Partial Differential Equations

The Diffusion Equation

Written by:

Jens Bratten Due
Mohamed Ismail

Department of physics UiO

December 19, 2019

Abstract

In this project, we have presented three different methods for solving partial differential equations. The project is divided into three parts, comprising of a numerical study of the 1D and 2D general diffusion equations, and lastly, the heat equation describing the thermal evolution of the lithosphere, both with and without taking into account heat from radioactive materials in the mantle. The first part consists of developing the explicit forward Euler method, as well as the implicit backward Euler and Crank-Nicolson schemes. We have compared our implementation of these methods with the derived analytical solution at two points in time and concluded that the implicit methods are the most accurate, with a slightly higher accuracy from Crank-Nicolson.

The second part is dedicated to studying the 2D diffusion equation using an implicit scheme solved through the use of Jacobi's method. Again, we find a close approximation to the analytical solution with the error being mostly dependent on positional step size Δx.

The third and final part presents a model of the temperature distribution of the lithosphere. We see, yet again, a good numerical approximation to the analytical solution for a low complexity. As we have introduced higher complexities consisting of radioactive materials without decay, we observe a decrease in accuracy, leading us to conclude that the case with no analytical solution, enrichment with decay, may not be perfectly approximated by our numerical method. We receive this set-back with humility and grace, leaving the study and implementation of a stronger method up to future articles on this matter.

Contents

1 Introduction

Numerous fields within science focuses on solving partial differential equations to better understand the laws of nature. Therefore, it is essential to have dependable and accurate methods of solving partial differential equations.

In this project we want to study the numerical stability of different methods for solving partial differential equations (PDEs) and apply them to real world applications. The methods we will explore is the explicit forward Euler algorithm, the implicit Backward Euler algorithm and the implicit Crank-Nicolson algorithm. First we will start off with the general one-dimensional diffusion equation and compare the numerical methods to the analytical solution to observe which method is the superior one. Then we will move to the two-dimensional case and further investigate the accuracy of the numerical methods.

Finally, with our new-found experience with the general diffusion equation, we will attempt to study the heat diffusion found in the lithosphere.

2 Theoretical Methods & Technicalities

2.1 Diffusion Equation in One Dimension

The one-dimensional PDE known as the diffusion equation is given as

$$\frac{\partial u(x,t)}{\partial t} = \frac{\partial^2 u(x,t)}{\partial x^2} \tag{1}$$

and with initial conditions, i.e., the conditions at $t = 0$,

$$u(x,0) = 0 \qquad 0 < x < L$$

with $L = 1$ the length of the x-region of interest and boundary conditions of

$$u(0,t) = 0 \qquad t \geq 0,$$

and

$$u(L,t) = 1 \qquad t \geq 0.$$

there exists analytical solutions for this problem, and when doing scientific research numerically, it is an invaluable resource to have analytical solutions to compare the numerical results with.

The analytical solution to this problem is

$$u(x,t) = \frac{x}{L} + \sum_{n=1}^{\infty} \frac{(-1)^n 2}{n\pi} \sin\big(\frac{n\pi}{L}x\big) e^{-\frac{n^2\pi^2}{L^2}t} \tag{2}$$

where the full derivation can be found in Appendix A.

2.1.1 Forward Euler - Explicit Scheme

One method for solving such a problem is the explicit forward Euler method. It starts by discretizing the time derivative by a forward formula

$$\frac{\partial u}{\partial t} = u_t \approx \frac{u(x, t + \Delta t) - u(x, t)}{\Delta t} = \frac{u(x_i, t_j + \Delta t) - u(x_i, t_j)}{\Delta t}$$

with a truncation error of $O(\Delta t)$.

From here, we discretize the position derivative using a centered difference

$$\frac{\partial^2 u}{\partial x^2} = u_{xx} \approx \frac{u(x + \Delta x, t) - 2u(x, t) + u(x - \Delta x, t)}{\Delta x^2} = \frac{u(x_i + \Delta x, t_j) - 2u(x_i, t_j) + u(x_i - \Delta x, t_j)}{\Delta x^2}$$

with a local approximation error off $O(\Delta x^2)$.

These equations can be further simplified to

$$u_t \approx \frac{u_{i,j+1} - u_{i,j}}{\Delta t}$$
$$u_{xx} \approx \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta x^2}$$

and the one-dimensional diffusion equation can then be rewritten in its discretized version as

$$\frac{u_{i,j+1} - u_{i,j}}{\Delta t} = \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta x^2}.$$

By defining the parameter $\alpha = \frac{\Delta t}{\Delta x^2}$, we arrive at the forward Euler explicit scheme

$$u_{i,j+1} = \alpha u_{i-1,j} + (1 - 2\alpha)u_{i,j} + \alpha u_{i+1,j}. \tag{3}$$

This system of equations can be represented in terms of a matrix-vector multiplication $U_{j+1} = \boldsymbol{A}U_j$, by defining a matrix $\boldsymbol{A}$

$$\boldsymbol{A} = \begin{bmatrix} 1-2\alpha & \alpha & 0 & 0 & \dots & 0 \\ \alpha & 1-2\alpha & \alpha & 0 & \dots & 0 \\ 0 & \alpha & 1-2\alpha & \alpha & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & \dots & \alpha & 1-2\alpha & \alpha \\ 0 & \dots & \dots & \dots & \alpha & 1-2\alpha \end{bmatrix}$$

and the vector U_j is

$$U_j = \begin{bmatrix} u_{1,j} \\ u_{2,j} \\ \dots \\ u_{n,j} \end{bmatrix}$$

For a computation of the system over j time steps, the matrix $\boldsymbol{A}$ is applied j times. Therefore, we can rewrite the problem as

$$U_j = \boldsymbol{A}U_{j-1} = \boldsymbol{A}\big(\boldsymbol{A}U_{j-2}\big) = \cdots = \boldsymbol{A}^j U_0 \tag{4}$$

2.1.2 Backward Euler - Implicit Scheme

Another well known method of solving PDE's is the implicit scheme known as backward Euler. Here we start off with the backward going Euler formula for the first derivative with respect to time.

$$u_t \approx \frac{u(x,t) - u(x,t-\Delta t)}{\Delta t} = \frac{u(x_i,t_j) - u(x_i,t_j-\Delta t)}{\Delta t}$$

This approximation also comes with a truncation error of $O(\Delta t)$.

From there, we use the approximated version for the second derivative with respect to position

$$u_{xx} \approx \frac{u(x+\Delta x,t) - 2u(x,t) + u(x-\Delta x,t)}{\Delta x^2} = \frac{u(x_i+\Delta x,t_j) - 2u(x_i,t_j) + u(x_i-\Delta x,t_j)}{\Delta x^2}.$$

We now obtain

$$\begin{aligned}
\frac{u(x_i,t_j) - u(x_i,t_j-\Delta t)}{\Delta t} &= \frac{u(x_i+\Delta x,t_j) - 2u(x_i,t_j) + u(x_i-\Delta x,t_j)}{\Delta x^2} \\
\frac{u_{i,j} - u_{i,j-1}}{\Delta t} &= \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta x^2} \\
u_{i,j-1} &= -\alpha u_{i-1,j} + (1+2\alpha)u_{i,j} - \alpha u_{i+1,j}
\end{aligned}$$

where $u_{i,j-1}$ is the only unknown quantity. By defining a matrix $\boldsymbol{A}$

$$\boldsymbol{A} = \begin{bmatrix}
1+2\alpha & -\alpha & 0 & 0 & \dots & 0 \\
-\alpha & 1+2\alpha & -\alpha & 0 & \dots & 0 \\
0 & -\alpha & 1+2\alpha & -\alpha & 0 & 0 \\
\dots & \dots & \dots & \dots & \dots & \dots \\
0 & \dots & \dots & -\alpha & 1+2\alpha & -\alpha \\
0 & \dots & \dots & \dots & -\alpha & 1+2\alpha
\end{bmatrix}$$

,

we can reformulate the problem as a matrix-vector multiplication.

$$\boldsymbol{A}V_j = V_{j-1} \tag{5}$$

This leads to the general formula

$$V_j = \boldsymbol{A}^{-1}V_{j-1} = \boldsymbol{A}^{-1}\big(\boldsymbol{A}^{-1}V_{j-2}\big) = \cdots = \boldsymbol{A}^{-j}V_0 \tag{6}$$

where the vector V_j is

$$V_j = \begin{bmatrix} u_{1,j} \\ u_{2,j} \\ \cdots \\ u_{n,j} \end{bmatrix}$$

and V_0 is the initial vector at time $t = 0$ defined by the initial value $u(x, 0)$.

This scheme is defined as implicit because it relies on determining the vector $u_{i,j-1}$ instead of $u_{i,j+1}$. If α does not depend on time t we only need to invert the matrix once. Alternatively,

we can solve this system of equations using methods from linear algebra. These are however very cumbersome, since they involve $\sim O(N^3)$ operations for a $N \times N$ matrix. It is much faster to solve these linear equations using methods for tridiagonal matrices, since these involve only $\sim O(N)$ operations [1].

2.1.3 Crank-Nicolson - Implicit Scheme

It is possible to combine the forward and backward Euler methods in a slightly more general approach. Introducing a parameter θ (the so-called θ-rule) we can set up an equation

$$\frac{\theta}{\Delta x^2}\big(u_{i-1,j} - 2u_{i,j} + u_{i+1,j}\big) + \frac{1-\theta}{\Delta x^2}\big(u_{i+1,j-1} - 2u_{i,j-1} + u_{i-1,j-1}\big) = \frac{1}{\Delta t}\big(u_{i,j} - u_{i,j-1}\big) \quad (7)$$

which for $\theta = 0$ yields the forward formula for the first derivative and the explicit scheme, while $\theta = 1$ yields the backward formula and the implicit scheme. These two schemes are called the backward and forward Euler schemes, respectively. For $\theta = 1/2$ we obtain a new scheme named after its inventors, Crank and Nicolson. This scheme yields a truncation error in time of $O(\Delta t^2)$ and is stable for all possible combinations of Δt and Δx.

To derive the Crank-Nicolson equation, one starts with the forward Euler scheme and Taylor expand $u(x, t + \Delta t)$, $u(x + \Delta x, t)$ and $u(x - \Delta x, t)$. However, for the Crank-Nicolson scheme one requires an additional Taylor expansion of $u(x, t + \Delta t)$, $u(x + \Delta x, t)$ and $u(x\Delta x, t)$ around $(x, t + \Delta t/2)$.

Omitting all the long and exciting derivation, which can be found here [1], we end up with the following equations

$$u_t \approx \frac{u(x, t + \Delta t) - u(x, t)}{\Delta t} = \frac{u(x_i, t_j + \Delta t) - u(x_i, t_j)}{\Delta t}.$$

The corresponding spatial second-order derivative reads

$$\begin{aligned} u_{xx} \approx & \frac{1}{2}\left(\frac{u(x_i + \Delta x, t_j) - 2u(x_i, t_j) + u(x_i - \Delta x, t_j)}{\Delta x^2} + \right. \\ & \left. \frac{u(x_i + \Delta x, t_j + \Delta t) - 2u(x_i, t_j + \Delta t) + u(x_i - \Delta x, t_j + \Delta t)}{\Delta x^2}\right). \end{aligned}$$

Note well that we are using a time-centered scheme wih $t + \Delta t/2$ as center. Using our previous definition of $\alpha = \frac{\Delta t}{\Delta x}$ we can rewrite Eq. 7 as

$$-\alpha u_{i-1,j} + (2 + 2\alpha)u_{i,j} - \alpha u_{i+1,j} = \alpha u_{i-1,j-1} + (2 - 2\alpha)u_{i,j-1} + \alpha u_{i+1,j-1} \quad (8)$$

or in matrix-vector form as

$$\Big(2\boldsymbol{I} + \alpha\boldsymbol{B}\Big)\boldsymbol{V}_j = \Big(2\boldsymbol{I} - \alpha\boldsymbol{B}\Big)\boldsymbol{V}_{j-1} \quad (9)$$

where the vector $\boldsymbol{V}_j$ is the same as defined in the implicit case (backward Euler) while the matrix $\boldsymbol{B}$ is

$$\boldsymbol{B} = \begin{bmatrix} 2 & -1 & 0 & 0 & \dots & 0 \\ -1 & 2 & -1 & 0 & \dots & 0 \\ 0 & -1 & 2 & -1 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & \dots & -1 & 2 & -1 \\ 0 & \dots & \dots & \dots & -1 & 2 \end{bmatrix}$$

And we can rewrite the Crank-Nicholson scheme as follows

$$\boldsymbol{V}_j = \Big(2\boldsymbol{I} + \alpha\boldsymbol{B}\Big)^{-1}\Big(2\boldsymbol{I} - \alpha\boldsymbol{B}\Big)\boldsymbol{V}_{j-1} \tag{10}$$

$$\boldsymbol{V}_j = \Big(2\boldsymbol{I} + \alpha\boldsymbol{B}\Big)^{-1}\tilde{\boldsymbol{V}}_{j-1} \tag{11}$$

where $\tilde{\boldsymbol{V}}_{j-1} = \Big(2\boldsymbol{I} - \alpha\boldsymbol{B}\Big)\boldsymbol{V}_{j-1}$.

2.2 Truncation errors and stability properties

The truncation errors of all the different methods is given in table 1 and the derivation for them can be found here [1], where it is meticulously shown.

The stability criteria depends on the spectral radius, ρ_{max} of the matrix $\boldsymbol{A}$ in all the methods. The spectral radius of the matrix is defined as the largest absolute eigenvalue of the matrix

$$\rho(\boldsymbol{A}) = max\big\{|\lambda| : det\big(\boldsymbol{A} - \lambda\boldsymbol{I}\big) = 0\big\} \tag{12}$$

The methods are stable when they fulfill the condition of $\rho(\boldsymbol{A}) < 1$. The explicit scheme only satisfies the spectral radius for $\Delta t \leq \frac{1}{2}\Delta x^2$. However, the implicit schemes are always stable since their spectral radius always satisfies $\rho(\boldsymbol{A}) < 1$. This has been visualized in table 1. The proof for this can be found here [1].

Table 1: Truncation errors and stability

Scheme	**Truncation Error**	**Stability Requirements**
Crank-Nicolson	$O(\Delta x^2)$ & $O(\Delta t^2)$	Stable for all Δt & Δx
Backward Euler	$O(\Delta x^2)$ & $O(\Delta t)$	Stable for all Δt & Δx
Forward Euler	$O(\Delta x^2)$ & $O(\Delta t)$	$\Delta t \leq \frac{1}{2}\Delta x^2$

2.3 Diffusion Equation in Two Dimensions

In 2-dimensions the diffusion equation takes the form

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

In this project we will be looking into the model with a square lattice for x and y.
In addition, we will use the simple boundary conditions of

$$u(0,y,t) = u(L,y,t) = 0 \qquad y \in (0,L) \qquad t \geq 0,$$

and

$$u(x,0,t) = u(x,L,t) = 0 \qquad x \in (0,L) \qquad t \geq 0,$$

with initial conditions

$$u(x,y,0) = \sin(\pi x)\sin(\pi y) \qquad 0 < x,y < L$$

with $L = 1$ the length of the x,y-region of interest.

For this case, there also exist analytical solutions.

$$u(x,y,t) = \sin(\pi x)\sin(\pi y)e^{-2\pi^2 t} \tag{13}$$

where the derivation can be found in Appendix B.

2.3.1 Jacobi's Method - Implicit

$$\frac{\partial^2 u(x,y,t)}{\partial x^2} + \frac{\partial^2 u(x,y,t)}{\partial y^2} = \frac{\partial u(x,y,t)}{\partial t}, t > 0, x, y \in [0,1] \tag{14}$$

We again discretize the position and time derivatives, and use the following approximation for the position derivatives

$$u_{xx} \approx \frac{u^n_{i+1,j} - 2u^n_{i,j} + u^n_{i-1,j}}{\Delta x^2} \tag{15}$$

$$u_{yy} \approx \frac{u^n_{i,j+1} - 2u^n_{i,j} + u^n_{i,j-1}}{\Delta y^2} \tag{16}$$

where we define

$$\alpha_x = \frac{\Delta t}{\Delta x^2}, \quad \alpha_y = \frac{\Delta t}{\Delta y^2}. \tag{17}$$

In this case we will be using a uniform mesh, meaning $\Delta x = \Delta y$, which results in $\alpha_x = \alpha_y = \alpha$.

Now, we use the backward going Euler formula for the first derivative in time. In its discretized form we have

$$u_t \approx \frac{u_{i,j}^n - u_{i,j}^{n-1}}{\Delta t} \tag{18}$$

and this leads to

$$u_{i,j}^{n-1} = u_{i,j}^n + 4\alpha u_{i,j}^n - \alpha(u_{i+1,j}^n + u_{i-1,j}^n + u_{i,j+1}^n + u_{i,j-1}^n) \tag{19}$$

$$u_{i,j}^n = \frac{1}{1+4\alpha}\Big[\alpha(u_{i+1,j}^n + u_{i-1,j}^n + u_{i,j+1}^n + u_{i,j-1}^n) + u_{i,j}^{n-1}\Big], \tag{20}$$

or in a more compact form

$$u_{i,j}^n = \frac{1}{1+4\alpha}\Big[\alpha\Delta_{i,j}^n + u_{i,j}^{n-1}\Big] \tag{21}$$

where $\Delta_{i,j}^n = u_{i+1,j}^n + u_{i-1,j}^n + u_{i,j+1}^n + u_{i,j-1}^n$.

It is possible to show that this equation can be transformed into a linear algebra problem of the type $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{w}$, with $\boldsymbol{A}$ being a matrix, $\boldsymbol{x}$ an unknown and $\boldsymbol{w}$ a known vector, this can be found here [1].

To solve this problem one can utilize iterative solvers, such as Jacobi's method. When setting up Jacobi's method, it is useful to rewrite the matrix $\boldsymbol{A}$ as

$$\boldsymbol{A} = \boldsymbol{D} + \boldsymbol{U} + \boldsymbol{L} \tag{22}$$

where $\boldsymbol{D}$ is a diagonal matrix, $\boldsymbol{U}$ is an upper triangular matrix and $\boldsymbol{L}$ is a lower triangular matrix. One can then rewrite the equation $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{w}$ in terms of the matrices $\boldsymbol{D}, \boldsymbol{U}$ and $\boldsymbol{L}$.

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{w} \tag{23}$$

$$(\boldsymbol{D} + \boldsymbol{U} + \boldsymbol{L})\boldsymbol{x} = \boldsymbol{w} \tag{24}$$

$$\boldsymbol{x} = \boldsymbol{D}^{-1}\big(\boldsymbol{w} - (\boldsymbol{U} + \boldsymbol{L})\boldsymbol{x}\big) \tag{25}$$

$$\boldsymbol{x}_{i+1} = \boldsymbol{D}^{-1}\big(\boldsymbol{w} - (\boldsymbol{U} + \boldsymbol{L})\boldsymbol{x}_i\big) \quad i = 0, 1, 2, \ldots \tag{26}$$

However, in order for Jacobi's method to converge, there are some conditions that must be met. This method requires that the matrix $\boldsymbol{A}$ must be a positive or semi-positive definite matrix as well as being diagonally dominant.

2.4 Temperature distribution in the lithosphere

The purpose of this part is to calculate the thermal evolution of the lithosphere up to the present, following the emplacement of radioactive elements in the mantle wedge 1 Gy ago.

What we need to compute is the heat equation, which models the time evolution of the temperature distribution in the lithosphere.

$$\vec{\nabla}(k\vec{\nabla}T) + Q = \rho c_p \frac{\partial T}{\partial t} \tag{27}$$

Here, T is the temperature, ρ is the density, k is the thermal conductivity, c_p is the specific heat capacity and Q is the heat production. In addition, we will only be studying the equation in 2D.

To simplify the system, we will separate the lithosphere in 3 layers: The upper crust, the lower crust and the mantle, all with different depth. Each areas heat production is listed below.

Table 2: The heat production (Q) of the different depth systems.

System	**Heat production** Q $[\mu W/m^3]$	**Depth** [km]
Upper Crust	1.4	0-20
Lower Crust	0.35	20-40
Mantle	0.05	40-120

Equation 27 has analytical solutions when studying a steady state situation, and the full derivation for this can be found in Appendix C. These solutions will be used as benchmark results to the numerical ones.

2.4.1 Heat production

Furthermore, we will study this system when the lower layer of the lithosphere (mantle) is affected by the heat production of the radioactive elements Uranium (U), Thorium (Th) and Potassium (K). We will look at the system in two situations. The first is when the heat production of the radioactive elements will remain constant over geological time and the second is when the radioactivity will decrease with time, due to decay.

The first situation is straight forward, but to study the system where radioactivity will decrease with time, we need to include the half-lives of the radioactive elements. To implement this, the radioactive half-life formula is utilized

$$m_t = m_0 e^{-kt} \tag{28}$$

where m_t is the mass of a material at time t, m_0 is the initial mass at time $t = 0$ and k is the material's characteristic decay constant.
The half-lives, concentrations, heat productions and decay constants of each material is presented in table 3.

Table 3: The distribution and heat production contributions of the radioactive elements U, Th, K in the mantle.

Radioactive Elements	**Concentration** [%]	**Halflife** [1Ga]	Q $[\mu W/m^3]$	**Decay constant** k $[Gy^{-1}]$
Uranium U	40	4.47	0.2	0.155
Thorium Th	40	14.0	0.2	0.0495
Potassium K	20	1.25	0.1	0.555

By utilizing equation 28 and the data in table 3, we can set up an equation that describes the heat production of the materials as a function of time.

$$Q_U(t) = Q_U e^{-kt} = 0.2e^{-0.155t}, \quad Q_{Th}(t) = Q_{Th}e^{-kt} = 0.2e^{-0.0495t} \quad \& \quad Q_K(t) = Q_K e^{-kt} = 0.1e^{-0.555t} \tag{29}$$

$$Q_{tot}(t) = Q_U(t) + Q_{Th}(t) + Q_K(t) = 0.2e^{-0.155t} + 0.2e^{-0.0495t} + 0.1e^{-0.555t} \tag{30}$$

$Q_{tot}(t)$ is the expression for the additional heat production caused by the radioactive elements.

2.4.2 Discretization & Scaling of Equations

To solve the PDE numerically, we discretize it in the following way:

$$T(x, y, t) = T(x_i, y_i, t_i) = T^n_{i,j}$$

Here, we use the same approximations for the derivatives as for the implicit case in 2D.

$$\begin{aligned}
\frac{\partial^2 T}{\partial x^2} = T^{xx} &\approx \frac{T(x_i + \Delta x, y_j, t_n) - 2T(x_i, y_j, t_n) + T(x_i - \Delta x, y_j, t_n)}{\Delta x^2} \\
\frac{\partial^2 T}{\partial y^2} = T^{yy} &\approx \frac{T(x_i, y_j + \Delta y_j, t_n) - 2T(x_i, y_j, t_n) + T(x_i, y_j - \Delta y, t_n)}{\Delta y^2} \\
\frac{\partial T}{\partial t} = T^t &\approx \frac{T(x_i, y_j, t_n) - T(x_i, y_j, t_n - \Delta t)}{\Delta t}
\end{aligned}$$

By inserting these expressions into equation 27, it can then be rewritten as

$$\begin{aligned}
k\Big(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}\Big) + Q &= \rho c_p \frac{\partial T}{\partial t} \\
k\big(T^{xx} + T^{yy}\big) + Q &= \rho c_p T^t
\end{aligned}$$

Using $\Delta x = \Delta y$ and insetting the discretized versions of the derivatives, the equation now reads

$$\begin{aligned}
\frac{k}{\Delta x^2}\big(T^n_{i+1,j} + T^n_{i-1,j} + T^n_{i,j+1} + T^n_{i,j-1} - 4T^n_{i,j}\big) + Q &= \rho c_p \Big(\frac{T^n_{i,j} - T^{n-1}_{i,j}}{\Delta t}\Big) \\
\frac{k}{\rho c_p}\frac{\Delta t}{\Delta x^2}\big(\Delta^n_{i,j} - 4T^n_{i,j}\big) + \frac{\Delta t}{\rho c_p} Q &= T^n_{i,j} - T^{n-1}_{i,j}
\end{aligned}$$

where $\Delta^n_{i,j} = T^n_{i+1,j} + T^n_{i-1,j} + T^n_{i,j+1} + T^n_{i,j-1}$.

Rewriting the expression above to isolate $T^n_{i,j}$ on one side, gives

$$T^n_{i,j} = \frac{1}{1 + 4\frac{k\Delta t}{\rho c_p \Delta x^2}} \Big[\frac{k\Delta t}{\rho c_p \Delta x^2} \Delta^n_{i,j} + \frac{\Delta t}{\rho c_p} Q + T^{n-1}_{i,j} \Big] \tag{31}$$

By introducing the parameter T_s that scales $T \in [0, 1]$ and the parameters x_s, t_s that scales the step sizes Δx and Δt, it follows that

$$T^n_{i,j} = \frac{1}{1 + 4\frac{kt_s\Delta t}{\rho c_p x_s^2 \Delta x^2}} \Big[\frac{kt_s\Delta t}{\rho c_p x_s^2 \Delta x^2} \Delta^n_{i,j} + \frac{t_s \Delta t}{T_c \rho c_p} Q + T^{n-1}_{i,j} \Big], \tag{32}$$

and defining the parameters

$$\alpha = \frac{\Delta t}{\Delta x^2}, \quad \beta = \frac{kt_s}{x_s^2 \rho c_p} \quad \& \quad Q_c = \frac{t_s \Delta t}{\rho c_p T_c} \tag{33}$$

returns the following expression

$$T^n_{i,j} = \frac{1}{1 + 4\alpha\beta} \Big[\alpha\beta \Delta^n_{i,j} + Q_c Q + T^{n-1}_{i,j} \Big]. \tag{34}$$

From the implicit scheme in 2D, we know that this kind of equation can be solved by transforming into a linear algebra problem and utilize an iterative solver.

3 Implementation

Programs used in this project can be found on https://github.com/jensbd/FYS4150/tree/master/Project5 and the Readme explains how to run the scripts. All calculations are done in C++, while the plotting is done in Python.
The C++ programs are mainly based on the program codes found in the course's github repository: https://github.com/CompPhysics/ComputationalPhysics [1]. We have also utilized the C++ library Armadillo [2] [3].

4 Results and Discussion

4.1 1D

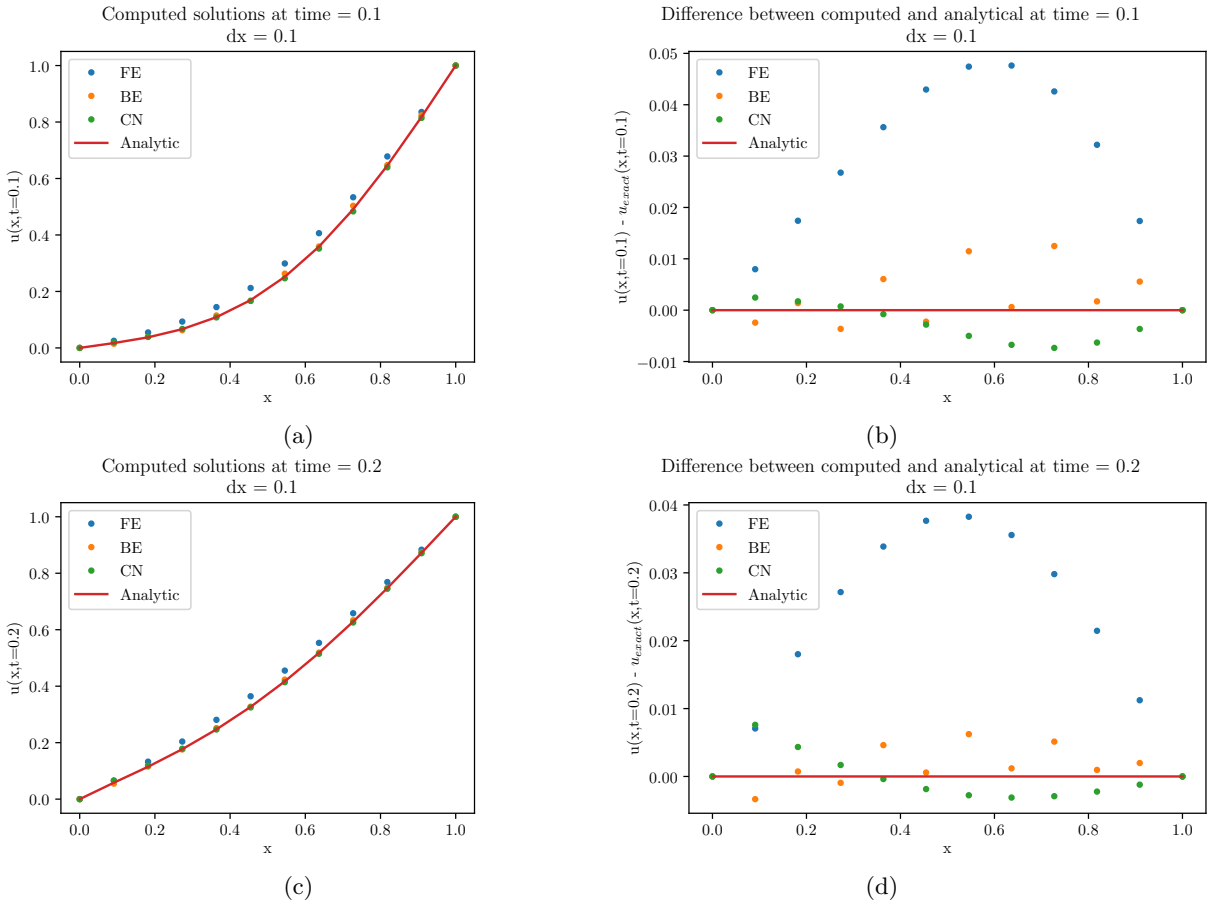


Figure 1: Comparison of the three schemes and analytic solution at time $t_1 = 0.1$ and $t_2 = 0.2$ with a step length of $\Delta x = 0.1$ and $\Delta t = 0.5\Delta x^2$.

From figure 1, we see that the explicit Forward Euler scheme is the worst performer, compared to the implicit schemes. Backward Euler and Crank-Nicolson are more closely matched, with a slightly better performance from Crank-Nicolson. These results match the truncation errors listed in table 1. Since Δt is so small compared to Δx the error will probably be dominated by Δx, which is confirmed by figure 1, where both implicit schemes give similar results.

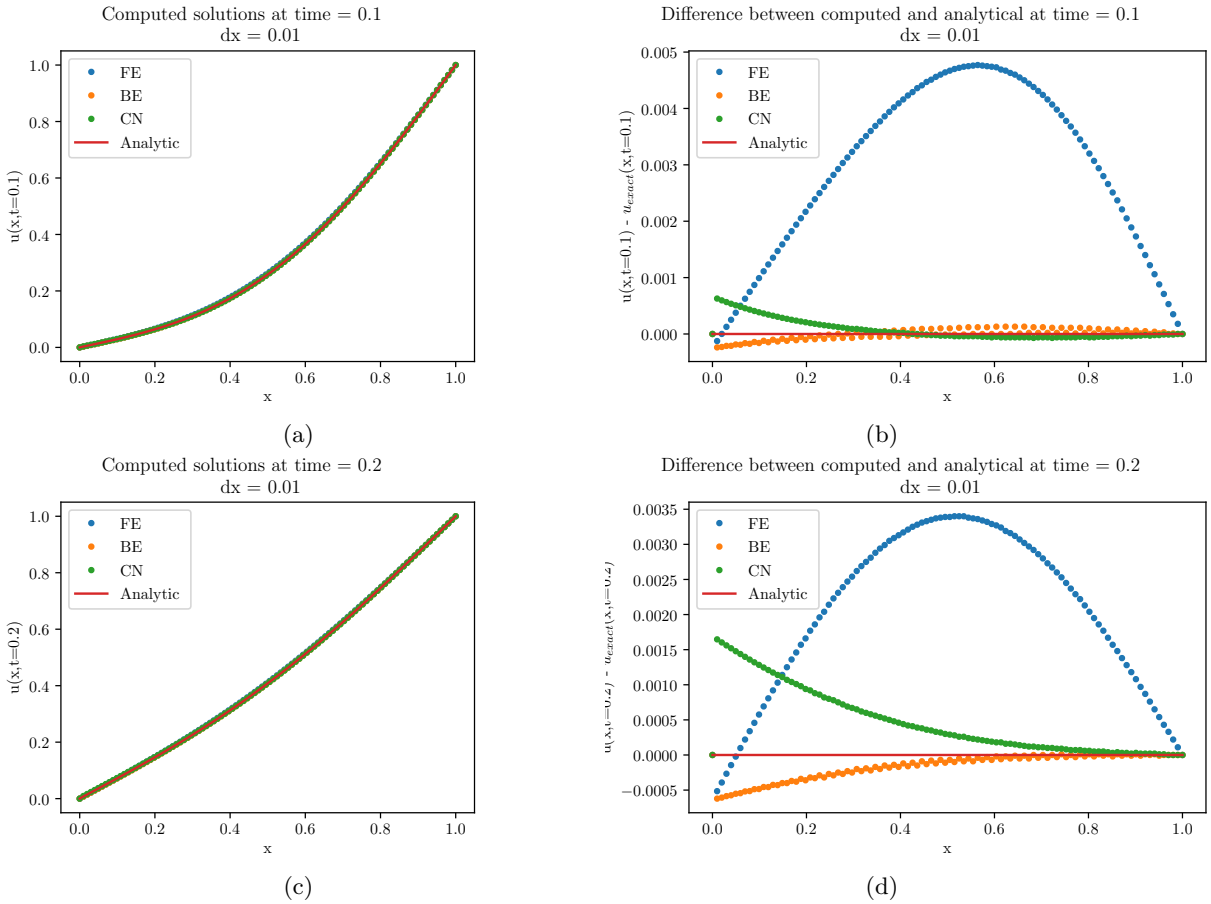
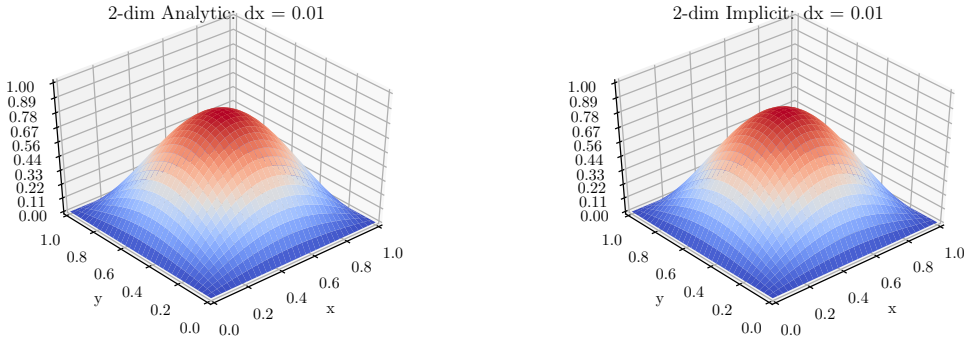


Figure 2: Comparison of the three schemes and analytic solution at time $t_1 = 0.1$ and $t_2 = 0.2$ with a step length of $\Delta x = 0.01$ and $\Delta t = 0.5\Delta x^2$.

In figure 2, where $\Delta x = 0.01$, we see the same trend as for $\Delta x = 0.1$, where forward Euler is still significantly more imprecise than the implicit schemes. We also observe that the errors for all schemes have been reduced by a factor 10, as expected from the change of Δx. However in this scenario, backward Euler seems to be the best performer, with an absolute maximum difference of ≈ 0.0005. The differences in this case are much smaller than for $\Delta x = 0.1$, showing that Crank-Nicolson seems to be the most accurate method of the three for a set of different step sizes.

From these results, we can conclude that the the implicit schemes have a much better performance than the explicit one, leading us to choose an implicit method when going forward in this project.

4.2 2D



(a) (b)

Figure 3: Comparison of analytical and numerical solution from the implicit scheme in 2 dimensions. We observe a very close match between the two figures.

Table 4: Absolute mean differences between the 2-dimensional analytical solution and implicit scheme for different combinations of Δx and Δt, evaluated at one point in time.

Absolute Mean Error		
Δt	$\Delta x = 10^{-1}$	$\Delta x = 10^{-2}$
Δx	$7.87 \cdot 10^{-2}$	$6.51 \cdot 10^{-3}$
$\Delta x/10$	$5.11 \cdot 10^{-2}$	$5.05 \cdot 10^{-3}$
$\Delta x/100$	$4.83 \cdot 10^{-2}$	$5.07 \cdot 10^{-3}$

From table 4, we see that the error, as expected, is lower for the lower values of Δx and Δt. Interestingly however, this does not seem to hold for when $\Delta t = \Delta x/100$. Here, the error is more or less unchanged compared to the larger step-size above. We seem to have reached a limit for how small we can make Δt compared to Δx, and there may be very little to gain by decreasing Δt any further. Reducing the error from here will probably require a reduction in Δx, which will lead to a significant increase in runtime.

4.3 Lithosphere

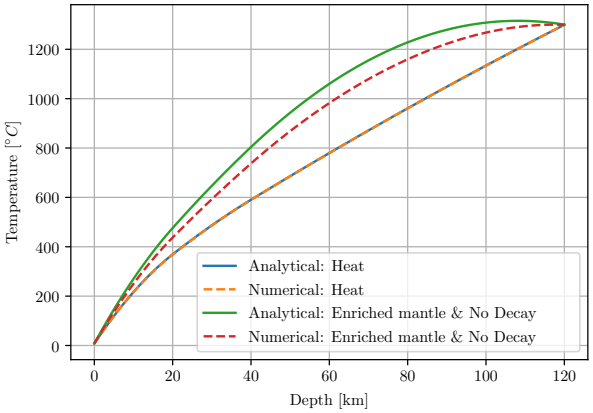


Figure 4: Comparison of the analytical solution and numerical simulation of heat distribution in the lithosphere for two different cases.
Case 1 shows the situation for heat production separated in each lithosphere layer.
Case 2 is the steady state situation for heat production and a constant enrichment of radioactive materials without decay.

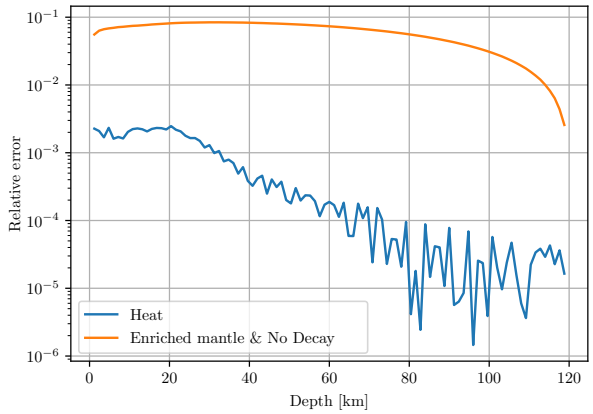


Figure 5: The relative error vs. depth between the analytical and numerical solution for case 1 & 2 shown above.

From figure 4, we see that the first case is almost perfectly approximated by the numerical simulation, but the second case shows a larger difference between the analytical solution and the numerical simulation. This is also illustrated by the relative error in figure 5, where the error is quite high for case 2 compared to case 1.

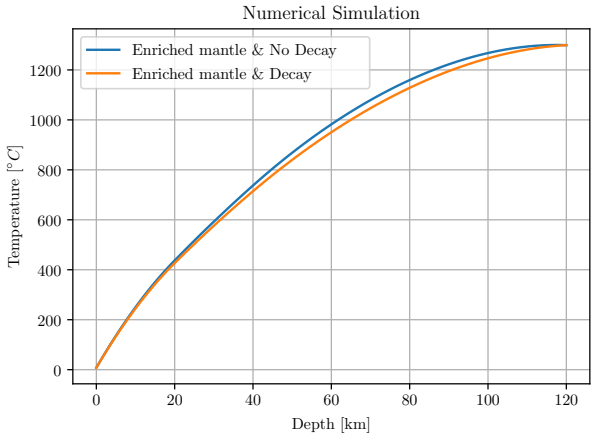


Figure 6: Comparison of numerical simulation of heat distribution in the lithosphere for two different cases. Case 2 is the steady state situation for heat production and a constant enrichment of radioactive materials without decay. Case 3 shows the situation with an enriched mantle with radioactive decay.

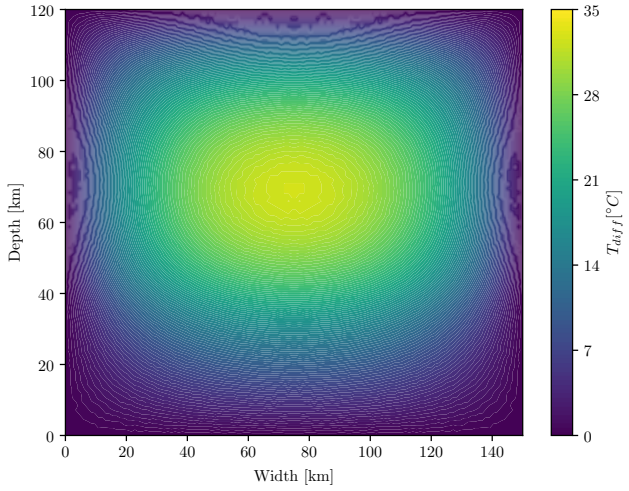


Figure 7: An illustration of the difference between case 2 & 3 shown above.
Largest temperature difference is 32.6°C and is located at width 70.8 km and depth 68.4 km.

Figure 6 shows that in the beginning the two cases are similar, and that is because the heat productions from the radioactive materials are the same during that period.
From figure 7 we observe that the temperature difference is highest in the mantle, and decreases towards zero in the boundaries. The maximum temperature difference between the cases is computed to be 32.6°C at a depth of 68.4 km, which is in the mantle, and this makes sense since the mantle is where we included the decay of the radioactive materials. These results indicate that the decay of the radioactive materials has an impact on the system.

5 Conclusion

In this project, we have developed several methods for solving the general 1D diffusion equation. Comparing with the analytical solution showed us the generally good performance of the explicit forward Euler scheme, as well as the implicit backward Euler and Crank-Nicolson schemes. However, we observed that the implicit schemes beat the explicit for both analysed spatial step sizes Δx, with Crank-Nicolson being the best for $\Delta x = 0.1$ and backward Euler edging out the victory at $\Delta x = 0.01$. For determining the best performing method of the three, we decided that the consistently strong performance of Crank-Nicolson for both step sizes to be the winning characteristic.

With the knowledge of the superior accuracy from the implicit methods, we decided to utilize an implicit method when tackling the 2D case. The resulting discretization showed again a very satisfactory approximation of the analytical solution. Reducing Δx and Δt resulted in a reduction of the mean error as well, and we observed that since Δt is dependent on Δx, the lowering of Δx would result in the largest gain of accuracy.

The last part of this project was to model the temperature distribution of the lithosphere for different levels of complexity. We observed an almost perfect approximation of the case with just heat, and a close, but not optimal performance for the case with the enriched mantle and no radioactive decay. For the last case, with decay included, our numerical simulation is quite similar to the no-decay case, with the exception in the mantle where the highest temperature difference of 32.6°C was discovered. This model may be representative of the general trend in the actual lithosphere, but due to the mentioned errors, can not be confidently assumed to describe the real situation perfectly. This leaves ample room for future studies into the further reduction of errors, and perfection of the model.

Appendices

A Analytical solution (1 dim)

We start of with the PDE

$$\frac{\partial u(x,t)}{\partial t} = \frac{\partial^2 u(x,t)}{\partial x^2}$$

with initial conditions, i.e., the conditions at $t = 0$,

$$u(x,0) = 0 \qquad 0 < x < L$$

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

$$u(0,t) = 0 \qquad t \geq 0,$$

and

$$u(L,t) = 1 \qquad t \geq 0.$$

Then we make an assumption that the function $u(x,t)$ can be written in the form

$$u(x,t) = v(x,t) + ax$$

where $v(x,t)$ is the solution with boundary conditions $v(0,t) = 0$ and $v(1,t) = 0$, and where a is just a constant.

Then we make an ansatz, where we assume that the function $v(x,t)$ can be seperated in terms of it's variables.

$$v(x,t) = F(x)G(t)$$

If we plug this into the equation above, we get

$$F(x)\frac{\partial G(t)}{\partial t} = G(t)\frac{\partial^2 F(x)}{\partial x^2}$$

Then we define the derivatives of the functions $F(x)$ and $G(t)$ as

$$\frac{\partial G(t)}{\partial t} = G'(t) \quad \& \quad \frac{\partial^2 F(x)}{\partial x^2} = F''(x)$$

Then, the equations can be rewritten as

$$F(x)G'(t) = G(t)F''(x)$$
$$\frac{G'(t)}{G(t)} = \frac{F''(x)}{F(x)}$$

where the derivative is with respect to x on the left hand side (lhs) and with respect to t on the right hand side (rhs). This equation should hold for all x and t . We must then, require the rhs and lhs to be equal to a constant. We call this constant $-\omega^2$. This gives us the two differential equations,

$$\begin{aligned}
&\frac{G'(t)}{G(t)} = -\omega^2 \quad \& \quad \frac{F''(x)}{F(x)} = -\omega^2 \\
&G'(t) = -\omega^2 G(t) \quad \& \quad F''(x) + \omega^2 F(x) = 0
\end{aligned}$$

The general solutions to these differential equations are trivial and are equal to

$$G(t) = Ce^{-\omega^2 t} \quad \& \quad F(x) = A\sin(\omega x) + B\cos(\omega x)$$

By applying the boundary conditions of $v(x,t)$, we get

$$\begin{aligned}
&v(x,t) = F(x)G(t) \\
&v(0,t) = F(0)G(t) = 0 \quad \rightarrow \quad \big(A\sin(\omega \times 0) + B\cos(\omega \times 0)\big)G(t) = 0 \quad \rightarrow \quad B = 0 \\
&v(L,t) = F(L)G(t) = 0 \quad \rightarrow \quad A\sin(\omega \times L)G(t) = 0 \quad \rightarrow \quad \sin(\omega \times L) = 0 \quad \rightarrow \quad \omega = \frac{n\pi}{L}
\end{aligned}$$

To satisfy the boundary conditions we require $B = 0$ and $\omega = \frac{n\pi}{L}$. One solution is therefore found to be

$$v(x,t) = F(x)G(t) = A\sin(wx)Ce^{-\omega^2 t} = A\sin\big(\frac{n\pi}{L}x\big)Ce^{-\frac{n^2\pi^2}{L^2}t} = A_n\sin\big(\frac{n\pi}{L}x\big)e^{-\frac{n^2\pi^2}{L^2}t}$$

where we define the constant $A_n = A \cdot C$.

But there are infinitely many possible n values (infinite number of solutions). Moreover, the diffusion equation is linear and because of this we know that a superposition of solutions will also be a solution of the equation. We may therefore write

$$v(x,t) = \sum_{n=1}^{\infty} A_n\sin\big(\frac{n\pi}{L}x\big)e^{-\frac{n^2\pi^2}{L^2}t}$$

Now if we use this in the definition of $u(x,t)$ and apply it's boundary conditions, we end up with

$$\begin{aligned}
&u(x,t) = ax + v(x,t) \\
&u(0,t) = a \times 0 + v(0,t) = 0 \\
&u(L,t) = a \times L + v(L,t) = 1 \quad \rightarrow \quad a = \frac{1}{L} \\
&u(x,t) = ax + v(x,t) = \frac{x}{L} + \sum_{n=1}^{\infty} A_n\sin\big(\frac{n\pi}{L}x\big)e^{-\frac{n^2\pi^2}{L^2}t}
\end{aligned}$$

The coefficient A_n is in turn determined from the initial condition, which in this case is $u(x,0) = 0$.

$$\begin{aligned}
u(x,0) &= \frac{x}{L} + \sum_{n=1}^{\infty} A_n\sin\big(\frac{n\pi}{L}x\big)e^{-\frac{n^2\pi^2}{L^2}\times 0} \\
0 &= \frac{x}{L} + \sum_{n=1}^{\infty} A_n\sin\big(\frac{n\pi}{L}x\big) \quad \rightarrow \quad \sum_{n=1}^{\infty} A_n\sin\big(\frac{n\pi}{L}x\big) = -\frac{x}{L}
\end{aligned}$$

The coefficient A_n is the Fourier coefficients for the function $g(x)$, which in this case is the expression $-\frac{x}{L}$. Because of this, A_n is given by (from the theory on Fourier series)

$$A_n = \frac{2}{L}\int_0^L g(x)\sin\big(\frac{n\pi x}{L}\big)dx$$

$$A_n = -\frac{2}{L}\int_0^L \frac{x}{L}\sin\big(\frac{n\pi x}{L}\big)dx, \quad u = \frac{n\pi}{L}x, \quad x = \frac{L}{n\pi}u \quad u(0) = 0, \quad u(L) = n\pi, \quad du = \frac{n\pi}{L}dx \quad \rightarrow \quad dx = \frac{L}{n\pi}du$$

$$A_n = -\frac{2}{L}\int_0^L \frac{x}{L}\sin\big(\frac{n\pi x}{L}\big)dx = -\frac{2}{L}\int_0^{n\pi} \frac{u}{n\pi}\sin u\frac{L}{n\pi}du = -\frac{2}{(n\pi)^2}\int_0^{n\pi} u\sin u du$$

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

$$A_n = (-1)^n\frac{2}{n\pi}$$

Then at last we have found the analytical solution to the PDE of $u(x,t)$ which is

$$u(x,t) = ax + v(x,t) = \frac{x}{L} + \sum_{n=1}^{\infty}\frac{(-1)^n 2}{n\pi}\sin\big(\frac{n\pi}{L}x\big)e^{-\frac{n^2\pi^2}{L^2}t}$$

B Analytical solution (2 dim)

In 2-dimensions the diffusion equation transforms into

$$\frac{\partial u(x,y,t)}{\partial t} = \frac{\partial^2 u(x,y,t)}{\partial x^2} + \frac{\partial^2 u(x,y,t)}{\partial y^2}$$

and in this project we will be looking into the model with a square lattice for x and y.
In addition, we will use the simple boundary conditions of

$$u(0,y,t) = u(L,y,t) = 0 \qquad y \in (0,L) \qquad t \geq 0,$$

and

$$u(x,0,t) = u(x,L,t) = 0 \qquad x \in (0,L) \qquad t \geq 0,$$

with initial conditions, i.e., the conditions at $t = 0$,

$$u(x,y,0) = \sin(\pi x)\sin(\pi y) \qquad 0 < x, y < L$$

with $L = 1$ the length of the x, y-region of interest.

Just as we did in the 1-dimensional case, we make an ansatz, where we assume that the function $u(x,y,t)$ can be seperated in terms of it's variables.

$$u(x,y,t) = F(x)G(y)H(t)$$

If we plug this into the equation above, we get

$$F(x)G(y)\frac{\partial H(t)}{\partial t} = G(y)H(y)\frac{\partial^2 F(x)}{\partial x^2} + F(x)H(t)\frac{\partial^2 G(y)}{\partial y^2}$$

Then we define the derivatives of the functions $F(x)G(t)$ and $H(t)$ as

$$\frac{\partial H(t)}{\partial t} = H'(t) \quad \& \quad \frac{\partial^2 F(x)}{\partial x^2} = F''(x) \quad \& \quad \frac{\partial^2 G(y)}{\partial y^2} = G''(y)$$

Then, the equations can be rewritten as

$$\begin{aligned} F(x)G(y)H'(t) &= G(y)H(t)F''(x) + F(x)H(t)G''(y) \\ \frac{H'(t)}{H(t)} &= \frac{F''(x)}{F(x)} + \frac{G''(y)}{G(y)} \end{aligned}$$

where the derivative is with respect to t on the left hand side (lhs) and with respect to x and y on the right hand side (rhs). This equation should hold for all x, y and t. We must then, require the rhs and lhs to be equal to a constant. We call this constant $-a^2$. This gives us the two differential equations,

$$\frac{H'(t)}{H(t)} = -a^2 \quad \& \quad \frac{F''(x)}{F(x)} + \frac{G''(y)}{G(y)} = -a^2 = -b^2 - c^2$$

We further assume that $\frac{F''(x)}{F(x)}$ is equal to a constant b and $\frac{G''(y)}{G(y)}$ is equal to a constant c, so that $a^2 = b^2 + c^2 \quad \rightarrow \quad a = \sqrt{b^2 + c^2}$.

$$H'(t) = -a^2 H(t) \quad \& \quad F''(x) + bF(x) = 0 \quad \& \quad G''(y) + cG(y) = 0$$

The general solutions to these differential equations are trivial and are equal to

$$H(t) = Ae^{-a^2 t} \quad \& \quad F(x) = B\sin(bx) + C\cos(bx) \quad \& \quad G(y) = D\sin(cy) + E\cos(cy)$$

To satisfy the boundary conditions we require $C = E = 0$, $b = \frac{n\pi}{L}$ and $c = \frac{m\pi}{L}$. One solution is therefore found to be

$$\begin{aligned} u(x,y,t) = F(x)G(y)H(t) &= B\sin(bx)D\sin(cy)Ae^{-a^2 t} = B\sin\big(\frac{n\pi}{L}x\big)D\sin\big(\frac{m\pi}{L}y\big)Ae^{-\frac{(n^2+m^2)\pi^2}{L^2}t} \\ &= A_n \sin\big(\frac{n\pi}{L}x\big)\sin\big(\frac{n\pi}{L}y\big)e^{-\frac{2n^2\pi^2}{L^2}t} \end{aligned}$$

where we define the constant $A_n = A \cdot B \cdot D$, and since we are working with a square lattice, then $n = m$, and that simplifies the expression as shown above.

The coefficient A_n is in turn determined from the initial condition. We require

$$u(x,y,0) = \sin(\pi x)\sin(\pi y) = A_n \sin\big(\frac{n\pi}{L}x\big)\sin\big(\frac{n\pi}{L}y\big)e^0$$

By utilizing $L = 1$, we see that $n = 1$ and $A_n = 1$ to satisfy the initial conditions. Then that means the function $u(x,y,t)$ can be read as

$$u(x,y,t) = A_n \sin\big(\frac{n\pi}{L}x\big)\sin\big(\frac{n\pi}{L}y\big)e^{-\frac{2n^2\pi^2}{L^2}t} = \sin(\pi x)\sin(\pi y)e^{-2\pi^2 t}$$

C Analytical steady state solutions to temperature distribution in the Lithosphere

Before radioactive enrichment, the temperature is steady-state $\frac{\partial T}{\partial t} = 0$ and depends only on the depth.

In this case, there exists analytical solutions.

$$\begin{aligned} k\frac{\partial^2 T}{\partial x^2} + Q &= \frac{\partial T}{\partial t} = 0 \\ \frac{\partial^2 T}{\partial x^2} &= -\frac{Q}{k} \\ \frac{\partial T}{\partial x} &= -\frac{Q}{k}x + b \\ T(x) &= -\frac{Q}{2k}x^2 + bx + c = ax^2 + bx + c \end{aligned}$$

where $a = -\frac{Q}{2k}$, b and c are integration constants, and x is the depth in km. But since we are looking at three different systems with the upper crust, $T_1(x)$, defined from 0 to 20 km depth, the lower crust ,$T_2(x)$, from 20 to 40 km depth and the mantle, $T_3(x)$, from 40 to 120 km depth, they will all have distinct functions with distinct integration constants regarding b and c.

By requiring continuous temperatures and derivatives in the transition between the different systems

$$\begin{aligned} T_1(x = -20) = T_2(x = -20) \quad \& \quad \frac{\partial}{\partial x}T_1(x = -20) = \frac{\partial}{\partial x}T_2(x = -20) \\ T_2(x = -40) = T_3(x = -40) \quad \& \quad \frac{\partial}{\partial x}T_2(x = -40) = \frac{\partial}{\partial x}T_3(x = -40) \end{aligned}$$

and applying the boundary conditions of the total system, mainly

$$\begin{aligned} T_1(0) &= c_1 = 8 \\ T_3(120) &= a_3 \cdot (120)^2 + b_3 \cdot (120) + c_3 = 1300 \end{aligned}$$

we end up with the following expressions for $T_1(x), T_2(x)$ and $T_3(x)$,

$$\begin{aligned} T_1(x) &= -0.28x^2 - 23.66x + 8 \\ T_2(x) &= -0.07x^2 - 15.26x + 92 \\ T_1(x) &= -0.01x^2 - 10.46x + 188. \end{aligned}$$

With the addition of the radioactive elements Uranium (U), Thorium (Th) and Potassium (K) that provides an additional total heat production of $Q_{tot} = 0.5\mu W/m^3$ leads to different coefficients of a, b and c in the analytic solutions of T_1, T_2 and T_3. When including the extra heat production of the radioactive elements, the functions T_1, T_2 and T_3 now reads

$$\begin{aligned} T_1(x) &= -0.28x^2 - 29x + 8 \\ T_2(x) &= -0.07x^2 - 20.6x + 92 \\ T_1(x) &= -0.11x^2 - 23.8x + 28 \end{aligned}$$

References

[1] Morten Hjorth-Jensen, Github Repository, http://compphysics.github.io/ComputationalPhysics/doc/web/course

[2] Conrad Sanderson and Ryan Curtin. Armadillo: a template-based C++ library for linear algebra. Journal of Open Source Software, Vol. 1, pp. 26, 2016.

[3] Conrad Sanderson and Ryan Curtin. A User-Friendly Hybrid Sparse Matrix Class in C++. Lecture Notes in Computer Science (LNCS), Vol. 10931, pp. 422-430, 2018.