previously calculated values would have to be stored - in this case using an $n \times n$ array to index back previous coefficients for certain orders of k.

Algorithm 1: Hermite Coefficients

```
Result: Generate coefficients of Hermite polynomials up to the k-th order
narray: n x n array filled with zeroes;
korder = value of k-th order;
narray[0][0] = 1;
narray[1][1] = 2;
while i = 2 ≤ n, i++ do
    while j = 0 ≤ korder, j++ do
        narray[i][j] = 2*narray[i][j − 1] − 2*n*narray[i − 1][j]
    end
end
```

Initialization begins by creating an $n \times n$ array filled with zeros, then setting the values of the 0-th and 1st Hermite polynomial as seen from equation (14). The rows and columns of *narray* represent the order of the polynomial and the term respectively. korder is also initialized to determine how many iterations of the following while loops will occur.

Two nested while loops are required to index through the $n \times n$ array. The first while loop indexes the rows of *narray*, and the second while loop indexes the columns. The while loop indexing the rows begins at i = 2, since the coefficients of the 0-th and 1st Hermite polynomials are defined. Once the while loop indexing through the columns is complete, i and j is iterated by 1 *(i++, j++)*, and begins indexing through the next set of columns.

narray	col 0	col 1	col 2	...	col n
row 0	1	0	0	...	0
row 1	0	2	0	...	0
row 2	-2	0	4	...	0
$\vdots$	...	...	...	...	...
row n	$a_{n,0}$	$a_{n,1}$	$a_{n,2}$	...	$a_{n,k}$

Figure 2: $n \times n$ array representing calculated coefficients of the Hermite Polynomials

In Figure 2, the first three orders of the Hermite polynomials are illustrated. The 0-th column represents constants, and the 1st column represents coefficients of terms ρ^1 and as the columns progress they represent ρ^k.

2.2 Algorithm: Generating Hermite Polynomials

Once the Hermite coefficients are generated, the Hermite polynomials can be generated following the logic of equation (12). The following algorithm produces the Hermite polynomials by recursively adding each term of each Hermite polynomial to each other.

Algorithm 2: Hermite Polynomials

Result: Generate Hermite Polynomials
xarray: *array of arbitrary x-values*;
narray: *n x n array with calculated coefficients*;
korder = value of k-th order;
n = n-th hermite polynomial;
hermiteSum = 0;
while $k = 0 \leq korder, k++$ **do**
| hermiteSum += narray$[n][k]*$ power(xarray,k);
end
if *n equals 0* **then**
| return hermiteSum / power(1,3);
else
| return hermiteSum / power(n,3);
end

Algorithm 2 generates the Hermite polynomial of n-th value individually. This functionality allows the user to individually select a Hermite polynomial to generate - a loop can later on be used to generate a range of Hermite polynomials. "$+=$" is an iterative shortcut that adds the previous value with the current value. Algorithm 2 also includes an if-statement for the condition of n = 0. If n = 0, then the sum is multiplied by a factor of $1/1^3$, else then a factor of $1/n^3$ is applied - if this condition is applied to n = 0, then the 0-th Hermite polynomial is undefined.

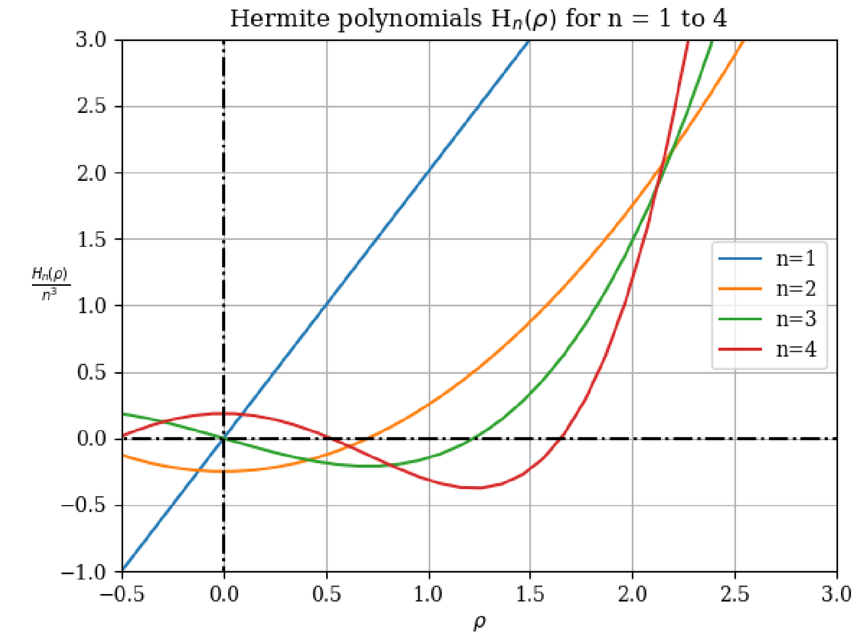

Figure 3: Hermite Polynomials multiplied by a factor of $1/n^3$

3 Discussion

Figure 3 illustrates first four Hermite polynomials. This illustrates the contribution to the general solution to the wave equation.
Since the Hermite polynomials are generated, equation (10) can be completed to plot the following the following wave functions:

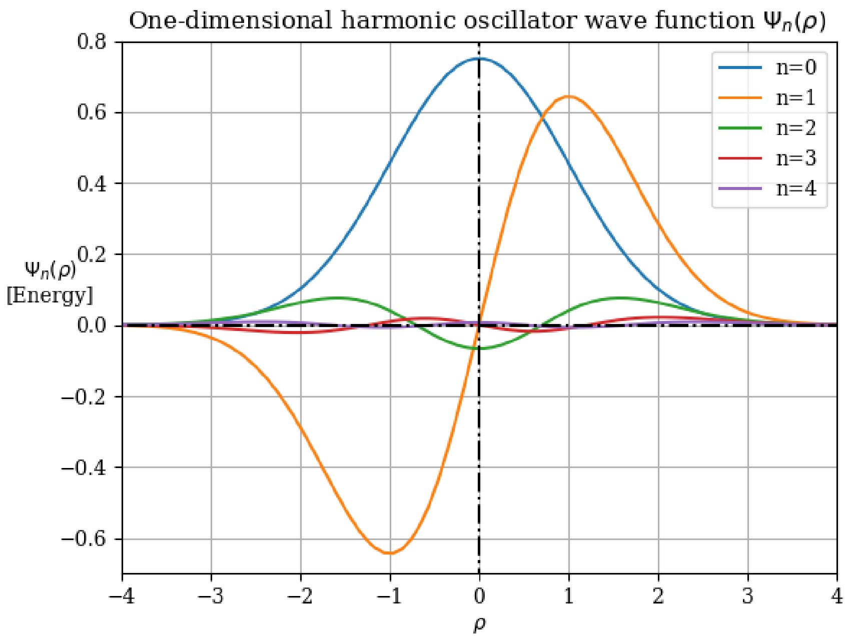

Figure 4: Wave equations for n = 0 to n = 4

Figure 4 illustrates the harmonic motion of a particle between a smooth potential at certain excitation states.

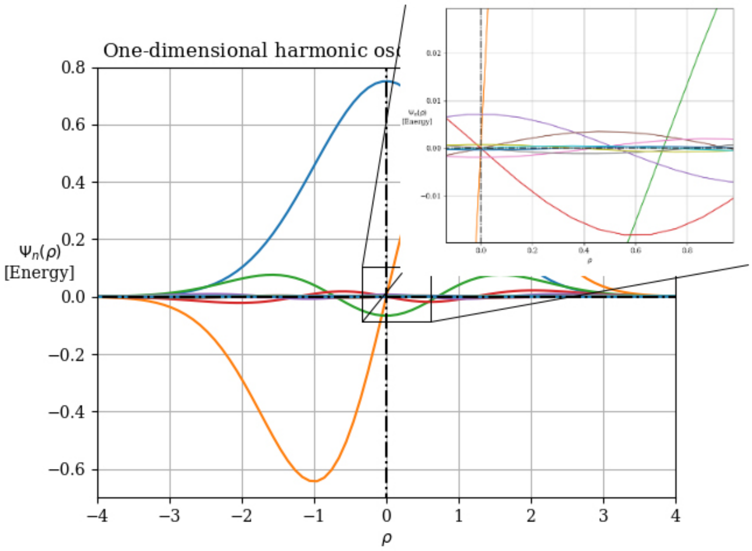

Figure 5: Wave equations for n = 0 to n = 10, zoomed in

It is worth noting that as excitation levels increase, the wave equation becomes discernible, as seen in figure 5. But if the wave equations of higher excitation states are isolated and compared to one another - as shown in figure 6 below - the correspondence principle becomes evident:

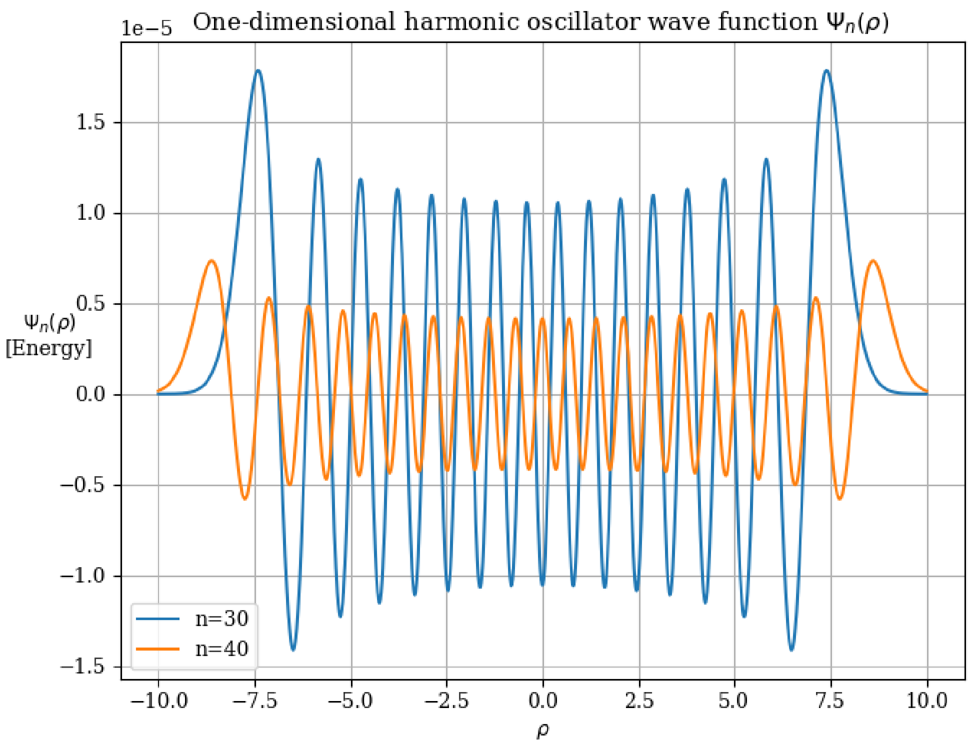

Figure 6: Wave equations for n = 30 and n = 40

4 Conclusion

Solutions to the one-dimensional quantum harmonic oscillator using Hermite Polynomials is created using algorithm 1 and 2. Algorithm 1 and 2 simulate the recursive calculation outlined by equation (13). In figure 6 plotting the wave equations of particles of higher excitation states illustrates the probability wave distribution following the correspondence principle - in which the probability of finding the oscillator at a value x begins to follow the classical system.

References

[1] *http://hyperphysics.phy-astr.gsu.edu/hbase/quantum/hosc6.htmlc1.*

[2] S.S.M Wong.*Computational Methods in Physics and Engineering*