PY421 – Introduction to Computational Physics

Project #3:

-solution of the time dependent Schrödinger equation;
-preservation of the norm, method based on the fast Fourier transform.

Solving the time dependent Schrödinger equation.

We would like to calculate the solution of the Schrödinger equation

$$\imath\hbar\partial_t\psi(x,t) = -\frac{\hbar^2}{2m}\partial_x^2\psi(x,t) + V(x)\psi(x,t) \tag{1}$$

We discretize space setting $x = n\,a$ where n ranges from 0 to $N-1$. $\psi(x,t)$ becomes then an array of values $\psi_n(t)$. We will assume periodic boundary conditions in space and we will leave these mostly implicit, i.e. if at some point we encounter the variable ψ_{n-1} and n takes value 0 we take ψ_{-1} to mean ψ_{N-1}. Similarly ψ_N is identified with ψ_0.

With the discretization

$$\partial_x^2\psi(x) \;\to\; \frac{\psi_{n+1} + \psi_{n-1} - 2\psi_n}{a^2} \tag{2}$$

our equation becomes

$$\imath\hbar\partial_t\psi_n = -\frac{\hbar^2}{2m}\frac{\psi_{n+1} + \psi_{n-1} - 2\psi_n}{a^2} + V_n\psi_n \tag{3}$$

It is clear that this equation is of the form

$$\imath\hbar\partial_t\psi_m = \sum_n A_{m,n}\psi_n \tag{4}$$

or, more compactly,

$$\imath\hbar\partial_t\psi = A\psi \tag{5}$$

where the matrix elements of A can be easily deduced from the discretized form of the Schrödinger equation, given above.

Now, if A were just a number, the equation $\imath\hbar\partial_t\psi = A\psi$ would be straightforward to integrate, with the result given by

$$\psi(t) = e^{-\frac{\imath}{\hbar}At}\psi(0) \tag{6}$$

This equation remains true even if A is a matrix, however the meaning of $\exp[-(\imath/\hbar)At]$ must be specified.

One possibility is to resort to the formula $\exp(x) = \lim_{M\to\infty}(1+\frac{x}{M})^M$. We divide t into M small intervals of magnitude dt (note, dt here stands for a finite, albeit small, number, not a differential)

$$t = M\,dt \tag{7}$$

and set

$$\psi(t) = \Big(I - \frac{\imath}{\hbar}A\,dt\Big)^M\psi(0) \tag{8}$$

In the limit $M\to\infty,\ dt = \frac{t}{M}\to 0$ the power will tend to the exponential $\exp(-\frac{\imath}{\hbar}At)$.

Notice that the above construction has a straightforward computational interpretation. It tells us that we obtain $\psi(t)$ from $\psi(0)$ by iterating M times a very simple step, namely

$$\psi \to \psi' = \psi - \frac{\imath}{\hbar}A\,dt\psi \tag{9}$$

On the other hand, looking at the matrix form of the evolution equation $\imath\hbar\partial_t\psi = A\psi$, it is clear that the formula just above corresponds to discretizing first

$$\imath\hbar\partial_t\psi \ \to\ \imath\hbar\frac{(\psi'-\psi)}{dt} \tag{10}$$

where $\psi = \psi(t)$ and $\psi' = \psi(t+dt)$ and solving for ψ' in terms of ψ.

However, this simple algorithm has some very serious drawbacks, which we will now discuss.

In order to illustrate the problems that the algorithm of Eq. 9 gives origin to we will consider the simplified case of the free motion of a particle, i.e. we will take $V = 0$. The matrix A reduces then to a constant times the double difference term which expresses the discretization of the derivative ∂_x^2, so that

$$(A\psi)_n = \frac{\hbar^2}{2m}\frac{2\psi_n - \psi_{n+1} - \psi_{n-1}}{a^2} \tag{11}$$

With this simplified form for A it is possible to find its eigenvectors. They are plane waves

$$\psi_n^{(k)} = \frac{1}{\sqrt{N}} e^{2\pi\imath \frac{kn}{N}} \tag{12}$$

where $k = 0, 1, \ldots, N-1$ is an integer labeling the eigenvectors.

Using the fact that, with the above form for ψ, $\psi_{n\pm1} = \exp(\frac{\pm 2\pi\imath k}{N})\psi_n$ one can easily show that

$$\begin{aligned} A\psi^{(k)} &= \frac{\hbar^2}{2ma^2}\Big(2 - e^{2\pi\imath\frac{k}{N}} - e^{-2\pi\imath\frac{k}{N}}\Big)\psi^{(k)} = \\ &= \frac{\hbar^2}{2ma^2} 2\Big[1 - \cos\Big(2\pi\frac{k}{N}\Big)\Big]\psi^{(k)} = \lambda_k \psi^{(k)} \end{aligned} \tag{13}$$

with

$$\lambda_k = \frac{\hbar^2}{2ma^2} 2\Big[1 - \cos\Big(2\pi\frac{k}{N}\Big)\Big] \tag{14}$$

Notice, incidentally, that, as one can see from a Taylor series expansion of the cosine in Eq. 14, for k small with respect to N the eigenvalues reduce to $\lambda = \frac{1}{2m}(\frac{2\pi\hbar k}{N\,a})^2$. Since the term with ∂_x^2 in the Schrödinger equation corresponds to the kinetic energy $\frac{p^2}{2m}$ of the wave, this shows that k itself is related to the momentum p by $p = \frac{2\pi\hbar k}{N\,a} = \frac{2\pi\hbar k}{L}$, where L is the total length of the system. In particular one recovers the important fact that there is one eigenvector, i.e. one state of the system, per every cell of volume $L\Delta p = 2\pi\hbar$ in phase space.

Every vector ψ_n can be expanded into components along the basis defined by a complete set of eigenvectors. With the eigenvectors of Eq. 12 the expansion takes the form of a discrete Fourier transform:

$$\psi_n = \sum_k c_k \psi_n^{(k)} = \sum_k c_k \frac{1}{\sqrt{N}} e^{2\pi\imath\frac{kn}{N}} \tag{15}$$

Substituting this expansion into the formula of Eq. (9) for the iterative algorithm we find

$$\psi_m \to \psi'_m = \psi_m - \sum_{n,k} \frac{\imath}{\hbar} A_{m,n}\, dt\, c_k \psi_n^{(k)} = \sum_k (1 - \frac{\imath}{\hbar}\lambda_k\, dt) c_k \psi_m^{(k)} \tag{16}$$

where we have used the fact that $\psi^{(k)}$ is an eigenvector of A (we are still considering the simplified case where $V = 0$).

What Eq. 16 is telling us is that the evolution algorithm is diagonal in the basis of eigenvectors of A given by the Fourier modes, with the coefficients of the expansion undergoing the simple change

$$c_k \rightarrow c'_k = (1 - \frac{\imath}{\hbar}\lambda_k \, dt)c_k \tag{17}$$

at each step.

But we see here the origin of the problem. Substituting the expansion of Eq. 15 into the continuum equation $\imath\hbar\partial_t\psi = A\psi$ we see that with continuous time the coefficients c_k should evolve according to

$$c_k(t) = e^{-\frac{\imath}{\hbar}\lambda_k t}c_k(0) \tag{18}$$

This equation implies that $|c_k|$ remains constant with time, i.e. that the evolution is unitary, but this fundamental condition is violated by the iterative step of Eq. 17, which gives

$$|c'_k| = \sqrt{(1 + \big(\frac{\lambda_k \, dt}{\hbar}\big)^2}\,|c_k| \tag{19}$$

Equation 19 not only tells us that the time discretized evolution is not unitary, but also, and this is where the real problem is, that the relative magnitudes of the Fourier components of the wave function will change with time. In particular, those with largest eigenvalues will grow faster: but these are typically artifacts of the space discretization (artifacts of the space lattice). From inspection of the formulae for eigenvectors and eigenvalues (Eqs. 12 and 14) one sees that the eigenvectors corresponding to the largest eigenvalues are those where the field varies rapidly from site to site. As the magnitude of the corresponding components grows, the outcome of the iterations becomes a wild, ragged field profile, which has nothing to do with the solution of our problem, and eventually grows so big to produce floating point errors.

Although we used the free evolution to illustrate the algorithmic difficulty, the problem is general. With a non-vanishing potential V_n it will not be possible to express the eigenvectors of the matrix A in closed form, but A will always have a set of N eigenvectors which can be chosen to form an orthonormal basis. (The matrix A becomes the energy operator, or Hamiltonian operator, in the continuum limit $a \rightarrow 0$ and its eigenvalues and eigenvectors become the energy eigenvalues and eigenstates). Expanding the wave function into the basis formed by such eigenvectors, we see that the components

would still evolve, in the iterative procedure, according to Eq. 17, albeit with different eigenvalues λ_k, and that their relative magnitudes would change, leading to a distortion of the wave function (even with a non-vanishing potential, moreover, the largest eigenvalues are still those associated with lattice artifacts).

Evolution algorithm based on the FFT (Fast Fourier Transform).

The problem illustrated above can be very easily remedied in the free case and the solution eventually leads also to a viable algorithm for the general case.

From Eq. 18 we see that in the free case all what we need to do to implement a unitary evolution is to expand the initial wave function $\psi(0)$ into eigenvectors of A, i.e. into energy eigenstates (we will frequently call A the energy operator or the Hamiltonian, even with a discretized space) and to evolve the coefficients according to that equation, setting $c_k(t) = \exp(-\frac{\imath}{\hbar}\lambda_k t)c_k(0)$. Thus we obtain the coefficients $c_k(0)$ by an inverse Fourier transform

$$c_k(0) = \sum_n \psi_n^{(0)} \frac{1}{\sqrt{N}} e^{-2\pi\imath\frac{kn}{N}} , \tag{20}$$

use Eq. 18 to obtain $c_k(t)$, and perform another Fourier transform (Eq. 15) to obtain the final wave function $\psi(t)$. If we wish to obtain the wave function at intervals dt, as is necessary to display the evolution, we will use

$$c_k \rightarrow c'_k = e^{-\frac{\imath}{\hbar}\lambda_k\, dt} c_k \tag{21}$$

as the basic iterative step, reconstructing the wave function by Fourier transform after each step.

In the general case we cannot proceed that simply, because calculating all the eigenvalues and eigenvectors of A, expanding the wave function into this basis of eigenvectors and reconstructing it at the end of the evolution or, even worse, after each step, while computationally feasible, is too time consuming even for systems of moderate size. However, we can proceed as follows. Let us express A as a sum of a kinetic term and a potential term:

$$A = K + V \tag{22}$$

where K is of course the expression that A takes with $V = 0$, defined by Eq. 11.

The formal iteration step we would like to implement is

$$\psi \to \psi' = e^{-\frac{\imath}{\hbar}(K+V)\,dt}\,\psi \tag{23}$$

This cannot be implemented exactly, because we would need to expand ψ into eigenvectors of $K+V$ to calculate the action of the exponential. However, we can approximate

$$e^{-\frac{\imath}{\hbar}(K+V)\,dt} \approx e^{-\frac{\imath}{\hbar}K\,dt}e^{-\frac{\imath}{\hbar}V\,dt} \tag{24}$$

This relation, which would be obviously exact if K and V were just numbers, is only approximate for non-commuting matrices. However, the error we make is of order $(dt)^2$ and so, for sufficiently small values of dt, replacing the exponential in Eq. 23 with the product of exponentials in Eq. 24 will provide a good approximation to the correct evolution. The great advantage of this approximation is that it gives origin to an iterative step which can be computationally implemented (with reasonable CP time) and which preserves the unitarity of the evolution.

Indeed, we will now evolve the wave function according to the following algorithm:

i) we evolve ψ in the ordinary space representation according to

$$\psi_n \to \psi'_n = e^{-\frac{\imath}{\hbar}V_n\,dt}\psi_n \tag{25}$$

(this implements the action of the second exponential term in Eq. (24));

ii) we perform an inverse Fourier transform to obtain the coefficients c'_k in the expansion of ψ',

iii) we evolve these coefficients according to

$$c'_k \to c''_k = e^{-\frac{\imath}{\hbar}\lambda_k\,dt}c'_k \tag{26}$$

iv) we reconstruct the final ψ'' from its expansion coefficients c''_k by a Fourier transform.

Of course, this algorithm will be computationally efficient only if the discrete Fourier transforms which must be performed at each iterative step are not too time consuming. Luckily, there exists a very efficient algorithm, the so called fast Fourier transform (FFT) algorithm, that can be used to perform a discrete Fourier transform of a vector of length N with a number of operations of order $N\log_2 N$. In our code we will use the FFT to performed the required direct and inverse Fourier transforms at each step of the evolution.