KTH Royal Institute of Technology

SF2520

Applied Numerical Methods

Computer Exercise 2

David Ahnlund
20000531-4778
dahnlund@kth.se

Emil Gestsson
20021030-3756
gestsson@kth.se

October 10, 2023

Part 1

In Part 1 of the computer exercise, we are solving the following convection-diffusion problem with finite differences:

$$-\frac{d^2T}{dz^2} + v\frac{dT}{dz} = Q(z), \text{ for } 0 < z < 1$$

Where T is the temperature in the cylinder described in the exercise description, z is the position along the cylinder, and v is the velocity of a fluid flowing across the cylinder.

The so-called driving function $Q(z)$, modeling the heat source, is defined as

$$Q(z) = Q_0 \sin\left(\frac{(z-a)\pi}{b-a}\right) \mathbb{1}_{[a,b]}(z)$$

The boundary conditions to the problem is given by:

$$T(0) = T_0$$

$$-\frac{dT(1)}{dz} = \alpha(v)(T(1) - T_{out}), \; \alpha(v) = \sqrt{\frac{v^2}{4} + \alpha_0^2} - \frac{v}{2}$$

Moreover, the constants are as follows:

$$a = 0.1; \;\; b = 0.4; \;\; Q_0 = 7000; \;\; T_{out} = 25; \;\; T_0 = 100$$

a)

For part 1a, v is set to equal 1, and the problem above was solved using finite differences:

$$\frac{d^2T}{dz^2} = \frac{T_{j+1} - 2T_j + T_{j-1}}{h^2} + \mathcal{O}(h^2)$$

$$\frac{dT}{dz} = \frac{T_{j+1} - T_{j-1}}{2h} + \mathcal{O}(h^2)$$

For the boundary $T(1)$ however, the skewed first derivative approximation was used:

$$\frac{dT(1)}{dz} = \frac{3T_N - 4T_{N-1} + T_{N-2}}{2h} + \mathcal{O}(h^2)$$

To construct the linear system to be solved $A\vec{T} = \vec{f}$, A was derived by inserting the finite differences into the PDE, ignoring the $\mathcal{O}(h^2)$:

$$-\frac{T_{j+1} - 2T_j + T_{j-1}}{h^2} + v\frac{T_{j+1} - T_{j-1}}{2h} = Q_0 \sin\left(\frac{(z-a)\pi}{b-a}\right) \mathbb{1}_{[a,b]}(z)$$

$$\Rightarrow \left(-\frac{1}{h^2} - \frac{v}{2h}\right) T_{j-1} + \frac{2}{h^2}T_j + \left(\frac{v}{2h} - \frac{1}{h^2}\right) T_{j+1} = Q_0 \sin\left(\frac{(z-a)\pi}{b-a}\right) \mathbb{1}_{[a,b]}(z)$$

In the case where $z = 0$ we have the following equation:

$$\frac{2}{h^2}T_1 + \left(\frac{v}{2h} - \frac{1}{h^2}\right) T_2 = Q_0 \sin\left(\frac{(z-a)\pi}{b-a}\right) \mathbb{1}_{[a,b]}(z) - \left(-\frac{1}{h^2} - \frac{v}{2h}\right) T_0$$

And for $z = 1$ we have the following equation system:

$$-\frac{3T_N - 4T_{N-1} + T_{N-2}}{2h} = \alpha(v)(T_N - T_{out})$$

$$\Rightarrow (1): \;\; T_N = \frac{1}{\frac{3}{2h} + \alpha(v)} \left(\frac{2T_{N-1}}{h} + \alpha(v)T_{out} - \frac{T_{N-2}}{2h}\right)$$

$$(2): \;\; \left(-\frac{1}{h^2} - \frac{v}{2h}\right) T_{N-2} + \frac{2}{h^2}T_{N-1} + \left(\frac{v}{2h} - \frac{1}{h^2}\right) T_N = Q_0 \sin\left(\frac{(z-a)\pi}{b-a}\right) \mathbb{1}_{[a,b]}(z)$$

Analytically solving the above system of equations, the following equation is obtained, which then can be inserted as the last row of the A-matrix:

$$(-\frac{1}{h^2} - \frac{v}{2h} - \frac{\theta}{2h})T_{N-2} + (\frac{2}{h^2} + \frac{2}{h}\theta)T_{N-1} = Q_0 \sin\left(\frac{(z-a)\pi}{b-a}\right) \mathbb{1}_{[a,b]}(z) - \theta\alpha(v)T_{out}$$

where

$$\theta = \frac{\frac{v}{2h} - \frac{1}{h^2}}{\frac{3}{2h} + \alpha(v)}$$

The matrix A finally is defined as:

$$A = \begin{bmatrix} \frac{2}{h^2} & \left(\frac{v}{2h} - \frac{1}{h^2}\right) & 0 & \cdots & & 0 \\ \left(-\frac{1}{h^2} - \frac{v}{2h}\right) & \frac{2}{h^2} & \left(\frac{v}{2h} - \frac{1}{h^2}\right) & 0 & \cdots & \vdots \\ 0 & \left(-\frac{1}{h^2} - \frac{v}{2h}\right) & \frac{2}{h^2} & \left(\frac{v}{2h} - \frac{1}{h^2}\right) & 0 & \\ \vdots & \ddots & \ddots & \ddots & & \\ & & \ddots & \ddots & \ddots & \\ & & & & & \\ & & & & & \vdots \\ \vdots & & & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & \left(-\frac{1}{h^2} - \frac{v}{2h} - \frac{\theta}{2h}\right) & \left(\frac{2}{h^2} + \frac{2}{h}\theta\right) \end{bmatrix}$$

and the vector $\vec{f}$ as

$$\vec{f} = \begin{bmatrix} Q_0 \sin\left(\frac{(h-a)\pi}{b-a}\right) \mathbb{1}_{[a,b]}(h) - \left(-\frac{1}{h^2} - \frac{v}{2h}\right) T_0 \\ Q_0 \sin\left(\frac{(2\cdot h)-a)\pi}{b-a}\right) \mathbb{1}_{[a,b]}(2\cdot h) \\ \vdots \\ \vdots \\ \vdots \\ Q_0 \sin\left(\frac{((N-2)\cdot h-a)\pi}{b-a}\right) \mathbb{1}_{[a,b]}((N-2)\cdot h) \\ Q_0 \sin\left(\frac{((N-1)\cdot h-a)\pi}{b-a}\right) \mathbb{1}_{[a,b]}((N-1)\cdot h) - \theta\alpha(v)T_{out} \end{bmatrix}$$

Now, by the use of Matlab's linear solver, the following results were obtained (Figure 1) :

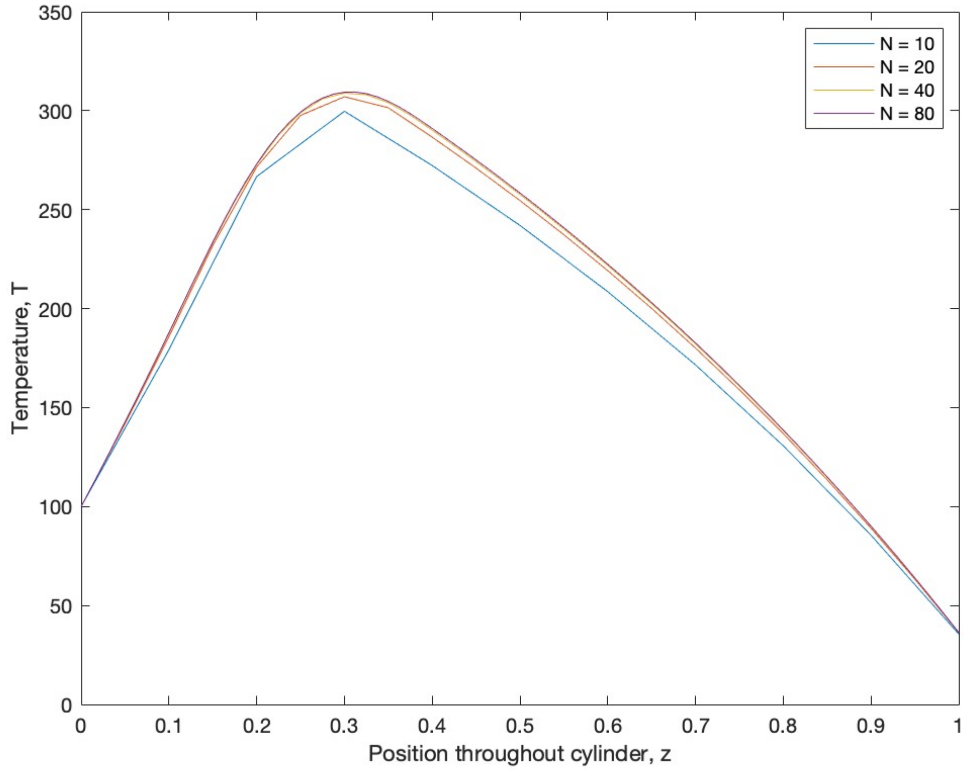

Figure 1: Plot of cylinder temperature for different Ns

When evaluating T at $z = 0.5$ for different values of N we got the following results:

$$T(z = 0.5) = 258.430, \text{ for } N = 80$$
$$T(z = 0.5) = 258.622, \text{ for } N = 160$$
$$T(z = 0.5) = 258.669, \text{ for } N = 320$$

The order of accuracy was confirmed by computing $E(N) = \text{diff}\left(\log_2 |T_N(z) - T_{2N}(z)|\right)$ for the different values of N, which gave the following values:

$(N, 2N)$	$(80, 160)$	$(160, 320)$	$(320, 640)$	$(640, 1280)$	$(1280, 2560)$	$(2560, 5120)$
$E(N), z = 0.5$	2.00177	2.00044	2.00011	2.00003	2.00001	1.99999
$E(N), z = 1$	2.00080	2.00002	1.99992	1.99994	1.99996	1.99997

b)

When performing the same numerical scheme as in Part 1a, but now instead with a v in the set $\{1, 5, 15, 100\}$ and $N = 10000$ the results in Figure 2 were obtained.

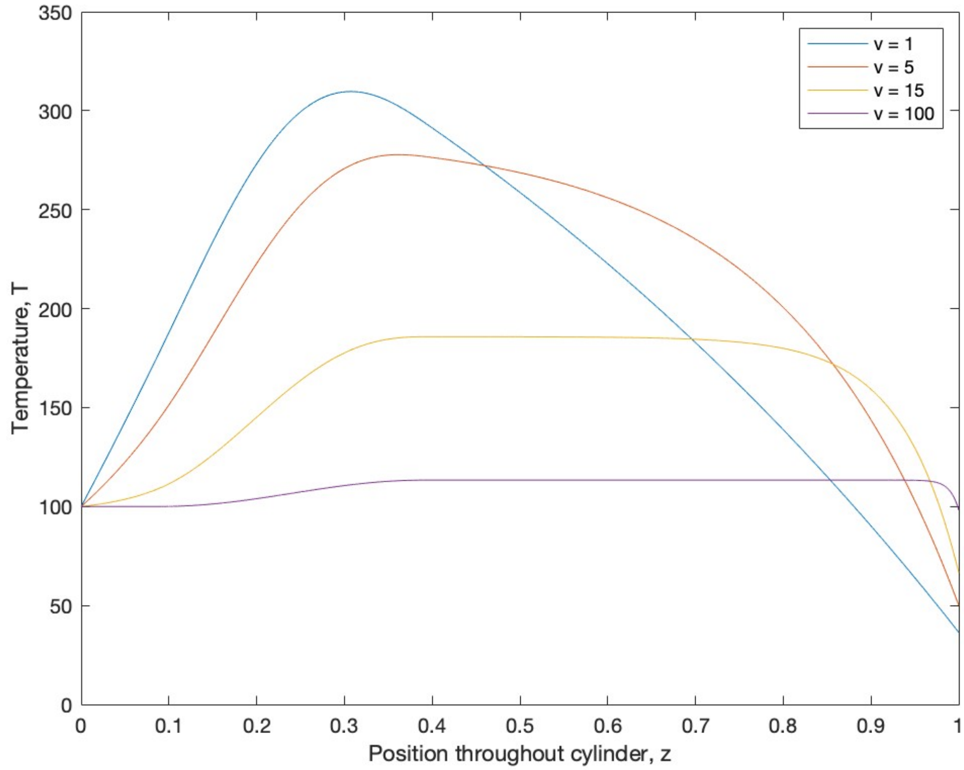

Figure 2: Plot of cylinder temperature for different v's

When evaluating T at $z = 0.5$ for the different values of v we got the following results:

$$T(z = 0.5) = 258.685, \text{ for } v = 1$$
$$T(z = 0.5) = 268.667, \text{ for } v = 5$$
$$T(z = 0.5) = 185.766, \text{ for } v = 15$$
$$T(z = 0.5) = 113.369, \text{ for } v = 100$$

As seen in Figure 2, the higher velocity we have for the fluid inside the cylinder, the less the heat source seems to affect the overall temperature, as we almost have a constant temperature of $T = T_0$ inside the cylinder for larger values of v. This seems intuitively right because the heat source has less time to increase the temperature inside the cylinder, as the fluid more quickly travels away from the heat source. At the same time, the outside temperature also has less time to cool the temperature further in the cylinder (i.e. values of $z < 1$) by the same principle. New fluid (which has been exposed to the heat source) is flowing towards the point where $z = 1$ at a faster rate if v is large, so as v increases the solution will converge to a constant temperature of T_0 across the z-axis.

Part 2

In Part 2 we were asked to solve the following elliptic problem:

$$-\Delta T = f, \qquad (x, y) \in \Omega \tag{1}$$

with boundary conditions:

$$T(x, 0) = T_{ext}, \qquad 0 < x < L_x \tag{2}$$
$$\frac{\partial T}{\partial x}(0, y) = 0, \qquad 0 < y < L_y \tag{3}$$
$$\frac{\partial T}{\partial x}(L_x, y) = 0, \qquad 0 < y < L_y \tag{4}$$
$$\frac{\partial T}{\partial y}(x, L_x) = 0, \qquad 0 < x < L_x \tag{5}$$

As in Part 1, we use a finite difference scheme for the laplace operator like the following:

$$\frac{\partial^2 T}{\partial x^2}(x_j, y_i) = \frac{T(x_{j+1}, y_i) - 2T(x_j, y_i) + T(x_{j-1}, y_i)}{h^2} + \mathcal{O}(h^2) \tag{6}$$
$$\frac{\partial^2 T}{\partial y^2}(x_j, y_i) = \frac{T(x_j, y_{i+1}) - 2T(x_j, y_i) + T(x_j, y_{i-1})}{h^2} + \mathcal{O}(h^2) \tag{7}$$

Moreover, the Neumann boundary conditions were applied by the use of the skewed first derivative approximations, forwards and backwards respectively:

$$\frac{\partial T}{\partial x}(x_j, y_i) = \frac{-3T(x_j, y_i) + 4T(x_{j+1}, y_i) - T(x_{j+2}, y_i)}{2h} + \mathcal{O}(h^2)$$
$$\frac{\partial T}{\partial x}(x_j, y_i) = \frac{3T(x_j, y_i) - 4T(x_{j-1}, y_i) + T(x_{j-2}, y_i)}{2h} + \mathcal{O}(h^2)$$

To then solve the 2D problem, finite difference matrices S_x and S_y were constructed separately, to use the Kronecker product approached presented during the lectures. Since the PDE problem is defined as $-\Delta T = f$ we therefore get a sign switch in S_x and S_y compared to what is seen in Equation 6 and 7:

$$S_x^* = \frac{1}{h^2}\begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & -1 & 2 & \ddots & \\ & & \ddots & \ddots & \\ & & & & \end{bmatrix} \qquad S_y^* = \frac{1}{h^2}\begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & -1 & 2 & \ddots & \\ & & \ddots & \ddots & \\ & & & & \end{bmatrix}$$

Note that the boundary conditions are not applied to the matrices yet, hence S_x^* and S_y^*.

For the boundary conditions, Equation 2 doesn't contribute to any changes in neither S_x^* nor S_y^*, but we have to add the $\frac{T_{ext}}{h^2}$-term to the resulting F-matrix's elements $F(x, 0)$.

For Equation 3 however, it leads to the following system:

$$\begin{cases} \frac{-3T(x_0,y_i)+4T(x_1,y_i)-T(x_2,y_i)}{2h} = 0 \\ -\frac{T(x_2,y_i)-2T(x_1,y_i)+T(x_0,y_i)}{h^2} = f \end{cases} \Rightarrow \frac{2T(x_1, y_i) - 2T(x_2, y_i)}{3h^2} = f \tag{8}$$

Similarly, for Equation 4 we get:

$$\begin{cases} \frac{3T(x_N,y_i)-4T(x_{N-1},y_i)+T(x_{N-2},y_i)}{2h} = 0 \\ -\frac{T(x_N,y_i)-2T(x_{N-1},y_i)+T(x_{N-2},y_i)}{h^2} = f \end{cases} \Rightarrow \frac{2T(x_{N-1}, y_i) - 2T(x_{N-2}, y_i)}{3h^2} = f \tag{9}$$

And for Equation 5 we get:

$$\begin{cases} \frac{3T(x_i,y_N)-4T(x_i,y_{N-1})+T(x_i,y_{N-2})}{2h} = 0 \\ -\frac{T(x_i,y_N)-2T(x_i,y_{N-1})+T(x_i,y_{N-2})}{h^2} = f \end{cases} \Rightarrow \frac{2T(x_i,y_{N-1})-2T(x_i,y_{N-2})}{3h^2} = f \tag{10}$$

Using the expressions yielded in Equations 8-10, S_x and S_y are defined as the following:

$$S_x = \frac{1}{h^2}\begin{bmatrix} \frac{2}{3} & -\frac{2}{3} & & & \\ -1 & 2 & -1 & & \\ & -1 & 2 & \ddots & \\ & & \ddots & \ddots & \\ & & & -\frac{2}{3} & \frac{2}{3} \end{bmatrix} \qquad S_y = \frac{1}{h^2}\begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & -1 & 2 & \ddots & \\ & & \ddots & \ddots & \\ & & & -\frac{2}{3} & \frac{2}{3} \end{bmatrix} \tag{11}$$

Finally the large A-matrix used to solve the linear system $A\vec{T} = \vec{F}$, where $\vec{T}$ and $\vec{F}$ are the reshaped F and T matrices to vectors of length $(N-1)\cdot(M-1)$, using the Kronecker product by the following Matlab command:

$$A = \text{kron}(\text{speye}(\text{size}(S_y)), S_x) + \text{kron}(S_y, \text{speye}(\text{size}(S_x)))$$

or equivalently

$$A = I_{S_y} \otimes S_x + S_y \otimes I_{S_x} \tag{12}$$

a)

In Part 2a, we use all the conclusions from the derivations above to numerically solve the PDE with $f = 2,\ \forall(x,y) \in \Omega$ and the constants

$$L_x = 12; \qquad L_y = 5; \qquad T_{ext} = 25;$$

and $h = L_x/N = L_y/M = 0.2$, i.e. $N = 60$. Using Matlab's backslash command, the following results were obtained, together with the plot in Figure 3

$$T(6,2) = 41.000, \text{ for } N = 60$$

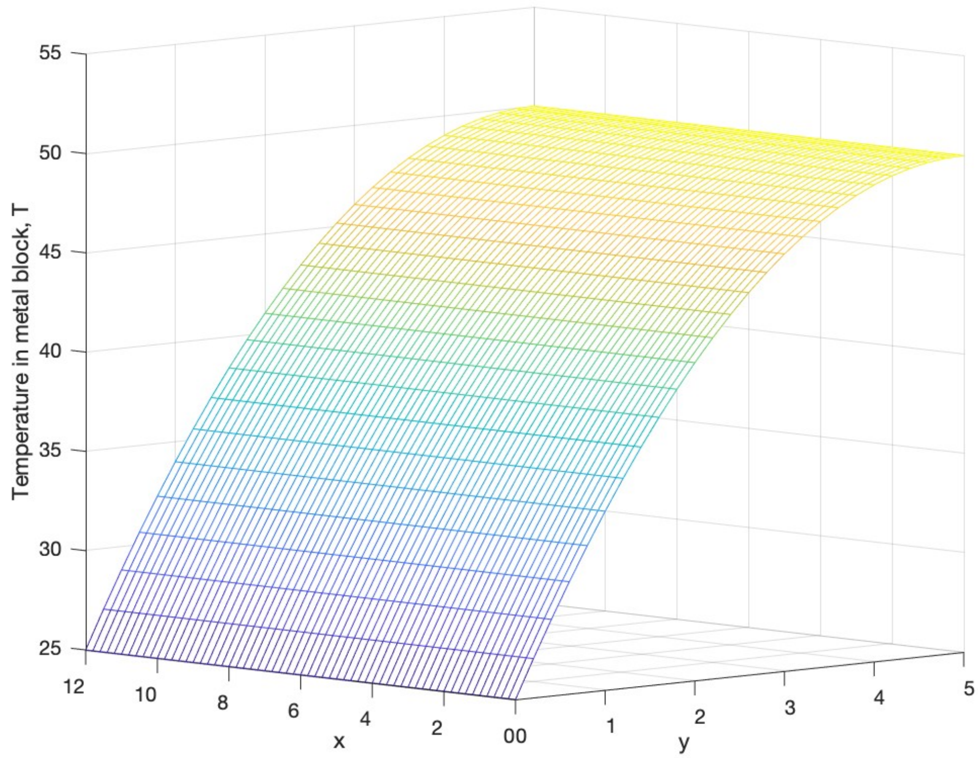

Figure 3: Plot of temperature in metal block, using finite differences

b)

Inserting $T(x, y) = c_0 + c_1 y + c_2 y^2$ into the PDE and boundary conditions yields $c_0 = T_{ext}$, $c_1 = fL_y$, $c_2 = -\frac{f}{2}$.

$$\begin{aligned} T(6,2) &= T_{ext} + fL_y \cdot 2 - \frac{f}{2} \cdot 2^2 \\ &= 25 + 2 \cdot 5 \cdot 2 - \frac{2}{2} \cdot 4 \\ &= 41 \end{aligned}$$

this equals the result in Part 2a.

c)

Expanding each g in

$$\frac{g(x+h) - 2g(x) + g(x-h)}{h^2}$$

with their corresponding Taylor expansions, every odd derivative term will cancel out, and thus

$$R(x) = 2 \sum_{n=2} \frac{h^{2n-2}}{(2n)!} g^{(2n)}(x)$$

when $g^{(2n)} = 0$ for $n \geq 2$, like in the analytical solution, since it is a 2nd degree polynomial, then $R(x) = 0$, which implies the approximate solution is equal to the analytical solution.

d)

In Part 2d, we seek to find the numerical solution to the PDE in Equation 1, but with another function for f. Here f is instead of 2, as in Part 2a, defined to be the following:

$$f(x, y) = 100 \exp(-\frac{1}{2}(x-4)^2 - 4(y-1)^2)$$

Using the same procedure as in Part 2a, the following results were obtained, together with the plot in Figure 4, as well as Figure 5 and Figure 6:

$$T(6, 2) = 47.220, \text{ for } N = 60$$
$$T(6, 2) = 47.224, \text{ for } N = 120$$
$$T(6, 2) = 47.225, \text{ for } N = 240$$

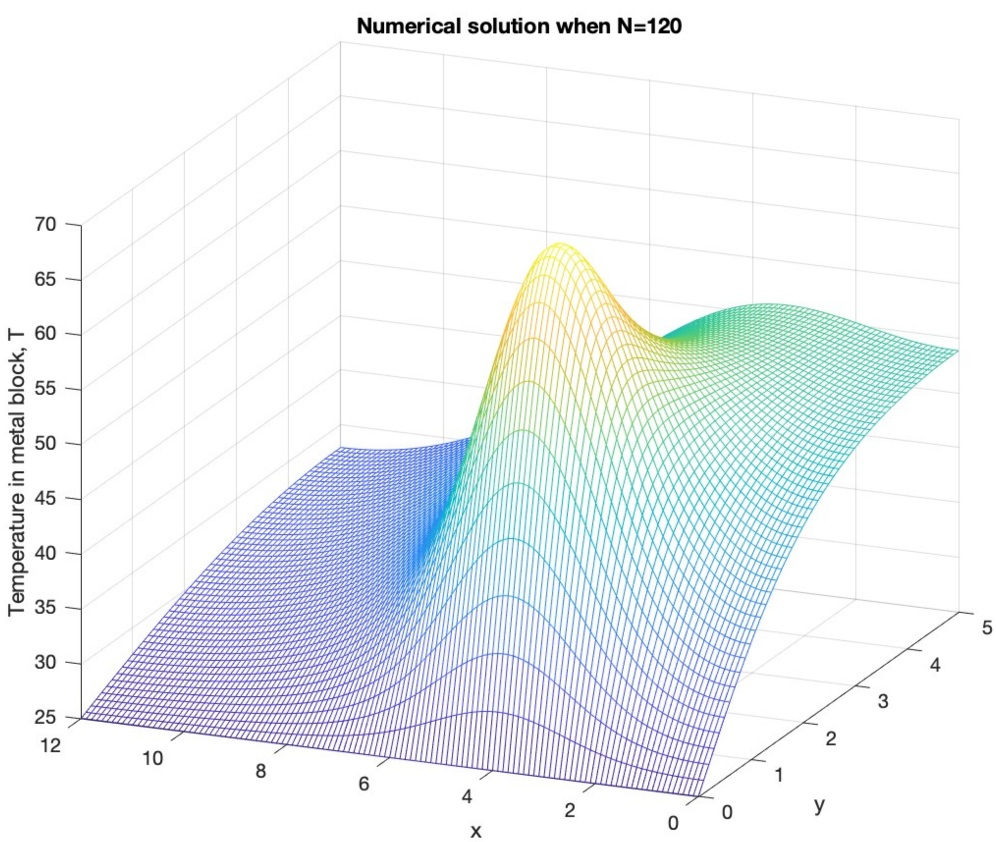

Figure 4: Plot of temperature in metal block, using finite differences, N = 120

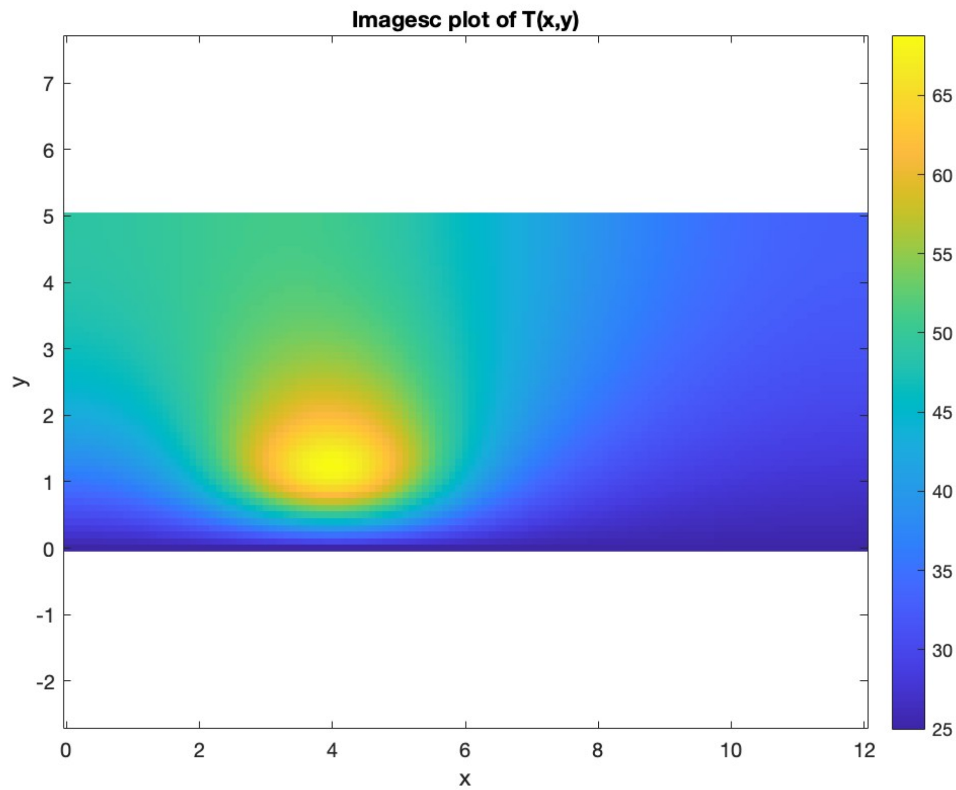

Figure 5: Plot of the numerical solution $T(x, y)$ using Matlabs "imagesc", N = 120

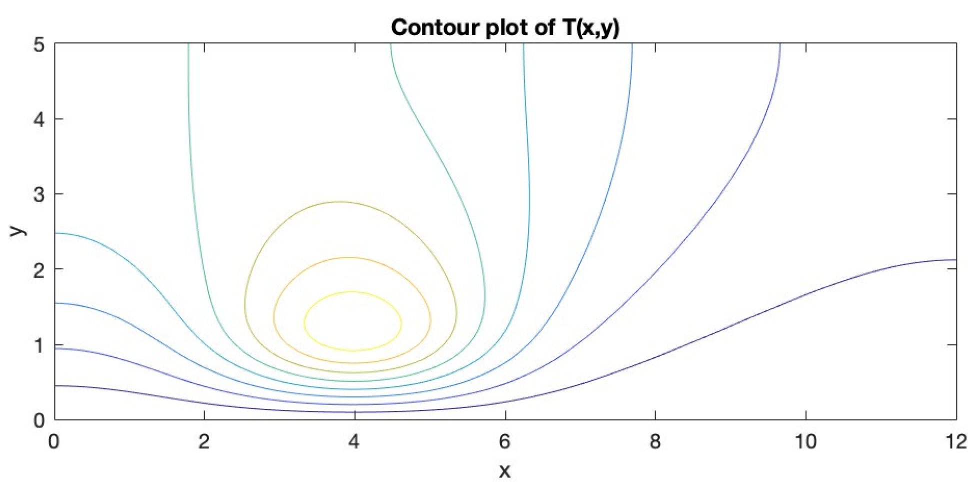

Figure 6: Plot of the numerical solution $T(x, y)$ using Matlabs "contour", N = 120

To verify that an accuracy convergence rate of order 2 is accomplished. We do, as in Part 1, a computation of diff $(\log_2|T_N(6,2) - T_{2N}(6,2)|)$. To get a good approximation of the convergence rate, we let N vary in the set $\{120, 240, 480, 960, 1920\}$. This gave the following values:

$(N, 2N)$	$(120, 240)$	$(240, 480)$	$(480, 960)$	$(960, 1920)$
Convergence rate at (6,2)	2.05979	2.02708	2.01285	2.00625

1 Part 3

In part 3 we used COMSOL to solve the PDE, and in particular we used the Poisson's Equation (poeq) physics interface, which solves the PDE equation

$$\nabla \cdot (-c\nabla u) = f$$

which corresponds to the desired PDE with the settings $c = 1$ and $f = 100\exp\left(-\frac{1}{2}(x-4)^2 - 4(y-1)^2\right)$.

a)

For a), we created a rectangle in accordance with the dirichlet condition $u = T_{ext}$ on the $y = 0$ line, and zero flux on the rest (since the normal derivative $= 0$). To measure the result, a point was added at (6, 2) to be used as a point probe. This ended up affecting the generated mesh, ensuring that (6, 2) was included as a point.

Here are a few of the probe values and triangle counts for different Mesh Settings.

Element Size	Triangles	Value
Normal	264	47.2241
Fine	390	47.2289
Finer	802	47.2247
Extra Fine	2640	47.2253
Extremely Fine	10484	47.2253

At Extremely Fine, we are certain that the first 3 digits are correct, and it also matches the result found in Part 2.

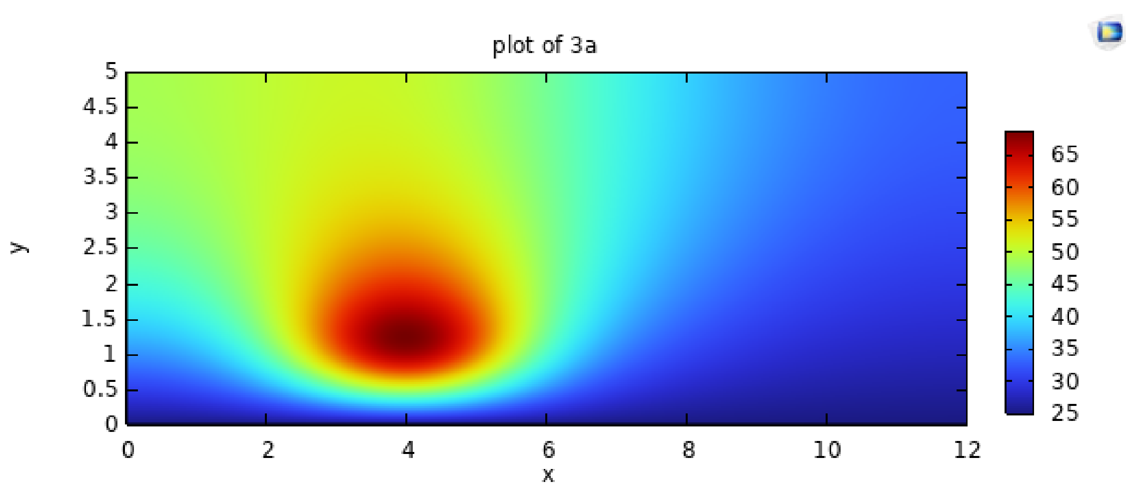

Figure 7: Plotted result from 3a

b)

The flux/source boundary condition is expressed as

$$-\mathbf{n} \cdot (-c\nabla u) = g - qu$$

observe that the left hand side is positive (since c = 1), so the right hand side of the robin condition must be multiplied by -1 to match. Therefore, $q = 0$ and $g = -\alpha(u - T_{ext})$ or $g = \alpha(T_{ext} - u)$.

This condition was then applied to all but the $y = 0$ line, which was set to zero flux. We got an average value of 193.9779 along the y = 5 by using it as a probe line.

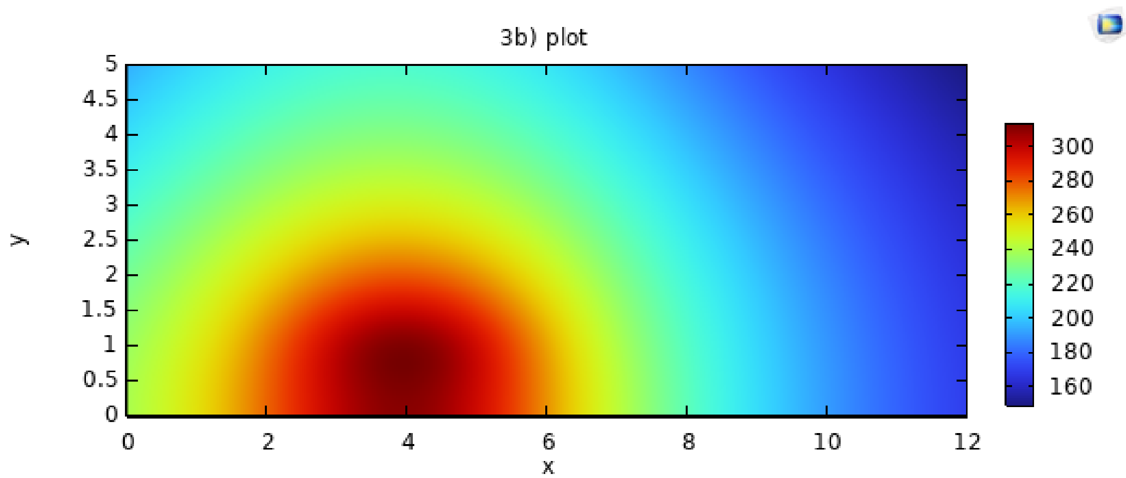

Figure 8: Plotted result from 3b

c)

To construct the geometry, a circle was added and the difference operation was used to cut it out of the rectangle. The average along the $y = 5$ line now became 147.8523, which means the hole reduced the average temperature by about 50.

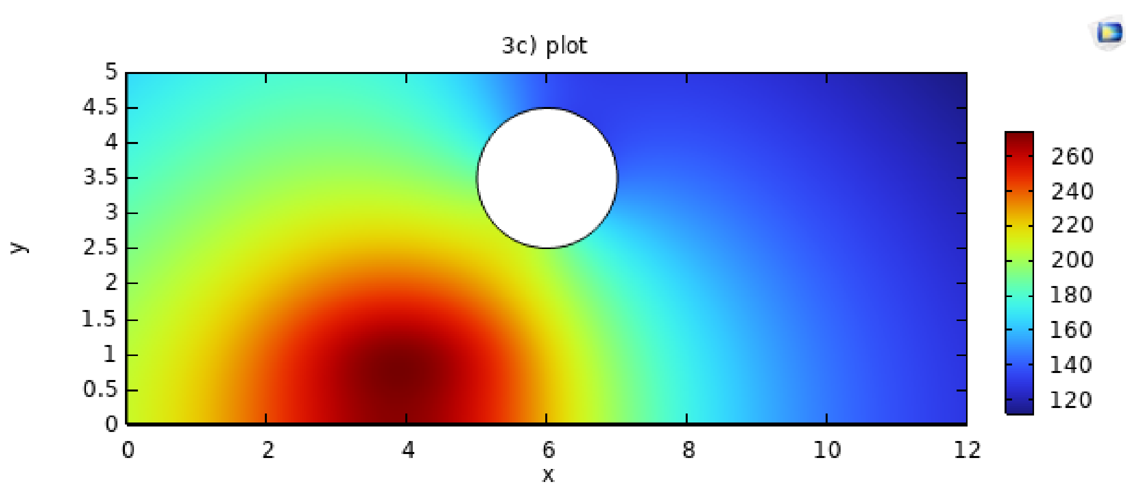

Figure 9: Plotted result with 1 hole

d)

With 4 holes the average became 134.0991.

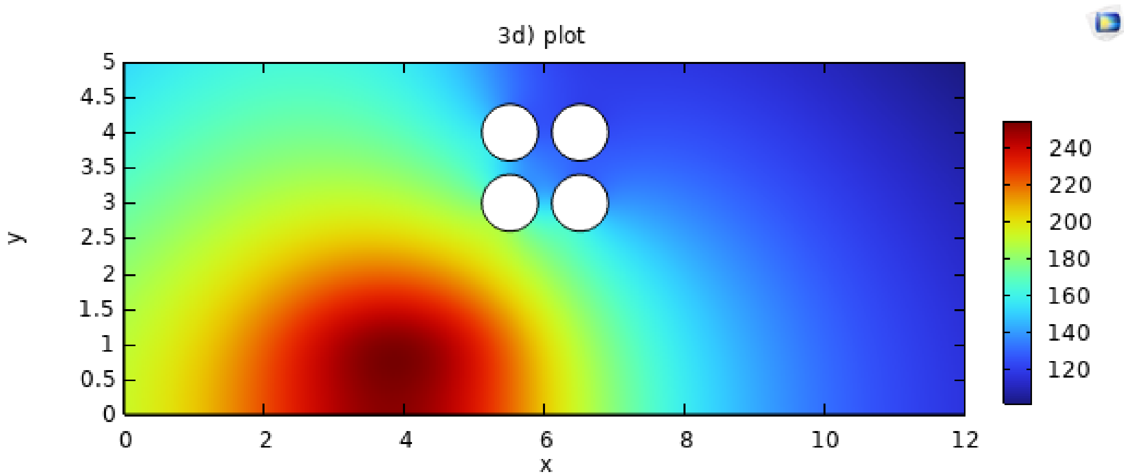

Figure 10: Plotted result with 4 holes.

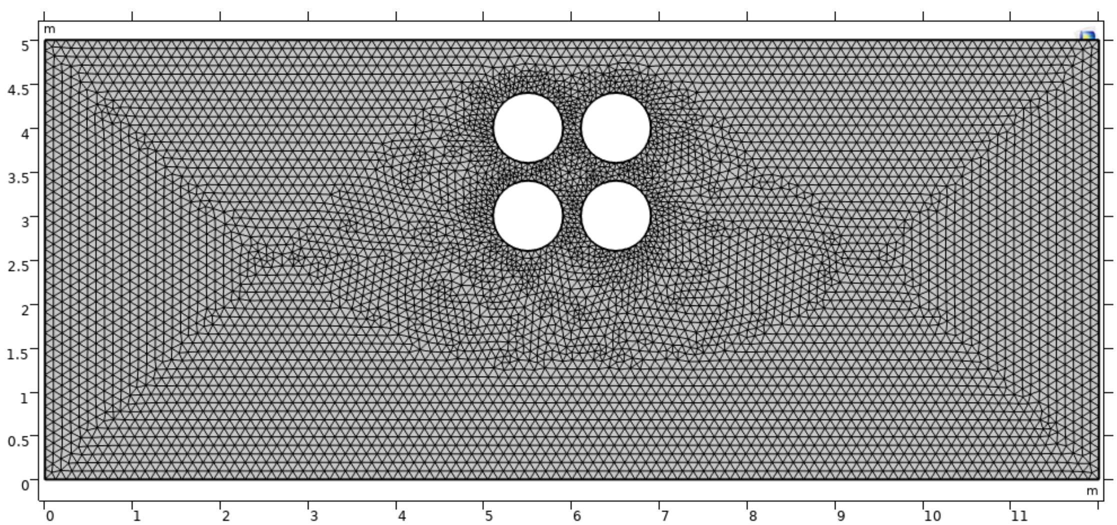

Figure 11: Resulting mesh when using the extremely fine setting

We see that the size of the triangles are substantially smaller near the circles, in this case its because more triangles are needed to adequately approximate the curved perimeter of circles.

e)

The idea to solve this was to fit lots of small holes in the allowed area, this is because a hole will get rid of the most area per circumference.

To implement this, we created a circle with radius r in the corner of the allowed boundary $(4.5+r, 2.5+r)$, and repeated them in an $N_x \times N_y$ array such that the distances $d_x, d_y > 4r$ satisfy

$N_x d_x \leq 3 - 2r$ and $N_y d_y \leq 2 - 2r$, which is required for the circles to stay in within the allowed rectangle. By checking the "Resulting objects selection" it was easy to apply both the difference operation and boundary conditions to the holes generated by the array.

Arbitrarily choosing $Nx = 10$, $Ny = 6$, $r = 0.1$, and solving for dx, dy; we get an average temperature of 85.3020, which is less than 100.

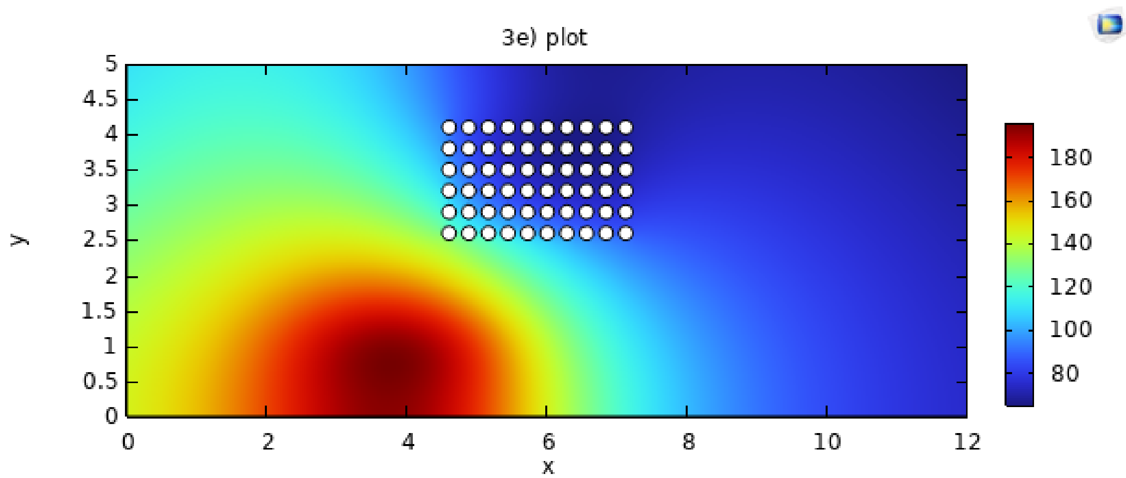

Figure 12: Plotted result with 60 holes.

Since these are 1mm holes, that are generously spaced, and there exists 1mm drills (even thread tap for extra surface area), it should be feasible to manufacture in real life with a sturdy and reasonably thick metal. To increase the perimeter/area ratio we could use optimal packing of the circles, and have their radius approach some very small ϵ where the laws of analysis still work as intended.