Final

1 Introduction

Numerical methods are often used to solve problems when analytic approaches prove to be impractical. This final presents two multi-part problems that make use of a couple of numerical methods in order to determine the cooking time of a potato and the concentration of oil after an oil spill. For the potato, it was given that its temperature satisfied the heat equation:

$$\frac{\partial T}{\partial t} = \lambda \Delta T, \qquad (x, y) \in \Omega \tag{1}$$

with the boundary conditions

$$T(t, x, y) = T_{water}(t), \qquad (x, y) \in \partial\Omega$$

and the initial conditions

$$T(0, x, y) = T_{room}, \qquad (x, y) \in \Omega$$

where λ is the thermal diffusivity of the potato's material, Ω and $\partial\ \Omega$ denote the domain in space occupied by the potato and the boundary of that domain.

For the oil spill, it was given that the diffusion of oil could be simulated using the advection-diffusion equation:

$$\frac{\partial T}{\partial t} + \vec{v} \bullet \nabla c = \lambda \Delta T + f, \qquad (x, y) \in \Omega \tag{2}$$

with the boundary conditions

$$T(t, x, y) = T_{bc}(t, x, y), \qquad if x = x_l, x = x_r or y = y_t,$$

$$\lambda \frac{\partial T}{\partial y} - v_y T = g(t, x, y), \qquad if y = y_b,$$

and the initial conditions

$$T(t_{start}, x, y) = T_{start}(x, y), \qquad (x, y) \in \Omega$$

2 Heat Equation

2.1 Generalized Heat Equation

Before solving for the time at which the potato was fully cooked, I first considered a generalized heat transfer problem and used the implicit scheme to discretize and then solve the generalized heat equation. The generalized heat equation looks as follows:

$$\begin{gathered}\tfrac{\partial T}{\partial t} = \lambda \Delta T + f, \qquad (x, y) \in \Omega \\ T(t, x, y) = T_{bc}(t, x, y), \qquad (x, y) \in \partial\Omega \\ T(t, x, y) = T_{start}(x, y), \qquad (x, y) \in \Omega\end{gathered} \tag{3}$$

where the rectangular domain $\Omega = [x_l; x_r]\text{x}[y_b; y_t]$, λ is the thermal diffusivity of the material, and $f = f(t, x, y), T_{bc}(t, x, y)$ and $T_{start}(x, y)$ are given functions describing the source term, boundary conditions and initial conditions, respectively. Using this information, we can attempt to consider this generalized heat equation at the moment of time t_{n+1} and location (x_i, y_j):

$$\frac{\partial T}{\partial t}(t_{n+1}, x_i, y_j) = \lambda \frac{\partial^2 T}{\partial x^2}(t_{n+1}, x_i, y_j) + \lambda \frac{\partial^2 T}{\partial y^2}(t_{n+1}, x_i, y_j) + f(t_{n+1}, x_i, y_j)$$

Using the backward finite difference formula to approximate the time derivative

$$\frac{\partial T}{\partial t}(t_{n+1}, x_i, y_j) \approx \frac{T(t_{n+1}, x_i, y_j) - T(t_n, x_i, y_j)}{\Delta t} \approx \frac{T^{n+1}_{i,j} - T^n_{i,j}}{\Delta t},$$

and the central finite difference formula to approximate the spatial derivatives in the x and y directions

$$\frac{\partial^2 T}{\partial x^2}(t_{n+1}, x_i, y_j) \approx \frac{T(t_{n+1}, x_{i+1}, y_j) - 2T(t_{n+1}, x_i, y_j) + T(t_{n+1}, x_{i-1}, y_j)}{\Delta x^2} \approx \frac{T^{n+1}_{i+1,j} - 2T^{n+1}_{i,j} + T^{n+1}_{i-1,j}}{\Delta x^2},$$

$$\frac{\partial^2 T}{\partial y^2}(t_{n+1}, x_i, y_j) \approx \frac{T(t_{n+1}, x_i, y_{j+1}) - 2T(t_{n+1}, x_i, y_j) + T(t_{n+1}, x_i, y_{j-1})}{\Delta y^2} \approx \frac{T^{n+1}_{i,j+1} - 2T^{n+1}_{i,j} + T^{n+1}_{i,j-1}}{\Delta y^2},$$

produces the following expression

$$\frac{T^{n+1}_{i,j} - T^n_{i,j}}{\Delta t} = \lambda \frac{T^{n+1}_{i+1,j} - 2T^{n+1}_{i,j} + T^{n+1}_{i-1,j}}{\Delta x^2} + \lambda \frac{T^{n+1}_{i,j+1} - 2T^{n+1}_{i,j} + T^{n+1}_{i,j-1}}{\Delta y^2} + f(t_{n+1}, x_i, y_j)$$

Multiplying the above expression by Δt and consolidating unknown terms to the left-hand side yields

$$T^{n+1}_{i,j-1}(-\frac{\lambda \Delta t}{\Delta y^2} + T^{n+1}_{i-1,j}(-\frac{\lambda \Delta t}{\Delta x^2} + T^{n+1}_{i,j}(1 + 2\frac{\lambda \Delta t}{\Delta x^2} + 2\frac{\lambda \Delta t}{\Delta y^2})$$

$$+T_{i+1,j}^{n+1}(-\frac{\lambda\Delta t}{\Delta x^2}) + T_{i,j+1}^{n+1}(-\frac{\lambda\Delta t}{\Delta y^2}) = T_{i,j}^{n} + \Delta t f(t_{n+1}, x_i, y_j)$$

The above expression can be written for all internal points. Combining this with the set of equations that account for boundary conditions

$$T_{i,j}^{n+1} = T_{bc}(t_{n+1}, x_i, y_j)$$

and a linear system of size N_xxN_y is formed, where N_x and N_y are the number of x and y points respectively. This can be written in matrix notation using the following formatting:

$$s_p = T_{i,j}^{n+1}, \qquad p = (j-1)N_x + i, \quad i = 1, ..., N_x, \quad j = 1, ..., N_y.$$

This transforms the equations in the linear system to:

$$a_b s_{p-N_x} + a_l s_{p-1} + a_c s_p + a_r s_{p+1} + a_t s_{p+N_x} = T_{i,j}^{n} + \Delta t f(t_{n+1}, x_i, y_j)$$

for all internal grid points, and

$$s_p = c_{bc}(t_{n+1}, x_i, y_j)$$

for all boundary grid points

where

$$a_c = (1 + 2\frac{\lambda\Delta t}{\Delta x^2} + 2\frac{\lambda\Delta t}{\Delta y^2}), \quad a_l = a_r = -\frac{\lambda\Delta t}{\Delta x^2}, \quad a_b = a_t = -\frac{\lambda\Delta t}{\Delta y^2}$$

From here it is straightforward to write the linear system in standard form

$$A \bullet \vec{T}^{n+1} = \vec{r}$$

where the vector of unknowns $\vec{T}^{n+1}$ is defined as

$$\vec{T}^{n+1} = \begin{bmatrix} s_1 \\ s_2 \\ s_3 \\ . \\ . \\ . \\ s_{N_xN_y} \end{bmatrix} = \begin{bmatrix} T_{1,1}^{n+1} \\ T_{2,1}^{n+1} \\ T_{3,1}^{n+1} \\ . \\ . \\ . \\ T_{N_xN_y}^{n+1} \end{bmatrix}$$

Matrix A is of size N_xN_yxN_xN_y and vector $\vec{r}$ is of length N_xN_y. The vector $\vec{r}$ is going to be our vector of right hand side solutions while matrix A will contain 1's on all boundary grid corner points, 0's in all boundary grid points that are not a corner, and will contain a value of either a_c, a_l, a_r, a_b,or a_t depending on the position of the internal grid point.

2.2 Implementing Implicit Scheme

Using this exact method I solved the example in part b:

$$\text{Domain:} \quad \Omega = [-1;1]x[-0.5;1.7],$$

$$\text{Thermal Diffusivity:} \quad \lambda = 0.75,$$

$$\text{Exact Solution:} \quad T_{exact} = sin(x)cos(y)exp(-t),$$

where the initial conditions, boundary conditions, and source term were calculated from the given exact solution. Solving from $t_{start} = 0$ to $t_{final} = 1$ for grid resolutions $(N_x, N_y = (25,30), (50,60)$, and $(100,120)$ with a time-step $\Delta t = 0.5\Delta x$, produced the results in the following table.

Resolution	Error	Order
25x30	5.8397e-04	0
50x60	2.7845e-04	1.06844
100x120	1.3621e-04	1.03163

Table 1: This table shows the error at each grid size and the associated order of accuracy of the implicit scheme

As the *Table 1* demonstrates, the order of accuracy of this implicit scheme is order 1.

2.3 Cooking the Potato

Finally, the implicit method was implemented to simulate the process of boiling a potato where the parameters were:

$$\text{Domain:} \quad \Omega = [-1;1]x[-0.5;1.7],$$

$$\text{Thermal Diffusivity:} \quad \lambda = 1.5x10^-3cm^2/s,$$

$$\text{Initial Conidtions:} \quad T_{start}(x,y) = 20°C,$$

$$\text{Boundary Conidtions:} \quad T_{bc}(t,x,y) = min(20 + 80\tfrac{t}{60}, 100)°C,$$

$$\text{Source term:} \quad f(t,x,y) = 0,$$

where $t_{start} = 0$s and $t_{final} = 1500$s using $N_x = 80$ and $N_y = 100$ and $\Delta t = 5$s. As demonstrated by the results, the potato was fully cooked in around 1,190 seconds or just under twenty minutes.

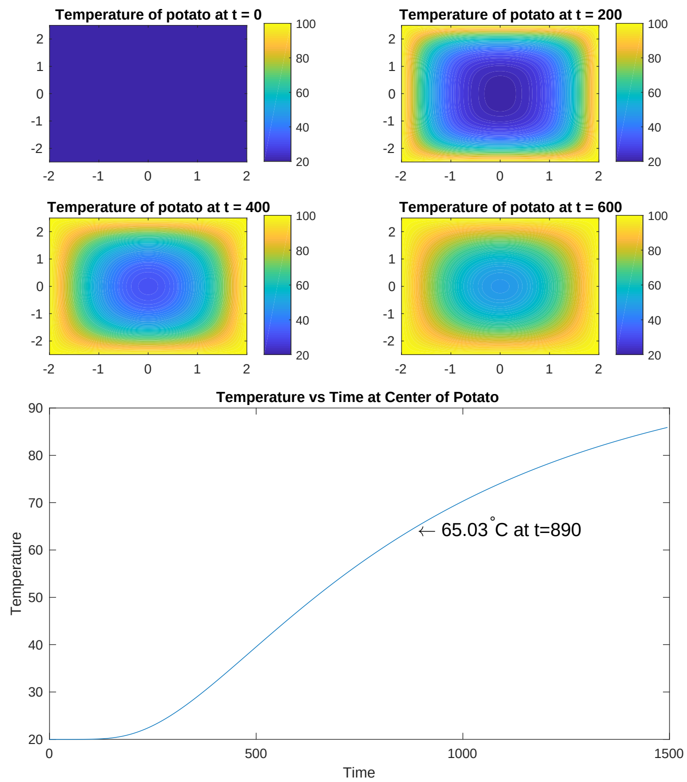

Figure 1: Cooking time of the potato is roughly 1,190 seconds

3 Advection-Diffusion Equation

3.1 Generalized Advection-Diffusion Equation

before solving the oil spill problem, I first considered a generalized advection-diffusion problem

$$\begin{aligned} \text{PDE:} \quad & \tfrac{\partial T}{\partial t} + \vec{v} \bullet \nabla c = \lambda \Delta T + f, \quad (x,y) \in \Omega \\ \text{BC:} \quad & T(t,x,y) = T_{bc}(t,x,y), \quad if x = x_l, x = x_r or y = y_t, \\ & \lambda \tfrac{\partial T}{\partial y} - v_y T = g(t,x,y), \quad if y = y_b, \\ \text{IC:} \quad & T(t_{start},x,y) = T_{start}(x,y), \quad (x,y) \in \Omega \end{aligned} \tag{4}$$

where the rectangular domain is $\Omega = [x_l; x_r]\text{x}[y_b; y_t]$, λ is the thermal diffusivity, and $f = f(t,x,y), T_{bc} = T_{bc}(t,x,y)$, $g = g(t,x,y)$, and $T_{start} = T_{start}(x,y)$ are given functions describing the source term, boundary conditions and initial conditions, respectively.

Using this information, we can attempt to consider this generalized advection-diffusion equation by explicitly approximating the advection term at the moment of time t_n and implicitly approximating the diffusion term at time t_{n+1}. Doing so gives us the following approximation to the advection-diffusion equation:

$$\frac{T_{i,j}^{n+1} - T_{i,j}^{n}}{\Delta t} + v_x \frac{T_{i+1,j}^{n} - T_{i,j}^{n}}{\Delta x} + v_y \frac{T_{i,j+1}^{n} - T_{i,j}^{n}}{\Delta x}$$

$$= \lambda \frac{T_{i+1,j}^{n+1} - 2T_{i,j}^{n+1} + T_{i-1,j}^{n+1}}{\Delta x^2} + \lambda \frac{T_{i,j+1}^{n+1} - 2T_{i,j}^{n+1} + T_{i,j-1}^{n+1}}{\Delta y^2}$$

consolidating unknown terms to the left-hand side yields

$$T_{i,j-1}^{n+1}(-\frac{\lambda \Delta t}{\Delta y^2}) + T_{i-1,j}^{n+1}(-\frac{\lambda \Delta t}{\Delta x^2}) + T_{i,j}^{n+1}(1 + 2\frac{\lambda \Delta t}{\Delta x^2} + 2\frac{\lambda \Delta t}{\Delta y^2}) + T_{i+1,j}^{n+1}(-\frac{\lambda \Delta t}{\Delta x^2})$$

$$+T_{i,j+1}^{n+1}(-\frac{\lambda \Delta t}{\Delta y^2}) = T_{i,j}^{n} + \Delta t f(t_{n+1}, x_i, y_j) - v_x \Delta t \frac{T_{i+1,j}^{n} - T_{i,j}^{n}}{\Delta x} - v_y \Delta t \frac{T_{i,j+1}^{n} - T_{i,j}^{n}}{\Delta y}$$

Similarly to when I solved the generalized heat equation before for the first portion of this final, this can be transformed into a system of linear equations. The system of linear equations, however, is not complete, because there are certain grid points at which the approximation cannot be calculated, because it is not contained within the grid. In this case, the so-called ghost point must be calculated. This can be done by implementing the Robin boundary condition:

$$\lambda \frac{\partial T}{\partial y}(t_{n+1}, x_i, y_1) - v_y T(t_{n+1}, x_i, y_1) = g(t_{n+1}, x_i, y_1)$$

Now we can use the central finite difference formula to approximate the spatial derivative

$$\frac{\partial T}{\partial y}(t_{n+1}, x_i, y_1) = \frac{T_{i,2}^{n+1} - T_{i,0}^{n+1}}{2\Delta y} + O(\Delta y^2)$$

Using this approximation in the formula for the Robin boundary conditions, we get:

$$\lambda \frac{T_{i,2}^{n+1} - T_{i,0}^{n+1}}{2\Delta y} - v_y T_{i,1}^{n+1} = g(t_{n+1}, x_i, y_1)$$

and therefore

$$T_{i,0}^{n+1} = T_{i,2}^{n+1} - \frac{2v_y \Delta y}{\lambda} T_{i,1}^{n+1} - \frac{2\Delta y}{\lambda} g(t_{n+1}, x_i, y_1)$$

substituting this estimation for $T_{i,0}^{n+1}$ and then rearranging terms so that unknowns are on the left provides us with one final equation:

$$T_{i-1,1}^{n+1}(-\frac{\lambda \Delta t}{\Delta x^2}) + T_{i,1}^{n+1}(1 + 2\frac{\lambda \Delta t}{\Delta x^2} + 2\frac{\lambda \Delta t}{\Delta y^2} + \frac{2v_y \Delta t}{\Delta y}) + T_{i+1,1}^{n+1}(-\frac{\lambda \Delta t}{\Delta x^2}) + T_{i,2}^{n+1}(-2\frac{\lambda \Delta t}{\Delta y^2})$$

$$= T_{i,j}^n + \Delta t f(t_{n+1}, x_i, y_j) - v_x \Delta t \frac{T_{i+1,1}^n - T_{i,1}^n}{\Delta x} - v_y \Delta t \frac{T_{i,2}^n - T_{i,1}^n}{\Delta y} - \frac{2\Delta t}{\Delta y} g(t_{n+1}, x_i, y_1)$$

Now there is a valid approximation of the advection-diffusion equation from all boundary points of the grid. It can now be solved as a linear system in standard form in the same manner as the heat equation for the first part of the final except that boundary points on the bottom boundary will now be solved using the results from the Robin boundary condition. Thus, there will be three different calculations for determining the values in matrix A and vector $\vec{r}$. One for internal points, one for boundary points that are not on the bottom, and one for boundary points on the bottom.

$$A \bullet \vec{T}^{n+1} = \vec{r}$$

where the vector of unknowns $\vec{T}^{n+1}$ is defined as

$$\vec{T}^{n+1} = \begin{bmatrix} s_1 \\ s_2 \\ s_3 \\ . \\ . \\ . \\ s_{N_x N_y} \end{bmatrix} = \begin{bmatrix} T_{1,1}^{n+1} \\ T_{2,1}^{n+1} \\ T_{3,1}^{n+1} \\ . \\ . \\ . \\ T_{N_x N_y}^{n+1} \end{bmatrix}$$

3.2 Implementing Both Explicit and Implicit Schemes

Using this exact method I solved the example in part b:

$$\text{Domain:} \quad \Omega = [-1;3]x[-1.5;1.5],$$

$$\text{Thermal Diffusivity:} \quad \lambda = 0.7,$$

$$\text{Velocity field:} \quad v_x = -0.8,$$

$$v_y = -0.4,$$

$$\text{Exact Solution:} \quad T_{exact} = sin(x)cos(y)exp(-t),$$

where the initial conditions, boundary conditions, and source term were calculated from the given exact solution. Solving from $t_{start} = 0$ to $t_{final} = 1$ for grid resolutions $(N_x, N_y = (25, 15), (40, 30)$, and $(160, 120)$ with a time-step $\Delta t = 0.5\Delta x$, produced the results in the following table.

Resolution	Error	Order
20x15	2.4041e-02	0
40x30	1.1986e-02	1.00424
80x60	0.5985e-02	1.00193
160x120	0.2991e-02	1.00037

Table 2: This table shows the error at each grid size and the associated order of accuracy

Looking at the results gathered by explicitly approximating the advection term at the moment of time t_n and implicitly approximating the diffusion term at time t_{n+1}, it is clear that the order of accuracy for this method is of order 1.

3.3 Simulating the Oil Spill

Finally, the method that involved explicitly approximating the advection term and implicitly approximating the diffusion term was used to simulate an oil spill where the parameters were:

$$\text{Domain:} \quad \Omega = [0;12]x[0;3],$$

$$\text{Diffusivity:} \quad \lambda = 0.2,$$

$$\text{Velocity field:} \quad v_x = -0.8,$$

$$v_y = -0.4,$$

$$\text{Exact Solution:} \quad T_{exact} = sin(x)cos(y)exp(-t),$$

$$\text{Initial Conidtions:} \quad T_{start}(x,y) = 0,$$

$$\text{Boundary Conidtions:} \quad T_{bc}(t,x,y) = 0,$$

$$g(t,x,y) = 0,$$

$$\text{Source term:} \quad f(t,x,y) = \frac{1}{2}(1 - tanh(\frac{\sqrt{((x-x_s)^2+y^2)-r_s}}{\epsilon})), \quad \text{if } t < 0.5,$$
$$0, \text{if} \quad t > 0.5$$

where $x_s = 10$, $r_s = 0.1$, $\epsilon = 0.1$, $t_{start} = 0$s and $t_{final} = 10$ using $N_x = 160$, $N_y = 40$ and $\Delta t = 0.1$.

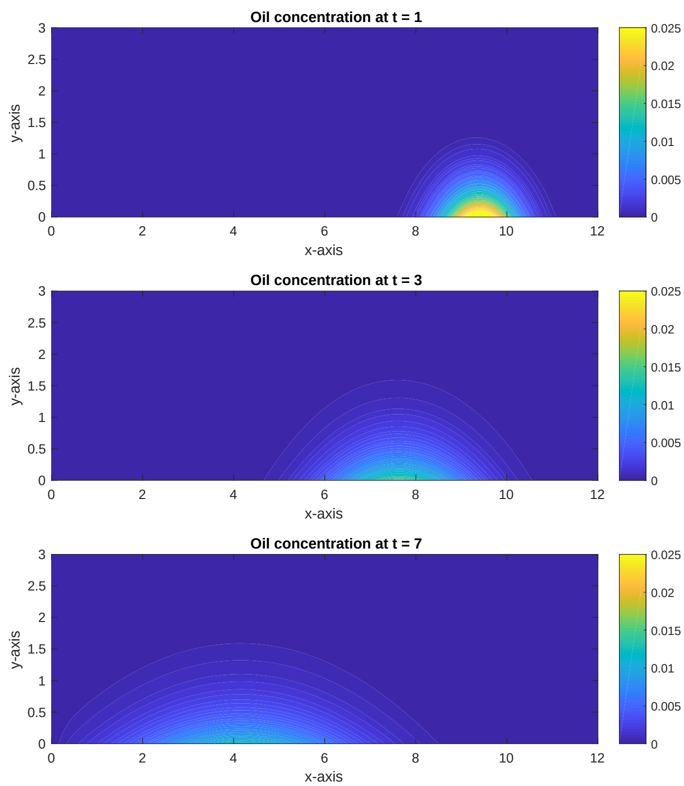

Figure 2: Snapshots of Oil Diffusion

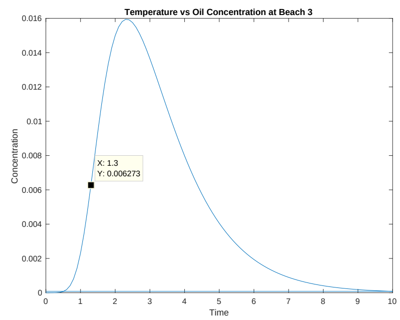

Figure 3: Beach 3 must be closed at around t = 1.3

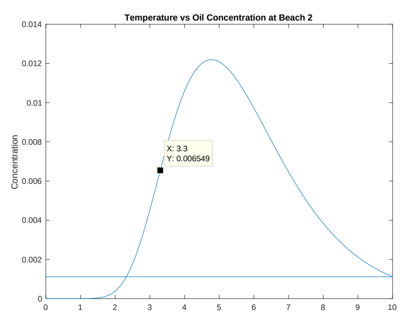

Figure 4: Beach 2 must be closed at around t = 3.3

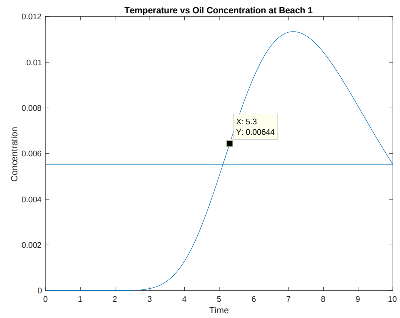

Figure 5: Beach 1 must be closed at around t = 5.3

4 Conclusion

Through numerical analysis both of the problems on this final were solved. A potato as described by the parameters in this final will cook in approximately twenty minutes, and all three beaches will have to be closed by approximately t = 5.3. Both of these problems were solved using a numerical method with accuracy of order 1. Using numerical methods such as these makes it much easier to solve difficult problems by finding a close approximation.