A finite difference method code for solving energy transfer problems

Norberto Marcelo Nigro [a,1] Mario Alberto Storti [a,2]

[a] *Centro Internacional de Métodos Computacionales en Ingeniería (CIMEC), INTEC-CONICET-UNL, Güemes 3450, 3000 Santa Fe, ARGENTINA*

Abstract

Key words: finite difference, energy transfer, advection-reaction-diffusion

---

1 INTRODUCTION

This note serves as a guide for using and extending a finite difference code used for the course titled *Energy Transfer* at INTEC

2 PROBLEM DEFINITION

The problem to be solved is defined as:

Find $\varphi(\mathbf{x}, \mathbf{t})$ *in* Ω *such that*

$$\rho C_p \frac{\partial \varphi}{\partial t} + \mathbf{v} \cdot \nabla \varphi + c\varphi = \nabla \cdot (\kappa \nabla \varphi) + \mathcal{G} \tag{1}$$

with φ normally the temperature, $\mathbf{v}$ the velocity vector normally associated with the *advection term*, c the reaction coefficient representing a linearized

[1] Corresponding author e-mail nnigro@intec.unl.edu.ar
[2] e-mail mstorti@intec.unl.edu.ar

16 November 2009

version of the heat released during chemical reactions, κ the conductivity, ρ the density and C_p the specific heat at constant pressure of the media and $\mathcal{G}$ a heat source term treated separately from the reaction term to consider sources that not depend on the temperature.

Ω represents the domain of definition of our problem, in our case we restrict only to cartesian coordinates on rectangular domains aligned with the cartesian axes, i.e. $\Omega = [x_0, x_1] \times [y_0, y_1]$, with (x_0, y_0) the lower left corner of the reactangular domain and (x_1, y_1) the upper right corner. The boundary of the domain Ω is represented by $\Gamma = \Gamma_\varphi \cup \Gamma_q \cup \Gamma_h$, being the boundary part where Dirichlet, Neumann or mixed (Robin) boundary conditions are applied respectively.

The boundary conditions are the following:

$$\begin{aligned} \varphi &= \overline{\varphi} && \forall \mathbf{x} \in \Gamma_\varphi \\ -\kappa \nabla \varphi \cdot \eta &= \overline{q} && \forall \mathbf{x} \in \Gamma_q \\ -\kappa \nabla \varphi \cdot \eta + h(\varphi - \varphi_\infty) &= 0 && \forall \mathbf{x} \in \Gamma_h \end{aligned} \tag{2}$$

The initial conditions are $\varphi(x, t = 0) = \overline{\varphi}^0$ from which the solution evolves in time.

Steady state solutions may be reached dropping the temporal term and removing the initial condition that in the linear case is irrelevant.

3 NUMERICAL APPROACH - DISCRETIZATION BY FINITE DIFFERENCES

The following statement represent the discrete version of 1 using finite differences.

$$\begin{aligned} \rho_{i,j}(C_p)_{i,j} \frac{\varphi^{n+1}_{i,j} - \varphi^n_{i,j}}{\Delta t} + (v_x)_{i,j} \frac{\varphi^{n+\theta}_{i+1,j} - \varphi^{n+\theta}_{i-1,j}}{x_{i+1,j} - x_{i-1,j}} + (v_y)_{i,j} \frac{\varphi^{n+\theta}_{i,j+1} - \varphi^{n+\theta}_{i,j-1}}{y_{i,j+1} - y_{i,j-1}} \\ + c_{i,j}\varphi^{n+\theta}_{i,j} = \frac{(q^x_{i+\frac{1}{2},j})^{n+\theta} - (q^x_{i-\frac{1}{2},j})^{n+\theta}}{x_{i+\frac{1}{2},j} - x_{i-\frac{1}{2},j}} + \frac{(q^y_{i,j+\frac{1}{2}})^{n+\theta} - (q^y_{i,j-\frac{1}{2}})^{n+\theta}}{y_{i,j+\frac{1}{2}} - y_{i,j-\frac{1}{2}}} + \mathcal{G}^{n+\theta}_{i,j} \end{aligned} \tag{3}$$

with $\varphi^{n+\theta}_{i,j} = \varphi(x = x_{i,j}, y = y_{i,j}, t = t^n + \theta\Delta t)$ the discrete approximation of the continuous variable φ at some point in the space $(x_{i,j}, y = y_{i,j})$ and at the time given by the last time $t = t^n$ added by a fraction $\theta \in [0, 1]$ of the time step Δt.

In the above equation (3) the heat flux is put inside, therefore two equations arise from it. To reduce the problem to only one unknown we assume a *constitutive law* that relates the heat flux with the energy, in this case with the temperature, in the following form:

$$
\begin{aligned}
q^x_{i+\frac{1}{2},j})^{n+\theta} &= (\kappa)_{i+\frac{1}{2},j} \frac{\varphi^{n+\theta}_{i+1,j} - \varphi^{n+\theta}_{i,j}}{x_{i+1,j} - x_{i,j}} \\
q^y_{i+\frac{1}{2},j})^{n+\theta} &= (\kappa)_{i,j+\frac{1}{2}} \frac{\varphi^{n+\theta}_{i,j+1} - \varphi^{n+\theta}_{i,j}}{y_{i,j+1} - y_{i,j}}
\end{aligned}
\tag{4}
$$

Taking $\theta = 0$ we adopt the *explicit* formulation, a very simple scheme where the new variable $\varphi^{n+\theta}_{i,j}$ may be calculated directly (without any matrix system) from the data and from the variable already known at older time steps, $\varphi^n_{i,j}$. If $\theta = 1$ the scheme is called *implicit* that needs the resolution of a matrix system and obviously is more expensive that the explicit one, but with an implicit treatment the numerical stability is larger than that with the explicit scheme. This last feature implies that the time step for the explcit schem to evolve in time is smaller than that of an implicit scheme. Finally to choose between them we need to analyze the relation between physical characteristics time steps with numerical one and if you need accuracy or not for the evolution in time.

Both, $\theta = 0, \theta = 1$ are first order in time, it means that the error in time grows as $\mathcal{O}(\Delta t)$ If we adopt $\theta = \frac{1}{2}$ the scheme changes to second order in time ($\mathcal{O}(\Delta t^2)$ and this strategy is normally chosen to enhance the time evolution of our unknown.

3.1 Explicit scheme

In this case the numerical equation to be solved is written as:

$$
\begin{aligned}
\frac{\rho_{i,j}(C_p)_{i,j}}{\Delta t} \varphi^{n+1}_{i,j} = {} & \frac{\rho_{i,j}(C_p)_{i,j}}{\Delta t} \varphi^n_{i,j} - (v_x)_{i,j} \frac{\varphi^n_{i+1,j} - \varphi^n_{i-1,j}}{x_{i+1,j} - x_{i-1,j}} - (v_y)_{i,j} \frac{\varphi^n_{i,j+1} - \varphi^n_{i,j-1}}{y_{i,j+1} - y_{i,j-1}} \\
& - c_{i,j}\varphi^n_{i,j} + \frac{(q^x_{i+\frac{1}{2},j})^n - (q^x_{i-\frac{1}{2},j})^n}{x_{i+\frac{1}{2},j} - x_{i-\frac{1}{2},j}} + \frac{(q^y_{i,j+\frac{1}{2}})^n - (q^y_{i,j-\frac{1}{2}})^n}{y_{i,j+\frac{1}{2}} - y_{i,j-\frac{1}{2}}} + \mathcal{G}^n_{i,j}
\end{aligned}
\tag{5}
$$

where the whole *righ hand size term* is known at the beginning of each time step and therefore making straighforward the computation of the variable updating $\varphi_{i,j}^{n+1}$.

3.2 Implicit scheme

The implicit case is much more difficult but however is in some situations much more attractive than the explicit scheme. The implicit scheme needs the computation of a matrix because in the original numerical equation 3 several terms depends on the unknown variable $\varphi_{i,j}^{n+1}$ and its neighbours $\varphi_{i+1,j}^{n+1}, \varphi_{i-1,j}^{n+1}, \varphi_{i,j+1}^{n+1}, \varphi_{i,j-1}^{n+1}$ for the simplest schemes.

Thinking in the nonlinear case for generality where normally one has to solve $\mathcal{F}(\varphi) = 0$, Newton strategy may be one possibility to ge the solution. Even the linear case may be included inside this problem doing the strategy enough general to consider several application problems. The strategy consists in computing an increment of the unknown solution (the roots of $\mathcal{F}(\varphi) = 0$) using a linearized version of the Taylor series representation of the function $\mathcal{F}$ in an iterative way, so, starting from a guess solution $\mathcal{F}(\varphi)^\nu \neq 0$ compute the increment $\Delta\varphi$ in the following form

$$
\begin{aligned}
\mathcal{F}(\varphi^{\nu+1}) &= \mathcal{F}(\varphi^\nu + \Delta\varphi) = 0 \\
\mathcal{F}(\varphi^{\nu+1}) = \mathcal{F}(\varphi^\nu + \Delta\varphi) &\approx \mathcal{F}(\varphi^\nu) + \frac{\partial \mathcal{F}}{\partial \varphi}|_{\varphi^\nu} \Delta\varphi = 0 \\
\Delta\varphi &= -\Big(\frac{\partial \mathcal{F}}{\partial \varphi}|_{\varphi^\nu}\Big)^{-1} \mathcal{F}(\varphi^\nu) \\
\varphi^{\nu+1} &= \varphi^\nu + \Delta\varphi
\end{aligned}
\tag{6}
$$

until convergence, normally $\|\mathcal{F}(\varphi^{\nu+1})\| < \epsilon$. In the above expression 6 one may distinguish the residual $\mathcal{R} = \mathcal{F}(\varphi^\nu)$ that is the non satisfaction of the original equation when it is evaluated with an already rough approximation φ^ν. Moreover, the updating requires the evaluation of a matrix, called *tangent matrix* ($\mathcal{K}$) in resemblance to the one dimensional non linear solution by Newton Raphson. That matrix is built from the derivative of each equation present in $\mathcal{F}$ respect to each variable present in the vector φ.

In our context the residual of the original equation taken with $\theta = 1$ for simplicity may be written as:

$$
\mathcal{R}(\varphi_{i,j}^{n+1,\nu}) = -\rho_{i,j}(C_p)_{i,j}(\frac{\varphi_{i,j}^{n+1,\nu} - \varphi_{i,j}^{n}}{\Delta t}) - (v_x)_{i,j}\frac{\varphi_{i+1,j}^{n+1,\nu} - \varphi_{i-1,j}^{n+1,\nu}}{x_{i+1,j} - x_{i-1,j}} - (v_y)_{i,j}\frac{\varphi_{i,j+1}^{n+1,\nu} - \varphi_{i,j-1}^{n+1,\nu}}{y_{i,j+1} - y_{i,j-1}}
-c_{i,j}\varphi_{i,j}^{n+1,\nu} + \frac{(q^x_{i+\frac{1}{2},j})^{n+1,\nu} - (q^x_{i-\frac{1}{2},j})^{n+1}}{x_{i+\frac{1}{2},j} - x_{i-\frac{1}{2},j}} + \frac{(q^y_{i,j+\frac{1}{2}})^{n+1,\nu} - (q^y_{i,j-\frac{1}{2}})^{n+1,\nu}}{y_{i,j+\frac{1}{2}} - y_{i,j-\frac{1}{2}}} + \mathcal{G}_{i,j}^{n+1} \tag{7}
$$

where $\varphi_{i,j}^{n+1}$ is our unknown in the Newton procedure and we use the iteration ν to reach the solution of the nonlinear equation at each time step.

To compute the tangent matrix we derive the residual respect to each component of the variable vector, i.e.:

$$
\mathcal{K}_{\alpha,(i,j)} = \frac{\partial \mathcal{R}_\alpha}{\partial \varphi_{(i,j)}^{n+1}} = -\frac{\rho_{i,j}(C_p)_{i,j}}{\Delta t} - c_{i,j} + \frac{\frac{\partial (q^x)^{n+1}_{i+\frac{1}{2},j}}{\partial \varphi_{(i,j)}^{n+1}} - \frac{\partial (q^x)^{n+1}_{i-\frac{1}{2},j}}{\partial \varphi_{(i,j)}^{n+1}}}{x_{i+\frac{1}{2},j} - x_{i-\frac{1}{2},j}} + \frac{\frac{\partial (q^y)^{n+1}_{i,j+\frac{1}{2}}}{\partial \varphi_{(i,j)}^{n+1}} - \frac{\partial (q^y)^{n+1}_{i,j-\frac{1}{2}}}{\partial \varphi_{(i,j)}^{n+1}}}{y_{i,j+\frac{1}{2}} - y_{i,j-\frac{1}{2}}} \tag{8}
$$

with here we use α as (i,j) to simplify the notation and where the partial derivative of the heat fluxes respect to the variable should be computed from 4.

$$
\begin{aligned}
\frac{\partial (q^x)^{n+1}_{i+\frac{1}{2},j}}{\partial \varphi_{i,j}^{n+1}} &= \frac{-\kappa_{i+\frac{1}{2},j}}{x_{i+1,j} - x_{i,j}} \\
\frac{\partial (q^y)^{n+1}_{i,j+\frac{1}{2}}}{\partial \varphi_{i,j}^{n+1}} &= \frac{-\kappa_{i,j+\frac{1}{2}}}{y_{i,j+1} - y_{i,j}} \\
\\
\frac{\partial (q^x)^{n+1}_{i-\frac{1}{2},j}}{\partial \varphi_{i,j}^{n+1}} &= \frac{-\kappa_{i-\frac{1}{2},j}}{x_{i,j} - x_{i-1,j}} \\
\frac{\partial (q^y)^{n+1}_{i,j-\frac{1}{2}}}{\partial \varphi_{i,j}^{n+1}} &= \frac{-\kappa_{i,j-\frac{1}{2}}}{y_{i,j} - y_{i,j-1}}
\end{aligned} \tag{9}
$$

The same with the other components of the tangent matrix

$$
\begin{aligned}
\mathcal{K}_{\alpha,(i+1,j)} &= \frac{\partial \mathcal{R}_\alpha}{\partial \varphi^{n+1}_{(i+1,j)}} \\
\mathcal{K}_{\alpha,(i-1,j)} &= \frac{\partial \mathcal{R}_\alpha}{\partial \varphi^{n+1}_{(i-1,j)}} \\
\\
\mathcal{K}_{\alpha,(i,j+1)} &= \frac{\partial \mathcal{R}_\alpha}{\partial \varphi^{n+1}_{(i,j+1)}} \\
\mathcal{K}_{\alpha,(i,j-1)} &= \frac{\partial \mathcal{R}_\alpha}{\partial \varphi^{n+1}_{(i,j-1)}}
\end{aligned} \tag{10}
$$

3.3 Boundary condition treatment

Without enter in too much details about this (readers may see some of the notes of the course put in `http://www.cimec.org.ar/twiki/courses` here we summarize how to write the boundary conditions and how these extra equations modify the numerical system to solve.

3.3.1 Dirichlet boundary conditions

As the solution of the numerical problem is written in terms of increments of a general Newton procedure, if the initial guess satisfy the Dirichlet boundary conditions the next increments should be always null in order to maintain the satisfaction of the boundary condition. This is a simple way avoiding the modification of the linear system built during the procedure, only reducing this system dropping the rows and columns of those nodes where this boundary condition is applied.

$$
\begin{aligned}
\mathcal{R}_\alpha &= [] \\
\\
\mathcal{K}_{\alpha,:} &= [] \\
\mathcal{K}_{:,\alpha} &= []
\end{aligned} \tag{11}
$$

for all $\alpha = (i,j)$ where Dirichlet boundary conditions are applied. [] means making empty the corresponding rows and or columns of the array.

Other possibility may be

$$
\begin{aligned}
\mathcal{R}_{\alpha} &= 0 \\
\\
\mathcal{K}_{\alpha,\beta} &= 0 \qquad\qquad \beta \neq \alpha \\
\mathcal{K}_{\alpha,\alpha} &= 1
\end{aligned}
\tag{12}
$$

3.3.2 Neumann boundary conditions

For applying such a boundary condition we use ficticious nodes. This is one of the different alternatives. Using the code the user should not worry about this, the program makes it by itself. The strategy consist in defining an extra node for each node of the boundary where Neumann boundary condition is applied following the direction provided by the normal defined in this type of boundary condition. In general as it was shown in 2 we have

$$
\mathcal{R}(\varphi^{n+1}_{ext(i,j,\eta)}) = -\kappa_{(i,j)} \frac{\varphi^{n+1}_{ext(i,j,\eta)} - \varphi^{n+1}_{int(i,j,\eta)}}{(x^{n+1}_{ext(i,j,\eta)} - x^{n+1}_{int(i,j,\eta)})\eta_x + (y^{n+1}_{ext(i,j,\eta)} - y^{n+1}_{int(i,j,\eta)})\eta_y} - \overline{q}
\tag{13}
$$

where $ext(i,j,\eta)$ means the ficticious node corresponding to the node (i,j) where the boundary conditions is applied following the η direction (unit normal is external oriented) and $int(i,j,\eta)$ is the corresponding interior node in the opposite direction of the external normal.

The contribution of these equations to the matrix is computed in a similar way previously shown.

3.3.3 mixed boundary conditions

Left as exercise.

4 INFORMATIC DETAILS

See the text file *my_notes.txt* for these details.

5 SOME EXAMPLES TO VALIDATE THE CODE

All the examples consist of rectangular domains aligned with cartesian axes. Boundary left, right, left and bottom are immediately identified without a sketch.

5.1 Example # 1

Solve

$$
\begin{aligned}
\nabla \cdot (\kappa \nabla \varphi) + \mathcal{G} = 0 & \\
\varphi = 0 & \qquad \forall \mathbf{x} \in \Gamma_{left} \\
\varphi = 1 & \qquad \forall \mathbf{x} \in \Gamma_{right} \\
-\kappa \nabla \varphi \cdot \eta = 0 & \qquad \forall \mathbf{x} \in \Gamma_{top} \cup \Gamma_{bottom}
\end{aligned}
\tag{14}
$$

This example is a quasi one dimensional problem where analytic solution may be obtained.

5.1.1 Caso # 1-(a)

For $\mathcal{G} = 0$ the solution is a straight line between the left and right temperature values.

First we prove this example and then we follow with non null source terms

5.1.2 Caso # 1-(b)

$\mathcal{G} = x^p$ with different values for p, $p = 1, 2, 3$

5.1.3 Caso # 1-(c)

$\mathcal{G} = e^{-x^2}$.

5.2 *Example # 2*

Solve

$$\nabla \cdot (\kappa \nabla \varphi) + \mathcal{G} = 0 \tag{15}$$

with the temperature of all the boundaries imposed to a given expression. Propose an expression for the solution, like $\varphi = \sum a_{p,q} x^p y^q$, match the boundary conditions to these field. After this get the source expression that balance the above partial differential equation and after that treat to solve numerically the problem analyzing how the error depend on the grid size.

6 PROBLEMS TO BE SOLVED WITH THE CODE

6.1 Problem # 1

Solve the advection-diffusion equation in a rectangular domain considering that the fluid go into the domain through the left boundary with a temperature of 100 Celsius degrees and found a heat bottom boundary at 300 Celsius. The upper boundary is consider to far and then it does not transfer heat through it. The right boundary is the output of the flow and it may be consider that the thermal profile is developed. Compute the temperature all around the domain at steady state for a fluid that have a thermal diffusivity of $0.001m^2/seg$ and the velocity field is contant, aligned with the x axis with a magnitude of $1mm/seg$. After getting this solution solve another similar problem but changing the velocity magnitude from $1mm/seg$ to $1m/seg$. Which changes do you note in the solution ? Explain the quality of the solutions obtained and if necessary explain how to improve them. Compare the numerical solution with the analytical solution to the boundary layer approximation given in the course theory.

6.2 Problem # 2

Consider a fin of a heat exhanger, a flat plate of rectangular shape that receive at its root a fixed temperature of 100 Celsius coming from the liquid being transported inside the tubes. Being the whole fin wetted by the air used to cool it that it is at 20 Celsius of temperature in its bulk and being the velocity of the air such that the convective film coefficient is around $h_{film} = 100Watt/(m^2K)$ applied to the whole plate (up and down). Determine the efficiency of the heat exchanger as a function of the fin length. The efficiency may be defined as the heat flux removed relative to the maximum heat flux produced by an infinite length fin.

As the convective heat transfer produced by the air at the up and down surface may be computed as a reactive term, then, this problem may be reduced to a 2D problem reaction-diffusion equation (1) taking $\kappa = 100W/K$ and a width of 10 cm for the fin.

Hint: take into account that the reaction term should be twiced because of the up and down faces exposed to the convective heat transfer.

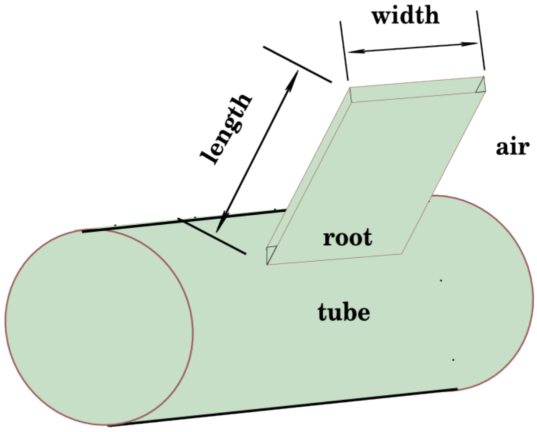

Fig. 1. Problem 2

6.3 Problem # 3

Solve the time evolution of a thermal wave produced at the left boundary of a rectangular domain and compare it with an analytical solution for such a problem. Remember that a thermal wave may be produced using a time varying Dirichlet boundary condition, i.e.

$$\begin{aligned} &\varphi = \overline{\varphi} \sin(\omega t) \sin(ky) \qquad \forall \mathbf{x} \in \Gamma_{left} \\ &\mathbf{q} = 0 \qquad \forall \mathbf{x} \in \Gamma_{top} \cup \Gamma_{bot} \cup \Gamma_{rig} \end{aligned} \tag{16}$$

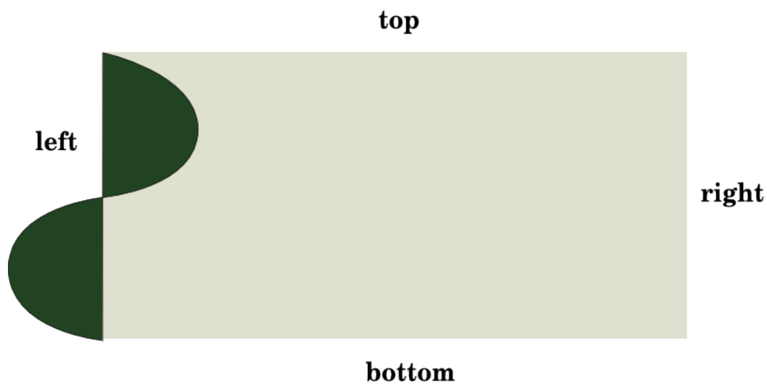

Fig. 2. Problem 3

6.4 Problem # 4

Given a rectangular domain formed by two materials with different conductivities. The less conductive material is placed at the middle vertically in two parts, one at the upper part and the other at the lower part leaving a space with the more conductive material in between. Analyze the influence of the dimensions of the application of less conductive material on the global thermal resistance of the device imposing two temperatures, one at the left boundary and one at the right boundary, as it is shown in the corresponding figure. Its width is fixed to a 20% of the domain length and the channel length is varied to get the corresponding $q = f(length)$ figure. Use $T = 300$ Celsius at left, $T = 30$ Celsius at right, $\kappa\|_1 = 43W/m/K$ (steel) and $kappa\|_2 = 1.5W/m/K$ (porcelain).

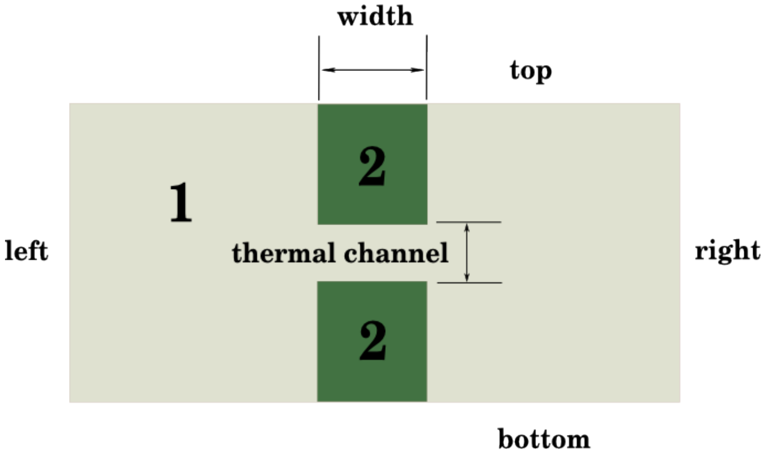

Fig. 3. Problem 4