



KTH Engineering Sciences

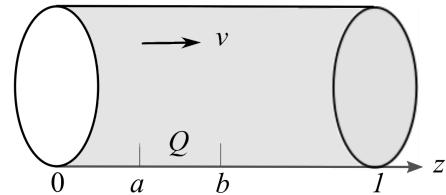
## Computer Exercise 2

### Elliptic equations

In this exercise you will solve elliptic PDEs in one and two dimensions. In the first two parts you will use the finite difference method on a simple geometry (an interval and a rectangle). In the third part you will use Comsol Multiphysics to find solutions in more complicated geometries.

#### Part 1: Finite difference approximation in 1D

Consider a short cylinder with a small cross section. In the cylinder there is a slowly moving fluid with high thermal diffusivity. The fluid is heated in a small section and the heat convects with the fluid along the cylinder. In suitable units for  $z$  and the fluid velocity  $v$ , the steady state temperature distribution  $T(z)$  is then determined by the convection–diffusion equation



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

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

$$Q(z) = \begin{cases} 0, & 0 \leq z < a, \\ Q_0 \sin\left(\frac{(z-a)\pi}{b-a}\right), & a \leq z \leq b, \\ 0, & b < z \leq 1. \end{cases}$$

At  $z = 0$  the fluid has the inlet temperature  $T_0$ ,

$$T(0) = T_0.$$

Beyond  $z = 1$  the pipe is poorly insulated and the liquid is cooled down since heat is leaking out to the exterior, which has the temperature  $T_{\text{out}}$ . This assumption can be modeled by the following boundary condition at  $z = 1$ :

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

where  $\alpha_0$  is the heat transfer coefficient for the non-convective ( $v = 0$ ) case. Use the following values of the parameters:  $a = 0.1$ ,  $b = 0.4$ ,  $Q_0 = 7000$ ,  $\alpha_0 = 50$ ,  $T_{\text{out}} = 25$  and  $T_0 = 100$ .

- (a) Consider first the case  $v = 1$ . Solve the boundary value problem with the finite difference method using MATLAB. Discretize the  $z$ -interval  $[0, 1]$  with grid points  $z_j = jh$  and  $h = 1/N$ . Hence,  $z_0 = 0$  and  $z_N = 1$ , so that  $N - 1$  is the number of inner grid points in the interval, which may or may not be the same as the number of unknowns in the finite difference method, depending on how you implement the boundary conditions. Use at least second order accurate approximations for the differential equation and the BCs.
- Plot the solutions  $T(z)$  you get with  $N = 10, 20, 40$  and  $80$  in the same graph.
  - Report the temperature values computed at  $z = 0.5$  for  $N = 80, 160, 320$  (three  $T$ -values).
- (b) Now solve the problem for  $v = 1, 5, 15, 100$ , and plot the solutions in the same graph. Interpret the curves based on the physical problem they model. Why do they look like they do? Are the results as expected? Note: For higher  $v$  you will need to take a smaller  $h$  (larger  $N$ ) to get the correct behavior of the solution close to  $z = 1$ . Report which  $h$  you take for the different  $v$ .

OBS! For full credit your implementation should exhibit second order accuracy when the grid is refined as in (a). You may need to take even larger  $N$  to verify that it is second order.

*Hints:* If you have used a second order stencil and second order approximation of the boundary conditions but still only observe a first order convergence rate it is quite likely that you have made a small mistake somewhere. (This is very easy to make!) Here are two common minor mistakes that will turn a second order method into a first order method:

- Your implementation of the number of unknowns is not compatible with your  $N$  and step length  $h$ . You may have used  $N$  unknowns when your implementation of the boundary conditions is based on using  $N - 1$  unknowns, for instance.
- You may evaluate the source function  $Q$  in slightly the wrong points. You might use  $Q(z_j - h)$  instead of  $Q(z_j)$  or something similar.

Additionally, even if you have a second order method, you may observe first order convergence if you do not read off the  $T$  value in exactly the same point for the different grids. Instead of  $z = 0.5$  you might be looking at  $z = 0.5 - h/2$  for instance.

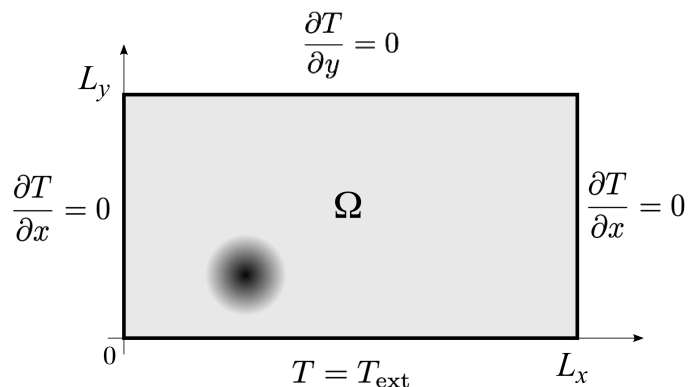
## Part 2: Finite difference approximation in 2D

In this part we consider heat conducted through a 2D rectangular metal block occupying the region  $\Omega = [0 < x < L_x, 0 < y < L_y]$  in the  $xy$ -plane. At  $y = 0$  the block is kept at the same temperature as the surrounding air  $T = T_{\text{ext}}$ . It is insulated at the other three sides. An external source modeled by the function  $f(x, y)$  heats the block. The following elliptic problem for the temperature distribution  $T(x, y)$  can then be formulated:

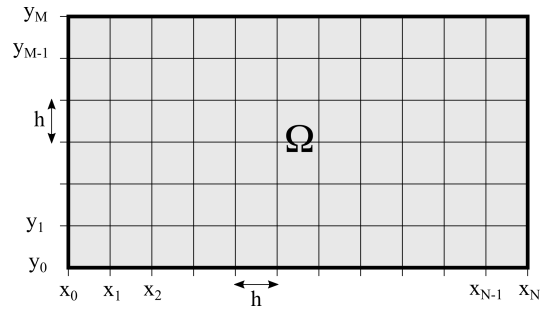
$$-\Delta T = f, \quad (x, y) \in \Omega,$$

with boundary conditions

$$\begin{aligned} T(x, 0) &= T_{\text{ext}}, & 0 < x < L_x, \\ \frac{\partial T}{\partial x}(0, y) &= 0, & 0 < y < L_y, \\ \frac{\partial T}{\partial x}(L_x, y) &= 0, & 0 < y < L_y, \\ \frac{\partial T}{\partial y}(x, L_y) &= 0, & 0 < x < L_x. \end{aligned}$$



Use the finite difference method to solve this problem with the parameter values  $L_x = 12$ ,  $L_y = 5$  and  $T_{\text{ext}} = 25$ . The approximation should be of at least order two. Discretize the rectangular domain into a quadratic mesh with the same, uniform, stepsize  $h = L_x/N = L_y/M$  in the  $x$ - and  $y$ -directions<sup>1</sup>. Similar to Part 1,  $M - 1$  and  $N - 1$  are the number of inner grid points in each direction, which may or may not be the same as the number of unknowns in the finite difference method, depending on how you implement the boundary conditions. Solve the resulting linear system with backslash in MATLAB.



- Compute the solution  $T(x, y)$  with  $f \equiv 2$  and  $h = 0.2$  ( $N = 60$ ). (Be careful with the sign here! Do not miss the minus sign in front of  $\Delta$ .) Visualize  $T(x, y)$  using the MATLAB function `mesh`. What is the computed  $T$ -value in the point  $(x, y) = (6, 2)$  inside the block?
- When  $f$  is constant, the exact solution to the PDE is a second order polynomial in  $y$ , i.e. of the form  $T(x, y) = c_0 + c_1 y + c_2 y^2$ . Use the PDE and the boundary conditions to find the coefficients  $c_j$  for the solution in (a). (Note that the Neumann BC on the left and right sides of the block are satisfied for all choices of coefficients.)
- For the case in (a) the numerical method actually gives the *exact* solution of the PDE. First compare the solutions from (a) and (b) in the point  $(x, y) = (6, 2)$  to verify this. Then find an expression for the error term  $R(x)$  of the central difference approximation

$$\frac{g(x+h) - 2g(x) + g(x-h)}{h^2} = g''(x) + R(x),$$

and use it to explain why the numerical solution is exact!

- Solve the problem (numerically) with the following localized heat source

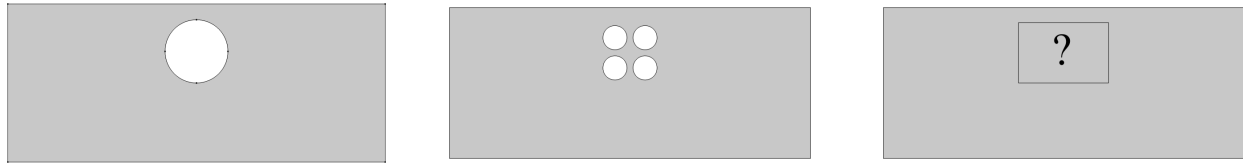
$$f(x, y) = 100 \exp \left( -\frac{1}{2}(x-4)^2 - 4(y-1)^2 \right).$$

Report the the three  $T$ -values obtained at  $(x, y) = (6, 2)$  when you use  $h = 0.2$  ( $N = 60$ ),  $h = 0.1$  ( $N = 120$ ) and  $h = 0.05$  ( $N = 240$ ). Here it is important to use MATLAB's `sparse` format for the matrices. Otherwise the problems with small  $h$  take very long time. Visualize the solution as above for  $h = 0.1$ . This time also plot it with the `imagesc` and `contour` commands. (You do not need to find the analytic solution for this case!)

OBS! For full credit your implementation should exhibit second order accuracy when the grid is refined. (You may want to check values at  $(6, 2)$  for additional  $h$  to verify your convergence rate.)

*Hint:* If you have problem getting second order accuracy, please look at the hints in Part 1 for the 1D case above. In 2D it is even easier to make small mistakes of the types mentioned there. Be particularly careful to read off the value in the right point,  $(x, y) = (6, 2)$  and not something like  $(x, y) = (6 + h/2, 2 - h/2)$ . Also make sure  $M = 5N/12$  is an integer, so that the step size is indeed the same in the  $x$ - and  $y$ -directions.

<sup>1</sup>This means that  $M = \frac{L_y}{L_x} N = 5N/12$  must be an integer.



**Figure 1.** Configuration in (c) with one hole (left), in (d) with four holes (middle) and your own design (right).

### Part 3: Comsol Multiphysics

- (a) (Warm-up.) Solve the problem in Part 2d with Comsol Multiphysics. Draw the geometry. Set the PDE coefficients. Specify boundary conditions. Generate the mesh. Compute the solution and plot it.

Check your numerical result: what is the  $T$ -value at the point<sup>2</sup>  $(6, 2)$ ? How many domain elements (triangles) have been generated in the mesh? Make a refinement of the grid and find again  $T(6, 2)$ . How many elements are there now? Continue to refine until you are sure that the value has 3 correct decimals. Compare with your own solutions above.

- (b) To make the problem more physically relevant, change the boundary conditions as follows: Let the bottom side  $y = 0$  be insulating, so that

$$\frac{\partial T}{\partial y}(x, 0) = 0, \quad 0 < x < L_x,$$

and use the Robin conditions

$$-\frac{\partial T}{\partial n} = \alpha(T - T_{\text{ext}}), \quad \alpha = 0.06, \quad T_{\text{ext}} = 25, \quad (1)$$

on the remaining three sides. (Here  $\partial/\partial n = \hat{n} \cdot \nabla$  is the normal derivative.) This models a block that is being cooled by convection (e.g. air flow) on those sides. The coefficient  $\alpha$  is the heat transfer coefficient and  $T_{\text{ext}}$  is the temperature of the surrounding air as before. Use the boundary condition **Flux/Source** to set this in Comsol.

Compute and plot the solution. Also report the average temperature on the top boundary  $y = 5$ . Use a Boundary Probe, which you find next to the Domain Point Probe used above.

- (c) To cool down the top boundary a hole is drilled in the block, where air can flow through. The hole is centered at  $(x, y) = (6.0, 3.5)$  and has radius 1.0. See left figure above. The cooling is modeled by the same boundary condition as in (1) on the edge of the hole.

Again, plot the solution and compute the average temperature on the top boundary  $y = 5$ . What is the temperature?

- (d) The cooling will be more efficient if the single hole is replaced by four smaller holes. See middle figure. Those holes have radius 0.4 and are centered in  $(x, y) = (6.0 \pm 0.5, 3.5 \pm 0.5)$ .

Solve the equation for this case and, as in (c), plot the solution and compute the average temperature on the top boundary  $y = 5$ . What is the new temperature?

Also plot the mesh for this case and comment on the triangle sizes: Why are they big in some parts and small in others?

*Tip:* To set the flux/source boundary condition on all those boundaries, it is easiest to first set it on *all* boundaries, selecting **All boundaries** in the menu, and then deselect the  $y = 0$  boundary.

<sup>2</sup>To read off the value at a point  $(x, y)$  in Comsol you can use a Domain Point Probe. Right click on **Definition** in the Model builder and select **Probes/DomainPointProbe** in the popup menu.

- (e) Finally, design a configuration of your own which reduces the temperature even more. The average temperature on the top boundary  $y = 5$  should be less than 100 degrees.

You are only allowed<sup>3</sup> to make changes to the geometry inside the rectangle  $[6.0 \pm 1.5, 3.5 \pm 1]$ . See right figure. The Comsol feature **Transforms/Array** (right click on Geometry) can be useful.

Plot the solution of your configuration and report the average boundary temperature.

Comment on the configuration, the result and its limitations, e.g.: How did you come up with it? How can it be further improved? Are the results physically relevant? Is it possible to manufacture? (You can e.g. assume that the length unit is centimeters.) Are there any numerical limitations for simulating it?

*Hint:* Keep in mind a key tenet in thermal physics, that the surface area to volume ratio is an important determinant of the cooling rate.

Feel free to improve the design further once you have satisfied the 100 degree requirement. You can for instance try to add cooling flanges on the sides of the block, like in the figure to the right. How cool can you make the block?



---

<sup>3</sup>It is even possible to come down below 50 degrees with this limitation, but it requires quite a lot of effort!