A fast explicit solver for Gross-Pitaevskii equation

Jeffrey M. Brown

May 6, 2015

1 Introduction

Here we demonstrate a fast explicit solver for the Gross-Pitaevskii equation (GPE) based on Visscher's method for time propagating Schrödinger's equation. Visscher's method was first published[1] in 1991, but still remains relatively unknown compared to comparable methods, such as Crank-Nicholson or the split step method. Visscher's method has the advantage that it offers the same relative order of accuracy, but is faster running and much easier to code due to its simple update scheme. This makes it a good candidate for pedagogical demonstration of quantum system dynamics. In the following sections we will explain Visscher's method for the Schrödinger equation, then modify the update scheme to include the nonlinear term found in GPE and finally show that the method reproduces some well known examples of Bose-Einstein condensate (BEC) time dynamics.

2 Visscher's Method

We begin by separating the wavefunction into its real R and imaginary I parts. Schrödinger's equation now becomes a set of coupled equations

$$\frac{dR}{dt} = HI \tag{1}$$

$$\frac{dI}{dt} = -HR \tag{2}$$

where the real part is updated by the Hamiltonian acting on the imaginary part and vice versa. Visscher suggests that (1) and (2) could be solved more easily by placing the real and imaginary parts at offset times. By approximating the time derivative with a simple finite difference and rearranging the equations, we produce an update scheme

$$R\left(t + \tfrac{1}{2}\Delta t\right) = R\left(t - \tfrac{1}{2}\Delta t\right) + \Delta t H I\left(t\right) \tag{3}$$

$$I\left(t + \tfrac{1}{2}\Delta t\right) = I\left(t - \tfrac{1}{2}\Delta t\right) - \Delta t H R\left(t\right) \tag{4}$$

[1] A fast explicit algorithm for the time-dependent Schrödinger equation, P.B. Visscher, Computers in Physics 5, 596 (1991); doi: 10.1063/1.168415

that evolves the system forward one time step Δt by applying (3) and (4) sequentially. From the scheme, we see that the time evolution of one part of the wavefunction is determined by its counterpart at an offset time, approximating wavefunction evolution.

Due to the temporal offset, we may not see at first how to represent the probability distribution or calculate an expectation value. Visscher suggests that at the times corresponding to the real part, we choose

$$|\psi(t)|^2 = R\left(t\right)^2 + I\left(t + \tfrac{1}{2}\Delta t\right) I\left(t - \tfrac{1}{2}\Delta t\right) \tag{5}$$

and for the times corresponding to the imaginary part

$$|\psi(t)|^2 = R\left(t + \tfrac{1}{2}\Delta t\right) R\left(t - \tfrac{1}{2}\Delta t\right) + I\left(t\right)^2. \tag{6}$$

The only constraint for these choices of probability is that we recover the continuum equations $|\psi(t)|^2 = R(t)^2 + I(t)^2$ as $\Delta t \to 0$. Therefore the above equations are a reasonable choice.

The final piece needed is the initial conditions $R(0)$ and $I\left(\frac{1}{2}\Delta t\right)$. If we have an explicit formula for the time dependent wavefunction, then there is no problem in creating these initial conditions. However, in most cases the time dependent nature of the initial wavefunction is not known, such as a numerically computed ground state of a particular potential. Visscher's method has the extra burden of knowing the real and imaginary parts at different times. A similar problem exists for initializing a finite difference Maxwell solver. This of course does not prevent this scheme from being useful, but one must take care to give accurate initial conditions.

One final note on offsetting the wavefunction's real and imaginary parts in time. Visscher's method loses its explicit scheme (and becomes an implicit method) if one chooses to propagate in complex valued time. Therefore, if one wants to start a simulation in the ground state, it is valuable to use the spectral method (or something similar) and propagate in complex time. Then once the ground state is found, change the update scheme to Visscher's and evolve the system normally. Note that solving for the ground state is a one time simulation for a given set of parameters. More will be said about this in the Remarks section.

3 Adaptation for GPE

Now that we know the basic scheme it will have to be modified in order to handle the nonlinear term in GPE. For the Schrödinger equation, the update scheme is written explicitly as

$$R\left(t + \tfrac{1}{2}\Delta t\right) = R\left(t - \tfrac{1}{2}\Delta t\right) + \Delta t\left(-\tfrac{1}{2}\nabla^2 + V\right) I\left(t\right) \tag{7}$$

$$I\left(t + \tfrac{1}{2}\Delta t\right) = I\left(t - \tfrac{1}{2}\Delta t\right) - \Delta t\left(-\tfrac{1}{2}\nabla^2 + V\right) R\left(t\right) \tag{8}$$

where we approximate the laplacian at a point in a discrete array for the 1D case as $\nabla^2 f[x_i] = (f[x_{i-1}] + f[x_{i+1}] - 2f[x_i])/\Delta x^2$ and for the 2D case $\nabla^2 f[x_i, y_i] =$

$(f[x_{i-1},y]+f[x_{i+1},y]+f[x,y_{i+1}]+f[x,y_{i-1}]-4f[x_i,y_i])/\Delta x^2$. The scheme is explicit since the future values of R and I only depend on quantities that were calculated in the past. By adding in the nonlinear term $\eta|\psi(t)|^2$ for the appropriate time we lose the explicit-ness.

$$R\left(t+\tfrac{1}{2}\Delta t\right)=R\left(t-\tfrac{1}{2}\Delta t\right)+\Delta t\left(-\tfrac{1}{2}\nabla^2+V+\eta\left[R\left(t+\tfrac{1}{2}\Delta t\right)R\left(t-\tfrac{1}{2}\Delta t\right)+I\left(t\right)\right]\right)I\left(t\right) \tag{9}$$

$$I\left(t+\tfrac{1}{2}\Delta t\right)=I\left(t-\tfrac{1}{2}\Delta t\right)-\Delta t\left(-\tfrac{1}{2}\nabla^2+V+\eta\left[R(t)+I\left(t+\tfrac{1}{2}\Delta t\right)I\left(t-\tfrac{1}{2}\Delta t\right)\right]\right)R\left(t\right) \tag{10}$$

Luckily, the equations are still linear allowing us to solve for $R(t+\frac{1}{2}\Delta t)$ and $I(t+\frac{1}{2}\Delta t)$.

$$R\left(t+\tfrac{1}{2}\Delta t\right)=\frac{R\left(t-\frac{1}{2}\Delta t\right)+\Delta t\left[-\frac{1}{2}\nabla^2+V+\eta I(t)^2\right]I\left(t\right)}{1-\Delta t\eta R\left(t-\frac{1}{2}\Delta t\right)I\left(t\right)} \tag{11}$$

$$I\left(t+\tfrac{1}{2}\Delta t\right)=\frac{I\left(t-\frac{1}{2}\Delta t\right)-\Delta t\left[-\frac{1}{2}\nabla^2+V+\eta R(t)^2\right]R\left(t\right)}{1+\Delta t\eta R\left(t-\frac{1}{2}\Delta t\right)I\left(t\right)} \tag{12}$$

We now have an update scheme that takes into account the nonlinearity and fortunately the LHS values are calculated from only past values, therefore no iterative methods are needed. A good sanity check is, if we take $\eta\to 0$, we recover Visscher's update scheme for the Schrödinger equation. These update equations are only slightly more complicated that the original ones and should provide a fast update scheme to model the time evolution of a BEC.

Now that we have an update scheme for GPE, how small of time steps must we take given our spatial discretization? A rough estimate of the stability criterion for this algorithm can be estimated using the criterion in Visscher's paper, which is $\Delta t<\Delta x^2$. This criterion assumes the value of the potential $V=0$ and ignores the effects of nonlinearity. This of course is an upper bound for Δt, therefore in practice $\Delta t<\frac{1}{4}\Delta x^2$ seems to be a reasonable choice for the types of potentials and strengths of nonlinearity found in BEC systems. Deriving a more rigorous criterion equation is impractical due to the nonlinear term in GPE, since it would require knowing the max value of $|\psi|^2$ for all times in the simulation. Therefore it's impossible to know whether a set of parameters would produce a stable evolution a priori. Luckily when the simulation becomes unstable, the wavefunction values grow extremely fast in magnitude and quickly produce nan values. A simple periodic check could be implemented to check for nan's, then halt the time evolution, and restart with a new smaller step size.

4 Results

Now we will display some well known systems dynamics to show that the update scheme works. We should first check that as the nonlinearity is turned off $\eta=0$, that we recover Visscher's update scheme for the Schrödinger equation. We know the exact stationary solutions for a harmonic potential $V=\frac{1}{2}x^2$ are the

harmonic oscillator states $\psi(x,t) = \exp[-\frac{1}{2}x^2 - iEt]H_n(x)/\sqrt{2^n n!\pi^{1/4}}$. If we initialize with the ground state, $n = 0$ and $E = 0.5$, we see in Figure 1 that it is stationary after propagating 10 atomic units of time. This is just a simple check of our code to make sure we are accurately reproducing the linear part of GPE.

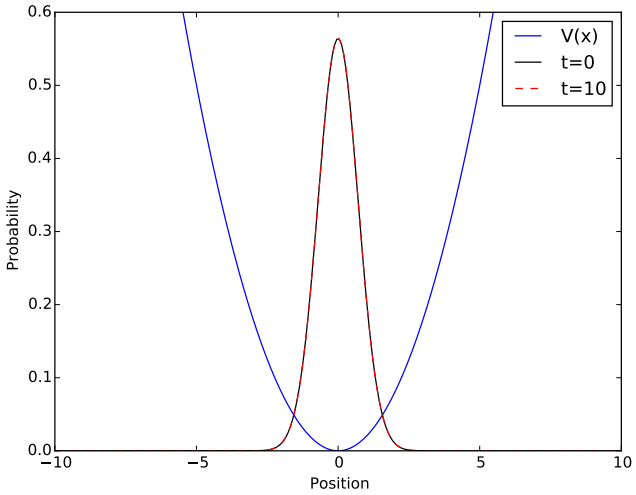

Figure 1: Quantum harmonic oscillator solution

Now we check that the nonlinear portion of GPE is reproduced accurately. For the case of 1D and the potential $V = 0$, the GPE becomes the nonlinear Schrödinger equation and therefore has a stationary solution of $\psi(x,t) = \frac{1}{\sqrt{2}}\text{sech}(x)\exp(-i\mu t)$ with energy $\mu = -\frac{1}{2}$ for a nonlinearity of $\eta = -2$. We see in Figure 2 that if we initialize the BEC wavefunction with the *sech* equation and evolve 10 atomic units of time forward, the wavefunction is unchanged. We can now be reasonably certain that we have coded our method correctly and that the update scheme is stable and accurately reproduces known exact solutions for our given time step.

Now let us test a quasi-2D BEC for which we do not have an exact wavefunction solution. If we keep the same QHO initial conditions, but turn on the nonlinearity with value $\eta = 4$, we now have a wavefunction that is not exactly the ground state of the potential and nonlinearity. We should see some periodic, oscillatory behavior in the BEC, whose theory was laid on in Section 4.2.4 of the class notes. Since the harmonic trap and initial condition are radially symmetric, we should see the BEC have periodic "breathing" as demonstrated in Figure 3.

For the final demonstration we will embedded a vortex in the ground state of a quasi-2D BEC. First, we march in complex valued time using the spectral method and decay to the ground state of a harmonic trap with a nonlinearity of $\eta = 4$. Within this wavefunction we place a phase singularity using equation

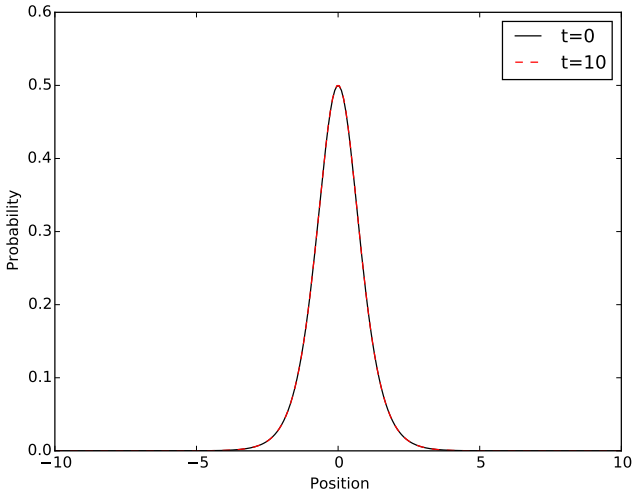

Figure 2: Sech(x) soliton solution for 1D GPE

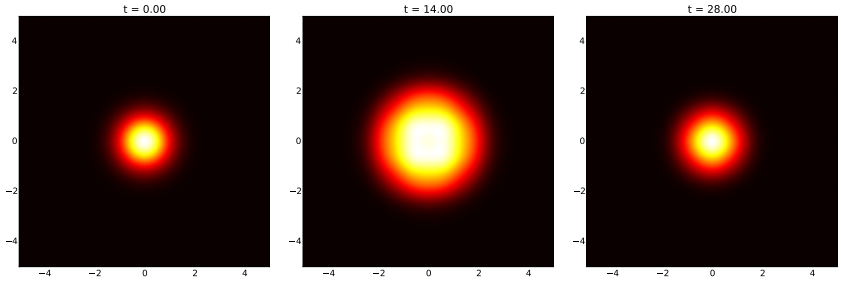

Figure 3: Breathing mode

(4.122) in the class notes Section 4.3.1. We expect from theory that the vortex will precess around the outer boundary and indeed it does as shown in Figure 4. We can change the phase to be of the opposite charge and the precession of the vortex change direction around the center of the BEC.

Now that we know the method reproduces known phenomena, we must show whether it is faster that the split step method. From Figure 5, we see that it indeed is. This is due to the computational complexity of the underlying update scheme, or in simpler terms at what rate does the runtime increase as we increase the number of points. The most computationally intensive part of the update for the split step method, is the Fourier transform. Since the FFT algorithm is of order $\mathcal{O}(N \log_2 N)$, whereas our discrete laplacian is $\mathcal{O}(N)$, for larger grids our method will have drastically reduced runtimes. In Figure 5, I've plotted the runtimes for Python implementation of split step and our method and I've also plotted a C++ implementation of the code. The Python

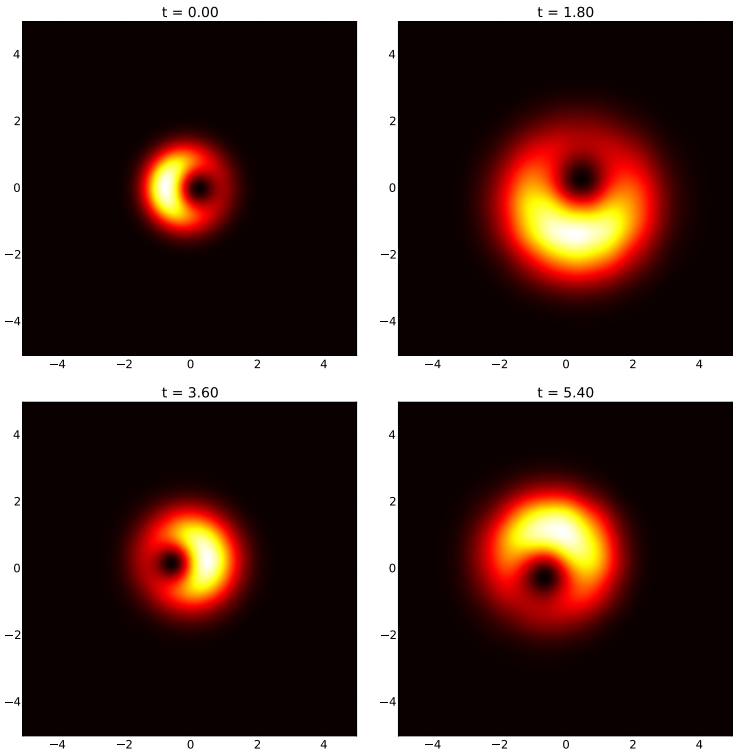

Figure 4: A vortex inside a quasi-2D BEC precesses around the perimeter.

code suffers from garbage collection issues due to the temporary arrays being generated and removed during the update (most interpreted languages will have the same issues). This can be seen by the slight variation away from quadratic (an increase in runtime) around 1000 Points, where the temporary arrays no longer fit into the smallest CPU cache. The C++ implementation does not have any temporary arrays and therefore plays nice with the CPU's caches. The method developed here allows for larger grid, higher resolution demonstrations which run within the attention span of students.

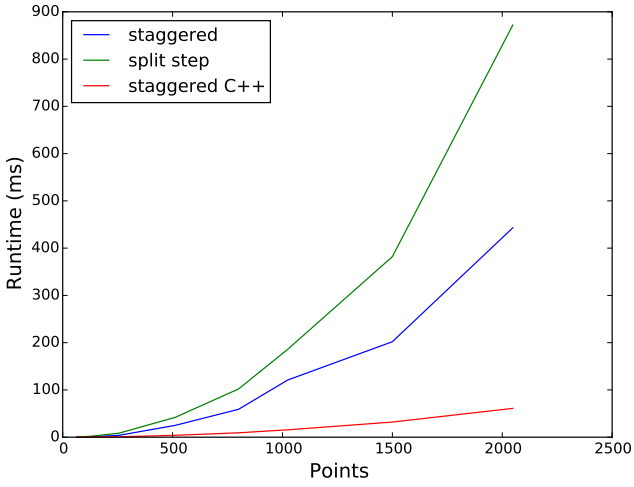

Figure 5: Plotted here is the time taking to make a single nonlinear step forward in time for a 2D grid with number of points along a side of N (total number of points in the grid is N^2). The method developed here scales better the than split step method using FFT.

5 Remarks

Here I have demonstrated a fast and explicit time evolution scheme for modeling BEC dynamics. The scheme reproduces many of the well known examples and is relatively simple to implement, making it ideal for pedagogical purposes. The scheme does have one disadvantage in that the initial conditions require the real and imaginary parts of the wavefunction to be known at offset times. Another disadvantage is that there is not a straightforward way to evolve in complex valued time, i.e. compute numerical ground states of arbitrary potentials. This is due to the nonlinear term $|\psi|^2$ and definition of probability at time t that mixes real and imaginary parts and produces an implicit, nonlinear update scheme when $H \to (1-\alpha)H$. One possible way around this, without resorting to the split step method, would be to extrapolate the offending term (something simple like $|\psi(t)^2| = 2|\psi(t-\Delta t)|^2 - |\psi(t-2\Delta t)|^2$) from previous values by storing a few past wavefunctions. This is probably robust enough to stay stable, but these types of improvements start to over shadow the simplicity of the current method. The most pragmatic solution is to use the spectral method to find the ground state for a given set of input parameters and save this to file. Then subsequent exploration can begin with this initial condition and proceed with the fast method demonstrated here.