Diffusion Equation

Simulating the diffusion equation in 1 and 2 dimensions

FYS3150: Computational Physics

Sigurd Sandvoll Sundberg*
Department of Geosciences, University of Oslo

(Dated: December 17, 2020)

We reviewed different finite-difference schemes for solving one- and two-dimensional diffusion equation, including a mathematical analysis of their truncation error and constraints put on Δx and Δt. The upper bound of $\alpha = \Delta t/\Delta x^2$, using von Neumann stability analysis and spectral analysis of a matrix. For the one-dimensional explicit scheme, this upper bound was found to be $\alpha = 1/2$ and for the two-dimensional scheme we have $\alpha = 1/4$. Whilst for the implicit schemes, they where found to be unconditionally stable, however no numerical analysis was done. We also found that the one-dimensional explicit scheme performed better than both the explicit schemes, having a faster convergence. Whilst for the two implicit schemes, the Crank-Nicholson scheme performed significantly better than the backward Euler at approximating the closed-form solutions. An attempt at finding the closed-form solutions for both spatial dimensions was done, with the one-dimensional closed-form solution, matching our analytical solution. The two-dimensional closed-form solution did not created the expected results from the analysis of the problem, and further study of the implementation and closed-form solutions could be interesting for further projects. In addition to study of implicit schemes for solving the two-dimensional diffusion equation and the stability and error of the schemes.

I. INTRODUCTION

Partial differential equations plays a large part numerous branches of the natural sciences and even social sciences. In this article, the diffusion equation is of interest, modeling everything from trace diffusion in standing water to radioactive heating of the mantel. The vast majority of these process do not allow themselves to be solved analytical, forcing us to adapt to the use of numerical solvers to tackle and study these problems. Especially for studying system which we have a hard time, or which could be impossible to gather accurate data throughout the entire system. Establishing functioning numerical methods for solving problems which we have no analytical solution, by first studying problems we have an analytical solution for, is key to for having a greater understanding of the nature around us.

We will review three finite difference schemes for solving the one-dimensional diffusion equation. One explicit method in the forward Euler and two implicit methods, the backward Euler and the Crank-Nicholson scheme. Through the use of the so-called θ-rule, based on Taylor expansions, we will derive the different methods, including short analysis of the constraint put on Δt and Δx in the different scheme. We will also look at the mathematical truncation error of the finite difference schemes and the absolute error, also known as the convergence of the different schemes. Before we extend our problem to two-dimensions, using the explicit scheme for which the same derivation is performed. We also try to find closed-form solutions for both the one-dimensional and two-dimensional case.

II. THEORY

The typical form for the one dimensional diffusion equation for a homogeneous medium is given by

$$\frac{\partial^2}{\partial x^2}u = D\frac{\partial}{\partial t}u,\ x \in [0, L],\ t \in [0, T], \tag{1}$$

where D the diffusion coefficient. D typically is a conglomerate of other physical values for the problem at hand.

The multi-dimensional extension of the diffusion equation takes the form

$$\frac{\partial}{\partial t}\vec{u} = D\Delta\vec{u},\ \vec{u} = (x, y, z) \in \Omega,\ t \in (0, T]. \tag{2}$$

A. Scaling the Diffusion Equation

We can further simplify the diffusion equation by scaling it. Making a dimensional problem into a dimensionless problem, where vastly different systems can be boiled down the same equation. Following the scaling proposed in Langtangen's book on *Scaling of Differential Equations* [1], using the dimensionless variables

$$\hat{x} = \frac{x}{x_c},\ \hat{t} = \frac{t}{t_c},\ \hat{u} = \frac{u}{u_c}. \tag{3}$$

If we insert the dimensionless variables denoted $\hat{x}$, we get

$$\frac{\partial}{\partial \hat{t}}\hat{u} = \frac{t_c D}{L^2}\frac{\partial^2}{\partial x^2}\hat{u},\ \hat{x} \in (0, 1),\ \hat{t} \in (0, \hat{T} = T/t_c]. \tag{4}$$

[*] https://github.com/SigurdSundberg/FYS3150

If we chose $t_c D/L^2 = 1$, it results in $t_c = L^2/D$, and our scaled diffusion equations then takes the form

$$\frac{\partial}{\partial t}u = \frac{\partial^2}{\partial x^2}u. \tag{5}$$

This equation is the one we will use from now on.

The multi-dimensional form follows almost the same scaling and the final form just adds the other dimensional terms.

B. Establishing the Problem

Our dimensionless problem follows equation (5) with $t > 0$, $x \in [0, L]$. This could be the temperature gradient of a rod of length L heated from one end. The initial conditions of the problem is given by

$$u(x, 0) = 0, \quad 0 < x < L \tag{6}$$

with $L = 1$, the length of the $x - region$. The boundary conditions are given as

$$\begin{aligned} u(0, t) = 0, &\quad t \geq 0, \\ u(L, t) = 1, &\quad t \geq 0. \end{aligned} \tag{7}$$

We can rewrite this as follows which will be useful for later

$$v(x, t) = u(x, t) + f(x) = 0. \tag{8}$$

This gives us a new set of boundary and initial conditions given as

$$\begin{aligned} v(x, 0) = u(x, 0) + f(x) = f(x) \\ v(0, t) = u(0, t) + f(0) = 0 \\ v(L, t) = u(L, t) + f(L) = 0 \end{aligned} \tag{9}$$

To determine our function require new initial condition $f(x)$ we can take the initial conditions from (9) which gives us

$$\begin{aligned} v(0, t) = u(0, t) + f(0) = 0 \rightarrow f(0) = 0 \\ v(L, t) = u(L, t) + f(L) = 0 \rightarrow f(L) = -1. \end{aligned} \tag{10}$$

From this we require

$$\frac{\partial^2}{\partial x^2}f = 0, \tag{11}$$

with the above constraints on $f(x)$. From this we find that

$$f(x) = c_1 + c_2 x, \tag{12}$$

applying the constraints on $f(x)$ we find

$$\begin{aligned} f(0) = 0 \rightarrow c_1 = 0 \\ f(L) = -1 \rightarrow c_2 = -\frac{1}{L} \end{aligned} \tag{13}$$

so we have

$$f(x) = -\frac{x}{L}. \tag{14}$$

C. Extending to Two Dimensions

There are multiple ways of extending the problems to two dimensions. One way would be to think of an infinitely wide square, i.e. we have no fluxes to either side[1]. For this we can fix two temperatures at the two remaining edges in short giving us a problem where we have two set of Dirichlet boundary conditions and two edges with Neumann boundary conditions.

By simply just adding a term to our one dimensional diffusion equation, we then have

$$\frac{\partial^2}{\partial x^2}u + \frac{\partial^2}{\partial y^2}u = \frac{\partial}{\partial t}u. \tag{15}$$

If we set our dimensions of the square in question to $x, y \in [0, L]$, we can have

$$\begin{aligned} u(0, y, t) &= 0 \\ u(L, y, t) &= 1 \\ \frac{\partial}{\partial y}u(x, 0, t) &= 0 \\ \frac{\partial}{\partial y}u(x, L, t) &= 0 \end{aligned} \tag{16}$$

Another way is to only use Neumann boundaries, or only Dirichlet boundaries, the options and the systems you can simulate with different choices of boundaries are plentiful. Some examples can be a metal sheet with no fluxes at the edges and a heat source at a specific point, or the heat evolution in a square with three edges at the same temperatures and the last edge at a different temperature. As mention the system we can study are plentiful.

In this article we will use the above mentioned boundary conditions, and they will also be used in finding the analytical solutions for the two dimensional problem.

D. Analytical Solution

For many numerical problems finding closed-form solution are not possible and comparing a numerical scheme to a closed-form solution is a good way to check whether the scheme is implemented correctly or not. This can serve as a baseline for studying how different algorithms differ and what algorithm is the correct choice of the problem, balancing between implementation time and accuracy.

The one- and two-dimensional diffusion equation do have closed-form solution which we can use to study the error of our scheme compared to the analytical solutions.

[1] The mantel of the earth is such a prospect or the atmosphere.

1. One-Dimensional

Starting off with the one-dimensional case we have the equation

$$\frac{\partial}{\partial t} u = \frac{\partial^2}{\partial x^2} u. \tag{17}$$

Studying the equation (8) with the new boundary conditions found in 9 we start off by using method of separation of variables. We start off with

$$v(x,t) = F(x)G(t), \tag{18}$$

inserting into the equation (5) it results in

$$\frac{F''}{F} = \frac{G'}{G}. \tag{19}$$

This equation should hold for all x and t. So we must require that the right-hand side and left-hand side to be equal to a constant. Lets call this constant $-\lambda^2$. This gives us two differential equations

$$F'' + \lambda^2 F = 0, \quad G' = -\lambda^2 G, \tag{20}$$

with general solutions

$$F(x) = A\sin(\lambda x) + B\cos(\lambda x), \quad G(t) = Ce^{-\lambda^2 t}. \tag{21}$$

Using our new boundary conditions (9), we require that $B = 0$ and $\lambda = n\pi/L$.

This gives us a set of solution on the form

$$v(x,t) = \sum_{n=1}^{\infty} A_n \sin(n\pi x/L) e^{-n^2\pi^2 t/L^2}. \tag{22}$$

We can determine the coefficient A_n using the initial condition. We require

$$v(x,0) = f(x) = \sum_{n=1}^{\infty} A_n \sin(n\pi x/L). \tag{23}$$

The coefficient A_n is the Fourier coefficients for the function $f(x)$[2], because of this, A_n is given by

$$A_n = \frac{2}{L}\int_0^L f(x)\sin(n\pi x/L)dx. \tag{24}$$

The integrated form is

$$A_n = \frac{2\pi n\cos(\pi n) - 2\sin(\pi n)}{\pi^2 n^2}. \tag{25}$$

So our analytical solution is given by

$$v(x,t) = \sum_{n=1}^{\infty} \frac{2\pi n\cos(\pi n) - 2\sin(\pi n)}{\pi^2 n^2} \sin(n\pi x/L) e^{-n^2\pi^2 t/L^2}. \tag{26}$$

2. Two-Dimensional

The two-dimensional case follows a similar approach to that of the 1-dimensional solution and we will simply state the result of the closed-form solution.

$$v(x,y,t) = \sum_{n=1}^{\infty}\sum_{m=1}^{\infty} (A_{nm}) \sin\left(\frac{n\pi x}{L}\right)\cos\left(\frac{n\pi y}{L}\right) e^{-\lambda_{nm} t} \tag{27}$$

where we have

$$\begin{aligned} A_{nm} &= \frac{4\left(n\pi\cos(n\pi) - \sin(n\pi)\right)\sin(n\pi)}{n^3\pi^3} \\ \lambda_{nm} &= \pi^2\left(\frac{n^2+m^2}{L^2}\right) \end{aligned} \tag{28}$$

For the full derivation see Appendix A.

E. Discretization

To apply numerical solutions to a continuous problem, we are required to discretize the continuous equations. Taking $x, t, u(x,t)$ as a baseline example we get

$$\begin{aligned} x &\to x_i \in [x_0, x_1, \ldots, x_i, \ldots, x_{n+1}] \\ t &\to t_j \in [t_0, t_1, \ldots, t_j, \ldots, t_n] \\ u(x,t) &\to u(x_i, t_j) = u_i^j \in [u_0^j, u_1^j, \ldots, u_i^j, \ldots, u_{n+1}^j] \end{aligned} \tag{29}$$

With j denoting time index of u, and i the spatial index. The grid-point $x_i = ih$, with $h = (x_{max} - x_{min})/(n+1)$.

This notation will be used when deriving the different numerical schemes.

F. θ-Rule

Taking the continuous equation (5), expanding it using forward in time, centered in space for transforming the continuous equation to a discretized version. We can write this as

$$\begin{aligned} \frac{\theta}{\Delta x^2}\left(u_{i-1}^j - 2u_i^j + u_{i+1}^j\right) + \frac{1-\theta}{\Delta x^2}\left(u_{i+1}^{j-1} - 2u_i^{j-1} + u_{i-1}^{j-1}\right) \\ = \frac{1}{\Delta t}\left(u_i^j - u_i^{j-1}\right). \end{aligned} \tag{30}$$

From this expansion we can create different formulas depending on our choice of θ.

For the following numerical methods we will be using the following Dirichlet boundary conditions

$$u(0,t) = u(L,t) = 0, \quad u_0 = u_n = 0 \tag{31}$$

G. Forward Euler

Choosing $\theta = 0$, we obtain the forward Euler formula,

$$\frac{u_i^{j+1} - u_i^j}{\Delta t} = \frac{u_{i+1}^j + u_{i-1}^j - 2u_i^j}{\Delta x^2} \tag{32}$$

$$u_i^{j+1} = \alpha \left(u_{i+1}^j + u_{i-1}^j\right) + (1 - 2\alpha)\, u_i^j \tag{33}$$

where $\alpha = dt/dx^2$. Following the similar expansion method as used in [3], we obtain the following matrix equation

$$\mathbf{A}u^j = u^{j+1} \tag{34}$$

with $\mathbf{A}$ for a 4 × 4 matrix given as

$$\mathbf{A} = \begin{bmatrix} 1-2\alpha & \alpha & 0 & 0 \\ \alpha & 1-2\alpha & \alpha & 0 \\ 0 & \alpha & 1-2\alpha & \alpha \\ 0 & 0 & \alpha & 1-2\alpha \end{bmatrix}, \tag{35}$$

and

$$u^{j+1} = \begin{bmatrix} u_0^{j+1} \\ u_1^{j+1} \\ u_2^{j+1} \\ u_3^{j+1} \end{bmatrix}, \quad u^j = \begin{bmatrix} u_0^j \\ u_1^j \\ u_2^j \\ u_3^j \end{bmatrix} \tag{36}$$

.

1. Truncation Error

From the Taylor expansions used to find the forward in time centered in space schemes, we know that the truncation error for the forward Euler scheme goes as $\mathcal{O}(\Delta x^2)$ and $\mathcal{O}(\Delta t)$.

2. Constraint

For the forward Euler we have a fairly strict limitation on values of α. The scheme is in other words not stable for values of $\alpha > 1/2$. This constraint comes from the study of the spectral radius of $\rho(\mathbf{A})$ of our matrix $\mathbf{A}$ satisfies the condition

$$\rho(\mathbf{A}) < 1. \tag{37}$$

The spectral radius is defined as[2]

$$\rho(\mathbf{A}) = \max\left\{|\lambda| : \det\left(\mathbf{A} - \lambda\mathcal{I}\right) = 0\right\}. \tag{38}$$

From this we have that if a matrix is positive definite, then the condition is always satisfied. We can rewrite $\mathbf{A}$ as

$$\mathbf{A} = \mathcal{I} - \alpha\mathbf{B}, \tag{39}$$

with

$$\begin{bmatrix} 2 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 2 \end{bmatrix}. \tag{40}$$

The eigenvalues of $\mathbf{A}$ are then given as $\lambda_i = 1 - \alpha\mu_i$, with μ_i being the eigenvalues of $\mathbf{B}$. Using the results found in [2], we find that the eigenvalues of $\mathbf{A}$ then results in

$$-1 < 1 - \alpha 2(1 - \cos(\theta)) < 1, \tag{41}$$

which puts our requirement for α to, $\alpha < (1 - \cos(\theta))^{-1}$, resulting in $\alpha < 1/2$.

H. Backward Euler

The backward Euler formula is obtained by choosing $\theta = 1$ in equation (30), obtaining

$$\frac{u_{i+1}^{j+1} + u_{i-1}^{j+1} - 2u_i^{j+1}}{\Delta x^2} = \frac{u_i^{j+1} - u_i^j}{\Delta t} \tag{42}$$

$$-\alpha \left(u_{i+1}^{j+1} + u_{i-1}^{j+1}\right) + (1 + 2\alpha)\, u_i^{j+1} = u_i^j \tag{43}$$

where $\alpha = \Delta t/\Delta x^2$. From this we can get the following matrix equation

$$\mathbf{A}u^{j+1} = u^j \tag{44}$$

expanding the equations for the backward Euler, the same was as for forward Euler. Simply by expanding it for $i = 3$, we can easily see the system emerge[2].

Our matrix $\mathbf{A}$ would then be given by

$$\mathbf{A} = \begin{bmatrix} 1+2\alpha & -\alpha & 0 & 0 \\ -\alpha & 1+2\alpha & -\alpha & 0 \\ 0 & -\alpha & 1+2\alpha & -\alpha \\ 0 & 0 & -\alpha & 1+2\alpha \end{bmatrix}, \tag{45}$$

and

$$u^{j+1} = \begin{bmatrix} u_0^{j+1} \\ u_1^{j+1} \\ u_2^{j+1} \\ u_3^{j+1} \end{bmatrix}, \quad u^j = \begin{bmatrix} u_0^j \\ u_1^j \\ u_2^j \\ u_3^j \end{bmatrix} \tag{46}$$

.

1. Truncation Error

The truncation error here goes the same as for the forward Euler, as we are expanding our problem using the same Taylor expansion, thus we have $\mathcal{O}(\Delta x^2)$ and $\mathcal{O}(\Delta t)$.

[2] We will show this expansion when considering the Crank-Nicholson scheme later.

2. Constraint

Following the same study of the spectral radius of $\mathbf{A}$, one can show that the constraint on α is unconditional, that is we can choose any value of α and the scheme remains stable[4]. However it is still beneficial to choose α, and thus $\Delta t \wedge \Delta x$ such that we get a good approximation.

I. Crank-Nicholson

The Crank-Nicholson scheme is obtained by choosing $\theta = 1/2$ in equation 30. This gives us

$$\begin{aligned}\frac{u_i^{j+1}-u_i^j}{\Delta t} &= \frac{u_{i+1}^{j+1}+u_{i-1}^{j+1}-2u_i^{j+1}}{2\Delta x^2} \\ &\quad + \frac{u_{i+1}^{j}+u_{i-1}^{j}-2u_i^{j}}{2\Delta x^2},\end{aligned} \tag{47}$$

if we collect common terms we have

$$\begin{aligned}&-\alpha\left(u_{i+1}^{j+1}+u_{i-1}^{j+1}\right)+(2+2\alpha)u_i^{j+1} \\ &= -\alpha\left(u_{i+1}^{j}+u_{i-1}^{j}\right)+(2-2\alpha)u_i^{j}\end{aligned} \tag{48}$$

using $\alpha = \Delta t/\Delta x^2$. Expanding this for $i = 0, 1, 2, 3$, using $u_{-1} = 0$ and $u_4 = 0$, we have

$$\begin{aligned}&-\alpha\left(u_1^{j+1}\right)+(2+2\alpha)u_0^{j+1} \\ &= -\alpha\left(u_1^{j}\right)+(2-2\alpha)u_0^{j}\end{aligned} \tag{49}$$

$$\begin{aligned}&-\alpha\left(u_2^{j+1}+u_0^{j+1}\right)+(2+2\alpha)u_1^{j+1} \\ &= -\alpha\left(u_2^{j}+u_0^{j}\right)+(2-2\alpha)u_1^{j}\end{aligned} \tag{50}$$

$$\begin{aligned}&-\alpha\left(u_3^{j+1}+u_1^{j+1}\right)+(2+2\alpha)u_2^{j+1} \\ &= -\alpha\left(u_3^{j}+u_1^{j}\right)+(2-2\alpha)u_2^{j}\end{aligned} \tag{51}$$

$$\begin{aligned}&-\alpha\left(u_2^{j+1}\right)+(2+2\alpha)u_3^{j+1} \\ &= -\alpha\left(u_2^{j}\right)+(2-2\alpha)u_3^{j}\end{aligned} \tag{52}$$

Again we can rewrite this as a matrix equation on the form

$$\mathbf{A}u^{j+1} = \mathbf{B}u^j. \tag{53}$$

Where we have

$$\mathbf{A} = \begin{bmatrix} 2+2\alpha & -\alpha & 0 & 0 \\ -\alpha & 2+2\alpha & -\alpha & 0 \\ 0 & -\alpha & 2+2\alpha & -\alpha \\ 0 & 0 & -\alpha & 2+2\alpha \end{bmatrix} \tag{54}$$

and

$$\mathbf{B} = \begin{bmatrix} 2-2\alpha & \alpha & 0 & 0 \\ \alpha & 2-2\alpha & \alpha & 0 \\ 0 & \alpha & 2-2\alpha & \alpha \\ 0 & 0 & \alpha & 2-2\alpha \end{bmatrix}. \tag{55}$$

And where we have u^{j+1} and u^j defined as before.

1. Truncation Error

Another way to look at the Crank-Nicholson method is as a half-time step approximation, a version of the midpoint rule for integration. But using the forward and backward Euler methods to for the set of points needed. Following the methods used in [5], we have

$$[D_t u = -au]^{n+\frac{1}{2}}, \tag{56}$$

where we can add a residual term R, thus giving us

$$[D_t u_e + a\overline{u_e} = R]^{n+\frac{1}{2}}. \tag{57}$$

where $_e$ is the exact solution, and $\overline{u_e}$ is the arithmetic mean. From this we have as per [5],

$$R^{n+\frac{1}{2}} = \left(\frac{1}{24}u_e'''(t_{n+\frac{1}{2}}) + \frac{1}{8}u''(t_n)\right)\Delta t^2 = \mathcal{O}(\Delta t^4) \tag{58}$$

from this we can see that the truncation error of Crank-Nicholson method is $\mathcal{O}(\Delta t^2)$. For the entire derivation see [5].

2. Constraint

The constraints for the Crank-Nicholson scheme is the same as for the backwards Euler, we are[4] free to choose Δt and Δx freely.

J. Forward Euler Two-Dimensions

For solving the two dimensional diffusion equation, we have chosen to use the forward Euler method. Whilst this put restrictions on our chosen values of α, it has the same truncation error as the backward Euler.

Assuming a square lattice, with equal many mesh points in either direction, we can discretize equation (15), as

$$u_{i,j}^{l+1} = u_{i,j}^l + \alpha\left[u_{i+1,j}^l + u_{i-1,j}^l + u_{i,j+1}^l + u_{i,j-1}^l - 4u_{i,j}^l\right] \tag{59}$$

using the same differentiation scheme as for the one dimensional counterparts. We also have α given as before by $\alpha = \Delta t/h^2$, with $h^2 = \Delta x^2 = \Delta y^2$.

1. Constraint

As is important with the forward Euler method is the constraint we have to put on Δt and Δx. The von Neumann stability analysis of the two-dimensional heat equation, is based on studying the growth of waves[6], e^{ikx}.

For two-dimensions we then have

$$u(x, y, t_n) = e^{i(k,l)\cdot\binom{x}{y}} = e^{ikx} \cdot e^{ily} \tag{60}$$

Writing our equation (15) on a different form then equation (59), we have

$$\begin{aligned} \frac{u^{l+1}_{i,j} - u^l_{i,j}}{\Delta t} &= \frac{u^l_{i+1,j} - 2u^l_{i,j} + u^l_{i-1,j}}{\Delta x^2} \\ &+ \frac{u^l_{i,j+1} - 2u^l_{i,j} + u^l_{i,j-1}}{\Delta y^2} \end{aligned} \tag{61}$$

which can be rewritten, using $u^l_{i,j} = x^{ikx}$, $x = 0$ for the current step and $G = u^{l+1}_{i,j}$. Where G is the growth factor, and if $|G(k)| < 1$, $\forall k$ the scheme is stable. We then have

$$\begin{aligned} \frac{G - 1}{\Delta t} &= \frac{e^{ik\Delta x} - 2 + e^{-ik\Delta x}}{\Delta x^2} \\ &+ \frac{e^{il\Delta y} - 2 + e^{-il\Delta y}}{\Delta y^2} \end{aligned} \tag{62}$$

this implies

$$G = 1 - 2\frac{\Delta t}{\Delta x^2}\left(1 - \cos\left(k\Delta x\right)\right) - 2\frac{\Delta t}{\Delta y^2}\left(1 - \cos\left(l\Delta y\right)\right). \tag{63}$$

This has a worst case at $k\Delta x = \pi = l\Delta y$ which implies

$$G = 1 - 4\frac{\Delta t}{\Delta x^2} - 4\frac{\Delta t}{\Delta y^2}, \tag{64}$$

thus we have the following restriction on Δt

$$\begin{aligned} \Delta t &\leq \frac{1}{2}\left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2}\right) \\ &= \frac{\Delta x^2 + \Delta y^2}{8} \underbrace{=}_{\Delta x = h = \Delta y} \frac{h^2}{4}, \end{aligned} \tag{65}$$

or $\alpha \leq 1/4$.

III. ALGORITHMS

We assume we have vectors of length N, which means that we have vectors such that we have $N - 2$ interior points and two fixed end points.

A. Algorithm Components

Algorithm 1, shows the simple implementation of the forward Euler method.

Algorithm 1: Forward Substitution

```
Initialization
for i = 1; i < N; i++ do
    $\hat{u}_i = \alpha(u_{i+1} + u_{i-1}) + eu_i$
end
```

Algorithm 2 shows the Tridiagonal Matrix algorithm, also known as the Thomas algorithm [7], implemented as in [3].

Algorithm 2: Tridiagonal Matrix Algorithm

```
Initialization
for i = 1; i < N − 1; i++ do
    $\hat{b}_i = b_i - a_{i-1}c_{i-1}/\hat{b}_{i-1}$
    $u'_i = u_i - u'_{i-1}a_{i-1}/\hat{b}_{i-1}$
end
$\hat{u}_n = u'_n/\hat{b}_n$
for i = N − 1; i > 0; i- do
    $\hat{u} = (u'_i - \hat{u}_{i+1}c_i)/\hat{b}_i$
end
```

1. Forward Euler

The forward Euler method for solving tridiagonal matrix equation on the form

$$u^{j+1} = \mathbf{A}u^j. \tag{66}$$

We can use algorithm 1, where from the matrix form in equation (35) we have $e = 1 - 2\alpha$ and $\alpha = dt/dx^2$.

2. Backward Euler

The backward Euler method for solving tridiagonal matrix equations on the form

$$\mathbf{A}u^{j+1} = u^j. \tag{67}$$

We can apply algorithm 2 using $a = \alpha$ and $b = 1 + 2\alpha$. This comes from the matrix form in equation (45).

3. Crank-Nicholson

The Crank-Nicholson method for solving tridiagonal matrix equations on the form

$$\mathbf{A}u^{j+1} = \mathbf{B}u^j. \tag{68}$$

To solve this equation, we can either go ahead with a matrix inversion, however as our matrices are sparse we can implement a partial approach. If we solve the right-hand side first the solve the left-hand side, we can apply both our algorithms in combination. We have

$$\mathbf{A}u^{j+1} = \mathbf{B}u^j \tag{69}$$

$$\mathbf{A}u^{j+1} = u'^j \tag{70}$$

where we solve equation (69) using algorithm 1 and solve our new equation (70) using algorithm 2 using the values from equations (54, 55) for our matrices $\mathbf{A}$ and $\mathbf{B}$.

B. Two-Dimensional Extension

An implementation of equation (59), for the two-dimensions using the forward Euler scheme may look as

Algorithm 3: Forward Euler Two Dimensions

```
Initialization
for i = 1; i < N; i++ do
    for j = 1; j<N; j++ do
        Q = $\alpha\left(\mathbf{U}[i+1,j] + \mathbf{U}[i-1,j]\right.$
            $\left.+\mathbf{U}[i,j+1] + \mathbf{U}[i,j-1] - 4\mathbf{U}[i,j]\right)$
        $\hat{\mathbf{U}}[i,j] = \mathbf{U}[i,j] +$ Q
    end
end
```

IV. RESULTS

A. One-Dimensional Solutions

1. Curved

In Figure 1 and Figure 3 a chosen time value for which the solution is still significantly curved for two different time steps. The time steps in questions is chosen by the constraint put on Δt in the forward Euler scheme, by a chosen value of Δx. Our two chosen values of Δx are $\Delta x = 1/10$ and $\Delta x = 1/100$, this corresponds to Figure 1 and Figure 3, respectively. The corresponding Δt values are then given as $\Delta t = 0.005$, and $\Delta t = 0.00005$, respectively.

A good separation of the different curves is visible for the largest time step, but separating the lines for the smaller time step is difficult.

2. Linear

In Figure 2 and Figure 4 a chosen time value for which the solution is almost linear for the same two time steps as earlier.

Almost no difference in the methods are visible in the graphs, all the methods are converging relatively quickly.

B. Error

1. Curved

In Figure 5 and Figure 7 the absolute error between the analytical solution and the numerical estimate for the curved solutions. The convergence for the different methods refer to how close they are to a straight line, thus we can see that the forward Euler method converges the quickest, whilst the backward Euler is fairly slow.

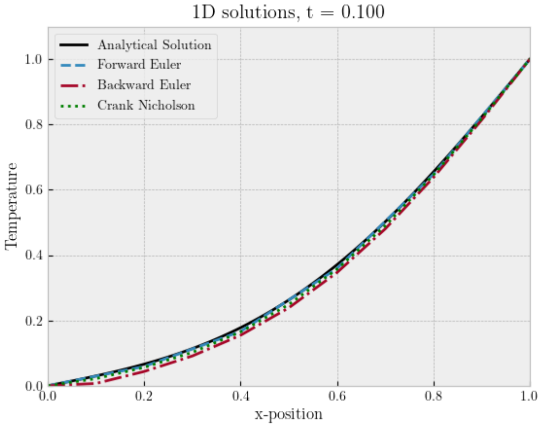

FIG. 1: The three finite difference schemes, using $\Delta x = 0.1$, for a curved solution. The numerical schemes are plotted against the analytical solution. Also $\alpha = 0.5$, is the forced condition such that $\Delta t = \alpha \Delta x^2$.

2. Linear

In Figure 6 and Figure 8 the absolute error between the analytical solution and the numerical estimate for the almost linear solutions. The convergence is again given by the same, and we can again see that the forward Euler converges the quickest whilst backward Euler is is slower.

C. Two-Dimensional Solutions

In Figure 9 we see the steady state solution for our two-dimensional scheme. Which shows a even gradient on the color mapping of the solution, whilst in Figure 10 we see a sharper gradient and a larger area for which we have a close to uniform solution. This also coincides with our results for the one-dimensional case, where we have a gradual evolution of the system, so this result in two-dimensions shows a significantly curved solution.

V. DISCUSSION

A. One-Dimension

The one-dimensional case is a good stepping stone for the implementation of the algorithms and studying the system evolution. Including it can serve as a stepping stone to analysis of certain two-dimensional problems where they can be turned into a quasi one-dimensional problem. One such case is studying systems that have zero flux at the edges.

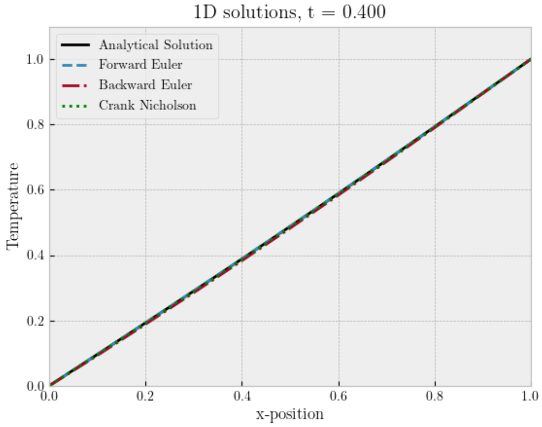

FIG. 2: The three finite difference schemes, using $\Delta x = 0.1$, for a near steady state solution. The numerical schemes are plotted against the analytical solution. Also $\alpha = 0.5$, is the forced condition such that $\Delta t = \alpha \Delta x^2$.

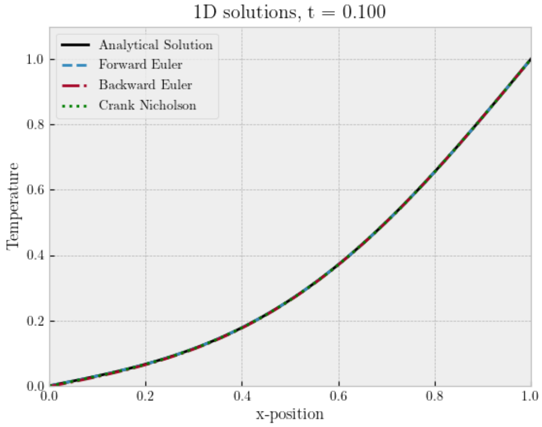

FIG. 3: The three finite difference schemes, using $\Delta x = 0.01$, for a curved solution. The numerical schemes are plotted against the analytical solution. Also $\alpha = 0.5$, is the forced condition such that $\Delta t = \alpha \Delta x^2$.

1. Methods

An important part when choosing what numerical scheme to apply to a certain problem, a mixture between speed of implementation and correctness is desirable. As more complex methods such as the Crank-Nicholson or going further such as Runge-Kutta4 or similar, the implementation time is significantly longer than that of the forward Euler especially if no premade program can be used. And even in some cases, they have a quicker convergence and are more accurate than their counterparts.

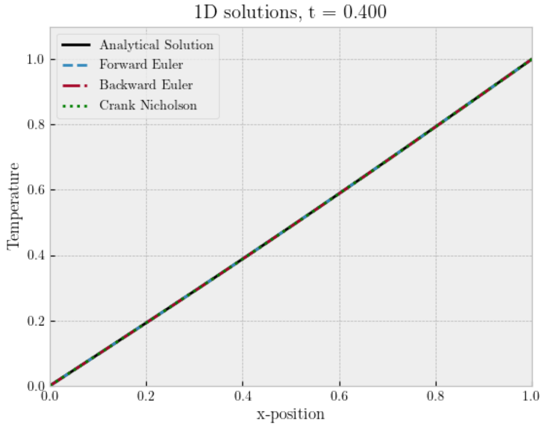

FIG. 4: The three finite difference schemes, using $\Delta x = 0.01$, for a near steady state solution. The numerical schemes are plotted against the analytical solution. Also $\alpha = 0.5$, is the forced condition such that $\Delta t = \alpha \Delta x^2$.

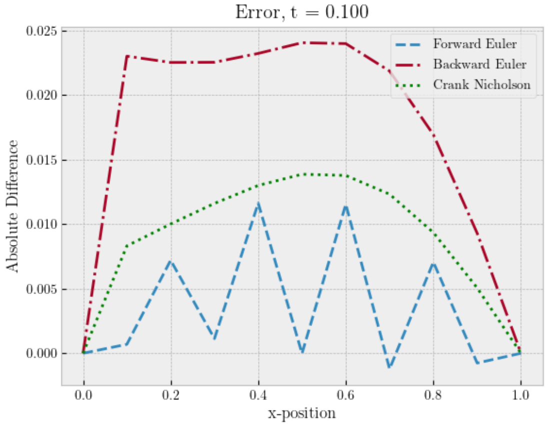

FIG. 5: The absolute error between the analytical solution and the numerical solutions. It is shown as the difference between the solutions, using $\alpha = 0.5$, and $\Delta x = 0.1$, up to t = 0.1.

In Figure 1 we can see that the forward Euler is the method closest to that of the analytical solution in black. The backward Euler is furthest from the analytical solution, whilst the Crank-Nicholson strikes a middle ground between the two schemes. Not much more can be deduced from the plots over the solutions themselves, as the naked eye is not able to see the difference between the solutions.

2. Error

As we can not deduce any noticeable difference between the schemes in the data plots themselves, by studying the absolute error we can see how the different methods behave compared to the analytical solution. These

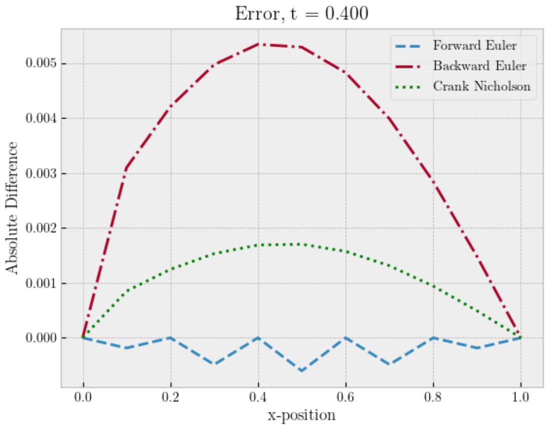

FIG. 6: The absolute error between the analytical solution and the numerical solutions. It is shown as the difference between the solutions, using $\alpha = 0.5$, and $\Delta x = 0.1$, up to t = 0.4.

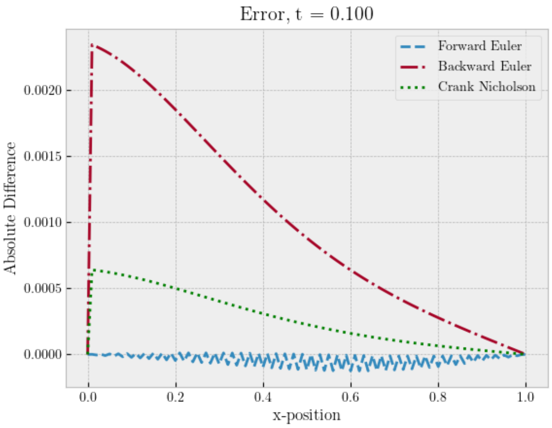

FIG. 7: The absolute error between the analytical solution and the numerical solutions. It is shown as the difference between the solutions, using $\alpha = 0.5$, and $\Delta x = 0.01$, up to t = 0.1.

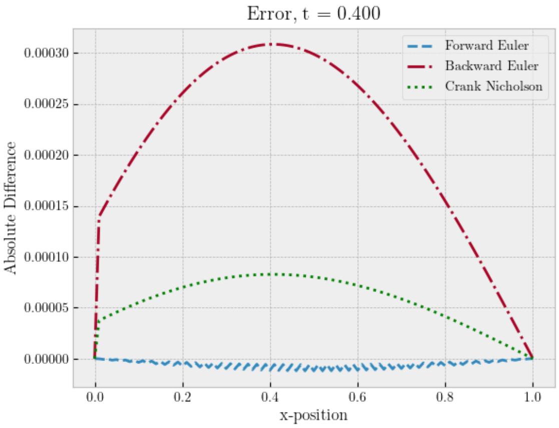

FIG. 8: The absolute error between the analytical solution and the numerical solutions. It is shown as the difference between the solutions, using $\alpha = 0.5$, and $\Delta x = 0.01$, up to t = 0.4.

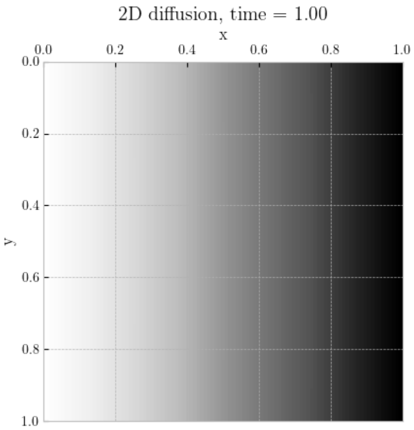

FIG. 9: The two-dimensional solution for the near steady state at t = 1. This is using the quasi one-dimensional solution approach and only depicts the numerical solution.

results can be seen in Figure 5, Figure 7, Figure 6 and Figure 8. What we can see is a consistent trend of the forward Euler converging the quickest to the analytical solution, and with the decrease time steps seen in Figure 7 and Figure 8, that the forward Euler is almost linear in shape. Only small fluctuations and a small curve is visible compared to its counterparts.

The backward Euler is by far the worst when it comes to convergence and is far off the analytical solution in comparison to the forward Euler and would be a less than ideal scheme to apply for this problem. Even thought the scheme is stable for all choices of Δt and Δx it comes at a cost, that the scheme tends to become more inaccurate with larger choices of Δt and Δx.

The Crank-Nicholson strikes the middle ground again, whilst slower computationally than backward and forward Euler, it has a time truncation error of $\mathcal{O}(\Delta t^2)$. This would prove to be the superior choice for longer time simulations, where the stable solution is not reached quickly.

3. Methods Revisited

After studying the error of the different methods, we can conclude that even though the forward Euler has a worse time accuracy than the Crank-Nicholson scheme, it converges significantly quicker and has a smaller error at the selected time steps. And for problems of a smaller size, especially scaled, using the forward Euler could be

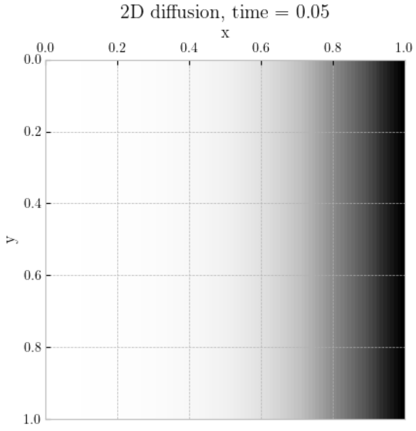

FIG. 10: The two-dimensional solution at t = 0.05, showing a curved solution, with large areas being near equal and a sharp peak. This is the same as we seen in Figure 3.

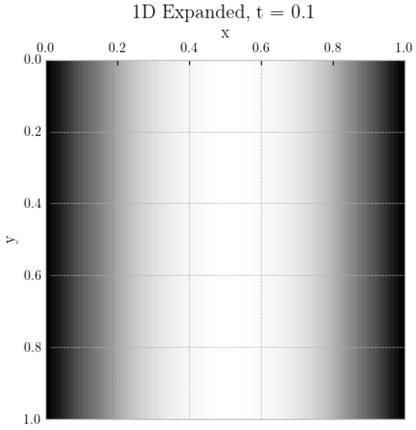

FIG. 11: The two-dimensional solution for $v(x, y, t)$, using the quasi one-dimensional approach. Showing what resembles a sine wave, which is expected from our one dimensional analytical solution.

the best choice. However if you want to study larger systems over longer time scales, using schemes that are more accurate in time is beneficial even tough they come at a computational cost. As the Crank-Nicholson scheme, combines both the backward Euler and forward Euler.

B. Two-Dimensions

The study of a two-dimensional problem, which has insulated edges is not always the most interesting, whilst having multiple real world applications, even a three dimensional problem, where you only have two edges for which you have any interaction with the area in question can be narrowed down to a simple one dimensional solution. Its counterpart is studying two-dimensional problems where you have different boundary conditions and perhaps different properties within the area in question.

From our study, we see that our two-dimensional case, in fact produces the same results as our one-dimensional case, which was what we expected. Furthermore from Figure 10, we actually have what we can assume to be the exact same behavior as for our one-dimensional case. This is what we would expect from our boundary conditions as we have no flux at the edges.

It would need more numerical study to ensure that the forward Euler in both cases, produces the same answers, however we can conclude that our results for studying a two-dimensional case with no fluxes at the edges can be simplified to a study of a quasi one-dimensional problem.

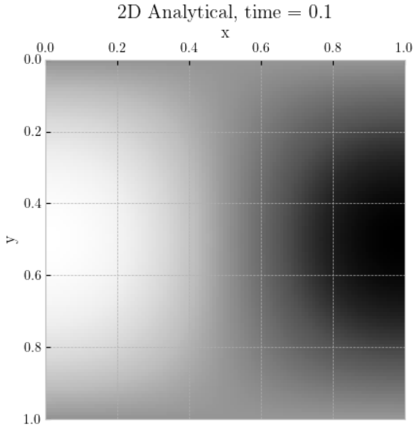

FIG. 12: The two-dimensional closed-form solution, plotted in the same way as above. This is not what we expected as a result, and shows one peak and a through with fixed edges.

1. *Quasi One-Dimensional*

As the boundary conditions for our two-dimensional problem is chosen such that there is no flux along the edges, we are dealing with a quasi one-dimensional problem, thus we can adapt our one-dimensional solution to also work for two-dimensions. This is because there are no changes in the y-direction and thus we have a problem where we are simply studying a one-dimensional problem in two-dimensions.

Our one-dimensional solution can then be used as a baseline for the two-dimensional closed-form solution, to understand whether the solution we found is correct. From this we can see that in Figure 12 is far from what is expected as seen in Figure 11 and some part of our two-dimensional closed-form solution or implementation is incorrect.

2. Analytical Solution

The study of the two-dimensional problem is based on the quasi one-dimensional approach, instead of using our closed-form two-dimensional solution. This is because as stated we our closed-form two-dimensional solution is not producing the results as expected. We observe a pattern in Figure 12 that is similar to that of a peak and a through with fixed boundary values which is not what we seen in Figure 11, which is resembles a wave with troughs on either side and a peak in the middle which is expected of a sine wave. The reason for the disagreement between our quasi one-dimensional solution and the two-dimensional analytical solution could be due to different factors.

The first, our closed-form solution may be wrong, and would never produce the results which we expect to see. However no such errors has been found when proof reading the solution, and it is left in as the closed-form solution to the two-dimensional problem.

The second, could be that our implementation of the double Fourier-series is wrong. This is equally likely to be the case, and no solution or improvement to the implementation that would result in the expected results was found.

VI. CONCLUSION

We have reviewed different finite difference schemes, studying their mathematical truncation error and stability for solving the diffusion equation in one and two spatial dimensions. Establishing functional numerical methods for both dimensions, in agreement with our one-dimensional solutions and quasi one-dimensional solutions. We found that the convergence of the forward Euler is significantly faster than both the implicit schemes. No study of the stability conditions, i.e. the constraint put on Δt and Δx was done. Instead the choice of Δt and Δx where made such that we always had $\alpha = 0.5$ for the one-dimensional case and $\alpha = 0.25$ for the two-dimensional case. Whilst the forward Euler has a quicker convergence, both the implicit schemes have no restrictions on their choices of Δt and Δx, and the Crank-Nicholson scheme having the best truncation error in the time dimension. This makes the Crank-Nicholson scheme, even though computationally expensive, the best choice for solving the diffusion equation. In two-dimensions we only developed the explicit scheme which reproduced the expected results from our assumptions of the setup problem. Reproducing the quasi one-dimensional problem we had, however we were not able to reproduce the correct results with our analytical solution. Either due to poor implementation or wrong closed-form solution, the solution found for the two-dimensional problem did not reproduce the numerical results that supported our expectation. Further study of the accuracy of the two-dimensional case and in-depth study of the stability of the different schemes in one-dimension. A further study of more advanced implicit methods in the two-dimensional case.

[1] H. P. Langtangen and G. K. Pedersen, "Scaling of Differential Equations," 2016. [Online]. Available: https://github.com/hplgit/scaling-book

[2] M. Hjorth-Jensen, "Computational physics, lecture notes fall 2015," *Department of Physics, University of Oslo*, 2015. [Online]. Available: https://github.com/CompPhysics/ComputationalPhysics/blob/master/doc/Lectures/lectures2015.pdf

[3] S. S. Sundberg, "Project 1," https://github.com/SigurdSundberg/FYS3150/tree/master/project1, 2020.

[4] A. MITCHELL and D. Griffiths, *The Finite Difference Method in Partial Differential Equations*, 01 1980.

[5] H. P. Langtangen, "Truncation error analysis," 2015. [Online]. Available: http://hplgit.github.io/num-methods-for-PDEs/doc/pub/trunc/pdf/trunc-4screen.pdf

[6] M. OpenCourseWare, "Von neumann stability analysis," *MIT OpenCourseWare*, 2009. [Online]. Available: https://ocw.mit.edu/courses/mathematics/18-336-numerical-methods-for-partial-differential-equations-spring-2009/lecture-notes/MIT18_336S09_lec14.pdf

[7] B. N. Datta, *Numerical linear algebra and applications.* Siam, 2010, vol. 116.

Appendix A: Analytical Solution for Two Dimensions

To find the analytical solution of equation (15), we start off with separation of variables

$$u(x,y,t) = X(x)Y(y)T(t). \tag{A1}$$

Inserting in the original PDE we get

$$\frac{T'(t)}{T(t)} = \frac{X''(x)}{X(x)} + \frac{Y''(y)}{Y(y)}. \tag{A2}$$

Since the l.h.s. depend on t and the r.h.s. on (x, y), both sides must equal a constant $-\lambda$. Such that we have

$$\begin{aligned} \frac{T'(t)}{T(t)} &= -\lambda \\ \frac{X''(x)}{X(x)} + \frac{Y''(y)}{Y(y)} &= -\lambda \end{aligned} \tag{A3}$$

Our boundary and initial conditions are as before, given by

$$\begin{aligned} u(0,y,t) = 0, &\quad \frac{\partial}{\partial y}u(x,0,t) = 0 \\ u(L,y,t) = 0, &\quad \frac{\partial}{\partial y}u(x,L,t) = 0 \\ u(x,y,0) &= f(x,y) \end{aligned} \tag{A4}$$

where we have $f(x, y) = -x/L$. This can be found the same way as for the one dimensional case, and as we have no variance in the y-direction, due to studying a quasi one dimensional problem, this equations holds for $f(x, y)$.

From equation (A3), we have that

$$T(t) = C_1 e^{-\lambda t} \tag{A5}$$

Separating the boundary conditions we have that

$$\begin{aligned} u(0,y,t) = 0 &\implies X(0) = 0 \\ u(L,y,t) = 0 &\implies X(L) = 0 \\ \frac{\partial}{\partial y}u(x,0,t) = 0 &\implies Y'(0) = 0 \\ \frac{\partial}{\partial y}u(x,L,t) = 0 &\implies Y'(L) = 0 \end{aligned}$$

Rearranging the spatial part of the equation gives

$$\frac{Y''(y)}{Y(y)} + \lambda = -\frac{X''(x)}{X(x)}. \tag{A6}$$

Since the l.h.s. depends on y and the r.h.s. depends on x, both sides must again equal a constant, say μ,

$$\frac{Y''(y)}{Y(y)} + \lambda = -\frac{X''(x)}{X(x)} = \mu. \tag{A7}$$

This gives us

$$X''(x) + \mu X(x) = 0, \quad X(0) = 0 = X(L), \tag{A8}$$

for which solution is the same as the one for one dimension, namely

$$X_n(x) = \sin\left(\frac{n\pi x}{L}\right), \quad \mu_n = \left(\frac{n\pi}{L}\right)^2 \tag{A9}$$

for $n = 1, 2, 3, \ldots$.

For $Y(y)$, we on the other hand have

$$Y''(y) + (\lambda - \mu_n)Y(y) = 0, \quad Y'(0) = 0 = Y'(L) \tag{A10}$$

which have the following solution

$$Y_m(y) = \cos\left(\frac{m\pi y}{L}\right), \quad (\lambda_{nm} - \mu_n) = \left(\frac{m\pi}{L}\right)^2 \tag{A11}$$

for $m = 1, 2, 3, \ldots$.

Which then gives us the general solution as

$$u(x,y,t) = \sin\left(\frac{n\pi x}{L}\right)\cos\left(\frac{m\pi y}{L}\right)e^{-\lambda_{nm}t} \tag{A12}$$

for $m, n = 1, 2, 3, \ldots$. Giving us a set of solutions given by

$$u(x,y,t) = \sum_{m=1}^{\infty}\sum_{n=1}^{\infty} A_{nm}u_{nm} \tag{A13}$$

Where A_{nm}, is given by

$$A_{nm} = \frac{4}{L^2}\int_{x=0}^{L}\int_{y=0}^{L} f(x,y)\sin\left(\frac{n\pi x}{L}\right)\cos\left(\frac{m\pi y}{L}\right)dy\,dx. \tag{A14}$$