École Normale Supérieure de Lyon
Master Sciences de la Matière 2011
Numerical Analysis Project

Numerical Resolution Of The Schrödinger Equation

Loren Jørgensen, David Lopes Cardozo, Etienne Thibierge

Abstract

In this work we solved the Schrödinger equation numerically in a few usual cases of physics. We first focused on finding suitable numerical tools. In that goal, we computed the free propagation of a Gaussian wave packet by different methods, and we compared the numerical results to the analytical one. We also solved numerically the static Schrödinger equation for different potentials using the finite difference method. Then we applied our code to several situations: tunnel effect in one and two dimensions, alpha radioactivity and particle diffraction.

Contents

1 Introduction: Framework and aims of our project

Since all of us are interested in Quantum Mechanics, both in Condensed Matter and Sub-Atomic Physics, we decided in our Numerical Analysis project to investigate numerically the fundamental equation of Quantum Mechanics: the Schrödinger equation. This postulate of Quantum Mechanics affirms that the time evolution of a particle of mass m, described by a wave function $\psi(\mathbf{r}, t)$, is linked to the potential in which it is moving *via* the relation:

$$-\frac{\hbar^2}{2m}\nabla^2\psi(\mathbf{r},t) + V(\mathbf{r},t)\psi(\mathbf{r},t) = \mathrm{i}\hbar\partial_t\psi(\mathbf{r},t) \tag{1}$$

In this equation, $V(\mathbf{r}, t)$ is the (local) potential, and $-\frac{\hbar^2}{2m}\nabla^2$ the kinetic energy operator. The sum of these two terms is of course the Hamiltonian of the particle. In the following, we choose our system of units such that the mass of the particle m becomes the unit of mass, and $\hbar$ becomes the unit of angular momentum. Furthermore, we always assume that the potential V does not depend on time. The reduced Schrödinger equation we focus on consequently reads:

$$-\frac{1}{2}\nabla^2\psi(\mathbf{r},t) + V(\mathbf{r})\psi(\mathbf{r},t) = \mathrm{i}\partial_t\psi(\mathbf{r},t) \tag{2}$$

Let us now detail the outline of the report, which corresponds roughly to the way we conducted the project. The first part of our work consisted in choosing a numerical method, both fast and precise, and valuable to attack some problems of physical interest. In this scope we implemented several algorithms described in the lectures of Prof. Rolf Walder [1], and analyzed their strength and weakness. Another important question was the choice of the used language. So we compared the speed of a program written on C/C++ with another one written in Matlab on the simplest example: the free propagation of a Gaussian wave packet in one dimension (1d). Then we focused on some cases in hand of Quantum Mechanics, both with our Schrödinger equation solver and with exact diagonalization algorithms, available on Matlab. Particle in a box, harmonic oscillator and 1d tunnel effect are namely studied. Then we extended the code to two dimensions (2d) to study free propagation and tunnel effect in a larger dimension. Last but not least, we briefly dealt with the Gamow model of α radioactivity, and then studied electronic diffraction by one and two splits.

We thank Prof. Rolf Walder for his advice, very helpful to optimize the project.

Please note that all the codes implemented for this project are available online[1].

2 Which numerical tools to choose?

2.1 Programming language

The choice of programming language is a crucial one for the success of the project. On the basis of our knowledge in programming languages, we had basically two options. On the one hand, we had a good experience in Matlab, which is furthermore relatively easy to use. But Matlab is a semi-interpreted language, and is known to be relatively slow. On the other hand we had a basic knowledge in C/C++, which is *a priori* quite faster.

As a result we started the project with some elementary codes in Matlab. Then we worked a lot to improve our knowledge in C++, download several libraries and learn how to use them. We compared then the results obtained with both languages, first to check that our programs were well implemented, and mostly to compare their performance. It appeared clearly that the C++ program was far from being optimized, because of our bad knowledge in this language. The Matlab program was indeed about twice faster than the C++ one ...

As a result we decided to program mainly in Matlab.

2.2 Free propagation of a Gaussian wave packet

We looked at the propagation of a free d-dimensionnal Gaussian wave packet of expression given in equation (3).

$$\psi(\mathbf{r}, t=0) = \mu \exp\left(-\frac{(\mathbf{r}-\mathbf{r_0})^2}{2\sigma^2}\right)\exp(\mathrm{i}\mathbf{k}\cdot\mathbf{r}) \tag{3}$$

[1] `http://perso.ens-lyon.fr/etienne.thibierge/ana_num`

μ is a coefficient so that the norm of ψ at time t, $N(t)$, defined as in (4) is set to 1.

$$N(t) = \int \mathrm{d}^d\mathbf{r} \, |\psi(\mathbf{r}, t)|^2 \tag{4}$$

First we restrict ourselves to the 1d case. This first simulation is compulsory to test numerical methods accuracy and our implementation of them by comparing our simulated results to the analytical solution found in [2].

A simulation obviously requires to discretize time and space in N_t and N_x steps (or cells). We adapted of course the discretization to the considered problem. For our first simulation, the mesh used for the spatial discretization had a step $\Delta x = 0.01$ and the grid was going from 0 to 10, that is to say 1001 points. We looked at free propagation during 300 time steps with $\Delta t = 0.0001$. The initial Gaussian we used had a width $\sigma = 0.1$.

The Laplace operator was calculated with a finite difference method, so that

$$\nabla^2 \psi(x, t) \approx \frac{\psi(x + \Delta x, t) + \psi(x - \Delta x, t) - 2\psi(x, t)}{\Delta x^2} \tag{5}$$

Then $\psi(x, t + \Delta t)$ was computed on each point of the grid using a Runge Kutta of 4th order method. This method is described in the Appendix, and we will explain Sec. 2.3 why we chose it.

Fig. 1 shows three instants of the free propagation of this one-dimensional wave packet, with a wave vector $k = 20$. We found relevant to compute the variation of the norm of our wave packet as an indicator of the validity of our simulation, since it has to be conserved during the Schrödinger's evolution. It was calculated using the rectangles method. The norm loss in this simulation is less than $2.10^{-6}\%$.

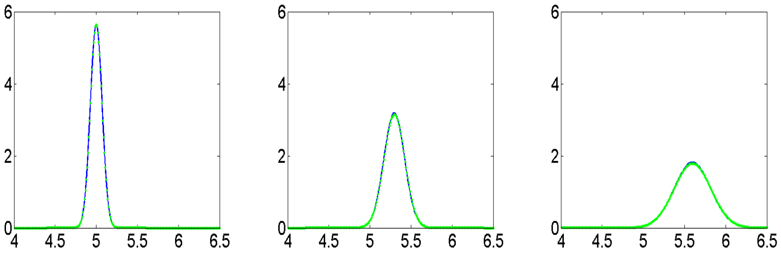

Fig. 1: Blue: simulation. Green: analytical solution. The three figures show $|\psi(x, t)|^2$ at time steps 1, 150 and 300 ($\Delta t = 0.0001$)

On Fig. 2 we see the propagation of the wave packet and its reflexion on a infinite wall that we put at $x = 6$. If qualitatively this reflexion looks normal, we see that it fosters an important loss on the norm of the wave packet of 3 %.

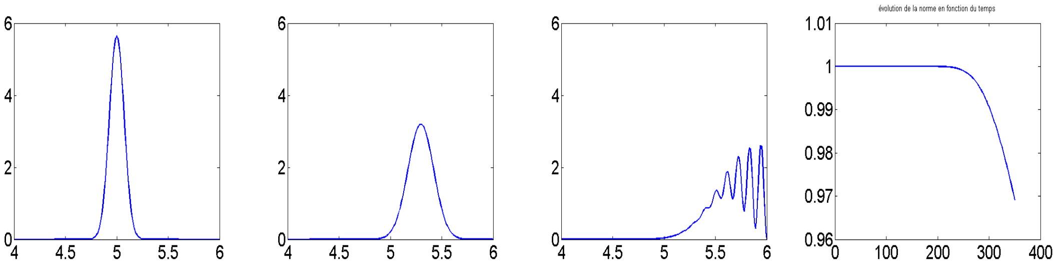

Fig. 2: The three figures show $|\psi(x, t)|^2$ at time steps 1, 150, 350 ($\Delta t = 0.0001$). Far right: evolution of the norm during propagation, the reflexion fosters a loss of norm.

This problem came from the way the boundary conditions were imposed in our code: the value of the wave function on boundaries was put at zero at every time step. Consequently it generated an infinite barrier at both ends of our space interval but a discontinuity in the wave function that was of course non-physical. We solved this problem by putting at boundaries a quickly increasing exponential potential that allowed us to impose smoother boundary conditions and avoid discontinuities but still confine our wave packet in a box. Fig. 3 shows the success of this method: the loss of norm is only less than $3.10^{-5}\%$.

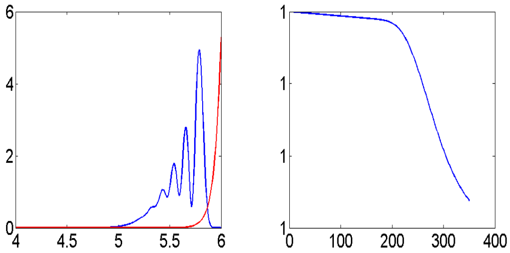

Fig. 3: Left: $|\psi(x,t)|^2$ in blue, and the exponential potential (rescaled to be visible) in red. Right: evolution of the norm during the simulation.

2.3 Comparison of several numerical methods

We repeated the calculation of the Gaussian wave packet free propagation with five different numerical methods in order to choose the best one. We tested :

- an explicit forward Euler method, see [3] p. 120.
- an implicit Crank-Nicholson method, based on a trapezoidal rule, see [4].
- an explicit second order Runge Kutta method (RK2), see [3] p. 124.
- an explicit fourth order Runge Kutta method (RK4), see [3] p. 126 and Appendix.
- an implicit Runge Kutta method called TR-BDF2, with a first stage that is a trapezoidal rule step and a second stage that is a backward differentiation formula of second order, see [3] p. 127.

2.3.1 Stability

We found that for Euler and Runge Kutta methods, values of time step and space step had to be chosen such that $\Delta t/\Delta x^2$ was small enough. If this ratio was larger than a critical value, the solution diverged rapidly. In general, we used a fixed space step $\Delta x = 0.01$. With Euler, the solution diverged for Δt greater than 10^{-6}. With RK2, Δt had to be smaller than 10^{-4} and for RK4, smaller than 10^{-3}. Obviously, to reach the same final evolution when reducing Δt, we had to increase the total number of time steps as much as needed. For this reason, Euler, which does not demand a lot of operations at each time step, turned out to be the longest method, because we waited for 30000 steps instead of only 300 for RK4.

Crank-Nicholson and TR-BDF2 did not have such divergence problems and they remained stable whatever the time step.

2.3.2 Accuracy

For each method, we compared the numerical solution to the analytical solution at $t = 0.3$, for different values of the wave vector. We estimated the difference to the analytical solution by measuring the difference of position and amplitude of the maximum of $|\psi|^2$. We noticed that this difference was larger when the wave vector had a greater value. For example, at $k = 60$ there is a shift of the maximum abscissa of about 10 space steps and the value of this maximum is between 20% and 30% greater than the analytical solution, depending on the method. Conversely at $k = 20$, for all the methods, there is no shift (the maxima are on the same space cell) and the error in the maximum value is about 2% only (see Fig. 4). We interpreted this divergence from the analytical solution by the fact that for a great k, propagation is quick (k being the group velocity since $\hbar = m = 1$) and therefore needs a refined discretization for the simulation to be accurate. This proved relevant when, by taking a smaller Δx, we saw the simulation of free propagation with $k = 60$ fit back to the analytical solution.

For each method, we also tested the conservation of the norm of the wave function during the propagation. The norm is not well conserved by the explicit Euler method at $k = 60$ (10.5% of norm gain) but except for this extreme case the gain or loss in the norm is always below 1% after $t = 0.3$. The conservation of the norm by the Crank-Nicholson method is remarkable (variation always less than 10^{-11}%).

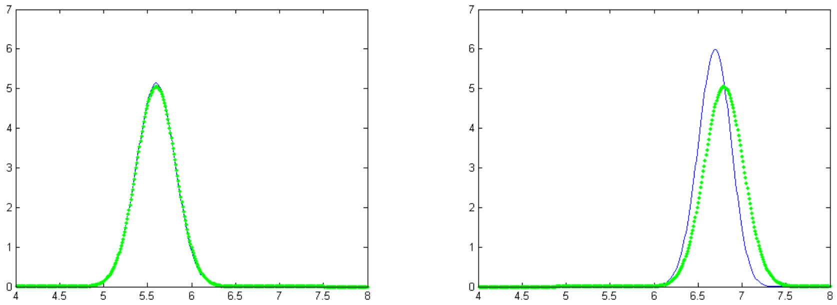

Fig. 4: Blue: simulation. Green: analytical solution. These figures show $|\psi(x,t)|^2$ at time $t = 0.3$ for $k = 20$ on the left and $k = 60$ on the right.

2.3.3 Speed

The explicit methods are the fastest ones because they need fewer operations at each time step than the implicit methods. In the latter, at each time step you need to solve at least one $N_x \times N_x$ linear system, which takes a lot of time. But as we explained before, explicit methods that need a very fine time step are also globally slow because you need a lot of time steps to see a good evolution of your system. So, we compared the time needed to get a similar accuracy with all the methods.

Method	Time step	Calculation duration
Explicit Euler	$\Delta t = 10^{-6}$	83 s
TR-BDF2	$\Delta t = 10^{-4}$	45 s
Crank-Nicholson	$\Delta t = 10^{-4}$	22 s
Runge Kutta 2	$\Delta t = 10^{-5}$	6.5 s
Runge Kutta 4	$\Delta t = 10^{-4}$	1.3 s

Finally, we chose to use the fourth order Runge Kutta method, which was stable for a time step $\Delta t = 0.0001$, worked fast and was accurate enough for what we wanted to do.

3 Cases in hand of Quantum Mechanics

3.1 Eigenvectors for a confined particle

Finding the eigenvectors and eigenvalues of a given Hamiltonian, that is to say solving the static Schrödinger equation, is a fundamental element to understand a problem of quantum mechanics. Here we describe a simple method at 1d that proved efficient to do this task numerically: the finite difference method. The idea is simple: one discretizes the one dimensional space by N_x points and the spatial derivative of Schrödinger equation as shown in equation (6) and imposes strict boundary conditions (7). In this section $x \in \mathbb{N}$ is the point index instead of its position.

$$\psi''(x) = \psi(x+1) + \psi(x-1) - 2\psi(x) \tag{6}$$

$$\psi(1) = \psi(N) = 0 \tag{7}$$

We then generate the Hamiltonian matrix H of elements (8), in which V is the potential (let remind that $\hbar$ and m are equal to 1). By diagonalizing this matrix with Matlab's exact diagonalization algorithm, we get the N_x first eigenvectors and eigenvalues of our problem.

$$\begin{aligned} H_{i,i} &= 1 + V(i) \\ H_{i,i\pm1} &= -\frac{1}{2} \\ H_{i,j} &= 0 \quad \text{otherwise} \end{aligned} \tag{8}$$

To test the accuracy of this method, we applied it on two well known problems of quantum mechanics: the particle confined in a box and the harmonic oscillator. For the first one, we simply put the potential V to zero and matrix elements $(1, N_x)$ and $(N_x, 1)$ to zero, thus imposing the condition that wave function should be zero at both ends of our space. Fig. 5 shows the result for $N_x = 1000$: we see that we get the expected eigenvectors that are simply sinuses without knot for the first one, with one knot for the second, two for the third ... and the eigenvalues are $k^2/2$ with $k = n\pi/N_x$, n being a strictly positive integer.

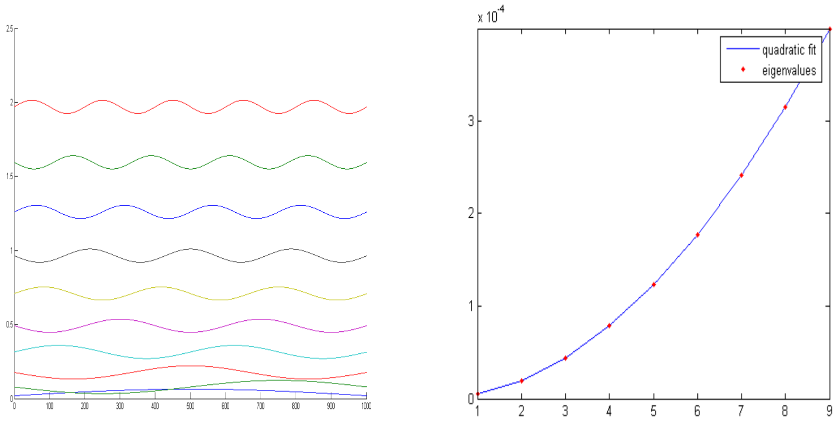

Fig. 5: Left: 10 first eigenvectors of the particle confined in a box problem. The eigenvectors are shifted by their eigenvalues and plotted with respect to the 1000 points space we used for the calculation. Right: red points represent the 10 first eigenvalues calculated numerically and the blue curve is the function $x^2(\pi/N_x)^2/2$ which is the expected evolution of the eigenvalues.

We also found the expected results for the harmonic oscillator problem as shown Fig. 6. We used $N_x = 1000$ again and a harmonic potential for which $\omega = 3 \cdot 10^{-6}$. The eigenvectors found are the expected Hermite polynomials, see [2] and the difference between two consecutive eigenvalues is constant at $\pm 10^{-3}$ percent of its value.

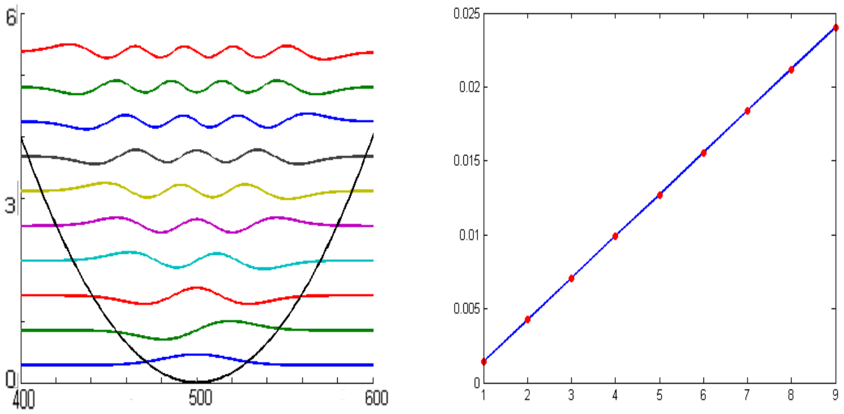

Fig. 6: Left: 10 first eigenvectors of the harmonic oscillator. The quadratic potential is represented in blue. The eigenvectors are shifted by their eigenvalues. Right: the red dots represent the 10 first numerically calculated eigenvalues and the blue curve is the expected analytical evolution $E = \omega(n + 1/2)$ and $n =$ 0,1,2 ...

3.2 Unidimensional tunnel effect

3.2.1 Introduction

Tunnel effect is probably the most famous quantum effect. In a few words, a particle can go through a potential barrier despite its energy is smaller than the barrier height, what would be strictly forbidden classically. This is quantum mechanically allowed due to the wave nature of the particle.

First we solved the eigenvectors problem at 1d by the exact diagonalization method described Sec. 3.1 to emphasize the existence of tunneling phenomena and the link with the barrier height. Then we used our RK4 Schrödinger equation solver to study the dynamics of 1d tunneling.

3.2.2 Eigenvectors problem

We consider a 1d system of length $N_x = 1000$, and a four point wide barrier. Height is not relevant, since it is just a shift of the eigenenergies, as explained in Ref. [2]. On Fig. 7, some eigenvectors are displayed. Tunneling appears clearly.

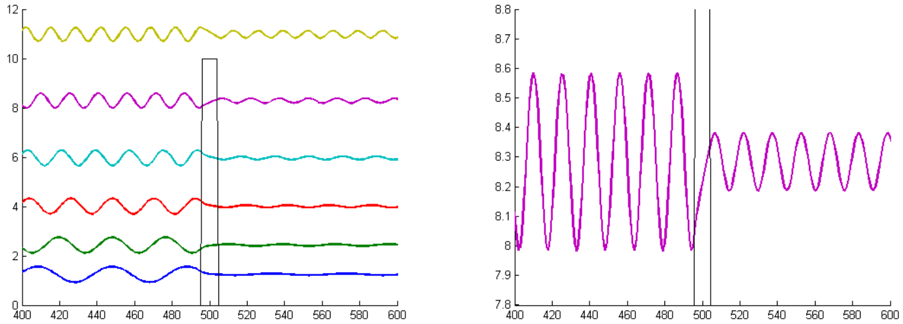

Fig. 7: Some eigenvectors of tunneling problem obtained by the procedure described in Sec. 3.1 with Matlab's exact diagonalization algorithm.

3.2.3 Dynamics of 1d tunneling

We investigate the dynamics of tunneling of a Gaussian wave packet through a Gaussian potential barrier. We could have studied tunneling through a square barrier. Nonetheless our RK4 solver can not deal correctly with discontinuities, so we decided to work on a smoother potential.

In this run, space is divided in 2800 cells of length $\Delta x = 5 \cdot 10^{-3}$, and time in 4000 time steps of length $\Delta t = 2.5 \cdot 10^{-5}$. The incident wave packet is a normalized Gaussian of standard deviation 0.1, and with wave vector $k = 30$. The potential barrier is a Gaussian also of standard deviation 0.01. The normalization of the potential has been chosen such that tunneling occurs.

The squared modulus of the wave packet $|\psi(x,t)|^2$ has been plotted versus x on Fig. 8 at time steps $t_1 = 1$, $t_2 = 950$, $t_3 = 2000$, $t_4 = 3000$ and $t_5 = 4500$.

Let comment these results. First between t_1 and t_2 the wave packet propagates freely and attenuates, as described in Sec. 3.1. At t_2, and still more clearly at t_3 the incident and reflected wave packets interfere with each other, giving rise to the oscillating pattern one can observe. At t_3, the transmitted wave packet appears clearly. Finally at t_4 and t_5 one can see that the reflected and transmitted wave packets evolve freely, and notably attenuate.

Furthermore, to check the precision of our simulation, we computed at each time step the norm of the wave packet. The total loss of norm during the simulating is about 1/10000000, which is extremely good!

3.3 Bidimensional free propagation

We can extend our simulation of free propagation of a Gaussian wave packet to two dimensions. On Fig. 9 is shown the propagation and reflexion on boundary walls of a 2d Gaussian wave packet. We used a Gaussian of form shown Eq. (3) with $\sigma = 0.1$, which is the same along both directions. We chose a wave vector of components $k_x = 0$ and $k_y = 20$. Here, calculation time should be much more taken into account than at 1d. The space grid we used was of length 1 in x direction with a discretization step of 0.025 in this direction and of length 1.5 in y direction and discretization 0.005. We took the discretization in y direction smaller than in the x one because it is the propagation direction. The time step had to be quite small for it is quadratic in the two space discretization steps, so we took $\Delta t = 2.5 \cdot 10^{-5}$. With this configuration, the calculation time was still quite short (a few seconds) and there was no significant improve of the result with a smaller discretization. The loss on the norm of the wave packet during free propagation was less than 0.02 %, which is most satisfactory.

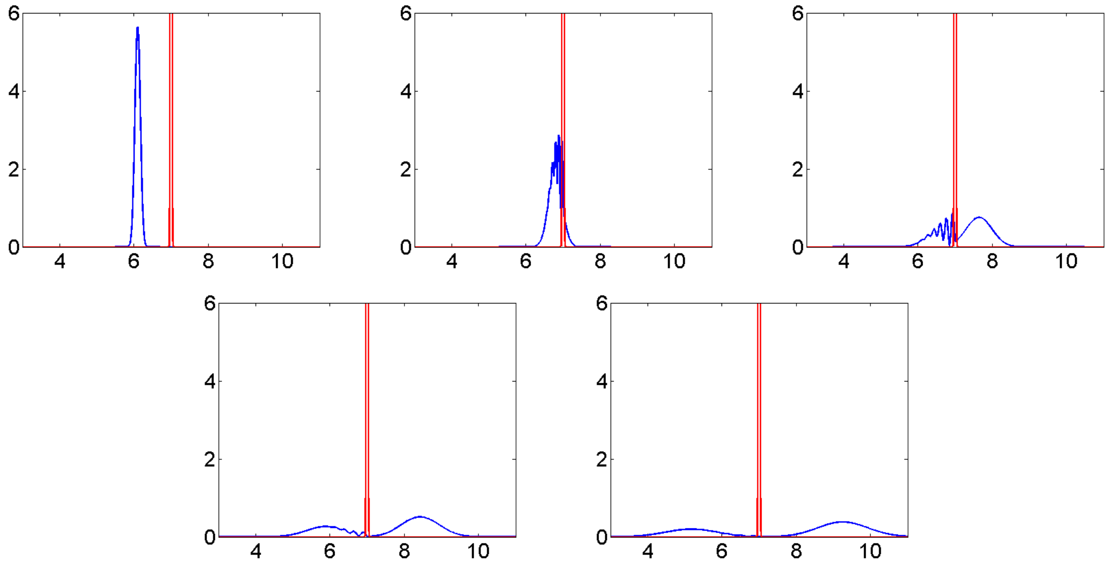

Fig. 8: Dynamics of tunnel effect, investigated with RK4 solver. Here is displayed $|\psi(x,t)|^2$ at time steps $t_1 = 1$, $t_2 = 950$, $t_3 = 2000$, $t_4 = 3000$ and $t_5 = 4500$.

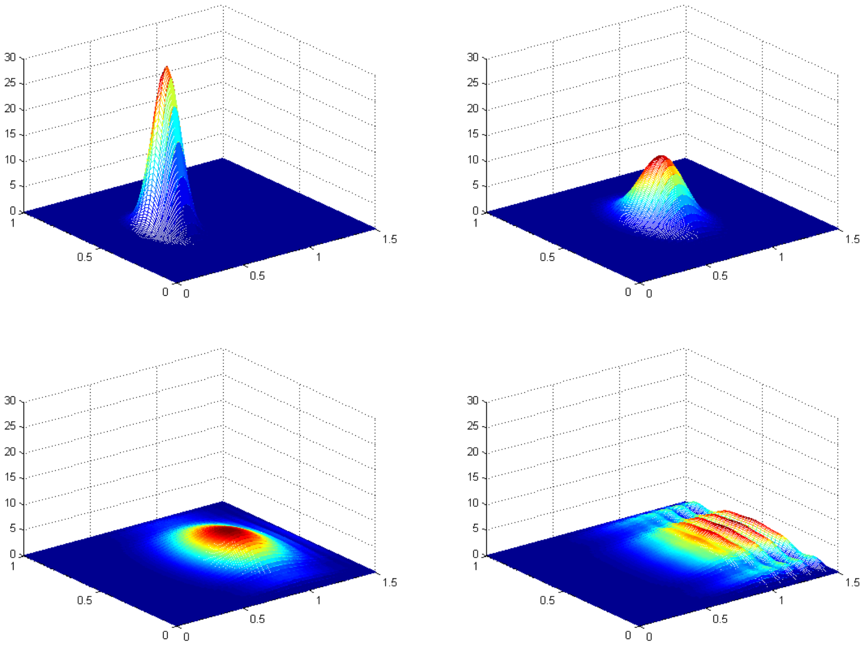

Fig. 9: Free propagation and reflexion of a 2D Gaussian wave packet. The figures from left to right show $|\psi(x,y,t)|^2$ at time steps 1, 500, 1000 and 1300 when reflexion on boundary infinite walls occurs.

3.4 Bidimensional tunnel effect

Let us now focus on 2d tunneling. This effect is investigated with the 2d RK4 solver we implemented.

3.4.1 Studied situation

The initial wave packet is Gaussian both on x and y directions, with the same standard deviation in each direction, as shown on Fig. 9 far left. The barrier we consider is oriented in the y direction, and Gaussian with small standard deviation in the x one. It is simply an extension to higher dimension of the barrier considered previously.

3.4.2 Normal incidence

First we considered the case of normal incidence on the barrier, meaning that $k_y = 0$. A plot of the evolution of the incident wave packet is displayed on Fig. 10. The parameters for this run are $\Delta x = \Delta y = 0.01$ and $\Delta t = 2.5 \cdot 10^{-6}$. As we can expect, nothing really interesting happens by comparison with the 1d case: since the incidence is normal, the y direction is essentially irrelevant. As a result we only observe a juxtaposition of 1d problems, one for each value of y.

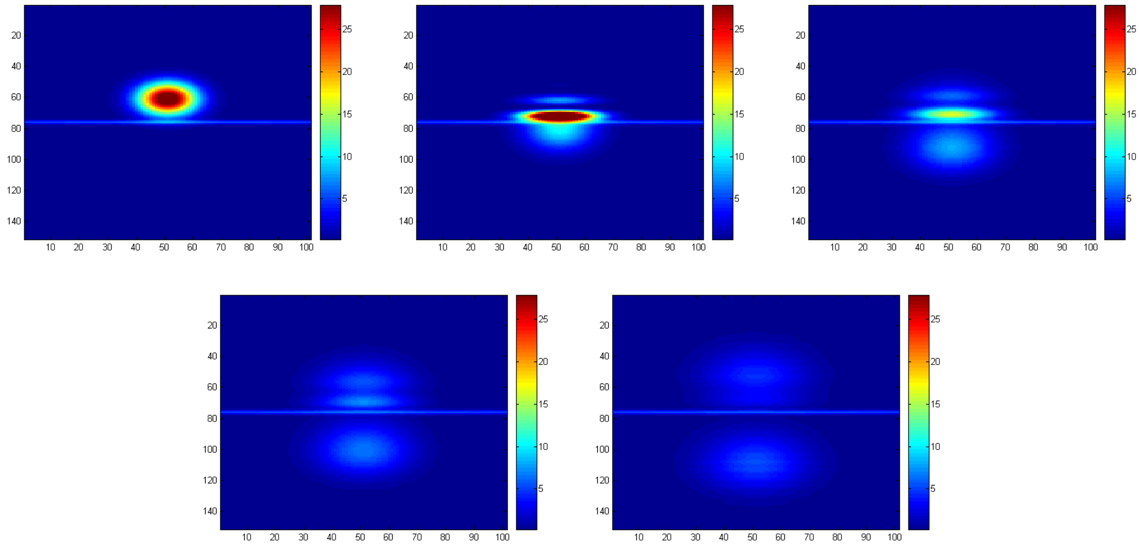

Fig. 10: Dynamics of 2d tunnel effect in normal incidence, investigated with 2d RK4 solver. Here is displayed $|\psi(x, y, t)|^2$ at time steps 1, 250, 400, 500 and 600.

3.4.3 Non-peculiar incidence

Then we looked at the less trivial case of a non-peculiar incidence: now, $k_y \neq 0$. A plot of the evolution of the incident wave packet is displayed on Fig. 11.

An inspection of the plots shows that the wave packet propagates freely along the y direction, despite it is both reflected and transmitted through the barrier along the x direction. This result is not particularly surprising, but nonetheless not completely obvious too. We could have imagined that the reflected part interferes with the incident one in a different way due to the propagation along the y axis, such that the interference pattern would have been disturbed. Our simulation shows that this is not true.

4 Application to atomic and condensed matter physics

4.1 Gamow model of alpha radioactivity

The first concrete case we wished to deal with is the one of the Gamow model of α radioactivity, first introduced in 1928. Briefly speaking, Gamow assumed that the α particles are already bound in the nucleus. The potential they feel is the sum of two terms. The first is due to the averaged nuclear force,

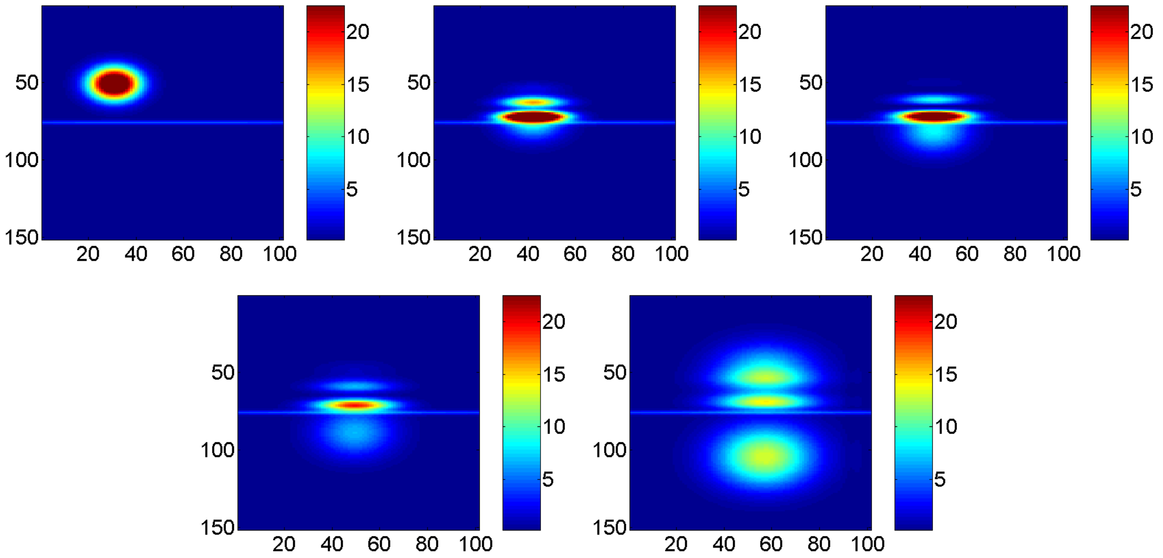

Fig. 11: Dynamics of 2d tunnel effect in non-peculiar incidence, investigated with 2d RK4 solver. Here is displayed $|\psi(x,y,t)|^2$ at time steps 1, 300, 400, 500 and 700. On the last plot, the amplitude has been multiplied by 10 in order to be more visible.

and can be described as a Woods-Saxon potential $V_{\mathrm{WS}}(r)$, see Eq. (9). The second one is simply the Coulomb potential $V_{\mathrm{C}}(r)$, see Eq. (10), due to the electromagnetic interaction between the α particle and the other nucleons.

$$V_{\mathrm{WS}}(r) = -\frac{V_0}{1 - \exp\left(\frac{r-R}{a}\right)} \tag{9}$$

$$V_{\mathrm{C}}(r) = -\frac{V_0}{r} \tag{10}$$

In a first approximation, we assumed that the net potential felt by the α particle reads $V(r) = \kappa V_{\mathrm{WS}}(r) + (1-\kappa)V_{\mathrm{C}}(r)$. It yields to a potential of reasonable shape, which can still be improved by choosing κ as a function of r, e.g. a linear function.

We observed tunneling as described Sec. 3.2, but nonetheless the results are a bit disappointing because they do not enable us to fit concrete data. The relevant physical parameters, such as transmission rate, are actually not easily accessible to our simulation. We decided consequently to spend little time on that, and to focus on more accessible and interesting effects, such as electronic diffraction.

4.2 Particle diffraction

The problem of the diffraction of particles is of great importance: the diffraction of a beam of electrons or atoms is a fundamental experiment to show the quantum wave-particle duality. We here present a simple first approach to this problem: the diffraction of a Gaussian wave packet (representing a particle) by one or two slits. We previously presented our quite successful simulations of free propagation and tunnel effect with a 2d wave packet using the RK4 algorithm. We used the same form of barrier as for the tunnel effect, a 2d Gaussian to avoid discontinuities, but higher in energy to prevent tunneling and with one or two slits cut open into it.

We tried with these simulations to push the spatial precision quite far to obtain a precise diffraction pattern and to use a quite large grid to give space to the wave packet to evolve. Therefore, we used a spatial discretization of 0.01 in both directions with 200 points in x direction (the one along the barrier) and 400 points in the y direction (the one of the propagation of the wave packet). Consequently, the time discretization had to be done with a quite small time step of 5.10^{-6}. This resulted in a huge amount of data and, because these simulations were performed with Matlab, it caused many problems of memory that were solved by saving only 1 over 16 of the time steps. Moreover, the calculation time was long (more than half an hour) for these simulations, which improved the difficulty of working on them. We used 4800 time steps, so 300 were saved.

For both simulations, our initial condition was a Gaussian, as in the previous 2d simulations, with $\sigma = 0.2$ and with a wave vector of components $k_x = 60$ and $k_y = 0$. The loss of norm during our simulations was of the order of 0.01 %, which was quite satisfactory. The width of the apertures was of $0.1 \approx 2\pi/k$(that is to say 10 points), and in the Young slits case the two were separated by this same distance, this allowed to have interferences.

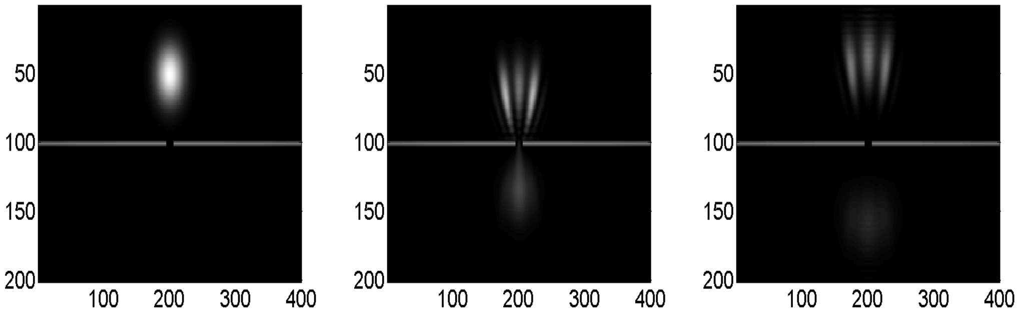

Fig. 12: Diffraction of a Gaussian wave packet by a simple slit. The gray levels represent $|\psi(x,y,t)|^2$. From left to right: time steps 1, 3200 and 4000.

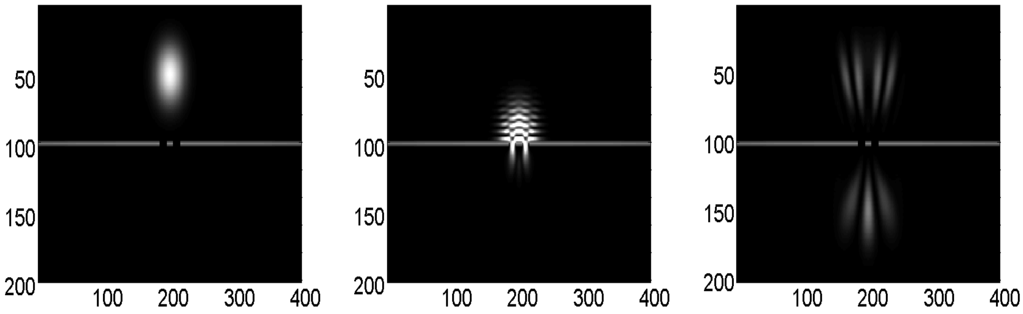

Fig. 13: Diffraction of a Gaussian wave packet by Young slits. The gray levels represent $|\psi(x,y,t)|^2$. From left to right: time steps 1, 1600 and 3680.

The evolution of our wave packet is displayed when diffracted by a simple (Fig. 12) and double slit (Fig. 13). As we can see on these figures, the result looks quite satisfactory in a qualitative manner. The diffracted wave packet in the Young slits obviously shows interferences with three visible peaks. But, if we look more carefully at the diffraction pattern we can see that we should avoid concluding too quickly. Fig. 14 shows the diffraction patterns for the two simulations at the same distance of 60 points away from the slits. These patterns raise many questions. As we can see, the diffraction pattern for the simple slit is not easy to interpret, we do not get a squared sine cardinal as in the well known light diffraction experiment but neither a simple peak (two secondary maxima seem to be rising). We also see that the pattern obtained for the Young slits is qualitatively satisfactory with a central peak and two symmetric secondary peaks. But, in the pattern obtained with light diffraction on Young slits we see an infinity of other smaller peaks on the sides, which is not the case here.

Actually, the diffraction patterns obtained in our simulations are difficult to analyze. The results we first expected were the diffraction patterns obtained with light diffraction, because they are experimentally obtained in experiments of electron or atoms diffraction. However, the light pattern is obtained by the diffraction of a plane wave, infinitely delocalized, which is not the case here: the diffraction of a Gaussian wave packet on Young slits logically results in a spatially finite diffraction pattern. Moreover, our wave packet is time dependent, which makes it even more different from the light case. We can add that, when we compare our pattern obtained with a Gaussian wave packet to the diffraction pattern obtained with light, the secondary peaks are more important in the Gaussian case. Moreover, the light diffraction patterns are calculated in Fraunhofer approximation, which means that we should look at the pattern at a distance d such that $a^2/d \ll 2\pi/k$ with a the size of the aperture. Diffusion of a quantum particle on a potential should also satisfy such conditions for the results to be comparable.

To obtain simulations that satisfy this inequality we should perform our simulation on a space grid at least ten times bigger. This would have been a challenge itself to have such an important grid

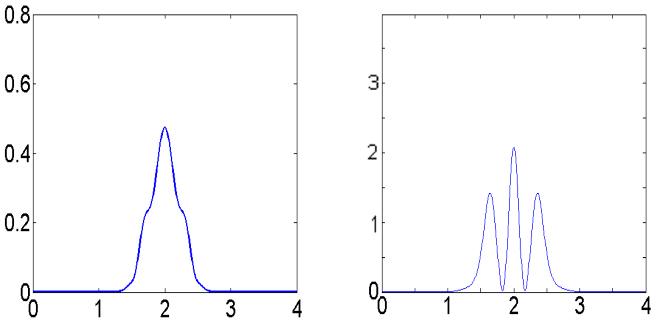

Fig. 14: Diffraction figures taken at time step 3200 and at a distance of 60 points away from the slit (or slits). So, we display the amplitude of the wave packet with respect to position along y direction. Left: diffraction by a simple slit. Right: diffraction by Young slits.

with our discretization because of memory problems and would have required a very long computation time (probably several hours). A possible solution to our memory issue could have been to enlarge the space grid on which the wave packet evolves but with a reduced discretization (larger discretization steps). This seemed to be risky because the numerical solution for the simple slit diffraction showed significant differences depending on the quality of the discretization. Therefore, to be able to perform the numerical experiment in Fraunhofer conditions it would have been necessary to find more adapted numerical methods. This would be a most interesting following to our work and a natural development of it, which we did not tackle by lack of time.

To try to interpret more precisely our results, we also tried to use an analytical solution to the problem of diffraction of a time-dependent Gaussian wave packet outside Fraunhofer conditions. It was a challenge for us to derive on our own a relevant analytical solution to the problem, and we did not fully succeed. The articles we found on the subject [5, 6] were quite technical and would have needed a more thorough bibliography to be used. This is then another natural possible extension of our work, which could prove to be out of our reach anyway.

5 Conclusion and possible future work

In this project we solved numerically the 1d and 2d Schrödinger equation in several cases both of pedagogical and physical interest.

First we had to choose a programming language and an efficient and fast algorithm. In this scope, we implemented several algorithms and compared their performance on the easy example of free propagation in a box. We chose finally to work with the Runge Kutta of 4th order algorithm. In addition to that we implemented this algorithm both in C++ and Matlab to compare. Our little knowledge in C++ made it hard to optimize our code in C++, and so we worked mostly on Matlab.

Our study of cases in hand of Quantum Mechanics (exact diagonalization for confined particle and 1d tunnel effect; dynamics of tunnel effect in one and two dimensions) was then quite succesful. Comparison of our simulations to analytical results made us confident on their accuracy and we used norm conservation as an indicator of the solution convergence.

Last but not least, we applied our code to more concrete (and complex) cases. First we studied the Gamow model of α radioactivity but our results were not easy to interpret quantitatively. Then we focused on the diffraction of a Gaussian wave packet by one and two slits. Our results are satisfactory on a qualitative point of view, since we clearly see diffraction occurrence. The validity of these results was difficult, if not impossible, to check quantitatively. Our numerical method did not allow us to be in Fraunhofer conditions in which the diffraction patterns are well known. We tried without success to derive by ourselves and to find in the literature an analytic solution to this problem.

That might be a first natural extension to our work: derive an analytic solution, and e.g. try to fit our numerical result by the analytical one. The second possible extension is trying to find a more efficient numerical way to compute our solution within Fraunhofer conditions. Data storage and computation time are the most critical points. Another aspect we could have studied is the case of an atomic lattice, in which the potential is sinusoidal both in x and y directions. And more generally there must exist plenty of cases physically interesting in which our Schrödinger equation solver could work!

Appendix: Runge Kutta of 4th order algorithm

Here we describe the Runge Kutta 4 algorithm that was used to solve our differential equation.

Let

$$\frac{\partial \psi}{\partial t} = f(x, \psi) \tag{11}$$

be the differential equation, with here

$$f(x, \psi) = \mathrm{i}\left(\frac{\psi(x+\Delta x, t) + \psi(x-\Delta x, t) - 2\psi(x, t)}{2\Delta x^2} - V(x)\psi(x, t)\right) \tag{12}$$

This fourth order Runge Kutta method consists in four stages, described below :

$$k_1 = f(x, \psi) \tag{13}$$

$$k_2 = f(\psi(x, t) + \frac{\Delta t}{2} k_1) \tag{14}$$

$$k_3 = f(\psi(x, t) + \frac{\Delta t}{2} k_2) \tag{15}$$

$$k_4 = f(\psi(x, t) + \Delta t k_3) \tag{16}$$

and finally:

$$\psi(x, t + \Delta t) = \psi(x, t) + \frac{\Delta t}{6}(k_1 + 2k_2 + 2k_3 + k_4) \tag{17}$$

References

[1] Prof. Rolf Walder and Prof. Doris Follini. Lecture Notes of the course *Numerical Analysis*. École Normale Supérieure de Lyon, Spring semester 2011.

[2] Claude Cohen-Tannoudji, Bernard Diu, and Franck Laloë. *Mécanique Quantique*, volume 1 and 2. Hermann, 1973.

[3] Randall J. LeVeque. *Methods for Ordinary and Partial Differential Equations: Steady-State and Time-Dependent Problems*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, USA, 2007.

[4] Wikipedia - The free encyclopedia. Crank-Nicholson method. `http://en.wikipedia.org/wiki/Crank-Nicolson_method`.

[5] Antonio Zecca. Two-Slit Diffraction Pattern for Gaussian Wave Packets. *International Journal of Theoretical Physics*, 38:911–918, 1999.

[6] Antonio Zecca and Giancarlo Cavalleri. Gaussian Wave Packets Passing Through a Slit: a Comparison between the Predictions of the Schrödinger QM and of Stochastic Electrodynamics with Spin. *Il Nuovo Cimento B*, 112:1–9, 1997.