The quantum harmonic oscillator with two electrons

Greg von Winckel

August 8, 2014

1 Problem statement

The quantum harmonic oscillator is one of the standard problems covered in all introductory courses in quantum mechanics. It is one of the examples which where the eigenfunctions are known exactly and the operator can be factored to give the interesting representation using ladder operators. In this note, we will look at how to numerically compute the energy levels of the quantum harmonic oscillator when it contains two interacting electrons in one spatial dimension, with the usual quadratic potential. The time independent Schrödinger equation for the system is

$$\left\{-\frac{1}{2}(\partial_{q_1}^2+\partial_{q_2}^2)+\frac{1}{2}(q_1^2+q_2^2)+\frac{1}{|q_1-q_2|}\right\}\psi(q_1,q_2)=\lambda\psi(q_1,q_2) \tag{1}$$

The electronic wavefunction must satisfy the antisymmetry property

$$\psi(q_1,q_2)=-\psi(q_2,q_1) \tag{2}$$

Which is to say that the solution must be antisymmetric about the line $q_1=q_2$ and must be identically zero along that line. First we use a change of coordinates (rotation).

$$\begin{pmatrix}x\\y\end{pmatrix}=\frac{1}{\sqrt{2}}\begin{pmatrix}1&1\\1&-1\end{pmatrix}\begin{pmatrix}q_1\\q_2\end{pmatrix},\quad \begin{pmatrix}q_1\\q_2\end{pmatrix}=\frac{1}{\sqrt{2}}\begin{pmatrix}1&1\\1&-1\end{pmatrix}\begin{pmatrix}x\\y\end{pmatrix} \tag{3}$$

Using this transformation, we have that the quadratic potential term is

$$(q_1^2+q_2^2)=\left(\frac{x+y}{\sqrt{2}}\right)^2+\left(\frac{x-y}{\sqrt{2}}\right)^2=x^2+y^2 \tag{4}$$

the interaction potential is

$$\frac{1}{|q_1-q_2|}=\frac{1}{|y|} \tag{5}$$

and the kinetic term is

$$\partial_{q_1}^2+\partial_{q_2}^2=\partial_x^2+\partial_y^2 \tag{6}$$

We can rewrite the Schrödinger equation in the new coordinate system as

$$\left\{-\frac{1}{2}(\partial_x^2+\partial_y^2)+\frac{1}{2}(x^2+y^2)+\frac{1}{|y|}\right\}\psi(x,y)=\lambda\psi(x,y) \tag{7}$$

In this coordinate system, the problem is separable. Let $\psi(x,y)=u(x)v(y)$. Substitute this into (7)

$$\frac{v}{2}(-u''+x^2u)+\frac{u}{2}\left(-v''+y^2v+\frac{2}{|y|}v\right)=\lambda uv \tag{8}$$

Dividing by uv and moving all x-dependent terms to the left and all y-dependent terms to the right, we have

$$-\frac{1}{2u}(-u''+x^2u)+\lambda=\frac{1}{2v}\left(-v''+y^2v+\frac{2}{|y|}v\right)=\nu \tag{9}$$

We now have the uncoupled problems

$$\frac{1}{2}(-u''+x^2u)=\mu u \tag{10}$$

$$\frac{1}{2}\left(-v''+y^2v+\frac{2}{|y|}v\right)=\nu v \tag{11}$$

Where $\lambda=\mu+\nu$. The first equation for $u(x)$ is the standard quantum harmonic oscillator problem which has the exact solution eigenpairs μ_j, u_j

$$\frac{1}{2}(-u_j''+x^2u_j)=\mu_ju_j,\quad \mu_j=\frac{j+1}{2},\quad u_j(x)=H_j(x)e^{-x/2} \tag{12}$$

Where $H_j(x)$ is the jth Hermite polynomial. The question is how to compute (approximate) solutions to v-equation.

2 Behavior of solutions

It is clear that the solutions for large y, v equation looks very much like the u equation, so we should expect the asymptotic solution to be a Hermite function. What about the solution for small y?

$$yv''-2v=0 \tag{13}$$

The solutions to the small y problem are

$$v(y)=c_1\sqrt{y}I_1(\sqrt{8y})+c_2\sqrt{y}K_1(\sqrt{8y})=c_1v_1(y)+c_2v_2(y) \tag{14}$$

where I and K are the modified Bessel functions of the first and second kind, respectively. If we analyze the behavior of these two types of solutions as $y\to 0$, we have that

$$\lim_{y\to 0}v_1(y)=0,\quad \lim_{y\to 0}v_2(y)=\frac{1}{\sqrt{8}} \tag{15}$$

Note that for our solution to be antisymmetric, we need $v(0) = 0$, so let $c_2 = 0$. The differential equation looks not very nice due to the singularity and although the v_1 solution is bounded and goes to zero at $y = 0$, it is also sort of nasty looking since it involves square roots and Bessel functions. It is worth trying to study the solution some so that we know what a good numerical method would be to compute the eigenfunctions to the full problem.

We can also obtain the small y solution in series form, using the method of Frobenius. We begin with the Ansatz

$$v(y) = y^r \sum_{k=0}^{\infty} a_k x^k \tag{16}$$

Therefore

$$yv'' = \sum_{k=0}^{\infty} (k+r)(k+r-1) a_k y^{k+r-1} \tag{17}$$

Substituting the series into the differential quation we get

$$r(r-1)a_0 y^{r-1} + \sum_{k=1}^{\infty} [(r+k)(r+k+1)a_k - 2a_{k-1}] y^{k+r-1} = 0 \tag{18}$$

If this is to be zero for all y, it must be the case that $r(r-1) = 0$ so the roots are $r_1 = 1$ and $r_2 = 0$. Since the difference between these roots is an integer, we can expect that one of the solutions may contain a logarithm. We obtain the first solution using r_1 and discover that the coefficients of the expansion must satisfy the recursion relation

$$a_k = \frac{2a_{k-1}}{(k+1)k}, \quad k \geq 1 \tag{19}$$

Choosing $a_0 = 1$, the coefficients may be written in closed form as

$$a_k = \frac{2^k}{(k+1)k!} \tag{20}$$

Using the ratio test

$$\lim_{k\to\infty} \frac{|a_k|}{|a_{k-1}|} = \lim_{k\to\infty} \frac{2}{k^2+k} = 0 \tag{21}$$

we see that our series converges and that the radius of convergence is infinite. This means we have found a "nice" solution to a "nasty" differential equation.

$$v_1(y) = \sum_{k=0}^{\infty} \frac{2^k y^{k+1}}{(k+1)k!} \tag{22}$$

It turns out that $\sqrt{y} I_1(\sqrt{8y})$ a nice function after all. Now let's look at the other solution.

$$v_2(y) = y^{r_2} \sum_{k=0}^{\infty} b_k y^k \tag{23}$$

The b_k have to satisfy the same recursion relation as the a_k

$$b_k = \frac{2}{k(k-1)} b_{k-1}, \quad k \geq 1 \tag{24}$$

However, this fails for $k = 1$, so we must try a solution of the form

$$v_2(y) = cv_1 \log y + \sum_{k=0}^{\infty} b_k y^k \tag{25}$$

It follows that

$$yv_2'' = -c\frac{v_1}{y} + 2cv_1' + cyv_1'' \log y + y\frac{d^2}{dy^2} \sum_{k=0}^{\infty} b_k y^k \tag{26}$$

Using $\mathcal{L} = y\frac{d^2}{dy^2} - 2$

$$v \log y \mathcal{L} v_1 + 2cv_1' - c\frac{v_1}{y} + \sum_{k=0}^{\infty} b_k \mathcal{L} y^k = 0 \tag{27}$$

Since $\mathcal{L}v_1 = 0$, this simplifies to

$$\sum_{k=0}^{\infty} b_k \mathcal{L} y^k = -2cv_1' + c\frac{v_1}{y} \tag{28}$$

which we can write as

$$-c\sum_{k=0}^{\infty} \frac{2^k(2k+1)y^k}{(k+1)k}! = \sum_{k=0}^{\infty} [k(k+1)b_{k+1} - 2b_k] y^k \tag{29}$$

Since this holds for any value of y, the two sums must be equal term by term

$$k(k+1)b_{k+1} - 2b_k = -c\frac{2^k(2k+1)}{(k+1)}k! \tag{30}$$

When $k = 0$, we have $b_0 = \frac{c}{2}$ and we can generate rest of the series. This, however, is not the solution we want. We know that our solution should be antisymmetric about $y = 0$ and analytic.

3 Discretization Method 1

The Galerkin discretization of the eigenvalue problem is

$$\sum_k \left\{ \frac{1}{2}(\phi_j', \phi_k') + \frac{1}{2}(\phi_j, y^2\phi_k) + \left(\phi_j, \frac{1}{|y|}\phi_k\right) \right\} \hat{v}_k = \nu \sum_k \hat{v}_k(\phi_j, \phi_k) \tag{31}$$

Assuming that $\phi_j(y)$ is a polynomial times $e^{-y^2/2}$, these inner products could be computed using Gauss-Hermite quadrature. However, the integrand including $|y|^{-1}$ is not analytic and the Gauss-Hermite rule will converge algebraically. Instead, we will determine closed form expressions for the matrix elements. Suppose we use the Galerkin basis

$$v(y) = \sum_{k=0}^{\infty} \hat{v}_k \phi_k(y), \quad \phi_k(y) = yH_k(y)e^{-y^2/2} \tag{32}$$

The Hermite polynomials have the properties

$$H_0(y) = 1 \tag{33}$$
$$H_1(y) = 2y \tag{34}$$
$$H_{j+1}(y) = 2yH_j(y) - 2jH_{j-1}(y) \tag{35}$$
$$H_j'(y) = 2jH_{j-1}(y) \tag{36}$$
$$\int_{-\infty}^{\infty} H_j(y)H_k(y)e^{-y^2}\,\mathrm{d}y = \gamma_k\delta_{jk}, \quad \gamma_k = \sqrt{\pi}2^k k! \tag{37}$$

3.1 Computing the mass matrix

The mass matrix, which appears on the right-hand-side of the eigenvalue problem has the matrix elements

$$m_{jk} = (\phi_j, \phi_k) = \int_{-\infty}^{\infty} y^2 H_j(y)H_k(y)e^{-y^2}\,\mathrm{d}y \tag{38}$$

Using the recurrence relation

$$yH_j(y) = \frac{1}{2}H_{j+1}(y) + jH_{j-1}(y) \tag{39}$$

Plugging this into the mass matrix formula, we get

$$m_{jk} = \frac{1}{4}\int_{-\infty}^{\infty} [H_{j+1}(y) + 2jH_{j-1}(y)][H_{k+1}(y) + 2kH_{k-1}(y)]e^{-y^2}\,\mathrm{d}y \tag{40}$$

The elements of this matrix are

$$m_{jk} = \begin{cases} \frac{1}{4}\gamma_{j+1} + j^2\gamma_{j-1} & \text{if } k = j \\ \frac{j+2}{2}\gamma_{j+1} & \text{if } k = j+2 \\ \frac{k+2}{2}\gamma_{k+1} & \text{if } j = k+2 \\ \frac{(j+3)(j+4)}{4}\gamma_{j+2} & \text{if } k = j+4 \\ \frac{(k+3)(k+4)}{4}\gamma_{k+2} & \text{if } j = k+4 \\ 0 & \text{otherwise} \end{cases} \tag{41}$$

3.2 Computing the kinetic term

Using these properties, the derivative of the trial function can be written as

$$\phi_j'(x) = \begin{cases} \frac{1}{4}[-H_2(y) + 2H_0(y)]e^{-y^2/2} & \text{if } j = 0 \\ \frac{1}{4}[-H_3(y) + 2H_1(y)]e^{-y^2/2} & \text{if } j = 1 \\ \frac{1}{4}[-H_{j+2}(y) + 2H_j(y) + 4j(j-1)H_{j-2}(y)]e^{-y^2/2} & \text{if } j \geq 2 \end{cases} \tag{42}$$

The matrix elements are $\kappa_{jk} = (\phi_j', \phi_k')$

$$\kappa_{jk} = \begin{cases} \frac{1}{16}[\gamma_{j+2} + 4\gamma_j + 16j^2(j-1)^2\gamma_{j-2}] & \text{if } k = j \\ -\frac{(j+4)(j+3)}{4}\gamma_{j+2} & \text{if } k = j+4 \\ -\frac{(k+4)(k+3)}{4}\gamma_{k+2} & \text{if } j = k+4 \\ 0 & \text{otherwise} \end{cases} \tag{43}$$

3.3 Computing the quadratic term

The quadratic term is

$$(\phi_j, x^2\phi_k) = \int_{-\infty}^{\infty} H_j(y)H_k(y)y^4 e^{-y^2}\,\mathrm{d}y \tag{44}$$

Multiplying the Hermite polynomials by y^2 gives us

$$y^2 H_j(y) = \frac{y}{2}H_{j+1}(y) + jyH_{j-1}(y) = \frac{1}{4}H_{j+2}(y) + \frac{2j+1}{2}H_j(y) + j(j-1)H_{j-2}(y) \tag{45}$$

The matrix elements for the quadratic term are

$$q_{jk} = \begin{cases} \frac{1}{16}\gamma_{j+2} + \left(\frac{2j+1}{2}\right)^2\gamma_j + j^2(j-1)^2\gamma_{j-2} & \text{if } k = j \\ \left(\frac{2j+5}{8}\right)\gamma_{j+2} + \frac{1}{2}(2j+1)(j+1)(j+2)\gamma_j & \text{if } k = j+2 \\ \left(\frac{2k+5}{8}\right)\gamma_{k+2} + \frac{1}{2}(2k+1)(k+1)(k+2)\gamma_k & \text{if } j = k+2 \end{cases} \tag{46}$$

3.4 Computing the Coulombic term

To proceed it would be useful to calculate inner products of Hermite polynomials on $[0, \infty)$. Consider the inner product

$$a_{jk} = \int_0^{\infty} H_j(x)H_k(x)e^{-x^2}\,\mathrm{dx} \tag{47}$$

Using the relation that $H_k(x) = 2xH_{k-1}(x) - 2(k-1)H_{k-2}(x)$, we have

$$a_{jk} = \int_0^{\infty} H_j(x)H_{k-1}(x)(2x)e^{-x^2}\,\mathrm{d}x - 2(k-1)\underbrace{\int_0^{\infty} H_j(x)H_{k-2}(x)\,\mathrm{d}x}_{a_{j,k-2}} \tag{48}$$

Looking more closely at the first integral in the above, we have

$$\int_0^\infty H_j(x)H_{k-1}(x)(2x)e^{-x^2}\,\mathrm{d}x = \tag{49}$$

$$-\int_0^\infty H_j(x)H_{k-1}(x)\frac{d}{dx}e^{-x^2}\,\mathrm{d}x = \tag{50}$$

$$H_j(0)H_{k-1}(0) + \int_0^\infty \frac{d}{dx}[H_j(x)H_{k-1}(x)]e^{-x^2}\,\mathrm{d}x = \tag{51}$$

$$H_j(0)H_{k-1}(0) + \int_0^\infty [H_j'(x)H_{k-1}(x) + H_j(x)H_{k-1}'(x)]e^{-x^2}\,\mathrm{d}x = \tag{52}$$

$$H_j(0)H_{k-1}(0) + \int_0^\infty [H_j'(x)H_{k-1}(x) + H_j(x)H_{k-1}'(x)]e^{-x^2}\,\mathrm{d}x = \tag{53}$$

$$H_j(0)H_{k-1}(0) + \int_0^\infty [2jH_{j-1}(x)H_{k-1}(x) + 2(k-1)H_j(x)H_{k-2}(x)]e^{-x^2}\,\mathrm{d}x = \tag{54}$$

$$H_j(0)H_{k-1}(0) + 2ja_{j-1,k-1} + 2(k-1)a_{j,k-2} \tag{55}$$

Putting it all together,

$$a_{jk} = H_j(0)H_{k-1}(0) + 2ja_{j-1,k-1} + 2(k-1)a_{j,k-2} - 2(k-1)a_{j,k-2} \tag{56}$$

Which simplifies to

$$a_{jk} = H_j(0)H_{k-1}(0) + 2ja_{j-1,k-1} \tag{57}$$

Where the "forcing term" is computed using $H_{k+1}(0) = 2kH_{k-1}(0)$ with the initial values being $H_0(0) = 1$ and $H_1(0) = 0$. In order to generate all of the terms in , we can use the formula above to determine a_{0k} for $k > 0$

$$a_{0k} = H_{k-1}(0) \tag{58}$$

and the first element is

$$a_{00} = \int_0^\infty e^{-x^2}\,\mathrm{d}x \tag{59}$$

The usual trick for evaluating this integral is

$$[a_{00}]^2 = \frac{1}{4}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-x^2-y^2}\,\mathrm{d}x\mathrm{d}y = \frac{1}{2}\int_0^{2\pi}\int_0^{\infty} e^{-r^2}\,r\mathrm{d}r\mathrm{d}\theta = \frac{\pi}{2}\int_0^\infty e^{-r^2}\,r\mathrm{d}r = \frac{\pi}{4} \tag{60}$$

Now we can compute all weighted inner products of Hermite polynomials on $[0,\infty)$ using the recurrence

$$a_{00} = \sqrt{\pi}/2 \tag{61}$$
$$a_{j0} = H_{j-1}(0), \quad \text{for } j > 0 \tag{62}$$
$$a_{0k} = H_{k-1}(0), \quad \text{for } k > 0 \tag{63}$$
$$a_{jk} = H_j(0)H_{k-1}(0) + 2ja_{j-1,k-1}, \quad \text{for } j,k > 0 \tag{64}$$

Now to evaluate the integrals with the Coulomb singularity. The matrix elements of interest are

$$\int_{-\infty}^{\infty} \phi_j(y)\frac{1}{|y|}\,\mathrm{d}y = \int_{-\infty}^{\infty} H_j(y)H_k(y)|y|e^{-y^2}\,\mathrm{d}y \tag{65}$$

Let

$$b_{jk} = 2\int_{0}^{\infty} H_j(x)H_k(x)|x|e^{-x^2}\,\mathrm{d}x \tag{66}$$

It follows that

$$\int_{-\infty}^{\infty} \phi_j(y)\phi_k(y)\frac{1}{|y|}\,\mathrm{d}y = \begin{cases} b_{jk} & j+k = \text{even} \\ 0 & j+k = \text{odd} \end{cases} \tag{67}$$

Now we wish to determine the b_{jk} in terms of things we already know how to calculate. Making a few manipulations,

$$\int_{-\infty}^{\infty} H_j(x)H_k(x)|x|e^{-x^2}\,\mathrm{d}x = \tag{68}$$

$$\int_{0}^{\infty} H_j(x)H_k(x)(2x)e^{-x^2}\,\mathrm{dx} = \tag{69}$$

$$-\int_{0}^{\infty} H_j(x)H_k(x)\frac{d}{dx}e^{-x^2}\,\mathrm{d}x = \tag{70}$$

$$-H_j(x)H_k(x)e^{-x^2}\Big|_0^\infty + \int_{0}^{\infty} \frac{d}{dx}[H_j(x)H_k(x)]e^{-x^2}\,\mathrm{d}x = \tag{71}$$

$$H_j(0)H_k(0) + \int_{0}^{\infty} [H_j'(x)H_k(x) + H_j(x)H_k'(x)]e^{-x^2}\,\mathrm{d}x = \tag{72}$$

$$H_j(0)H_k(0) + \int_{0}^{\infty} 2[jH_{j-1}(x)H_k(x) + kH_j(x)H_{k-1}(x)]e^{-x^2}\,\mathrm{d}x \tag{73}$$

The coefficients are therefore

$$b_{jk} = H_j(0)H_k(0) + 2ja_{j-1,k} + 2ka_{j,k-1} \tag{74}$$

In the special cases

$$b_{00} = 1 \tag{75}$$
$$b_{0k} = 2a_{0,k-1} \quad \text{if } k \geq 1 \tag{76}$$

The matrix elements are $c_{jk} = (\phi'_j, \phi'_k)$ where

$$c_{jk} = \begin{cases} b_{jk} & \text{if } j+k = \text{even} \\ 0 & \text{if } j+k = \text{odd} \end{cases} \tag{77}$$

4 Discretization Method 2

An alternate approach is to use the antisymmetric normalized Hermite functions as the Galerkin basis

$$\varphi_k(y) = \psi_{2k+1}(x) = \frac{1}{\sqrt{\gamma_{2k+1}}} H_{2k+1}(y) e^{-y^2/2} \tag{78}$$

The Hermite functions satisfy the relationships

$$\psi'_k(x) = \alpha_k \psi_{k-1}(x) - \alpha_{k+1}\psi_{k+1}(x) \tag{79}$$
$$x\psi_k(x) = \alpha_k \psi_{k-1}(x) + \alpha_{k+1}\psi_{k+1}(x) \tag{80}$$
$$\tag{81}$$

Where $\alpha_k = \sqrt{\frac{k}{2}}$.

This choice makes discretizing the mass, kinetic, and quadratic matrices very easy, but the Coulombic term still requires some effort to obtain.

The mass matrix is the identity

$$m_{jk} = \int_{-\infty}^{\infty} \varphi_j(x)\varphi_k(x)\,\mathrm{d}x = \delta_{jk} \tag{82}$$

The off-diagonal elements of the quadratic term exactly cancel the off-diagonal elements of the kinetic term so that

$$\kappa_{jk} + q_{jk} = \int_{-\infty}^{\infty} \varphi'_j(x)\varphi'_k(x) + x^2\varphi_j(x)\varphi_k(x)\,\mathrm{d}x = (3+4j)\delta_{jk} \tag{83}$$

As for the Coulombic term, we have

$$c_{jk} = \int_{-\infty}^{\infty} \varphi_j(x)\frac{1}{|x|}\varphi_k(x)\,\mathrm{d}x \tag{84}$$

As before, symmetry requires that this be zero unless $j + k$ is even. Let us again introduce a related integral

$$b_{jk} = 2\int_0^\infty \varphi_j(x)\frac{1}{x}\varphi_k(x)\,\mathrm{d}x \tag{85}$$

We can write this more explicitly as

$$b_{jk} = \frac{2}{\sqrt{\gamma_{2j+1}\gamma_{2k+1}}}\int_0^\infty H_{2j+1}(x)\frac{1}{x}H_{2k+1}(x)e^{-x^2}\,\mathrm{d}x \tag{86}$$

This is equal to c_{jk} for $j + k$ even. Since we are only working with odd Hermite polynomials, it is perfectly legal to divide them by x

$$\frac{1}{x}H_{2k+1}(x) = 2H_k(x) - \frac{4k}{x}H_{2k-1}(x) \tag{87}$$

Plugging this into the integral gives us

$$b_{jk} = \frac{2}{\sqrt{\gamma_{2j+1}\gamma_{2k+1}}}\int_0^\infty H_{2j+1}(x)\left[2H_{2k}(x) - \frac{4k}{x}H_{2k-1}(x)\right]e^{-x^2}\,\mathrm{d}x \tag{88}$$

Splitting this into two integrals

$$b_{jk} = \frac{4}{\sqrt{\gamma_{2j+1}\gamma_{2k+1}}}\int_0^\infty H_{2j+1}(x)H_{2k}(x)e^{-x^2}\,\mathrm{d}x - \frac{8k}{\sqrt{\gamma_{2j+1}\gamma_{2k+1}}}\int_0^\infty H_{2j+1}(x)H_{2k-1}(x)\frac{1}{x}e^{-x^2}\,\mathrm{d}x \tag{89}$$

Part of this is something that appeared using Discretization Method 1, namely the matrix elements a_{jk} in (47) and the second term is just a previous term in the sequence.

$$b_{jk} = \frac{4}{\sqrt{\gamma_{2j+1}\gamma_{2k+1}}}a_{2j+1,2k} - 2\sqrt{\frac{k}{2(2k+1)}}b_{j,k-1} \tag{90}$$

The initial condition is

$$b_{00} = \frac{2}{\sqrt{\pi}} \tag{91}$$

Looking at this expression, it is a bit annoying to have to be working with such large numbers as γ_{2k+1}. It would be better if we rescaled things so all the terms we work with have more uniform magnitude. To that end, let's consider introducing a rescaled version of a_{jk} as follows:

$$\tilde{a}_{jk} = \int_0^\infty \psi_j(x)\psi_k(x)e^{-x^2}\,\mathrm{dx} = \frac{1}{\sqrt{\gamma_j\gamma_k}}a_{jk} \tag{92}$$

The recursion for $\tilde{a}_{jk}$ becomes

$$\tilde{a}_{jk} = \frac{H_j(0)H_{k-1}(0)}{\sqrt{\gamma_j\gamma_k}} + 2j\sqrt{\frac{\gamma_{j-1}\gamma_{k-1}}{\gamma_j\gamma_k}}\tilde{a}_{j-1,k-1} \tag{93}$$

We can express the values of the Hermite polynomials at zero (Hermite numbers) in terms of the values of the Hermite functions at zero

$$H_j(0) = \sqrt{\gamma_j}\psi_j(0) \tag{94}$$

We can rewrite the recursion as

$$\tilde{a}_{jk} = \sqrt{\frac{\gamma_{k-1}}{\gamma_k}}\psi_j(0)\psi_{k-1}(0) + 2j\sqrt{\frac{\gamma_{j-1}\gamma_{k-1}}{\gamma_j\gamma_k}}\tilde{a}_{j-1,k-1} \tag{95}$$

Using $\gamma_k = \sqrt{\pi}2^k k!$ we obtain the rules for computing the rescaled coefficients

$$\tilde{a}_{jj} = \frac{1}{2}, \quad \text{for all } j \tag{96}$$

$$\tilde{a}_{j0} = \sqrt{\frac{1}{2\sqrt{\pi}j}}\psi_{j-1}(0), \quad \text{for } j > 0 \tag{97}$$

$$\tilde{a}_{0k} = \sqrt{\frac{1}{2\sqrt{\pi}k}}\psi_{k-1}(0), \quad \text{for } k > 0 \tag{98}$$

$$\tilde{a}_{jk} = \sqrt{\frac{1}{2k}}\psi_j(0)\psi_{k-1}(0) + \sqrt{\frac{j}{k}}\tilde{a}_{j-1,k-1} \tag{99}$$

Where

$$\psi_0(0) = \pi^{-1/4} \tag{100}$$

$$\psi_1(0) = 0 \tag{101}$$

$$\psi_{k+1}(0) = -\sqrt{\frac{k}{k+1}}\psi_{k-1}(0) \tag{102}$$

The new expression for b_{jk} is

$$b_{jk} = \sqrt{\frac{1}{2(2k+1)}}\left(4\tilde{a}_{2j+1,2k} - \sqrt{k}b_{j,k-1}\right) \tag{103}$$

5 Implementation

At the end, we can write our eigenvalue problem as

$$\left\{\frac{1}{2}(\mathbf{K}+\mathbf{Q})+\mathbf{C}\right\}\mathbf{v}=\nu\mathbf{M}\mathbf{v} \tag{104}$$

Since Discretization Method 2 requires computing fewer matrix elements and we already have the nice scaling of the trial functions and coefficients, we will only write a Python code for this choice. First we import modules for plotting, computing eigenvalues and eigenvectors, and a couple small numerical things.

```python
from __future__ import division
from scipy.linalg import eigh
from mpl_toolkits.mplot3d import Axes3D
from matplotlib import cm
from matplotlib.tri import Triangulation
import numpy as np
import matplotlib.pyplot as plt
```

We can generate the normalized Hermite functions on a given grid using the recurence relation

```python
def hermite_functions(x,n):
    """
    Evaluate the first n normalized Hermite functions on a grid x
    """

    m = len(x)
    psi = np.zeros((m,n))
    psi[:,0] = np.pi**(-0.25)*np.exp(-x**2/2)
    psi[:,1] = np.sqrt(2)*x*psi[:,0]
    a = np.sqrt(np.arange(0,n)/2)

    for k in range(1,n-1):
        psi[:,k+1] = (x*psi[:,k] - a[k]*psi[:,k-1])/a[k+1]

    return psi
```

The elements of the Coulomb interaction matrix are also obtained using the recurrence relations derived in section 4.

```
def coulomb_matrix(n):
    """
    Compute the Coulomb interaction matrix which has matrix elements

    C[j][k] = (psi_{2j+1},psi_{2k+1}/|x|)

    Where psi is the normalized Hermite function
    """

    m = 2*n+1

    # Values of the Hermite functions at x=0
    psi0 = np.zeros(m)

    psi0[0] = np.pi**(-0.25)

    for k in range(1,m-1):
        psi0[k+1] = -np.sqrt(k/(k+1.0))*psi0[k-1]


    # The a-tilde coefficients of section 4
    A = np.eye(m)/2
    A[0,0] = 0.5
    A[0,1:] = psi0[:-1]/np.sqrt(2*np.sqrt(np.pi)*np.arange(1,m))
    A[1:,0] = A[0,1:]

    for j in range(1,m):
        for k in range(1,j):
            A[j,k] = psi0[j]*psi0[k-1]/np.sqrt(2*k) + \
                     np.sqrt(j/k)*A[j-1,k-1]
            A[k,j] = A[j,k]

    # The b coefficients in section 4
    B = np.zeros((n,n))

    B[0,0] = 2/np.sqrt(np.pi)

    for k in range(1,n):
        B[0,k] = (4*A[1,2*k]-2*np.sqrt(k)*B[0,k-1])/np.sqrt(2*(2*k+1))

    B[1:,0] = B[0,1:]
    for j in range(1,n):
        for k in range(1,n):
            B[j,k] = (4*A[2*j+1,2*k]-2*np.sqrt(k)*B[j,k-1])/ \
                     np.sqrt(2*(2*k+1))

    # Coulomb matrix
    C = np.zeros((n,n))
    for j in range(n):
        for k in range(n):
            if not (j+k) % 2:
                C[j,k] = B[j,k]

    return C
```

Finally, we can numerically solve the eigenvalue problem for the anti-symmetric equation and use the fact that we know exactly the eigenfunctions for the other dimenions to obtain eigenfunctions to the two-particle problem.

```
if __name__ == '__main__':

    x = np.linspace(-10,10,100)
    yy,xx = np.meshgrid(x,x)

    # Original coordinate system
    Q1 = (xx+yy)/np.sqrt(2)
    Q2 = (xx-yy)/np.sqrt(2)

    n = 16

    # Compute Hermite functions
    hf = hermite_functions(x,2*n+1)

    # Use anti-symmetric functions as basis
    phi = hf[:,1::2]

    # Non-Coulombic part of Hamiltonian H0 = (K+Q)/2
    H0 = np.diag((3+4*np.arange(n))/2)

    # Coulombic part of Hamiltonian
    C = coulomb_matrix(n)

    # Total Hamiltonian
    H = H0 + C

    nu,Vhat = eigh(H)
    # Evaluate antisymmetric Hermite-function series on grid
    V = np.dot(phi,Vhat)

    # Create an eigenfunction to the two-particle problem
    Psi = np.outer(hf[:,0],V[:,0])

    psi = Psi.flatten()
    q1 = Q1.flatten()
    q2 = Q2.flatten()

    dex = np.where((q1**2+q2**2)<10)[0]
    tri = Triangulation(q2[dex],q1[dex])
    fig = plt.figure(1)
    ax =  fig.gca(projection='3d')
    ax.plot_trisurf(tri,psi[dex], cmap=cm.jet, linewidth=0.2 )
    ax.set_xlabel(r'$q_1$',fontsize=18)
    ax.set_ylabel(r'$q_2$',fontsize=18)
    plt.show()
```

Note: the Delaunay triangulation used for `plot_trisurf` seems not to like the grid used very much. Changing the cutoff radius does seem to make it fail for some values. Probably it should be implemented differently.

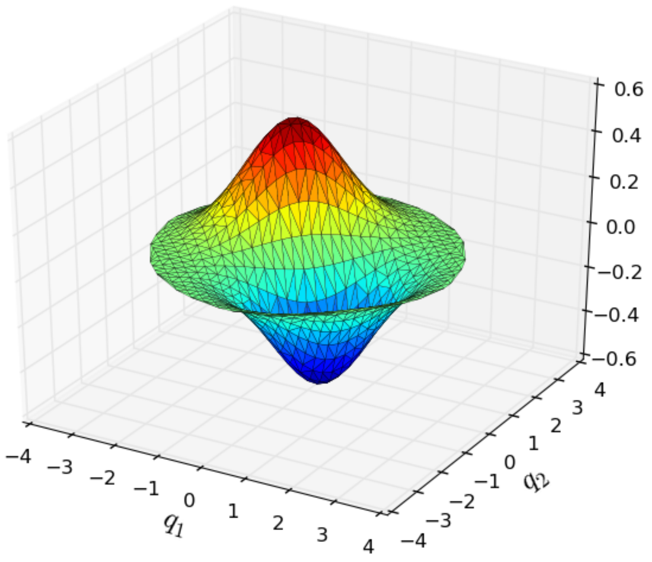

Figure 1: Ground state eigenfunction for the two-particle system