Simulating the Double Slit Phenomena in a Box

Kim Dahl-Hansen, Fridtjof Ronge Gjengset,
Sebastian Ranum Tenmann & Cecilie Klarpås
(Dated: December 15, 2021)

We have implemented the Crank-Nicolson method in 2+1 dimensions for the numerical estimation of the two-dimensional time-dependent Schrödinger equation in order to simulate the interaction of a Gaussian wave-packet with 0-3 slitted potential barriers. Animations and plots of the simulated quantum align well with the expected interference pattern behaviour of the simulation. The 1D plot of the quantum state observed at a virtual detection screen produced the expected interference pattern for the probability of detecting the particle at certain points along a detection screen for single, double and triple slit.

Our code and animations of different simulations can be found at GitHub[a].

I. INTRODUCTION

In 1801, almost a hundred years before Max Planck proposed his theory that is the basis of modern quantum mechanics[1], a young Thomas Young demonstrated the wave-particle duality of light through his famous interference experiment in which he split a beam of sunlight emerging from a small hole in two by using a paper card[2],and observing the resulting patterns projected by the split beam onto a canvas (Figure 1).

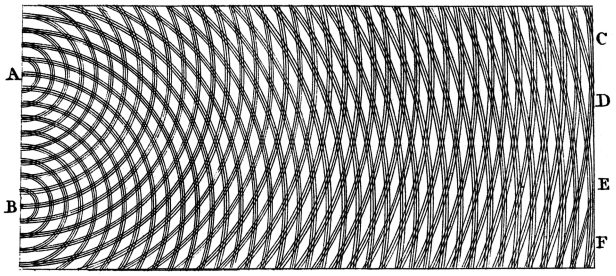

FIG. 1: Youngs' sketch of his experiment[2]. A and B show the split light beams. C, D, E and F indicates the wave-pattern projected onto a canvas.

Here we show our work on creating a numerical simulation of the behaviour of a Gaussian wave-packet evolving in time within a two dimensional box containing a separating barrier with one, two and three slits. The evolution of the Gaussian was calculated for each time step by using a discretised two-dimensional time-dependent Schrödinger equation with the Crank-Nicolson method in 2+1 dimensions. This method provides a good way to see the time-varying probability distribution of the system without having to do the heavy calculations needed to do it analytically. We do recognise that there exists more efficient methods of numerically simulating such a system, however it does provide a simple and unconditionally stable way to do the calculations in the simulation in a timely manner.[3][4]

[a] GitHub: https://github.com/cecilmkl/fys3150/tree/main/project5

II. METHODS

The general formulation of the Schrödinger equation can be written as

$$i\hbar\frac{d}{dt}|\Psi\rangle = \hat{H}|\Psi\rangle$$

Here $\hbar$ is the reduced Planck's constant, $\hat{H}$ is the Hamiltonian operator and $|\Psi\rangle$ is the quantum state.

A single, non-relativistic particle in two dimensions were considered. Position space was used such that the quantum state can be expressed as a complex valued function $\Psi(x, y, t)$, which is commonly referred to as the *wave function*. The Schrödinger equation becomes

$$i\hbar\frac{\partial}{\partial t}\Psi(x,y,t) = -\frac{\hbar^2}{2m}\Big(\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}\Big)\Psi(x,y,t) + V(x,y,t)\Psi(x,y,t).$$

where the potential V encodes the external environment. We explored the case of a potential V that is time-independent, $V = V(x, y)$.

Position space allowed us to take advantage of the *Born rule*

$$p(x,y;t) = |\Psi(x,y,t)|^2 = \Psi^*(x,y,t)\Psi(x,y,t),$$

which relates the quantum state Ψ to the probability, $p(x, y; t)$, of detecting a particle at a position (x, y), if measured at time t. Here, the Ψ^* denotes the complex conjugate.

For simplicity, the problem was scaled so that we were left with dimensionless variables. The "bare" Schrödinger equation can then be written as

$$i\frac{\partial u}{\partial t} = -\frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} + v(x,y)u. \tag{1}$$

where $u(x, y, t)$ denotes the *wave function* in the presence of potential $v(x, y)$. The Born rule can then be expressed as

$$p(x,y;t) = |u(x,y,t)|^2 = u^*(x,y,t)u(x,y,t) \tag{2}$$

assuming the wave function u is properly normalised so that the sum of discrete position probabilities is 1.

The values of the wave function that denotes the state of the system, $u(x, y, t)$ were stored in a matrix U^n of size $M \times M$ for each time step n.

A. Initial and boundary conditions

The box were defined by the discrete positions within $x \in [0, 1]$ and $y \in [0, 1]$, divided into M points in each direction. Dirichlet boundary conditions were assumed, with $u = 0$ along the borders, such that

$$\begin{aligned} u(x = 0, y, t) &= 0 \\ u(x = 1, y, t) &= 0 \\ u(x, y = 0, t) &= 0 \\ u(x, y = 1, t) &= 0 \end{aligned}$$

in the base code, which simplified the Crank-Nicolson algorithm.

The wave function u was be initialised using the Gaussian wave-packet described by

$$\begin{aligned} u(x, y, t = 0) = \exp\left\{-\frac{(x - x_c)^2}{2\sigma_x^2} - \frac{(y - y_c)^2}{2\sigma_y^2}\right\} \\ \cdot \exp\{ip_x(x - x_c) + ip_y(y - y_c)\}, \quad (3) \end{aligned}$$

where (x_c, y_c) are the center coordinates, σ_x and σ_y are the initial widths of the Gaussian, and p_x and p_y are the wave packet momenta.

In order to assure that the particle is trapped inside the box when the simulation is started, the initial conditions $u(x, y, t = 0)$ must adhere to the boundary conditions such that $u(x = 0, y, t = 0) = 0$ and so on.

Additionally, u was normalised in order to ensure that the total probability of the 2D quantum state of the system $p(x, y; t)$ sums to 1, like probabilities should. This means that the initial wave function had to satisfy

$$\sum_{x,y} p(x, y; t = 0) = u^*(x, y, t = 0)u(x, y, t = 0) = 1.$$

Without these initial conditions, we risked that the particle could be positioned outside the box.

B. Evolving the system in time

The algorithm used to evolve the system in time were obtained by discretising the Schrodinger equation(Eq.1) through the Crank-Nicolson approach such[1] that the discrete form of the equation becomes:

$$\begin{aligned} u_{i,j}^{n+1} - r[u_{i+1,j}^{n+1} - 2u_{i,j}^{n+1} + u_{i-1,j}^{n+1}] \\ - r[u_{i,j+1}^{n+1} - 2u_{i,j}^{n+1} + u_{i,j-1}^{n+1}] + \frac{i\Delta t}{2} v_{i,j} u_{i,j}^{n+1} \\ = u_{i,j}^n + r[u_{i+1,j}^n - 2u_{i,j}^n + u_{i-1,j}^n] \\ + r[u_{i,j+1}^n - 2u_{i,j}^n + u_{i,j-1}^n] - \frac{i\Delta t}{2} v_{i,j} u_{i,j}^n. \quad (4) \end{aligned}$$

Where $r \equiv \frac{i\Delta t}{2h^2}$ and subscript $i, j = 1, \ldots, M - 2$ are indices that should not be confused with the imaginary unit. Note also that the superscripts are not powers, but simply the indices for the time variable.

Due to our simple boundary conditions, the equation can be rewritten as 4 using matrices so that

$$A\vec{u}^{\,n+1} = B\vec{u}^{\,n}. \quad (5)$$

Here, $\vec{u}_{i,j}^n$ is a column vector containing all internal points $u_{i,j}^n$ of the xy grid at time step n. Due to space constraints, $\vec{u}_{i,j}^n$ is shown here as a row vector

$$\begin{aligned} \vec{u}^{\,n} = [(u_{1,1}^n, u_{2,1}^n, \ldots, u_{M-2,1}^n), ..., \\ (u_{1,M-2}^n, u_{2,M-2}^n, \ldots, u_{M-2,M-2}^n)], \end{aligned}$$

where the parentheses are added to indicate change in index j, but has no other mathematical meaning. The vector $\vec{u}^{\,n}$ is of size $(M-2)^2$ such that both A and B matrices are of size $(M-2)^2 \times (M-2)^2$ and on the form

$$A = \begin{bmatrix} a_0 & -r & 0 & -r & 0 & \\ -r & a_1 & -r & 0 & -r & \\ 0 & -r & a_2 & -r & 0 & \ldots \\ -r & 0 & -r & a_3 & -r & \\ 0 & -r & 0 & -r & a_4 & \\ & & \vdots & & & \ddots \end{bmatrix},$$

with $r = i\frac{\Delta t}{\Delta x^2 \Delta y^2} = i\frac{\Delta t}{2h^2}$. The B matrix has the same structure, but with positive r on the super- and sub-diagonals and the diagonal elements of both matrices are given by

$$a_k = 1 + 4r + i\frac{\Delta t}{2} v_{ij} \quad (6)$$

$$b_k = 1 - 4r - i\frac{\Delta t}{2} v_{ij}, \quad (7)$$

for $k = 0, ..., (M-2)^2 - 1$.

By using equation 5, the state of the system in the next time-step, n+1, is calculated by two relatively simple operations:

$$B\vec{u}^n = \vec{b}. \quad (8)$$

And the solution is given by

[1] Calculations for arriving to the discrete equation is shown in appendix V A

$$A\vec{u}^{n+1} = \vec{b}, \tag{9}$$

to find the state of the system $\vec{u}^{n+1}$ in the next time-step. As A and B are very large, sparse matrices with most entries equal to 0, this was solved using *Armadillo's* built-in solver for sparse matrices, spsolver. This solver implements the so-called superLU library which implements a parallelised (through CUDA, openMP and MPI) LU decomposition with partial pivoting and forward and backwards susbstitution in order to effectively solve systems involving sparse matrices.

C. Numerical solution

In our simulation, the Gaussian wave-packet was contained to positions $x, y \in [0, 1]$ on a grid consisting of $M \times M$ data-points, meaning that the Gaussian can evolve on an internal grid of $(M-2) \times (M-2)$ data-points that are not affected by the boundary conditions. In order to make the box boundaries and barrier impenetrable by the wave-packet, we set the potential of the boundary and the barrier to be a very large value given as v_0.

This was strictly not necessary as u is bound to be zero along the boundary. Thus, we chose to do this as a formality, because a high potential corresponds to a 'wall', and we wanted the particle to be inside the box. New "walls", i.e. slitted barriers, were constructed in the box by setting the potential to be as high as the boundary conditions for the box itself.

Parallelisation of the simulation code were achieved by letting simulations with different parameters run on different threads(note that this does not cause any speedup for the core algorithm, but makes it faster if we wish to run several simulations at once.), and through Armadillo's spsolver mentioned earlier.

D. Algorithm

The Crank-Nicolson scheme that were used to discretise the Schrödinger equation is a second order partial differential equation, and it is an implicit method that combines the explicit *forward difference scheme* with the *backward difference scheme*. Here the simplified one dimensional case is shown, but further details on how the Crank-Nicolson method is used in two dimensions, and in particular for the Schrödinger equation, is provided in Appendix V A.

The general form for a 1D partial differential equation is given as

$$\frac{\partial u(x,t)}{\partial t} = F(x,t)$$

Algorithm 1 Discretisation with Crank-Nicolson

procedure CRANK-NICOLSON(u, F, n)

$\frac{u_{i,j}^{n+1} - u_{i,j}^{n}}{\Delta t} = F_i^n$ ▷ Forward difference scheme

$\frac{u_{i,j}^{n+1} - u_{i,j}^{n}}{\Delta t} = F_i^{n+1}$ ▷ Backward difference scheme

$\frac{u_{i,j}^{n+1} - u_{i,j}^{n}}{\Delta t} = \theta F_i^{n+1} + (1-\theta)F_i^n$ for $\theta \in [0,1]$ ▷ Linear combination of F.D and B.D.

$\frac{u_{i,j}^{n+1} - u_{i,j}^{n}}{\Delta t} = \frac{1}{2}[F_i^{n+1} - F_i^n]$ ▷ C-N with $\theta = \frac{1}{2}$

The spatial discretisation with Crank-Nicolson as shown in algorithm 1, is applied to the algorithm through the rendering of the matrices A and B used for solving the matrix equation $A\vec{u}_{i,j}^{n+1} = B\vec{u}_{i,j}^{n}$ as described in Section II B. This is particularly visible when the definition of a_k(6) and b_k(7) in Section II B is compared to the discretised Schrödinger equation(4).

The matrix equation is then solved for $\vec{u}_{i,j}^{n+1}$ using the algorithm shown in Algorithm 2. It is important to note that the matrix equation requires the quantum state at time step n to be described as a column vector, not as a state matrix, so the quantum state must be converted between column vector and matrix form in order to evolve the system in time and to represent the state of the system at a given time.

Algorithm 2 Solving differential equation with C-N

procedure SOLVING FOR NEXT STATE $u_{i,j}^{n+1}$ WITH CN(matrix U^n, A, B, time-step n, total time T)

$U^n \to \vec{u}_{i,j}^{n}$ ▷ Transform quantum state

for i in range$(0, T, \Delta t)$ **do**

$\vec{b} = B\vec{u}_{i,j}^{n}$

$\vec{u}_{i,j}^{n} = A^{-1}\vec{b}$ ▷ using a built-in solver or doing it manually

$\vec{u}_{i,j}^{n} \to U^n$ ▷ Transform quantum state back

This algorithm will has a truncation error of $\mathcal{O}(\Delta t^2)$ + $\mathcal{O}(\Delta x^2)$ (+ $\mathcal{O}(\Delta y^2)$ for 2D case), which is fairly good for very small step sizes Δt, Δx and Δy. The last two are equal and defined as h in this project. Additionally, it does not have any stability requirement [3].

E. Control checks

As a control for our simulation, we made sure that there were no gain or loss of probability. As the particle is expected to be found somewhere inside the box, we expect the probability across all data points in the xy-

plane of a time slice to sum to one, i.e.

$$\sum_{i,j} u^*_{i,j} u_{i,j} = 1. \tag{10}$$

We controlled this by making a plot of the calculated total probability's deviation from one. We did this for both a simulation without any barrier, and for a double slit barrier.

It is also worth noting that in general:

$$\begin{aligned} p^n_{ij} &= (a - bi)(a + bi) \\ &= aa - (-b)b + ((-b)a + ab)i \\ &= a^2 + b^2 + (ab - ab)i \\ &= a^2 + b^2. \qquad (11) \\ &= Re(c)^2 + Im(c)^2, \qquad (12) \end{aligned}$$

where c is the complex number.

III. RESULTS & DISCUSSION

A. Controlling simulation an total probability

To obtain a control check on the total probability of our system we used the following parameters $h = 0.005$, $\Delta t = 2.5 \times 10^{-5}$, $T = 0.008$, $x_c = 0.25$, $\sigma_x = 0.05$, $p_x = 200$, $y_c = 0.5$, $\sigma_y = 0.20$, $p_y = 0$ and $v_0 = 0$ for the simulation with no barrier. We then used $v_0 = 1 \times 10^{10}$ when simulating with the double slit barrier. The results are shown in figure 2 and 3 respectively.

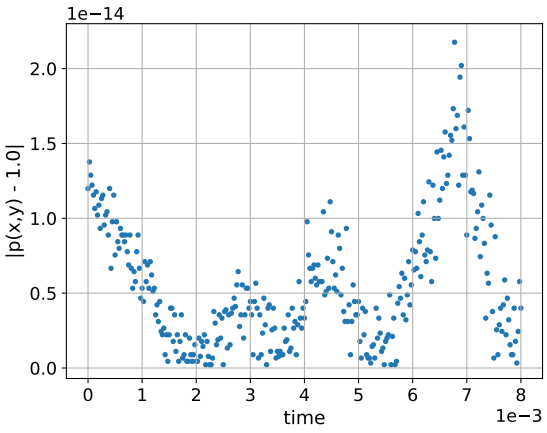

FIG. 2: This plot shows the deviation from 1 of the total probability for an empty box without slits.

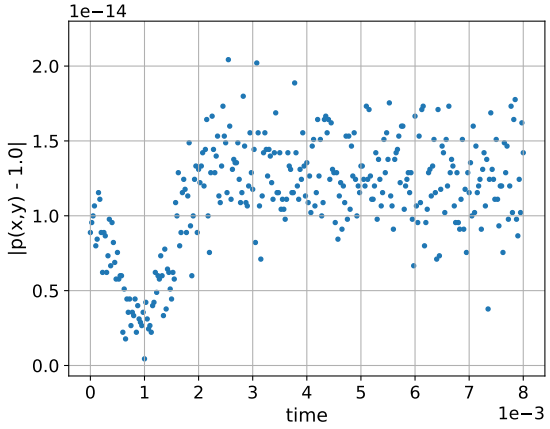

FIG. 3: This plot shows the deviation from 1.0 of the total probability for an empty box with a double slit.

Figure 2 and 3 show that the deviation from 1.0 is of 10^{-14} order of magnitude in both cases. This indicates that our simulation does experience some loss of probability, but the loss itself is negligible.

In the case with no slit in figure 2, we observe that the deviation from 1.0 seems to decrease and increase periodically as the wave-packet interacts with the barrier walls, with a larger amplitude, or deviation, as the simulation time increase.

In contrast, in the probability loss plot of the double slit simulation shown in figure 3, the peaks from the wall bouncing disappear, and the probability deviation from unity is smallest as the wave-packet hits the barrier wall with the double slit at time $t = 0.001$, and then increases and becomes less coherent as the wave-packet looses coherence and bounces chaotically around inside the box.

B. The 2D probability function[2]

Here we present the probability density state p^n_{ij} of the system with a double slit barrier at 3 time points. We ran the simulation with the following parameters: $h = 0.005$, $\Delta t = 2.5 \times 10^{-5}$, $T = 0.002$, $x_c = 0.25$, $\sigma_x = 0.05$, $p_x = 200$, $y_c = 0.5$, $\sigma_y = 0.20$, $p_y = 0$ and $v_0 = 1 \times 10^{10}$. The double slit barrier is located at $x = 0.5$ with a width=0.02, the slits have an aperture of 0.05 in y-direction and are separated by a wall piece with length 0.05 in y-direction. The separation piece is centred at $y = 0.5$.

Figure 4 shows the quantum state at time $t = 0.0$, the initial Gaussian wave-packet visible to the left. The

[2] All the plots that show the quantum state of the system also have an animation, which you can find on our GitHub page. Looking at the animations might give you a better understanding of how the system evolves over time, in comparison to static plots.

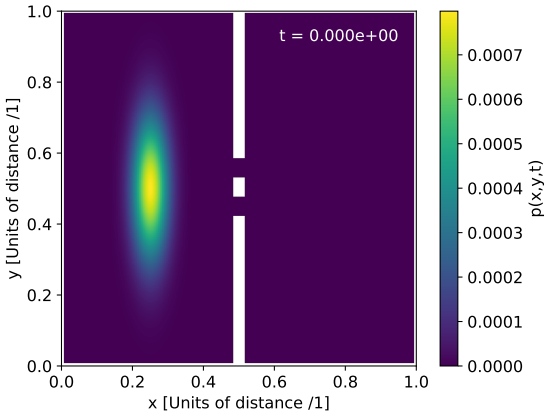

FIG. 4: Shows p_{ij}^n at the initial state $t = 0.00$. x and y shows the spacial dimensions, while the color-bar shows the magnitude of p_{ij}^n. The double slit barrier, marked in white, is located at $x = 0.5$.

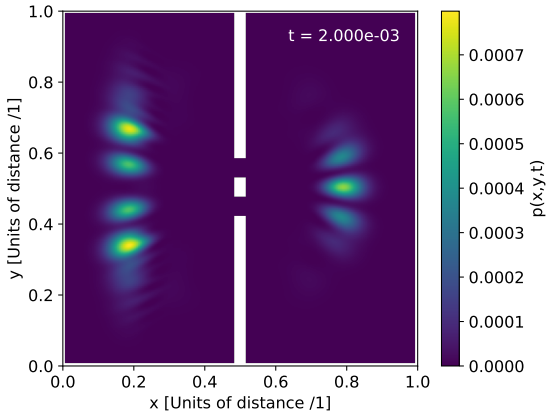

FIG. 6: Shows p_{ij}^n at $t = 0.002$, after the wave-packet has first hit the barrier. x and y shows the spacial dimensions, while the color-bar shows the magnitude of p_{ij}^n. The double slit barrier, marked in white, is located at $x = 0.5$.

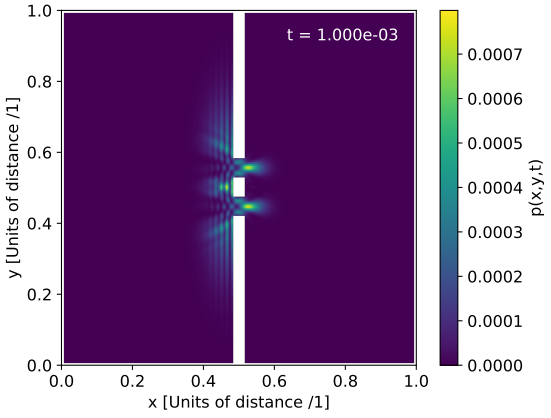

FIG. 5: Shows p_{ij}^n at $t = 0.001$, when the wave packet 'hits' the double slit barrier. x and y shows the spacial dimensions, while the color-bar shows the magnitude of p_{ij}^n. The double slit barrier, marked in white, is located at $x = 0.5$.

initial state looks as expected with the parameters we set for the simulation matching the form and position of the wave-packet we see in 4.

When the wave-packet passes through the slits at time $t = 0.001$ and $t = 0.002$, shown in figure 5 and 6 respectively, we see several interesting observations. Firstly, we can see the interference patters forming on the right hand side of the barrier as a result of the wave-packet passing through both slits at the same time. This pattern is exactly what we expect when plotting the quantum state and shows that our algorithm works as intended, as this reflects what would show up on a detector screen if we conducted the double-slit experiment in real life and is akin to the pattern shown in Young's sketch in figure 1. We also see part of the wave-packet reflected back by hitting the wall, as well as a slight reflected interference pattern caused by the separation piece on the left side of the wall (also observed by G. Kälbermann [5]). This effect is clear in our simulation, due to our code not taking into account possible energy loss in collisions with walls or barriers, and hence the wave can scatter back and forth, while still being observable. However, we have not done a test on possible energy loss that the wave might experience due to truncation errors, so this might be observed by a simulation taking place over a longer time-period.

C. $Re(u_{ij})$ and $Im(u_{ij})$

We can also plot the real and imaginary part of u_{ij} to get a better sense of how they combine to form wavelike interference patterns in the probability function. You can see the imaginary part of u plotted in figure 7-9, and the real part of u in figure 10-12. The plots show the imaginary and real part of u at the same time-steps as we did for the probability function, that is $t = 0.00$, $t = 0.001$ and $t = 0.002$.

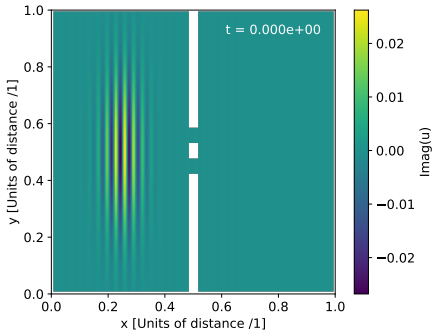

FIG. 7: Shows $Im(u_{ij})$ at the initial state $t = 0.00$. x and y shows the spacial dimensions, while the color-bar shows the magnitude of $Im(u_{ij})$. The double slit barrier, marked in white, is located at $x = 0.5$.

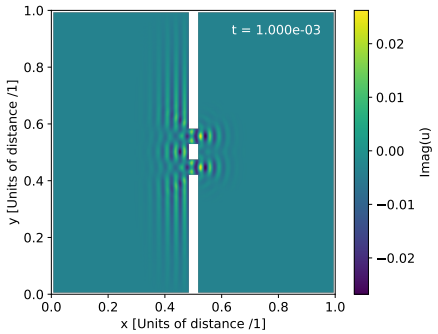

FIG. 8: Shows $Im(u_{ij})$ at $t = 0.001$. x and y shows the spacial dimensions, while the color-bar shows the magnitude of $Im(u_{ij})$. The double slit barrier, marked in white, is located at $x = 0.5$.

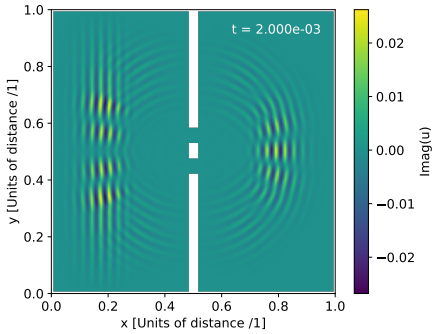

FIG. 9: Shows $Im(u_{ij})$ at $t = 0.002$. x and y shows the spacial dimensions, while the color-bar shows the magnitude of $Im(u_{ij})$. The double slit barrier, marked in white, is located at $x = 0.5$.

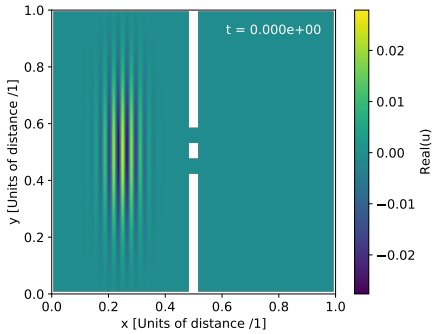

FIG. 10: Shows $Re(u_{ij})$ at the initial state $t = 0.00$. x and y shows the spacial dimensions, while the color-bar shows the magnitude of $Re(u_{ij})$. The double slit barrier, marked in white, is located at $x = 0.5$.

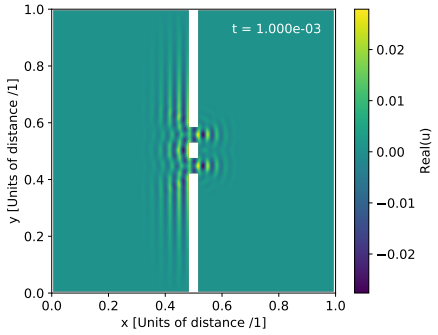

FIG. 11: Shows $Re(u_{ij})$ at $t = 0.001$. x and y shows the spacial dimensions, while the color-bar shows the magnitude of $Re(u_{ij})$. The double slit barrier, marked in white, is located at $x = 0.5$.

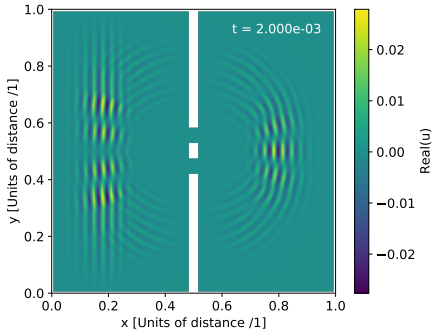

FIG. 12: Shows $Re(u_{ij})$ at $t = 0.002$. x and y shows the spacial dimensions, while the color-bar shows the magnitude of $Re(u_{ij})$. The double slit barrier, marked in white, is located at $x = 0.5$.

Looking at the figures we have for the real and imaginary part of u_{ij}, 7-12 respectively, we see that they look similar to each other, at the same time-steps. We know that $p_{ij}^n = u_{ij}^{n*} u_{ij}^n$, and as we showed in equation 12, p_{ij}^n is just the sum of the squares of the imaginary part and the real part. In other words, p_{ij}^n should show constructive interference between the two, never be negative and only be close to zero if both the imaginary and real part is close to zero. We see that the imaginary and real part are both positive and negative in the region of the wave-packet, and zero everywhere else. As we know that $p = Re(u)^2 + Im(u)^2$, we see that the plots for p, that is figure 4-6, gives the sum of the imaginary and real part, removing the positive/negative wave features from the real/imaginary plots, resulting in a more "smooth" probability plot.

D. Detector screen

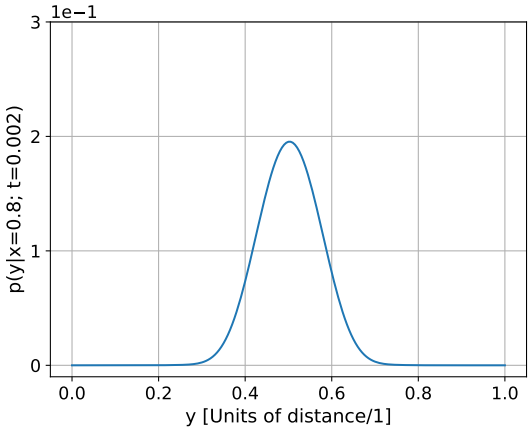

FIG. 13: Shows the probability of finding the particle on the detector screen given $x = 0.8$ and $t = 0.002$, using a single slit barrier.

As we can see from figure 6, we can see the particle has a non-zero probability at $x = 0.8$ at time $t = 0.002$. If we assume that there is a detector screen placed at $x = 0.8$, which spans the entire y-direction, and we extract the y-data at $x = 0.8$ and $t = 0.002$, we get the probability of the particle being detected by the screen at that time. However, as previously discussed the probability of finding the particle inside the box is one. As we want the detector screen to *actually* detect the particle, we need to normalise this probability once again to 1. The resulting pdf's for finding the particle at different points along the y-axis is $p(y|x = 0.8, t = 0.002)$ (and not $p(y|t = 0.002)$ along $x = 0.8$) is shown in figures 13, 14 and 15 for single, double and triple slits respectively. The potential barrier containing the slits is located at $x = 0.5$ with a width=0.02, the slits have an

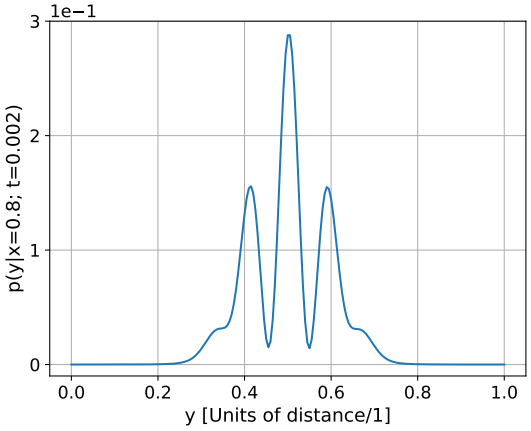

FIG. 14: Shows the probability of finding the particle on the detector screen given $x = 0.8$ and $t = 0.002$, using a double slit barrier.

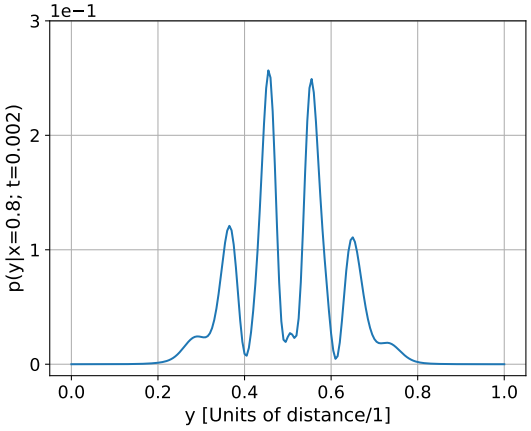

FIG. 15: Shows the probability of finding the particle on the detector screen given $x = 0.8$ and $t = 0.002$, using a triple slit barrier.

aperture of 0.05 in y-direction and are separated by a wall piece with length 0.05 in y-direction. In the single slit case, the slit is centered at $y = 0.5$, while for the triple slit the centre slit is located at $y = 0.5$. In the double slit case, the two slits are separated by a wall piece which is centred at $y = 0.5$.

We see in figure 13 the exact pattern we would expect if we conducted the singe slit experiment. The probability of detecting the particle peaks at $y = 0.5$, which is where the slit is located. We see that the probability declines as we start from $y = 0.5$ and is close to zero for $y = 0.5 \pm 0.2$. If we moved the barrier further back, for example to $x = 0.9$, we would expect a lower peak probability at the centre, and higher probability for detecting the particle further from the center of the detection screen. If we moved the detection screen to $x = 0.6$,

we would expect a higher probability for detecting the particle at $y = 0.5$, but lower probability when straying from the center of the detector screen. The further the screen is located from the slit, the more time the particle has to dissipate outwards like a wave, causing this effect. For the double slit setup shown in figure 14 we see the interference pattern we expect. With the highest probability for detecting the particle directly behind the centre separation piece at $y = 0.5$. We then see a drop in probability caused by destructive interference, and the two second probability peaks caused by constructive interference. In the pdf for the triple slit setup shown in figure 15 we see two central peaks and again a drop in probability in the peaks further from the centre.

IV. CONCLUSION

We have successfully implemented a numerical approach to solving the time evolution of the quantum state of a system with a Gaussian wave-packet interacting with 0-3 slitted barriers in a two dimensional box. Simulations of the system's quantum state were conducted successfully, as well as interactions with a detector screen. The resulting animations and plots of the system at various time points shows interference patterns consistent with experimental data.

Our simulation was verified to not experience a significant gain or loss of probability and can thus be considered a fairly robust method for simulating systems such as this.

Further work could explore both other types of wave-packets than the Gaussian, as well as increasing the spatial and temporal resolution, and adding another spatial dimension.

[1] J. Mehra, *The Historical Development of Quantum Theory, Vol. 1: The Quantum Theory of Planck, Einstein, Bohr and Sommerfeld. Its Foundation and the Rise of Its Difficulties (1900–1925)* (New York: Springer-Verlag, 1982).
[2] T. Young, *The Bakerian lecture. Experiments and calculation relative to physical optics* (Philosophical Transactions of the Royal Society of London, 1804).
[3] M. Hjort-Jensen, *Computational Physics* (Department of Physics, University of Oslo, 2015) lecture notes Fall 2015.
[4] J. W. Thomas, *Numerical Partial Differential Equations: Finite Difference Methods* (Texts in Applied Mathematics, 1995).
[5] G. Kälbermann, Journal of Physics A: Mathematical and General **35**, 4599–4616 (2002).

V. APPENDIX

A. Discretising the Schrödinger equation

We have the 'bare' Schrödinger equation:

$$i\frac{\partial u}{\partial t} = -\frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} + v(x,y)u. \tag{13}$$

We can discretise the right side of equation 13 in the following way(not for time yet):

$$-\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta x^2} - \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\Delta y^2} + v_{i,j}u_{i,j}. \tag{14}$$

Note that all the is that appear in subscripts are not imaginary numbers, but indices. The Crank-Nicolson approach lets us discretise $i\frac{\partial u}{\partial t}$ and write it as:

$$i\frac{u_{i,j}^{n+1} - u_{i,j}^{n}}{\Delta t} = \frac{1}{2}[F_{i,j}^{n+1} + F_{i,j}^{n}] \tag{15}$$

Where $F_{i,j}$ is some arbitrary function, or in this case equal to the expression we have for the right side as shown in equation 14. However, we want to organise our discrete Schrödinger equation according to the time-steps, so we rewrite equation 15 in the following way:

$$i\frac{u_{i,j}^{n+1} - u_{i,j}^{n}}{\Delta t} = \frac{1}{2}[F_{i,j}^{n+1} + F_{i,j}^{n}] \tag{16}$$

$$iu_{i,j}^{n+1} - iu_{i,j}^{n} = \frac{\Delta t}{2}F_{i,j}^{n+1} + \frac{\Delta t}{2}F_{i,j}^{n} \tag{17}$$

$$iu_{i,j}^{n+1} - \frac{\Delta t}{2}F_{i,j}^{n+1} = iu_{i,j}^{n} + \frac{\Delta t}{2}F_{i,j}^{n} \tag{18}$$

$$\tag{19}$$

We now insert the expression we have for $F_{i,j}$, that is expression 14. We also recognize that having the same step-length in both x and y direction, we have $\Delta x = \Delta y = h$. We can now write the discrete Schrödinger equation as:

$$\begin{aligned} iu_{i,j}^{n+1} + \frac{\Delta t}{2}\left(\frac{u_{i+1,j}^{n+1} - 2u_{i,j}^{n+1} + u_{i-1,j}^{n+1}}{h^2} + \frac{u_{i,j+1}^{n+1} - 2u_{i,j}^{n+1} + u_{i,j-1}^{n+1}}{h^2} - v_{i,j}u_{i,j}^{n+1}\right) \\ = iu_{i,j}^{n} - \frac{\Delta t}{2}\left(\frac{u_{i+1,j}^{n} - 2u_{i,j}^{n} + u_{i-1,j}^{n}}{h^2} + \frac{u_{i,j+1}^{n} - 2u_{i,j}^{n} + u_{i,j-1}^{n}}{h^2} - v_{i,j}u_{i,j}^{n}\right) \end{aligned} \tag{20}$$

We then multiply the equation with $(-i)$ and rewrite it to be a little more tidy:

$$\begin{aligned} u_{i,j}^{n+1} - \frac{i\Delta t}{2h^2}[u_{i+1,j}^{n+1} - 2u_{i,j}^{n+1} + u_{i-1,j}^{n+1}] - \frac{i\Delta t}{2h^2}[u_{i,j+1}^{n+1} - 2u_{i,j}^{n+1} + u_{i,j-1}^{n+1}] + \frac{i\Delta t}{2}v_{i,j}u_{i,j}^{n+1} \\ = u_{i,j}^{n} + \frac{i\Delta t}{2h^2}[u_{i+1,j}^{n} - 2u_{i,j}^{n} + u_{i-1,j}^{n}] + \frac{i\Delta t}{2h^2}[u_{i,j+1}^{n} - 2u_{i,j}^{n} + u_{i,j-1}^{n}] - \frac{i\Delta t}{2}v_{i,j}u_{i,j}^{n}. \end{aligned} \tag{21}$$

As $\frac{i\Delta t}{2h^2}$ is only a constant we can define $r \equiv \frac{i\Delta t}{2h^2}$, and simply write:

$$\begin{aligned} u_{i,j}^{n+1} - r[u_{i+1,j}^{n+1} - 2u_{i,j}^{n+1} + u_{i-1,j}^{n+1}] - r[u_{i,j+1}^{n+1} - 2u_{i,j}^{n+1} + u_{i,j-1}^{n+1}] + \frac{i\Delta t}{2}v_{i,j}u_{i,j}^{n+1} \\ = u_{i,j}^{n} + r[u_{i+1,j}^{n} - 2u_{i,j}^{n} + u_{i-1,j}^{n}] + r[u_{i,j+1}^{n} - 2u_{i,j}^{n} + u_{i,j-1}^{n}] - \frac{i\Delta t}{2}v_{i,j}u_{i,j}^{n}. \end{aligned} \tag{22}$$

Where $r \equiv \frac{i\Delta t}{2h^2}$. Equation 22 now expresses the discrete version of the 'bare' Schrödinger equation showed in 13.'