Homework 4: 1D Time-Dependent Schrödinger Equation

Kory Beach
Department of Physics, Astronomy, and Applied Physics,
Rensselaer Polytechnic Institute
beachk2@rpi.edu

December 11, 2016

1.1 Introduction

The time-dependent Schrödinger equation (TDSE) is one of the most important equations in physics because its solutions provide precise information of how quantum mechanical wavefunctions will evolve in time. In its most general case it takes the form

$$i\hbar\frac{\partial}{\partial t}\Psi(\mathbf{r},t) = \hat{H}\Psi(\mathbf{r},t), \tag{1.1}$$

which for a single non-relativistic particle in a potential V becomes

$$i\hbar\frac{\partial}{\partial t}\Psi(\mathbf{r},t) = \left[\frac{-\hbar^2}{2\mu}\nabla^2 + V(\mathbf{r},t)\right]\Psi(\mathbf{r},t) \tag{1.2}$$

where μ is the reduced mass of the particle.

While it is possible for some systems to analytically consider solutions of the form

$$\Psi(\mathbf{r},t) = e^{-i\hat{H}t}\Psi(\mathbf{r},0), \tag{1.3}$$

in practice this becomes exceedingly difficult because for most physical systems the Hamiltonian is too complicated for an analytical solution. It is therefore necessary to adopt numerical approximation methods that can allow us to accurately and reliably predict the time evolution of quantum systems. In this report we will discuss the stability, accuracy, and implementation of three numerical schemes for solving the one-dimensional time-dependent Schrödinger equation: the fully explicit scheme, the fully implicit scheme, and the Crank-Nicolson method.

1.2 Methods

All three methods which we will discuss are finite difference methods that reformulate the first derivative in time and second derivative in position using either forward-, backward-, or central-difference approximations. To start we will consider a free particle in one dimension:

$$i\frac{\partial}{\partial t}\psi = -\frac{\partial^2}{\partial x^2}\psi \tag{1.4}$$

where we have omitted the constants $\hbar$ and m for simplicity.

1.2.1 Fully Explicit Scheme

In the fully explicit scheme, a forward difference in time and a central difference in position are employed. If we expand out the derivatives from Equation 1.4 we obtain

$$i\left[\frac{\psi_j^{n+1} - \psi_j^n}{\Delta t}\right] = -\left[\frac{\psi_{j+1}^n - 2\psi_j^n + \psi_{j-1}^n}{(\Delta x)^2}\right] \tag{1.5}$$

where the indicies j and n indicate space and time respectively. Rearranging to solve for ψ_j^{n+1} we obtain an expression that can be solved iteratively to give the wavefunction forward in time:

$$\psi_j^{n+1} = \psi_j^n + \frac{i\Delta t}{(\Delta x)^2}(\psi_{j+1}^n - 2\psi_j^n + \psi_{j-1}^n). \tag{1.6}$$

Although writing the finite difference in this way appears relatively intuitive because we want to see forward in time and take a symmetric approach in position, this algorithm is actually highly unstable when implemented, as will be examined later.

1.2.2 Fully Implicit Scheme

Another approach that can be implemented to counter the instability of the fully explicit scheme is to take a backward difference in time and a central difference in space.

$$i\left[\frac{\psi_j^{n+1} - \psi_j^n}{\Delta t}\right] = -\left[\frac{\psi_{j+1}^{n+1} - 2\psi_j^{n+1} + \psi_{j-1}^{n+1}}{(\Delta x)^2}\right] \tag{1.7}$$

Equation 1.7 is best solved in the matrix form

$$\begin{pmatrix} 2+\mu & -1 & 0 & \cdots & 0 \\ -1 & 2+\mu & -1 & \cdots & 0 \\ 0 & -1 & 2+\mu & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 2+\mu \end{pmatrix} \begin{pmatrix} \psi_1^{n+1} \\ \psi_2^{n+1} \\ \psi_3^{n+1} \\ \vdots \\ \psi_J^{n+1} \end{pmatrix} = \mu \begin{pmatrix} \psi_1^n \\ \psi_2^n \\ \psi_3^n \\ \vdots \\ \psi_J^n \end{pmatrix} \tag{1.8}$$

where $\mu = \frac{(\Delta x)^2}{i\Delta t}$, where we can obtain an array of updated array for ψ^{n+1} for every timestep by taking the inverse of the matrix and multiplying it with the right hand side.

While this method is algorithmically stable, the problem that emerges when using it to solve for the time evolution of a physical system is that it is not norm conserving because the matrix multiplication is not unitary. Hence, we can expect the total area under the curve of the probability density function to "leak" probability as the wavefunction evolves.

1.2.3 Crank-Nicolson

In order to correct for the instability of the fully explicit scheme and the lack of norm conservation in the fully implicit scheme, we can combine the two methods by taking their average in the following way:

$$i\left[\frac{\psi_j^{n+1} - \psi_j^n}{\Delta t}\right] = -\frac{1}{2}\left[\frac{\psi_{j+1}^n - 2\psi_j^n + \psi_{j-1}^n + \psi_{j+1}^{n+1} - 2\psi_j^{n+1} + \psi_{j-1}^{n+1}}{(\Delta x)^2}\right] \tag{1.9}$$

which can again be written in matrix form as

$$\begin{pmatrix} 2+\mu & -1 & 0 & \cdots & 0 \\ -1 & 2+\mu & -1 & \cdots & 0 \\ 0 & -1 & 2+\mu & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 2+\mu \end{pmatrix} \begin{pmatrix} \psi_1^{n+1} \\ \psi_2^{n+1} \\ \psi_3^{n+1} \\ \vdots \\ \psi_J^{n+1} \end{pmatrix} = \begin{pmatrix} \mu-2 & 1 & 0 & \cdots & 0 \\ 1 & \mu-2 & 1 & \cdots & 0 \\ 0 & 1 & \mu-2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \mu-2 \end{pmatrix} \begin{pmatrix} \psi_1^{n} \\ \psi_2^{n} \\ \psi_3^{n} \\ \vdots \\ \psi_J^{n} \end{pmatrix} \tag{1.10}$$

where $\mu = \frac{2(\Delta x)^2}{i\Delta t}$ here differs from that of Equation 1.8 by a factor of 2. As with the implicit scheme, we can solve for ψ^{n+1} by taking the inverse of the left hand side matrix and multiplying it on the right hand side. There is, however an extra step involved because of the second matrix which must first be multiplied in to the right hand side vector.

As we will show in the implementation, the Crank-Nicolson approach is unconditionally stable and norm conserving, making it a far more suitable choice than the other two schemes for simulating physical systems. It is also important to note that, because it employs both the forward and backward differences, the Crank-Nicolson method is second order in accuracy, whereas the other two schemes are only first order. This combination of improved stability and accuracy makes it an attractive choice for calculations that involve time evolution.

1.2.4 Including the Potential

While we have thus far not considered the potential term of the Hamiltonian, it is relatively simple to include in the Crank-Nicolson algorithm. If we add to the right side of Equation 1.9 an average of potentials acting on the wavefunction in the following form:

$$\frac{1}{2}(V_j^n \psi_j^n + V_j^{n+1} \psi_j^{n+1}) \tag{1.11}$$

we can rearrange to obtain a modified form of Equation 1.10:

$$\begin{pmatrix} 2+\mu-\frac{i\mu\Delta tV_1}{2} & -1 & 0 & \cdots & 0 \\ -1 & 2+\mu-\frac{i\mu\Delta tV_2}{2} & -1 & \cdots & 0 \\ 0 & -1 & 2+\mu-\frac{i\mu\Delta tV_3}{2} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 2+\mu-\frac{i\mu\Delta tV_J}{2} \end{pmatrix} \begin{pmatrix} \psi_1^{n+1} \\ \psi_2^{n+1} \\ \psi_3^{n+1} \\ \vdots \\ \psi_J^{n+1} \end{pmatrix}$$
$$= \begin{pmatrix} \mu-2+\frac{i\mu\Delta tV_1}{2} & 1 & 0 & \cdots & 0 \\ 1 & \mu-2+\frac{i\mu\Delta tV_2}{2} & 1 & \cdots & 0 \\ 0 & 1 & \mu-2+\frac{i\mu\Delta tV_3}{2} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \mu-2+\frac{i\mu\Delta tV_J}{2} \end{pmatrix} \begin{pmatrix} \psi_1^{n} \\ \psi_2^{n} \\ \psi_3^{n} \\ \vdots \\ \psi_J^{n} \end{pmatrix} \tag{1.12}$$

where we have assumed that V_j is constant in time. By implementing this scheme we can study the time evolution of a wavefunction that is interacting with a potential of any form.

1.3 Implementation

All three time evolution schemes were implemented in C++ using Visual Studio 2013. Because these equations contain complex numbers in all cases, values of the class **std::complex<double>** were used to populate vectors of variable type **VecComplex**, as defined by the Numerical Recipes library. The tridiagonal matrix inversion function **tridag.h** had to be modified to accomodate complex numbers.

For the explicit scheme a simple time evolution iteration was performed with no matrix multiplication necessary. For the implicit scheme the modified function **tridagComplex** was called for every timestep and the probability density $\psi^\star\psi$ was written out to a text file for plotting. For the Crank-Nicolson method, every iteration the right side of the equation was manually multiplied, exploiting the ease of multiplication with a tridiagonal matrix. This new transformed vector was then inserted into the **tridagComplex** function to obtain the new wavefunction. The probability density function was again computed and written to a text file. The evolution of the wavefunctions for the various schemse could be visualized using animations that were made using the gif terminal in Gnuplot.

1.4 Results and Discussion

In all tests the inital form of the wavefunction was a Gaussian wavepacket given by the equation:

$$\psi_o = \frac{1}{(\pi\sigma_0)^{\frac{1}{4}}} e^{-(x-x_0)^2/(2\sigma_0^2)} e^{ik_0x} \tag{1.13}$$

where σ_0 is a Gaussian width parameter and k_0 is a wavenumber that affects the propagation speed of the wave.

All three schemes were first implemented without a potential (other than the infinite potentials at the barriers) to test for norm conservation and diffusion dynamics. For the explicit scheme, shown in **explicit.gif**, we see a rapid explosion of the wavefunction that indicates unstable divergence after just a few timesteps. For the implicit scheme, shown in **implicit.gif**, we see that the wavefunction is well-behaved but that the norm is visibly decreasing with time. The Crank-Nicolson, used for comparison in both animations, is well-behaved and norm conserving. To properly confirm this norm conservation the area under the curve (i.e. $\int_0^L \psi^\star \psi dx$) was calculated for each of the schemes, as shown in Figure 1.1. From this plot it is evident that the Crank-Nicolson is extremely stable and norm-conserving, whereas, as expected, the implicit scheme "leaks" probability and the explicit scheme explodes exponentially.

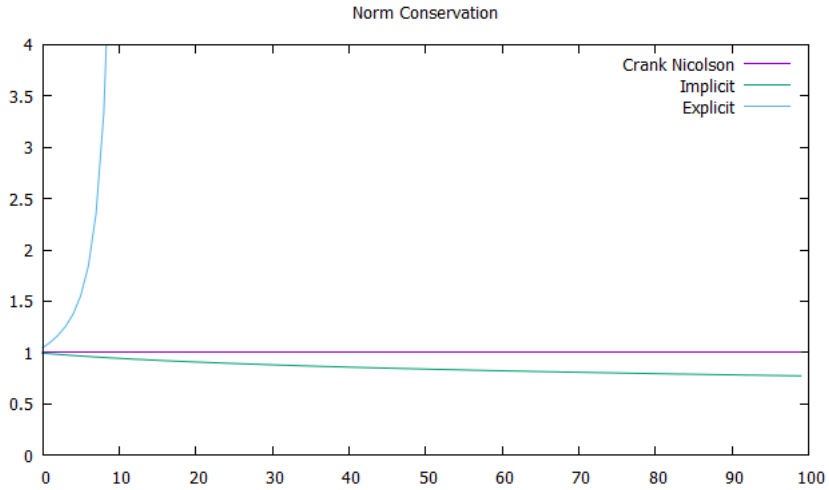

Figure 1.1: Calculated area under the curve of the probability distribution functions for all three schemes as a function of time. Vertical axis is the area (i.e. the total probability of the function) and horizontal axis is the number of timesteps elapsed.

Three different potential barriers were simulated using the Crank-Nicolson method as illustrated in Equation 1.12: $V_o = 3$, 100, and 10000. For all three, the box size was $L = 10$ and the inital position of the wavefunction was at $x_o = L/4$. In all cases the wavefunction propagated towards the barrier and tunneling occurred at varying degrees. At $V_o = 3$ the wavefunction was relatively unperturbed by the barrier because the inital momentum k_0 was high enough relative to the potential that the effect of the barrier was not very significant. In the high barrier limit $V_o = 10000$, the barrier acted much more like an infinite potential. However, some tunneling did occur, and tunneling

was maximized when the wavefunctions on both sides of the barrier coincided to reach a maximum at the edge of the barrier. In the intermediate potential region, $V_o = 100$, about half the wavefunction was reflected and half trasmitted through the barrier. The resulting effect was an interference pattern on both sides which fluctuated as the waves propagated on either side of the potential barrier.

1.5 Conclusion

In all cases, the dynamics of the system were consistent with what would be expected in a one-dimensional quantum system with a finite barrier. While it is important to remember that the exact wavefunction will differ because of the approximations made to create this model, it is clear that the Crank-Nicolson method is a potentially invaluable tool for modeling the dynamics of time-evolving systems. We have shown that while the implicit and explicit schemes are individually flawed, when we take the average of the two, the Crank-Nicolson model proves to be more than the sum of its parts. Whether or not this exact model would be used for more serious applications, using the Crank-Nicolson method in to solve the TDSE in this way provides an excellent framework for simulating the complex behavior of quantum systems.

Bibliography

[1] Meunier, Vincent. Advanced Computational Physics Lecture Slides; Lectures 7: Diffusion Equation