

Two dimensional heat equation

Praveen. C, Deep Ray, Jean-Pierre Raymond

July 16, 2013

1 The model

Let $z = z(x, y, t)$ denote the temperature. The shifted 2-D heat equation is given by

$$z_t = \mu \Delta z + \alpha z, \quad (x, y) \in \Omega = (0, 1) \times (0, 1), \quad t \in [0, T]$$

with boundary conditions

$$z(x, 0, t) = z(x, 1, t) = 0, \quad z(1, y, t) = u(y, t), \quad \frac{\partial z}{\partial x}(0, y, t) = 0$$

and initial condition

$$z(x, y, 0) = z_0(x, y)$$

Here $\alpha \geq 0$ and $\mu > 0$. Let us denote the Dirichlet part of the boundary by Γ_D

$$\Gamma_D = \{y = 0\} \cup \{y = 1\} \cup \{x = 1\}$$

the Neumann part as

$$\Gamma_N = \{x = 0\}$$

and the part on which the control is applied as

$$\Gamma_c = \{x = 1\}$$

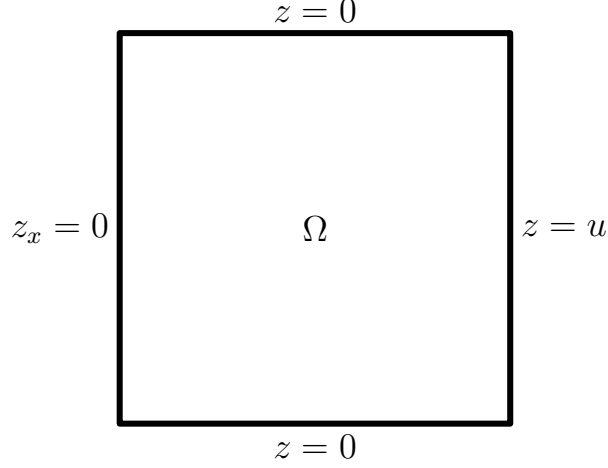


Figure 1: Problem definition

1.1 Observations

We will measure an average value of the temperature on strips along the left vertical boundary

$$I_i = [a_i, b_i]$$

as shown in figure. Thus the observations are

$$y_i(t) = \frac{1}{b_i - a_i} \int_{a_i}^{b_i} z(0, y, t) dy \quad (1)$$

Note: Do not confuse the observation y_i with the spatial coordinate y .

1.2 Weak formulation

We assume $z_0 \in L^2(\Omega)$. We wish to find $z \in L^2(0, T; H^1(\Omega))$ such that

$$\frac{d}{dt}(z(t), \phi)_{L^2} = -\mu \int_{\Omega} \nabla z \cdot \nabla \phi dx + \alpha \int_{\Omega} z \phi dx, \quad \forall \phi \in H_{\Gamma_D}^1(\Omega)$$

$$z(x, 0, t) = z(x, 1, t) = 0, \quad z(1, y, t) = u(y, t)$$

$$(z(0), \phi)_{L^2} = (z_0, \phi)_{L^2}$$

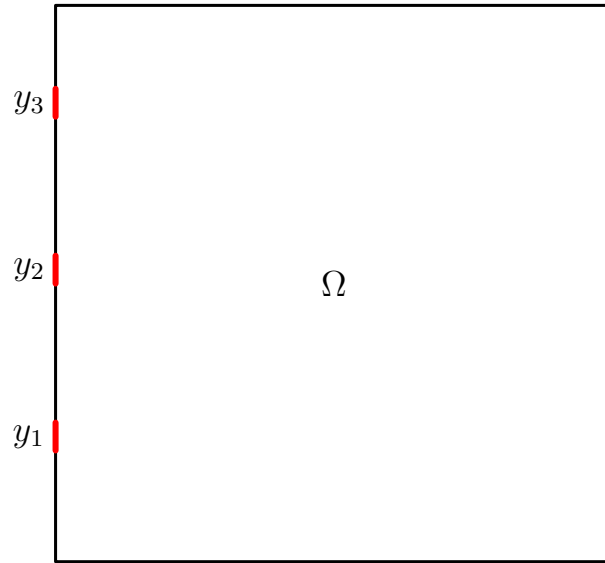


Figure 2: Observations

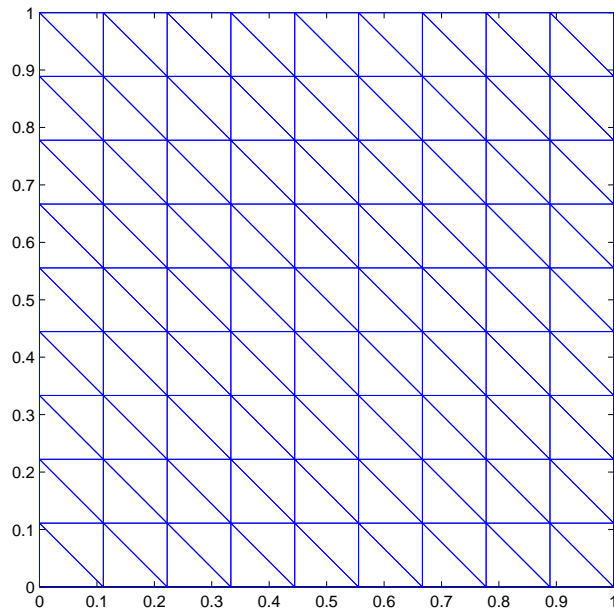


Figure 3: Example of a finite element mesh

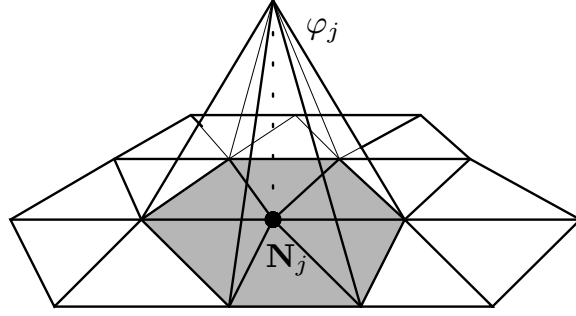


Figure 4: Piecewise affine basis functions

2 FEM approximation

Consider a division of Ω into disjoint triangles as shown in figure (xxx). We will assume that the vertices of the mesh are numbered in some manner. Let us define the following sets of vertices

$$\begin{aligned} N_c &= \text{vertices on } \Gamma_c \\ N_d &= \text{vertices on } \{y = 0\} \cup \{y = 1\} \\ N_f &= \text{remaining vertices} \end{aligned}$$

For each vertex i , define the piecewise affine functions $\phi_i(x, y)$ with the property that

$$\phi_i(x_j, y_j) = \delta_{ij}$$

We will take the control to be of the form

$$u(y, t) = v(t) \sin(\pi y)$$

Then the finite element solution is of the form

$$z(x, y, t) = \sum_{j \in N_f} z_j(t) \phi_j(x, y) + v(t) \sum_{j \in N_c} \sin(\pi y_j) \phi_j(x, y)$$

The approximate weak formulation is given as

$$\frac{d}{dt}(z(t), \phi_i)_{L^2} = -\mu \int_{\Omega} \nabla z \cdot \nabla \phi_i dx + \alpha \int_{\Omega} z \phi_i dx, \quad \forall i \in N_f$$

i.e.,

$$\begin{aligned}
& \sum_{j \in N_f} \frac{dz_j}{dt} \int_{\Omega} \phi_j \phi_i + \frac{dv}{dt} \sum_{j \in N_c} \sin(\pi y_j) \int_{\Omega} \phi_j \phi_i \\
&= -\mu \sum_{j \in N_f} z_j \int_{\Omega} \nabla \phi_j \cdot \nabla \phi_i - \mu v \sum_{j \in N_c} \sin(\pi y_j) \int_{\Omega} \nabla \phi_j \cdot \nabla \phi_i \\
& \quad + \alpha \sum_{j \in N_f} z_j \int_{\Omega} \phi_j \phi_i + \alpha v \sum_{j \in N_c} \sin(\pi y_j) \int_{\Omega} \phi_j \phi_i, \quad \forall i \in N_f
\end{aligned}$$

In order to simplify the presentation we will ignore the term containing $\frac{dv}{dt}$ ¹ and then we can write the FEM formulation as

$$\begin{aligned}
& \sum_{j \in N_f} \frac{dz_j}{dt} \int_{\Omega} \phi_j \phi_i \\
&= \sum_{j \in N_f} z_j \left[-\mu \int_{\Omega} \nabla \phi_j \cdot \nabla \phi_i + \alpha \int_{\Omega} \phi_j \phi_i \right] \\
& \quad v \sum_{j \in N_c} \left[-\mu \sin(\pi y_j) \int_{\Omega} \nabla \phi_j \cdot \nabla \phi_i + \alpha \sin(\pi y_j) \int_{\Omega} \phi_j \phi_i \right] \quad \forall i \in N_f
\end{aligned}$$

This can be written as a system of differential equations

$$\mathbf{M} \frac{d\mathbf{z}}{dt} = \mathbf{A}\mathbf{z} + \mathbf{B}v$$

2.1 Finite element assembly

The finite element basis functions have compact support. Hence we can compute the integrals by adding the contributions from a small number of triangles. For example, the elements of the mass matrix can be computed as

$$\int_{\Omega} \phi_i \phi_j = \sum_{K : i, j \in K} \int_K \phi_i \phi_j$$

and similarly the stiffness matrix is computed as

$$\int_{\Omega} \nabla \phi_i \cdot \nabla \phi_j = \sum_{K : i, j \in K} \int_K \nabla \phi_i \cdot \nabla \phi_j$$

¹This term vanishes if we use the trapezoidal rule for integration.

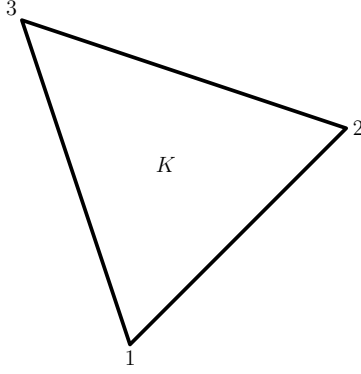


Figure 5: Triangle

The integrals on each triangle K will be evaluated exactly. For a triangle K with vertices labelled 1, 2, 3 as in figure (5), the local mass and stiffness matrices are given by

$$M^K = \frac{1}{24} \det \begin{bmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{bmatrix} \begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix}$$

$$A^K = \frac{1}{2} \det \begin{bmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{bmatrix} G G^\top \quad \text{where} \quad G = \begin{bmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{bmatrix}^{-1} \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

2.2 Computing the observation

We will assume that the intervals $[a_i, b_i]$ on which the observation is computed is exactly covered by the edges of the finite element mesh. Let E_i denote the edges on $[a_i, b_i]$

$$\begin{aligned} y_i &= \frac{1}{b_i - a_i} \int_{a_i}^{b_i} z(0, y, t) dy \\ &= \frac{1}{b_i - a_i} \sum_{e \in E_i} \int_e z(0, y, t) dy \\ &= \frac{1}{b_i - a_i} \sum_{e \in E_i} \frac{1}{2} (z_{e_1} + z_{e_2}) |e| \end{aligned}$$

The set of observations can be written as

$$\mathbf{y} = \mathbf{H}\mathbf{z}$$

The observation zones are defined by the following parameters

Value/ i	1	2	3
a	0.20	0.50	0.80
b	0.25	0.55	0.85

2.3 Grid information

The grid information consists of following files

- `coordinates.dat`: contains x, y coordinates of all the vertices
- `elements3.dat`: contains verices forming each triangle
- `dirichlet.dat`: contains boundary edges on the dirichlet boundary
- `neumann.dat`: contains boundary edges on the neumann boundary

An example of this type of data is given in figures (6), (7).

**Note: In our current programs we use a mesh consisting of only triangles.
Also, the serial numbers are not stored in the files.**

The domain Ω is the unit square. The program `square.m` generates the mesh and creates the above four files. You have to specify the number of vertices on the side of the square. E.g., to have 11 points on each side, you do the following in matlab

```
>> square(11)
```

When you run the above program, you can see the mesh being plotted. Examine the four files created by this program. For out actual computations, we will use 101 points on each side of the square. In this case, there are 5 edges which exactly cover each of the three observation zones.

1	0	0
2	1	0
3	1.59	0
4	2	1
5	3	1.41
6	3	2
7	3	3
8	2	3
9	1	3
10	0	3
11	0	2
12	0	1
13	1	1
14	1	2
15	2	2

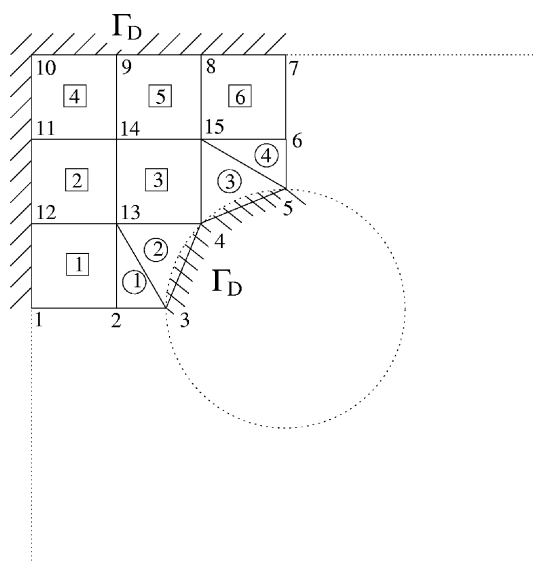


Figure 6: Example of a mesh

elements3.dat	elements4.dat
1 2 3 13	1 1 2 13 12
2 3 4 13	2 12 13 14 11
3 4 5 15	3 13 4 15 14
4 5 6 15	4 11 14 9 10
	5 14 15 8 9
	6 15 6 7 8

neumann.dat	dirichlet.dat
1 5 6	1 3 4
2 6 7	2 4 5
3 1 2	3 7 8
4 2 3	4 8 9
	5 9 10
	6 10 11
	7 11 12
	8 12 1

Figure 7: Structure of mesh files