Membre de:
XIT XARXA DE CENTRES DE SUPORT A LA INNOVACIÓ TECNOLÒGICA

CTTC UPC
Centre Tecnològic de Transferència de Calor
UNIVERSITAT POLITÈCNICA DE CATALUNYA

A Two-dimensional Transient Conduction Problem

by CTTC

1. Exercise

- Write a computer program (according to the specifications given in section 4) to solve the heat conduction equation in the situation described in section 2.
- Ensure that the code is correct
- Choose a suitable mesh and time step.
- Run the simulation and submit us the files with the information requested in section 3.
- We will check the code and the results:
  - If the code and results are correct, we will ask you to write a short report about the work made and then you passed the test.
  - If the code looks good but there are problems, such as not enough accuracy, wrong programming style, etc, we will help you to enhance it and we will give you more opportunities.
  - We will not accept candidates who have used software not developed by themselves OR have given their own codes to other persons.

Comments:

- You must write your own code, not use already available software.
- A normal PC is enough to do this exercise.
- This is a personal problem, you can not ask for help of other persons.
- Don't give this problem to other persons.
- If you have questions about how to do this exercise, we suggest you to read the chapters 1-4 of the book "Numerical Heat Transfer and Fluid Flow" [1].
- If you have a question about the exercise please ask us.

2. Basic problem definition

A very long rod is composed of four different materials (M1 to M4), represented with different colours in the figure below. All the lines are parallel to the coordinate axis. The coordinates of the points p1 to p3 are given in table 1. Please note that the drawing is not scaled to the true dimensions of the materials. The properties of the materials are given in table 2. Each of the four sides of the rod interacts with the surrounding in a different manner, as described in table 3. The initial temperature field is $T = 8.00^oC$.

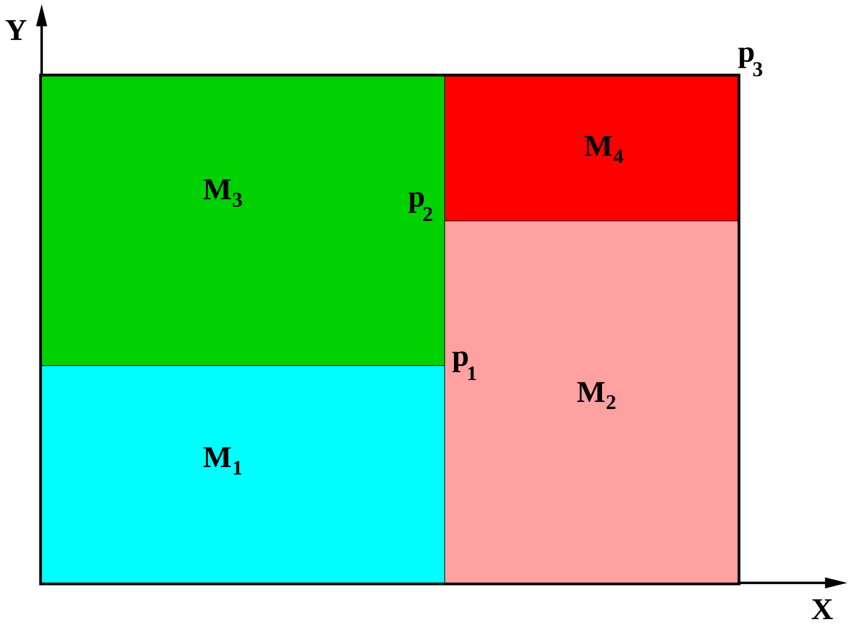

Figure 1: General schema of the proposed problem

	$x[m]$	$y[m]$
p_1	0.50	0.40
p_2	0.50	0.70
p_3	1.10	0.80

Table 1: Problem coordinates

	$\rho[kg/m^3]$	$c_p[J/kgK]$	$k[W/mK]$
M_1	1500.00	750.00	170.00
M_2	1600.00	770.00	140.00
M_3	1900.00	810.00	200.00
M_4	2500.00	930.00	140.00

Table 2: Physical properties

Cavity wall	Boundary condition
Bottom	Isotherm at $T = 23.00^oC$
Top	Uniform $Qflow = 60.00W/m$ length *
Left	In contact with a fluid at $T_g = 33.00^oC$ and heat transfer coefficient $9.00W/m^2K$
Right	Uniform temperature $T = 8.00 + 0.005t\ ^oC$ (where t is the time in seconds)

* This is an inlet heat flux, i.e. 60 W/m in the negative y-direction.

Table 3: Specific boundary conditions

3. Information requested

You must submit us two text files.

- Your code (see section 4.for details), named ”BNAME.c” where BNAME stands for your surname.
- The output file ”BNAME.dat”, that must contain 3 columns with the following information:

– Column 1: Time (s)
– Column 2: Temperature at the time specified in column 1, at location (0.65, 0.56).
– Column 3: Temperature at the time specified in column 1, at location (0.74, 0.72).

The file must contain 10000 rows, with increasing time. The first instant must be time equal to zero (the initial conditions) and the last instant time equal to 10000 s. No other information should be in the output file.

If you are familiar with gnuplot, you can check that the format of the output file is correct doing the following:

$gnuplot

gnuplot $>$ plot "BNAME.dat" using 1:2, "BNAME.dat" using 1:3

4. Code requirements

- You must write a C++, C or FORTRAN code that can be compiled in Linux environment.
- You can not use libraries (e.g. linear algebra libraries, PDE solvers, etc) not developed by you.
- The code should be modular and split in subroutines, presented in a single file and compile with no errors.
- The code must run without any input parameter and produce the requested solution file.

5. `Last comments

You can check that the results of your code are correct comparing with the instantaneous isotherms plot given below. You can also test steady 1D solutions of unsteady solutions in case of uniform temperature (0-dimensional) (e.g. impossing an extremely high thermal conductivity). After code verification, you can also test the influence of mesh density, time step, unsteady scheme, boundary conditions, etc.

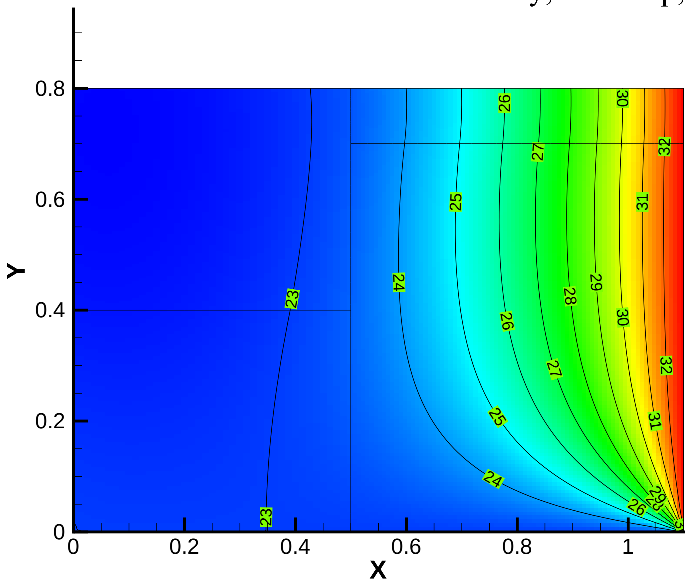

Figure 2: Instantaneous isotherms at $t = 10000/2$ s

References

1. Suhas V. Patankar. *Numerical Heat Transfer and Fluid Flow*. Hemisphere Publishing Corporation, McGraw-Hill Book Company, 1980.

Conduction heat transfer problem. Global algorithm

1. Input data: i) physical data (geometry, materials, boundary conditions, initial conditions); ii) numerical data (mesh, time step, convergence criteria, β, fr)

2. Previous calculation: mesh generation, geometry (surfaces, volumes), etc.

3. Initial temperature map ($t^n=0$): $T^n[i][j] = T_0(x,y)$

4. Calculation of the next time step is started: $t^{n+1} = t^n + \Delta t$
   Temperature field is estimated: $T^{n+1(*)}[i][j] = T^n[i][j]$

5. Evaluation of the discretization coefficients: a_E, a_W, a_N, a_S, b_P, a_P at each node $[i][j]$

6. Solve the set of equations: $a_P[i][j]\,T^{n+1}[i][j] = a_E[i][j]\,T[i+1][j] + a_W[i][j]\,T[i-1][j] + a_N[i][j]\,T[i][j+1] + a_S[i][j]\,T[i][j-1] + b_P[i][j]$
   (point-by-point or better line-by-line)

7. Is $\max\left|T^{n+1}[i][j] - T^{n+1*}[i][j]\right| < \delta$? $\xrightarrow{\text{YES}}$ go to 8
   $\xrightarrow{\text{NO}}$ $T^{n+1*}[i][j] = T^{n+1}[i][j]$, go to 5

8. New time step? $\xrightarrow{\text{NO}}$ go to 9
   $\xrightarrow{\text{YES}}$ $T^n[i][j] = T^{n+1}[i][j]$, go to 4
   $t^n = t^{n+1}$

9. Final calculations and print results

10. End.