Python activities - PHY456
Linear Variational Method

Federico Duque Gomez (Python code by Vijay Shankar)

May 21, 2013

1 Introduction

The variational method is a powerful tool to find solutions to the time-independent Schrodinger equation. As discussed in the lectures, the functional

$$E[\psi] = \frac{\langle\psi|H|\psi\rangle}{\langle\psi|\psi\rangle} \tag{1}$$

can never take a value smaller than the actual ground state energy for the Hamiltonian H. Using this fact the problem of finding the ground state becomes a minimization problem. In principle one would have to minimize over the Hilbert space of all the square integrable functions, which seems like a hopeless endeavor. However, usually we have some intuition of how the wavefunction should look like or we know some conditions that must be satisfied; therefore, we can restrict the search for the minimum to a certain family of functions, which can be parametrized by variational parameter. We have used this technique in the past (see for example Problem Set 2 - Question 1) to turn the complicated problem of minimizing a functional into a regular minimization of a function with respect to a parameter. Once we have found an approximation for the ground state we can start going up in energy finding the excited states by a similar technique.

In this activity you will learn a different and more powerful approach to minimize the functional $E[\psi]$ in a subspace of the Hilbert space (which, once again, is the result of some educated guess). One of the advantages of the new method is that you can get approximations not only for the ground state but also for some excited states in a single calculation. A numerical implementation of this technique is provided in the Python program *linearvariation_numerical.py*.

2 Method

In the *linear variational method* we expand the wavefunction

$$|\psi\rangle = \sum_{i=1}^{N} C_i\,|\phi_i\rangle \tag{2}$$

in terms of a basis of a subspace of the Hilbert space, $|\phi_i\rangle$, $i = 1, 2, ..., N$.

We can reformulate the minimization of the functional (1) as a constrained minimization problem for $\mathcal{L}$, given by

$$\mathcal{L} = \langle\psi|H|\psi\rangle - E(\langle\psi|\psi\rangle - 1), \tag{3}$$

where E is a lagrange multiplier that constrains the solution to be a function normalized to 1. Using the expansion (2) we can write

$$\mathcal{L} = \sum_{i,j=1}^{N} C_i^* C_j H_{ij} - E(\sum_{i,j=1}^{N} C_i^* C_j S_{ij} - 1), \tag{4}$$

where

$$H_{ij} = \langle\phi_i|H|\phi_j\rangle\,, \tag{5}$$
$$S_{ij} = \langle\phi_i|\phi_j\rangle\,. \tag{6}$$

Requiring that the derivative of $\mathcal{L}$ with respect to C_i vanishes we get

$$\sum_{j=1}^{N}(H_{ij} - ES_{ij})C_j = 0 \text{ for } i = 1, 2, ..., N, \tag{7}$$

which in matrix notation becomes

$$HC = ESC. \tag{8}$$

Notice that this is almost an ordinary eigenvalue equation, except that there is a matrix S on the right hand side. This is called a *generalized eigenvalue equation* and there are libraries for solving such problems numerically.

The eigenvalues and eigenvectors of this generalized eigenvalue problem provide an estimate for the N lowest energies and eigenstates of H. We will shortly discuss how are these solutions are and how they can be improved.

3 Implementation in Python

In the program *linearvariation_numerical.py* the linear variational method is used to find approximate solutions for the time-independent Schrödinger equation with an infinite well potential

$$V(x) = \begin{cases} 0 \text{ if } |x| \leq 1 \\ \infty \text{ if } |x| > 1 \end{cases} \tag{9}$$

which forces the wavefunctions to vanish at the boundaries of the well ($x = \pm 1$). The exact solution is well known so we can use this simple example to test the variational method. We use units where $\frac{\hbar^2}{2m} = 1$.

For basis functions, we choose simple polynomials that vanish on the boundaries of the well,

$$\phi_n(x) = x^n(x-1)(x+1) \text{ for } n = 0, 1, 2, ... \tag{10}$$

In the program we define the function **vwf(n,X)** to calculate them. The matrix elements (5) and (6) are very simple to calculate exactly but for generality the program calculates them numerically in the two functions **nhamiltonianmat(n,m,X)** and **noverlapmatrixmat(n,m,X)**. By redefining these functions you can make modify the program for other one-dimensional potentials.

Once the matrix elements are found, the code uses a Python library to solve the generalized eigenvalue problem (7). The solutions are organized by increasing energy.

4 Try it!

Open the script *linearvariation_numerical.py* from the Python Shell and run it. [1] The program will prompt you to enter the maximum number of variational states, that is the number N of states in the basis (10). After you have entered this number you will be asked to select which of the N calculated wavefunctions you want to be plotted and compared with the actual wavefunction (which can be computed exactly for the infinite well case). The program will display two plots. On the left you will see the wavefunction you selected plotted together with the corresponding exact solution. On the right you will find a comparison between the N lowest energies of the system found using

[1] If you don't know how to use the Python Shell or you have not installed a Python distribution in your computer, visit the compwiki web site **http://compwiki.physics.utoronto.ca** where you can find extensive information about how to install and use Python.

the variational method and their exact values.[2]

Try running the program a few times with different parameters. How do the approximate energy values behave compared to the exact solutions as you increase the number of variational states? What happens if you plot the approximate wavefunction for the same energy level with increasing number of basis states? For a fixed number of basis states, which calculated wavefunctions are closer to the actual solution?

Can you explain these observations? If you are interested in the details, they are described in J. K. L. MacDonald, Phys. Rev. **43**, 830 (1933). The main results are the following two theorems[3]:

- Between each pair of succesive solutions to (7), augmented by $-\infty$ and ∞, there occurs ate least one exact eigenvalue.
- If the number of variational states is increased by one, the new approximate solutions will be interleaved by the previous ones.

Using these two theorems explain what you saw in your trials.

[2]When you run the code you might get a warning message: " Complex Warning: Casting complex values to real discard the imaginary part". You can safely ignore it.

[3]See: http://demonstrations.wolfram.com/InterleavingTheoremsForTheRayleighRitzMethodInQuantumMechani/.