Studying the stability of discretisation schemes in solving partial differential equations

Gustav Bredal Erikstad, Eirik Elias Prydz Fredborg
& Marte Cecilie Wegger
(Dated: December 16, 2020)

This article aims to study the numerical stability of a selection of discretisation schemes in solving partial differential equations. The explicit forward Euler, implicit backward Euler and implicit Crank-Nicolson schemes are implemented in solving the one dimensional diffusion equation with various parameter values for spatial resolution, time step size and simulation time. The results are evaluated with respect to the closed form solution using a χ^2-test. The stability of the explicit forward Euler scheme is investigated more closely by solving the two-dimensional diffusion equation. The scheme is found to have significantly more variance for spatial resolution $h > 0.05$. We also find a stability criterion for time resolution of $r > 3.925$ in $\Delta t = h^2/r$.

I. INTRODUCTION

Physical and geophysical processes like dissipation of heat trough a material can be described and modeled using partial differential equations (PDEs). There are several different methods one can use when solving such equations numerically. We want to study the numerical stability of three methods, the explicit forward Euler algorithm, the implicit backward Euler algorithm and the Crank-Nicolson scheme, for solving PDEs, in order to investigate which one of the methods we will classify as the best. We will solve a diffusion equation for both one and two dimensions numerically. Before moving from one to two dimensions, we want to perform the calculations for two different time points, t_1 and t_2, and later compare the results with calculated analytic solutions. The methods are implemented into code by solving matrix equations and by use of the Thomas algorithm. We will also study the truncation errors in order to investigate the stability properties of the different methods. When moving to two dimensions, we only implement the explicit forward Euler algorithm, in order to study this method more closely. We want to study the numerical stability as a function of spatial resolution, $h = \Delta x = \Delta y$, and time resolution Δt. This we want to do in order to find out which Δt, Δx and Δy the method is unstable for. In addition, we will compare the numerical results with the results obtained by use of the analytic solution of the diffusion equation.

II. METHOD

A. Diffusion equation - 1D

We have that the first partial differential equation we want to solve, the diffusion equation for one dimension, is

$$\frac{\partial^2 u(x,t)}{\partial x^2} = \frac{\partial u(x,t)}{\partial t}, \quad t > 0, \quad x \in [0, L], \tag{1}$$

which can be written as

$$u_{xx} = u_t, \tag{2}$$

where we have that the initial condition at $t = 0$ is

$$u(x, 0) = 0, \quad 0 < x < L, \tag{3}$$

where $L = 1$, and is the length of the x-region we are interested in. Then we have that the boundary conditions are

$$u(0, t) = 0, \quad t \geq 0, u(L, t) = 1, \quad t \geq 0. \tag{4}$$

1. *Euler algorithm*

We will look at the explicit forward Euler algorithm with discretised versions of time given by a forward formula,

$$u_t \approx \frac{u(x_i, t_{j+1}) - u(x_i, t_j)}{\Delta t}, \tag{5}$$

and we will look at a centered difference in space, resulting in

$$u_{xx} \approx \frac{u(x_{i+1}, t_j) - 2u(x_i, t_j) + u(x_{i-1}, t_j)}{\Delta x^2}. \tag{6}$$

In addition, we will study the implicit backward Euler algorithm. The expression when we look at a centered difference in space is the same as for the forward Euler algorithm, shown in Equation 6, and when looking at the discretised version of time we have

$$u_t \approx \frac{u(x_i, t_j) - u(x_i, t_{j-1})}{\Delta t}. \tag{7}$$

2. *Crank-Nicolson scheme*

Another method we study is the implicit Crank-Nicolson scheme, with a time-centered scheme at $(x, t + \Delta t/2)$,

$$u_t \approx \frac{u(x_i, t_{j+1}) - u(x_i, t_j)}{\Delta t}, \tag{8}$$

and the spatial second-order derivative is presented as

$$u_{xx} \approx \frac{1}{2}\Big(\frac{u(x_{i+1},t_j)-2u(x_i,t_j)+u(x_{i-1},t_j)}{\Delta x^2} + \frac{u(x_{i+1},t_{j+1})-2u(x_i,t_{j+1})+u(x_{i-1},t_{j+1})}{\Delta x^2}\Big), \tag{9}$$

where we use a time-centered scheme with $t + \Delta t/2$ as center.

B. Matrix Systems

The next step is to insert the equations for the different schemes into the diffusion equation and solving for $u(x_i, t_{j+1})$. By doing this we can rewrite the equations into matrix systems.

1. Forward Euler

The forward Euler scheme gives a diffusion equation on the form

$$\frac{u(x_{i+1},t_j)-2u(x_i,t_j)+u(x_{i-1},t_j)}{\Delta x^2} = \frac{u(x_i,t_{j+1})-u(x_i,t_j)}{\Delta t}, \tag{10}$$

where we introduce $\alpha = \Delta t/\Delta x^2$ and solve for $u(x_i, t_{j+1})$,

$$u(x_i,t_{j+1}) = \alpha u(x_{i+1},t_j) + (1-2\alpha)u(x_i,y_j) + \alpha u(x_{i-1},t_j). \tag{11}$$

This can be written as a matrix equation on the form

$$\mathbf{v_{j+1}} = (\mathbf{I} + \alpha\mathbf{A})\mathbf{v_j}, \tag{12}$$

where $\mathbf{I}$ is the identity matrix, $\mathbf{v_j}$ is defined as

$$\mathbf{v_j} = \begin{bmatrix} u(x_1,t_j) \\ u(x_2,t_j) \\ \vdots \\ u(x_{n-1},t_j) \\ u(x_n,t_j) \end{bmatrix}, \tag{13}$$

and the matrix $\mathbf{A}$ is

$$\mathbf{A} = \begin{bmatrix} -2 & 1 & 0 & \cdots & \cdots & \cdots & 0 \\ 1 & -2 & 1 & 0 & \ddots & \ddots & \vdots \\ 0 & 1 & -2 & 1 & 0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & 0 & 1 & -2 & 1 & 0 \\ \vdots & \ddots & \ddots & 0 & 1 & -2 & 1 \\ 0 & \cdots & \cdots & \cdots & 0 & 1 & -2 \end{bmatrix}. \tag{14}$$

2. Backward Euler

Next we replicate the above analysis for the implicit backward Euler scheme, again by first inserting Equations 6 and 7 into 2

$$\frac{u(x_{i+1},t_j)-2u(x_i,t_j)+u(x_{i-1},t_j)}{\Delta x^2} = \frac{u(x_i,t_j)-u(x_i,t_{j-1})}{\Delta t} \tag{15}$$

$$\Rightarrow\ u(x_i,t_{j-1}) = -\alpha u(x_{i+1},t_j) + (1+2\alpha)u(x_i,t_j) - u(x_{i-1},t_j), \tag{16}$$

which is then written on matrix product form as

$$\mathbf{v_{j-1}} = (\mathbf{I} - \alpha\mathbf{A})\mathbf{v_j}, \tag{17}$$

which can be further simplified

$$\begin{aligned}\mathbf{v_j} &= (\mathbf{I}-\alpha\mathbf{A})^{-1}\mathbf{v_{j-1}} \\ &= (\mathbf{I}-\alpha\mathbf{A})^{-1}((\mathbf{I}-\alpha\mathbf{A})^{-1}\mathbf{v_{j-2}}) \\ &= \ldots = (\mathbf{I}-\alpha\mathbf{A})^{-j}\mathbf{v_0}.\end{aligned} \tag{18}$$

3. Crank-Nicolson scheme

Finally the same process is applied to the Crank-Nicolson scheme. First equations 8 and 9 are inserted into Equation 2,

$$\frac{1}{2\Delta x^2}\Big[u(x_{i+1},t_j)-2u(x_i,t_j)+u(x_{i-1},t_j) + u(x_{i+1},t_{j+1})-2u(x_i,t_{j+1})+u(x_{i-1},t_{j+1})\Big] = \frac{u(x_i,t_{j+1})-u(x_i,t_i)}{\Delta t}. \tag{19}$$

With this scheme we solve for the t_{j+1} terms,

$$-\alpha u(x_{i+1},t_{j+1}) + (2+2\alpha)u(x_i,t_{j+1}) - \alpha u(x_{i-1},t_{j+1}) = \alpha u(x_{i+1},t_j) + (2-2\alpha)u(x_i,t_j) + \alpha u(x_{i-1},t_j), \tag{20}$$

which is then written on matrix form

$$(2\mathbf{I} - \alpha\mathbf{A})\mathbf{v_{j+1}} = (2\mathbf{I} + \alpha\mathbf{A})\mathbf{v_j}, \tag{21}$$

which in turn means that

$$\mathbf{v_{j+1}} = (2\mathbf{I} - \alpha\mathbf{A})^{-1}(2\mathbf{I} + \alpha\mathbf{A})\mathbf{v_j}. \tag{22}$$

C. Truncation errors

We want to find the truncation errors of the three different schemes, in order to investigate the stability properties of the methods. This is done by Taylor expanding $u(x, y)$ for the two Euler schemes, and Taylor expanding around a central time point $t' \equiv t + \Delta t/2$ for the Crank-Nicolson scheme. These derivations are presented in the appendix (section VI).

D. Analytic solutions - 1D case

We also want to find the analytical solution to Equation 1, by use of the initial conditions and boundary conditions given in expressions 3 and 4, in order to later compare the numerical solutions with the analytical. It is beneficial to introduce an equation which we set to be the solution to the diffusion equation, Equation 1,

$$v(x,t) = u(x,t) + f(x), \tag{23}$$

and who has the Dirichlet boundary conditions,

$$v(0,t) = v(1,t) = 0. \tag{24}$$

We then need to find the initial condition. We get that

$$v(x,0) = u(x,0) + f(x) = f(x), \tag{25}$$

since we have that $u(x,0) = 0$. Now we can insert the expression for $v(x,t)$ into Equation 1, and get

$$\frac{d^2 f}{dx^2} = 0, \tag{26}$$

indicating that

$$f(x) = Ax + B, \tag{27}$$

where A and B are constants we need to find. First we find B,

$$\begin{aligned} &v(0,t) = u(0,t) + f(0) = 0, \\ &\Rightarrow f(0) = A \cdot 0 + B = 0, \\ &\Rightarrow B = 0, \end{aligned} \tag{28}$$

since $u(0,t) = 0$. Next, we find A,

$$\begin{aligned} &v(1,t) = u(1,t) + f(1) = 0, \\ &\Rightarrow f(1) = A \cdot 1 + 0 = -1, \\ &\Rightarrow A = -1, \end{aligned} \tag{29}$$

since $u(1,t) = 1$. We now have that $A = -1$ and $B = 0$, and we see that the initial condition is

$$v(x,0) = f(x) = -x. \tag{30}$$

It is now easier to first solve and find $v(x,t)$ and then find $u(x,t)$. The further calculations can be found in the appendix (section VI), and we obtain that the analytical solution to Equation 1 is

$$u(x,t) = \sum_{k=1}^{\infty} \frac{2}{k\pi}(-1)^k \sin(xk\pi) e^{-t(k\pi)^2} + x. \tag{31}$$

E. χ^2 analysis

The numerical results, in this study, are compared to the analytical expressions using a χ^2 analysis at given time steps in the simulations. This analysis uses the expression

$$\chi^2 = \frac{1}{N} \sum_{i=1}^{n} \frac{(u_i^{\text{num}} - u_i^{\text{ana}})^2}{u_i^{\text{ana}}}, \tag{32}$$

where N is the total number of evaluated points as defined by $N = 1/\Delta x$, u^{num} are the numerically calculated values and u^{ana} are the corresponding analytical values. This method returns the χ^2 value per evaluated point. The reason for using this and not a total χ^2 value is to better be able to compare the precision of the methods when using different values for Δx.

F. Implementation

After finding the analytic solution to the diffusion equation in one dimension, and finding the truncation errors of all three schemes, we want to implement these algorithms into code. We implement the forward Euler scheme and advance the 1D diffusion model by solving Equation 12. The forward Euler and Crank-Nicolson schemes are implemented with their respective matrix systems, which are solved using the Thomas algorithm as described in Project 1 [1]. The 1D diffusion model as described by Equation 1 is advanced up to times $t_1 = 0.02$ and $t_2 = 0.1$ with the three schemes described. For each of the modeling times, we use $\Delta x = (0.1, 0.01)$. The time-step length used is $\Delta t = \frac{(\Delta x)^2}{2}$. The analytic solution will also be computed for the parameter combinations, and the numerical solutions will be evaluated with respect to this using the χ^2-analysis described above.

G. Diffusion equation - 2D

The next step is to study the diffusion equation in two dimensions. We now have a differential equation which looks like this,

$$\frac{\partial^2 u(x,y,t)}{\partial x^2} + \frac{\partial^2 u(x,y,t)}{\partial y^2} = \frac{\partial u(x,y,t)}{\partial t}, \quad t > 0, \quad x, y \in [0,1]. \tag{33}$$

We will discretise this equation using the forward Euler scheme. In two dimensions the spatial part will look like

$$u_{xy} = \frac{u^m_{i+1,j} - 2u^m_{i,j} + u^m_{i-1,j}}{(\Delta x)^2} + \frac{u^m_{i,j+1} - 2u^m_{i,j} + u^m_{i,j-1}}{(\Delta y)^2}, \tag{34}$$

where u_{xy} is the spatial derivatives in two dimensions, given as

$$u_{xy} = \frac{\partial^2 u(x,y,t)}{\partial x^2} + \frac{\partial^2 u(x,y,t)}{\partial y^2}. \tag{35}$$

The time derivative is discretised the same way as before,

$$u_t = \frac{u^{m+1}_{i,j} - u^m_{i,j}}{\Delta t}. \tag{36}$$

We assume $\Delta x = \Delta y = h$, use $\alpha \equiv \frac{\Delta t}{h^2}$ and solve the diffusion equation given by $u_t = u_{xy}$ with respect to $u^{m+1}_{i,j}$,

$$u^{m+1}_{i,j} = \alpha\left(u^m_{i+1,j} + u^m_{i-1,j} + u^m_{i,j+1} + u^m_{i,j-1}\right) + (1-4\alpha)\, u^m_{i,j}. \tag{37}$$

H. Analytic solutions - 2D case

When solving Equation 33, we say that the boundary conditions and initial condition is

$$u(0,y,t) = u(1,y,t) = 0, \tag{38}$$
$$u(x,0,t) = u(x,1,t) = 0, \tag{39}$$
$$u(x,y,0) = f(x,y), \tag{40}$$

where we define $f(x,y)$ to as the first term in a general Fourier series, so we get that

$$f(x,y) = \sin(\pi x)\sin(\pi y). \tag{41}$$

After further calculations, which is presented in the appendix (section VI), we get that the analytical solution for the 2D case is

$$u(x,y,t) = e^{-2\pi^2 t}\sin(x\pi)\sin(y\pi). \tag{42}$$

I. Numerical stability analysis of the 2D diffusion model

We want to study the numerical stability of the explicit forward Euler scheme in solving the 2D diffusion equation by varying h and Δt.

We set the boundary conditions

$$u(0,y,t) = u(N_x - 1,y,t) = 0, \tag{43}$$
$$u(x,0,t) = u(x,N_y - 1,t) = 0, \tag{44}$$

and the initial condition,

$$u(x,y,0) = f(x,y), \tag{45}$$

where we define $f(x,y) = \sin(\pi x)\sin(\pi y)$.

The 2D diffusion model is computed using Equation 37 for a time $T = 0.02$ for varying h and Δt. When computing for different h-values, we use a static $\Delta t = \frac{h^2_{min}}{r}$, with $r = 4$. The stability of the forward Euler scheme for various Δt is analysed by varying the r-parameter in the area around $r = 4$. For each evaluated parameter set, the model is evaluated by computing the χ^2 per pixel with respect to the closed form solution (Equation 42).

III. ALGORITHMS

The algorithms used in this study is presented is presented below.

Algorithm 1 Explicit forward Euler

for $m = 0, 1, .., N_t$ **do**
  for $i = 1, 2, .., N_x$ **do**
    $u^{m+1}_i = \alpha u^m_{i-1} + (1-2\alpha)u^m_i + \alpha u^m_{i+1}$

Algorithm 2 Implicit backward Euler

Filling matrix $\mathbf{A}$
Filling right-hand side of equation
for $m = 1, 2, .., N_t$ **do**
  $\mathbf{v}^{m-1} = (\mathbf{I} - \alpha\mathbf{A})\mathbf{v}^m$ ▷ Solving the matrix equation using the Thomas algorithm

Algorithm 3 Implicit Crank-Nicolson

Filling matrix $\mathbf{A}$
Filling right-hand side of equation
for $m = 0, 1, .., N_t - 1$ **do**
  $\mathbf{v}^{m+1} = (2\mathbf{I} - \alpha\mathbf{A})^{-1}(2\mathbf{I} + \alpha\mathbf{A})\mathbf{v}^m$ ▷ Solving the matrix equation using the Thomas algorithm

Algorithm 4 2D forward Euler

for $m = 0, 1, .., N_t$ **do**
  for $i = 1, 2, ..., N_{xy} - 1$ **do**
    for $j = 1, 2, ..., N_{xy} - 1$ **do**
      $u^{m+1}_{i,j} = \alpha(u^m_{i+1,j} + u^m_{i-1,j} + u^m_{i,j+1} + u^m_{i,j-1}) + (1-4\alpha)u^m_{i,j}$

IV. RESULTS

A. The implementations in 1D

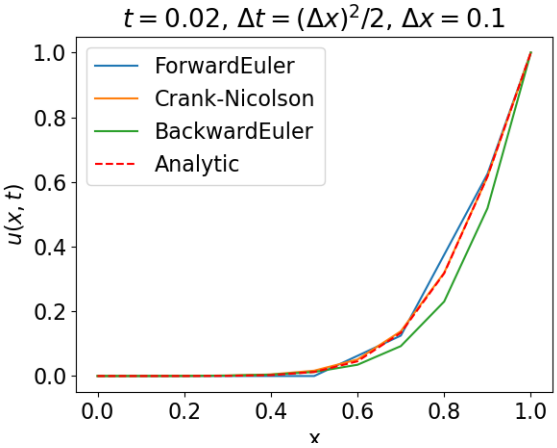

FIG. 1. The simulations using the different schemes for $\Delta x = 0.1$ at $t = t_1 = 0.02$ compared to the analytical solution in Equation 31.

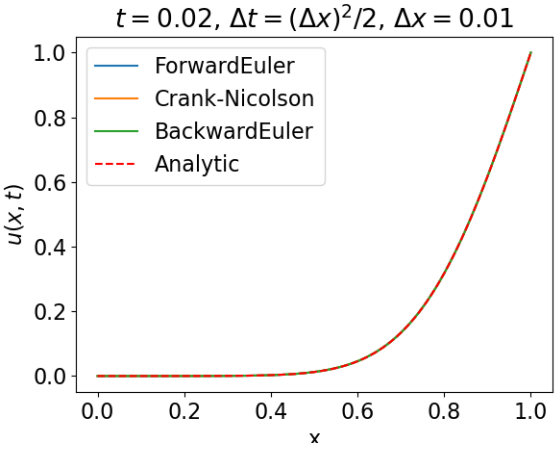

FIG. 2. The simulations using the different schemes for $\Delta x = 0.01$ at $t = t_1 = 0.02$ compared to the analytical solution in Equation 31.

Figure 1 and Figure 2 show the simulations of the different schemes advanced to a time $t_1 = 0.02$ with two different values of Δx, compared with a dashed line for the analytic solution. Figure 1 is produced with $\Delta x = 0.1$ whereas Figure 2 is produced using $\Delta x = 0.01$. There is a clear correlation between Δx and the accuracy of the simulations.

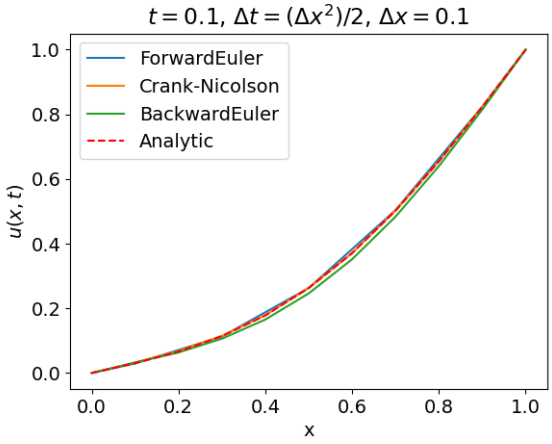

FIG. 3. The simulations using the different schemes for $\Delta x = 0.1$ at $t = t_2 = 0.1$ compared to the analytical solution in Equation 31.

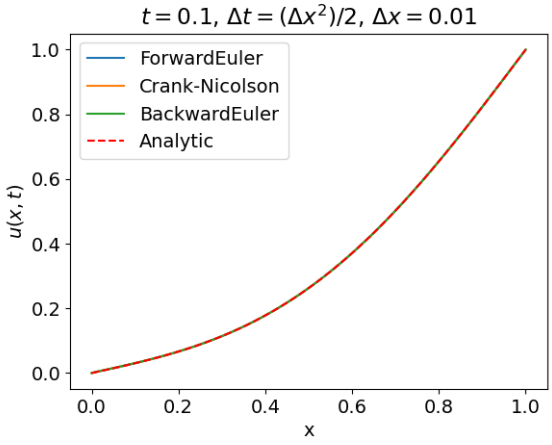

FIG. 4. The simulations using the different schemes for $\Delta x = 0.01$ at $t = t_2 = 0.1$ compared to the analytical solution in Equation 31.

Figure 3 and Figure 4 show the simulations of the different schemes advanced to a time $t_2 = 0.1$ with two different values of Δx, compared with a dashed line for the analytic solution. Figure 3 is produced with $\Delta x = 0.1$ whereas Figure 4 is produced using $\Delta x = 0.01$. There is, again, a clear correlation between Δx and the accuracy of the simulations.

The tables below present the χ^2/N values for the different simulations. Table I shows the accuracy when $\Delta x = 0.1$, and Table II when the simulations where performed with $\Delta x = 0.01$.

	Forward Euler	Backward Euler	Crank-Nicolson
t_1	$3.32 \cdot 10^{-3}$	$6.34 \cdot 10^{-3}$	$7.44 \cdot 10^{-4}$
t_2	$1.45 \cdot 10^{-4}$	$4.89 \cdot 10^{-4}$	$1.90 \cdot 10^{-5}$

TABLE I. χ^2/N of the different methods for $\Delta x = 0.1$

	Forward Euler	Backward Euler	Crank-Nicolson
t_1	$2.96 \cdot 10^{-7}$	$7.74 \cdot 10^{-7}$	$6.56 \cdot 10^{-8}$
t_2	$1.45 \cdot 10^{-8}$	$6.94 \cdot 10^{-6}$	$5.58 \cdot 10^{-7}$

TABLE II. χ^2/N of the different methods for $\Delta x = 0.01$

B. Explicit forward Euler in 2D

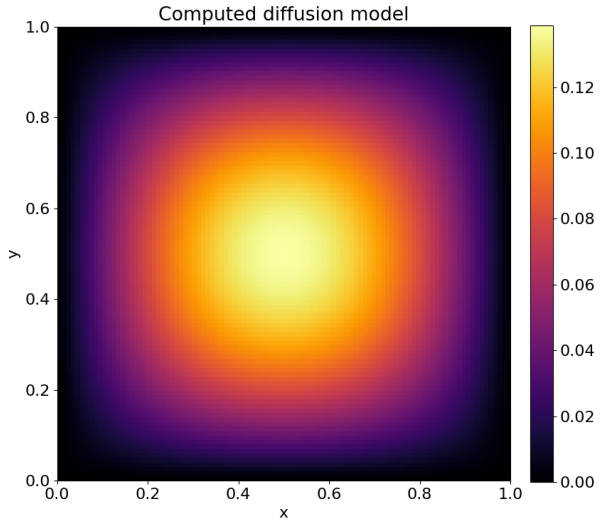

FIG. 5. Colour map of the 2D diffusion model computed by use of explicit forward Euler. The parameters used are $h = 0.01$ and $\Delta t = h^2/4$ for a time period of $t = 0.02$.

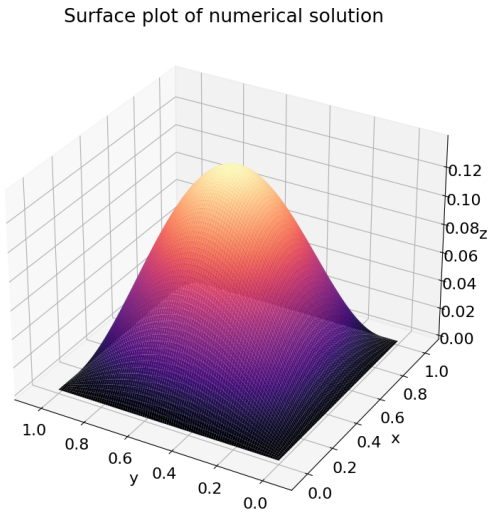

FIG. 6. 3D surface plot of the the 2D diffusion model computed with the explicit forward Euler algorithm, using $h = 0.01$ and $\Delta t = h^2/4$ for a time period of $t = 0.02$.

Figures 5 and 6 illustrate the 2D diffusion model computed by solving Equation 33 using the explicit forward Euler discretisation scheme. The model is advanced for a time of $t = 0.02$ with a spatial step size of $h = 0.01$ and a time step size of $\Delta t = h^2/4$.

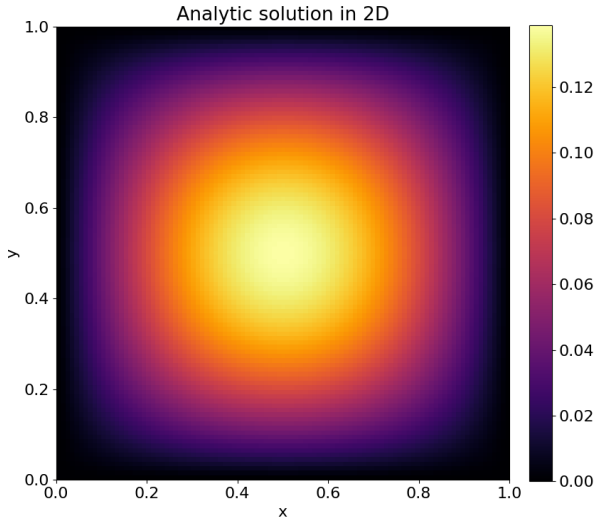

FIG. 7. Colour map of the analytic 2D diffusion model. The parameters used are $h = 0.01$ and $\Delta t = h^2/4$ for a time period of $t = 0.02$.

The closed form solution of the 2D diffusion equation (Equation 42) for $t = 0.02$ is presented in Figure 7.

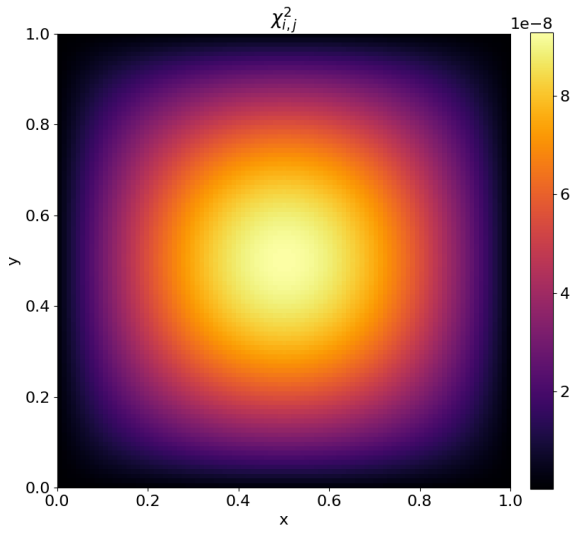

FIG. 8. Colour map of the χ^2-values per pixel for the 2D model.

The χ^2-values for each spaxel of the numerical solution Figure 5 with respect to the analytic solution Figure 7 is presented in Figure 8.

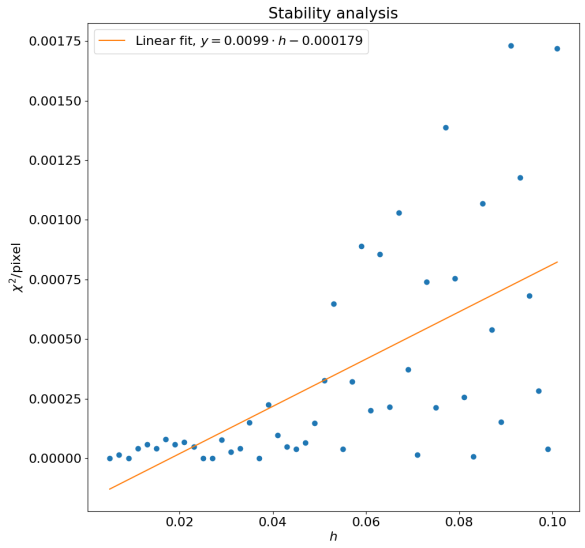

FIG. 9. Scatter plot with linear fit of the χ^2-values per pixel for the 2D model.

The numerical stability is studied in Figure 9 by plotting the χ^2 per pixel as a function of spatial step size, h. The range evaluated is $h \in [0.005, 0.1]$ with a resolution of 0.002. We have the same time step for all h, $\Delta t = 0.005^2/4$.

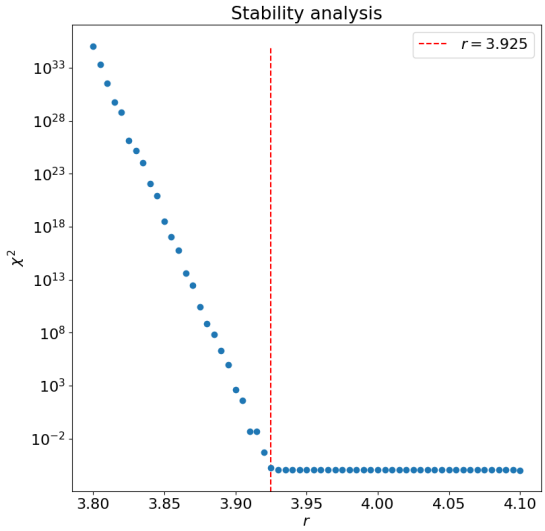

FIG. 10. Scatter plot of χ^2-values for varying r. The y-scale is logarithmic.

The stability of the Forward Euler scheme as a function of Δt is studied by varying the factor r in $\Delta t = h^2/r$. We consider $r \in [3.8, 4.1]$ with a resolution of 0.005. The the largest stable Δt for the Forward Euler scheme is found to be for $r = 3.925$.

All these results can also be found in the Github repository [2].

V. DISCUSSION AND CONCLUSION

When looking at the results of the implementations of the three different methods in one dimension, presented in figures 1, 2, 3 and 4, we see that there is a general tendency for the simulations to approach the analytic solution as time increases. The same is evident from tables I and II. This is expected as the analytic solution approaches an equilibrium of a constant linear function, so as time is progressed the function becomes simpler to simulate.

However in Table II we see that for the backward Euler and Crank-Nicolson schemes the accuracy decreases (as shown by increasing χ^2/N values) as time is progressed from t_1 to t_2. One possible explanation for this is that at $t = 0$ the simulations and the analytic solution are equal and as time progresses the accumulation of numerical error will cause the simulations to diverge. The question the arises for what values of t, Δx and Δt do these effects dominate, and for what schemes.

Therefore, a possible way to improve the results in this study for the one dimension analysis could possibly be to extend the simulations to contain several more values and combinations of t, Δx and Δt.

The numerical scheme with the overall best accuracy compared to the analytic solution seems to be the Crank-Nicolson scheme, only challenged by the forward Euler scheme in the $\Delta x = 0.01$ case.

The numerical solution of the 2D diffusion equation presented in figures 5 and 6, are consistent with the analytic solution presented in Figure 7. Combining this with the fact that the χ^2-values per spaxel in Figure 8 are all $< 10^{-7}$, we find indication that the forward Euler discretisation scheme can be used successfully to solve partial differential equations numerically.

In studying the stability of the scheme as a function of spatial resolution, Figure 9 seem to show a significantly larger variance in the χ^2 per pixel for $h > 0.05$. This indicates that the scheme is "stable" for $h < 0.05$, although a different Δt might have yielded a slightly different result. The stability analysis of the forward Euler scheme as a function of time resolution presented in Figure 10, indicates that the method becomes unstable for $r < 3.925$. Since we only look at the stability analysis as a function of h and r separately, it could be interesting to investigate the numerical stability in a 2D h, r-space.

In conclusion, the most accurate scheme in solving the one-dimensional diffusion equation is found to be the Crank-Nicolson scheme. A strong correlation between numerical accuracy and low Δx and Δt for all schemes is found from the χ^2 analysis. The explicit forward Euler scheme is found to be significantly more stable in solving the two-dimensional diffusion equation for spatial resolution $h < 0.05$. We also find that the method fails completely for $\Delta t > 0.01^2/r$ with $r = 3.925$

VI. APPENDIX

A. Truncation error calculations

In order to determine the truncation error of the two Euler schemes we first Taylor expand the different functions for u used in Equation 2

$$u(x+\Delta x,t) = u + \frac{\partial u}{\partial x}\Delta x + \frac{\partial^2 u}{2\partial x^2}\Delta x^2 + \mathcal{O}(\Delta x^3), \tag{46}$$

$$u(x-\Delta x,t) = u - \frac{\partial u}{\partial x}\Delta x + \frac{\partial^2 u}{2\partial x^2}\Delta x^2 + \mathcal{O}(\Delta x^3) \tag{47}$$

and

$$u(x,t+\Delta t) = u + \frac{\partial u}{\partial t}\Delta t + \mathcal{O}(\Delta t^2), \tag{48}$$

where $u = u(x,t)$ for easier notation.

1. Forward Euler

Inserting Equation 48 into Equation 5 gives

$$\left[\frac{\partial u(x,t)}{\partial t}\right]_{\text{approx}} = \frac{1}{\Delta t}\left[u + \frac{\partial u}{\partial t}\Delta t + \mathcal{O}(\Delta t^2) - u\right] = \frac{\partial u}{\partial t} + \mathcal{O}(\Delta t). \tag{49}$$

Next step is to insert equations 46 and 47 into Equation 6 which gives

$$\begin{aligned}\left[\frac{\partial^2 u(x,t)}{\partial x^2}\right]_{\text{approx}} = \frac{1}{\Delta x}\Big[&\cancel{u} + \cancel{\frac{\partial u}{\partial x}\Delta x} + \frac{\partial^2 u}{2\partial x^2}\Delta x^2 + \mathcal{O}(\Delta x^3) - \cancel{2u} + \cancel{u} - \cancel{\frac{\partial u}{\partial x}\Delta x} \\ &+ \frac{\partial^2 u}{2\partial x^2}\Delta x^2 + \mathcal{O}(\Delta x^3)\Big] = \frac{\partial^2 u}{\partial x^2} + \mathcal{O}(\Delta x^2).\end{aligned} \tag{50}$$

2. Backward Euler

It is quite simple to see that the backward Euler scheme will have truncation errors of the same magnitude as the forward Euler scheme as the only difference in the schemes is in the temporal part. and in this one would get a double negative that cancels out.

3. Crank-Nicolson scheme

For the Crank-Nicolson scheme we need Taylor expansions around a central time point $t' \equiv t + \Delta t/2$

$$u(x+\Delta x,t+\Delta t) = u' + \frac{\partial u'}{\partial x}\Delta x + \frac{\partial u'}{\partial t}\frac{\Delta t}{2} + \frac{\partial^2 u'}{2\partial x^2}\Delta x^2 + \frac{\partial^2 u'}{2\partial t^2}\frac{\Delta t^2}{4} + \frac{\partial^2 u'}{\partial x \partial t}\frac{\Delta t}{2}\Delta x + \mathcal{O}(\Delta t^3), \tag{51}$$

$$u(x-\Delta x,t+\Delta t) = u' - \frac{\partial u'}{\partial x}\Delta x + \frac{\partial u'}{\partial t}\frac{\Delta t}{2} + \frac{\partial^2 u'}{2\partial x^2}\Delta x^2 + \frac{\partial^2 u'}{2\partial t^2}\frac{\Delta t^2}{4} - \frac{\partial^2 u'}{\partial x \partial t}\frac{\Delta t}{2}\Delta x + \mathcal{O}(\Delta t^3), \tag{52}$$

$$u(x+\Delta x,t) = u' + \frac{\partial u'}{\partial x}\Delta x - \frac{\partial u'}{\partial t}\frac{\Delta t}{2} + \frac{\partial^2 u'}{2\partial x^2}\Delta x^2 + \frac{\partial^2 u'}{2\partial t^2}\frac{\Delta t^2}{4} - \frac{\partial^2 u'}{\partial x \partial t}\frac{\Delta t}{2}\Delta x + \mathcal{O}(\Delta t^3), \tag{53}$$

$$u(x-\Delta x,t) = u' - \frac{\partial u'}{\partial x}\Delta x - \frac{\partial u'}{\partial t}\frac{\Delta t}{2} + \frac{\partial^2 u'}{2\partial x^2}\Delta x^2 + \frac{\partial^2 u'}{2\partial t^2}\frac{\Delta t^2}{4} + \frac{\partial^2 u'}{\partial x \partial t}\frac{\Delta t}{2}\Delta x + \mathcal{O}(\Delta t^3) \tag{54}$$

and

$$u(x,t+\Delta t) = u' + \frac{\partial u'}{\partial t}\frac{\Delta t}{2} + \frac{\partial^2 u'}{2\partial t^2}\Delta t^2 + \mathcal{O}(\Delta t^3), \tag{55}$$

$$u(x,t) = u' - \frac{\partial u'}{\partial t}\frac{\Delta t}{2} + \frac{\partial^2 u'}{2\partial t^2}\Delta t^2 + \mathcal{O}(\Delta t^3), \tag{56}$$

where we have used $u' \equiv u(x,t')$.

We then start with the temporal side. Inserting equations 55 and 56 into Equation 8

$$\left[\frac{\partial u(x,t')}{\partial t}\right]_{\text{approx}} = \frac{1}{\Delta t}\left[\cancel{u'} + \frac{\partial u'}{\partial t}\frac{\Delta t}{2} + \cancel{\frac{\partial^2}{2\partial t^2}\Delta t} + \mathcal{O}(\Delta t^3) - \left(\cancel{u'} - \frac{\partial u'}{\partial t}\frac{\Delta t}{2} + \cancel{\frac{\partial^2 u'}{2\partial t^2}\Delta t^2} + \mathcal{O}(\Delta t^3)\right)\right] = \frac{\partial u(x,t')}{\partial t} + \mathcal{O}(\Delta t^2). \tag{57}$$

Next we do the same for the spatial side and obtain

$$\left[\frac{\partial^2 u(x,t')}{\partial x^2}\right]_{\text{approx}} = \frac{\partial^2 u(x,t')}{\partial x^2} - \frac{3}{4}\frac{\partial^2 u(x,t')}{\partial t^2}\frac{\Delta t}{\Delta x^2} + \mathcal{O}\left(\Delta t^3/\Delta x^2\right) \tag{58}$$

B. Derivation of analytic solution

1. 1D case

We say that the solution to Equation 1 is

$$v(x,t) = u(x,t) + f(x), \tag{59}$$

with Dirichlet boundary conditions,

$$v(0,t) = v(1,t) = 0, \tag{60}$$

and the initial condition

$$v(x,0) = f(x) = -x. \tag{61}$$

The next thing we need to do in order to find the analytical solution to Equation 1 is to guess the solution and make an ansatz,

$$v_k(x,t) = X_k(x)T_k(t), \tag{62}$$

where $X(x)$ and $T(t)$ are terms that depend on x and t respectively. We now say that $X(x) = X$ and $T(t) = T$, insert Equation 62 into Equation 1, do a small rewrite and obtain

$$\begin{aligned} X_k'' T_k &= X_k T_k', \\ \frac{X_k''}{X_k} &= \frac{T_k'}{T_k}. \end{aligned} \tag{63}$$

Since the term on the left hand side only depends on x and the term on the right hand side only depends on t, we can say that they are equal to $-C_k^2$, where C_k is a constant. We then get that

$$X_k'' = -X_k C_k^2, \tag{64}$$

$$T_k' = -T_k C_k^2. \tag{65}$$

We have that the general solution to Equation 65 goes as

$$T_k(t) \propto e^{-tC_k^2}, \tag{66}$$

and that the general solution to Equation 64 is

$$X_k(x) = D_k \cos(xC_k) + E_k \sin(xC_k), \tag{67}$$

which we observe is an expression for a harmonic oscillator, where D_k and E_k are constants. First, by use of the boundary conditions, we find D_k,

$$X_k(0) = D\cos(0) + E_k \sin(0) = 0 \;\rightarrow\; D_k = 0, \tag{68}$$

then we find,

$$X_k(1) = E_k \sin(C_k) = 0, \tag{69}$$

which is only true when $C_k = k\pi$ for $k = 1, 2, 3, ...$ Inserting these solutions for X_k and T_k into Equation 62, we then get

$$v(x,t) = E_k \sin(xC_k) e^{-tC_k^2}, \tag{70}$$

We have that every linear combination of the solution is a new solution, so we can write Equation 70 as

$$v(x,t) = \sum_{k=1}^{\infty} E_k \sin(xk\pi) e^{-t(k\pi)^2}. \tag{71}$$

Next, we need to decide the Fourier coefficient E_k, and this can be done by use of the initial condition in Equation 61, so we get that

$$\sum_{k=1}^{\infty} E_k \sin(xk\pi) = -x. \tag{72}$$

By use of Rottmann [3], we can rewrite this and get

$$E_k = -2\int_0^1 x\sin(xk\pi)dx = \frac{2}{k\pi}(-1)^k, \tag{73}$$

where in this case our $f(x)$ is $-x$. We then insert these results into Equation 59 and get that the analytical solution to the diffusion equation for the 1D case is

$$u(x,t) = \sum_{k=1}^{\infty} \frac{2}{k\pi}(-1)^k \sin(xk\pi) e^{-t(k\pi)^2} + x. \tag{74}$$

2. 2D case

We now want to solve Equation 33, where we have that the initial and boundary conditions are given as

$$u(0,y,t) = u(1,y,t) = 0, \tag{75}$$
$$u(x,0,t) = u(x,1,t) = 0, \tag{76}$$
$$u(x,y,0) = f(x,y), \tag{77}$$

where $f(x,y)$ is given as

$$f(x,y) = \sin(\pi x)\sin(\pi y). \tag{78}$$

As we did for the 1D case when finding the analytic solution, we make an educated guess and say that

$$u_{kl}(x,y,t) = W_{kl}(x,y)T_{kl}(t) \tag{79}$$

is the solution to Equation 33. We then insert this into Equation 33 and obtain

$$\frac{\partial^2}{\partial x^2}\Big[W_{kl}(x,y)\Big]T_{kl}(t) + \frac{\partial^2}{\partial y^2}\Big[W_{kl}(x,y)\Big]T_{kl}(t) = \frac{\partial}{\partial t}\Big[T_{kl}(t)\Big]W_{kl}(x,y). \tag{80}$$

By multiplying with $1/W_{kl}T_{kl}$ on both sides of the equation, we can rewrite this expression,

$$\frac{\partial^2}{\partial x^2}\Big[W_{kl}\Big]\frac{1}{W_{kl}} + \frac{\partial^2}{\partial y^2}\Big[W_{kl}\Big]\frac{1}{W_{kl}} = \frac{\partial}{\partial t}\Big[T_{kl}\Big]\frac{1}{T_{kl}} = -C_{kl}^2, \tag{81}$$

and say it is equal to a constant C_{kl} because of the same reasons as for the 1D case. We see that the time dependent term is the same as for the 1D case, so we then have that the general solution is

$$T_{kl}(t) \propto e^{-tC_{kl}^2}. \tag{82}$$

Next, we look at the x and y dependent terms,

$$\frac{1}{W_{kl}}\Big[\frac{\partial^2 W_{kl}}{\partial x^2} + \frac{\partial^2 W_{kl}}{\partial y^2}\Big] = -C_{kl}^2. \tag{83}$$

We can separate $W_{kl}(x,y) = X_k(x)Y_l(y)$, insert this into Equation 83 and get

$$\frac{X_k''}{X_k} = -C_k^2 \quad \text{and} \quad \frac{Y_l''}{Y_l} = -C_l^2, \tag{84}$$

where $C_{kl}^2 = C_k^2 + C_l^2$. We again recognise this as harmonic oscillators, and get that the general solutions are

$$X_k = D_k\cos(C_k x) + E_k\sin(C_k x) \quad \text{and} \quad Y_l = F_l\cos(C_l y) + G_l\sin(C_l y). \tag{85}$$

By doing the exact same approach as for the 1D case, by use of the given boundary conditions, we find that the constants $D_k = F_l = 0$, and that

$$X_k = E_k\sin(k\pi x) \quad \text{and} \quad Y_l = G_l\sin(l\pi y), \quad \Rightarrow C_k = k\pi,\ C_l = l\pi \tag{86}$$

and by inserting Equation 82 and Equation 86 into Equation 79, baking the constants E_k and G_l into one E_{kl} and taking the principle of superposition into account, we get that

$$u(x,y,t) = \sum_{k=1}^{\infty}\sum_{l=1}^{\infty} E_{kl}\sin(k\pi x)\sin(l\pi x)e^{-t[(k\pi)^2+(l\pi)^2]}. \tag{87}$$

Again, by use of Rottmann [3] we find the Fourier coefficient to be

$$E_{kl} = 4\int_0^1\int_0^1 f(x,y)\sin(k\pi x)\sin(l\pi x)dxdy. \tag{88}$$

We have that

$$E_{kl} = 1, \quad \text{when } k = l = 1, \tag{89}$$
$$E_{kl} = 0, \quad \text{otherwise}, \tag{90}$$

and by use of this when solving Equation 88, we get that the final analytic solution to the diffusion equation in the 2D case is

$$u(x,y,t) = e^{-2\pi^2 t}\sin(x\pi)\sin(y\pi). \tag{91}$$

---

[1] Wegger et al. Github FYS4150. Project1. https://github.com/martewegger/fys4150/tree/master/project1, 2020.
[2] Wegger et al. Github FYS4150. Project5. https://github.com/martewegger/fys4150/tree/master/project5, 2020.
[3] Karl Rottmann. Matematisk formelsamling, page 175. Spektrum forlag, 2017.