

# Case Study Modelling an Electronic Component

Azure Hutchings      Jean-Luc Danoy  
Faris Saad S Alsubaie

28 October 2019

## **Executive Summary**

Write Abstract Here

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Purpose of the Report . . . . .	3
1.2	The Maths of the Problem . . . . .	3
<b>2</b>	<b>Method for Discretising the Problem</b>	<b>5</b>
2.1	Constructing the Finite Difference Mesh . . . . .	5
2.1.1	Interior Nodes . . . . .	6
2.2	Node Ordering . . . . .	10
2.3	Matrix Construction . . . . .	10

# 1 Introduction

## 1.1 Purpose of the Report

The following report investigates the steady-state heat distribution in a newly designed component. We discuss how to numerically solve for the steady-state heat distribution, the efficiency of different methods and the effect of different ambient temperatures.

## 1.2 The Maths of the Problem

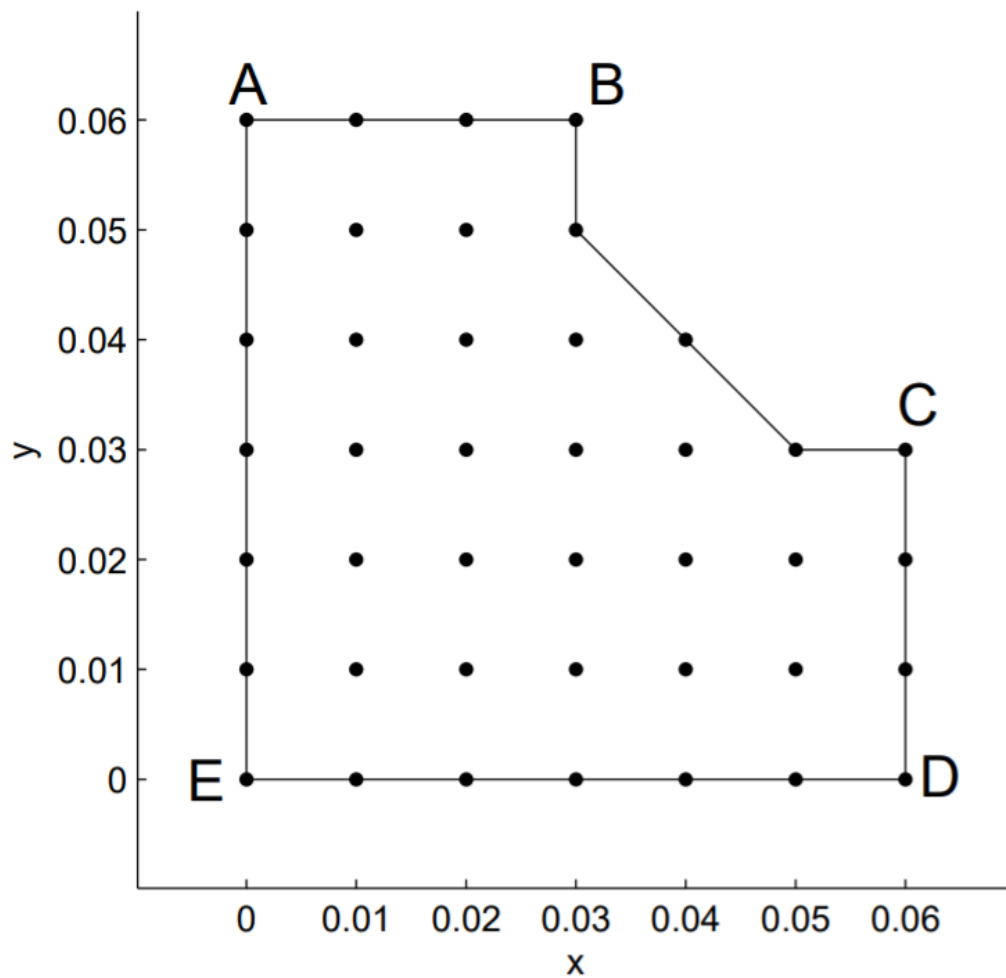


Figure 1: Schematic of electronic component.

The component schematic is shown in Figure 1. The location of the component within the device means it's subject to different temperature condition along it's boundaries. The boundary A-B is in perfect thermal contact with another component which the temperature is known to  $70^{\circ}\text{C}$ . The boundary C-D is also in perfect thermal contact with another component which the temperature is known to be  $40^{\circ}\text{C}$ . The boundary A-E-D is thermally insulated and the boundary B-C is exposed to the air at ambient temperature.

This type of model can be described with Laplace's equation. Letting  $T(x, y)$  represent the temperature of the component at point  $(x, y)$ , the model is as follows

$$\begin{aligned}
 \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} &= 0 && \text{in the interior} \\
 T &= 70 && \text{on boundary A-B} \\
 T &= 40 && \text{on boundary C-D} \\
 \nabla T \cdot \hat{\mathbf{n}} &= 0 && \text{on boundary A-E-D} \\
 k \nabla T \cdot \hat{\mathbf{n}} &= h(T_{\infty} - T) && \text{on boundary B-C}
 \end{aligned}$$

Where the thermal conductivity is  $k = 3\text{Wm}^{-1}\text{C}^{-1}$ , and the heat transfer coefficient is  $h = 20\text{Wm}^{-2}\text{C}^{-1}$ . To begin with, we will assume the ambient temperature is  $T_{\infty} = 20$ .

## 2 Method for Discretising the Problem

If this was left as a continuous partial differential equation, we would need to analytically solve it. However, it can be much quicker to numerically solve this problem and still retain a high degree of accuracy with the final answers. In this section we will show how we converted the analytical problem to a numerical problem, how the mesh was constructed, the node ordering used, the linear system we derived from the mesh and discuss the matrix that was created from it.

### 2.1 Constructing the Finite Difference Mesh

Suppose we had a function  $\phi(x, y, t)$  which gives the temperature at the point  $(x, y)$  on a 2-dimensional plane where  $t$  is the time since the start of the initial conditions. If the function has a steady-state solution, then it means at some point in time, any increase in time will not result in a change of temperature. This means that

$$\frac{d\phi}{dx}\bigg|_{x_i} \approx \frac{\phi(x_i + \Delta x) - \phi(x_i)}{\Delta x},$$

where the change in  $x$  along  $\phi$  at the point  $x_i$  is approximately the functions forward difference over the distance between nodes. We can then take the derivative of this again using backwards difference to get the second derivative centered around  $x_i$ .

$$\frac{d^2\phi}{dx^2}\bigg|_{x_i} \approx \frac{\phi(x_i + \Delta x) - 2\phi(x_i) + \phi(x_i - \Delta x)}{(\Delta x)^2},$$

If we do the same for the change in  $y$ , we arrive at an analogous equation

$$\frac{d^2\phi}{dy^2}\bigg|_{y_i} \approx \frac{\phi(y_i + \Delta y) - 2\phi(y_i) + \phi(y_i - \Delta y)}{(\Delta y)^2},$$

To numerically solve this problem, we must construct a finite difference mesh and create a matrix to represent this information.

By splitting the component up into intervals of 0.01 in both the  $x$  and  $y$  direction, we can split the component into a  $7 \times 7$  grid. Because we know that the temperature on the upper and right hand boundaries are fixed temperatures, we can ignore them in our discretisation as nodes. They will come into effect later as the right hand side matrix. Given the previous mesh from the introduction, we can label each node  $u_{i,j}$  for  $i, j = 0$  to  $6$  where  $i, j$  are given by the row and column of the node starting from the bottom left hand corner. Replacing  $\phi(x, y, t)$  with  $u_{i,j}(t)$  and noting that  $\Delta x = \Delta y$ , we arrive at the set of equations for the change in  $u_{i,j}(t)$ .

$$u'_{i,j}(t) \approx \frac{D}{\Delta x^2} (-u_{i+1,j}(t) - u_{i,j+1}(t) + 4u_{i,j}(t) - u_{i-1,j}(t) - u_{i,j-1}(t)).$$

Note that since we are finding a state where the temperature does not change in time ( $u'_{i,j}(t) = 0$ ), and assuming that  $D \neq 0$ , we can rewrite this to be

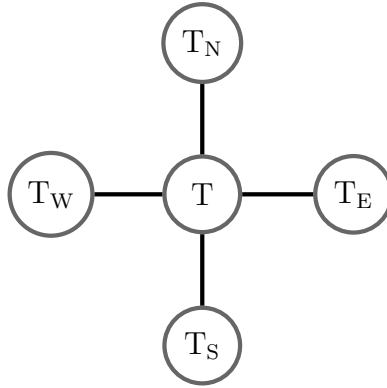
$$0 = (-u_{i+1,j} - u_{i,j+1} + 4u_{i,j} - u_{i-1,j} - u_{i,j-1}).$$

There are 3 different types of mesh that need to be created. These are for the A-E-D boundary, the interior of the mesh, and on the B-C boundary.

### 2.1.1 Interior Nodes

On the interior of the mesh, each node contains 4 other nodes adjacent to it, and the change in temperature at that node is given by the equation

$$\begin{aligned} \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} &= 0 \\ \frac{2T - T_W - T_E}{(0.01)^2} + \frac{2T - T_N - T_S}{(0.01)^2} &= 0 \\ 4T - T_W - T_E - T_N - T_S &= 0 \end{aligned}$$



As an example, we can examine the node  $T_{(1,1)}$ .

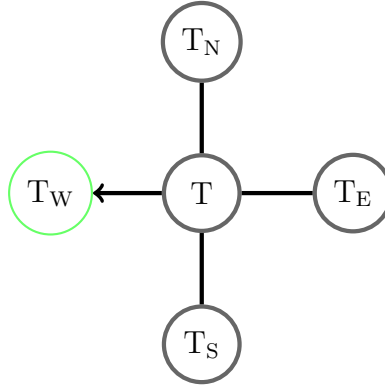
$$4T_{(1,1)} - T_{(0,1)} - T_{(2,1)} - T_{(1,2)} - T_{(1,0)} = 0$$

### Boundary A-E-D Nodes

On the boundary A-E-D, there are no nodes outside of the matrix. We can approximate them using first order forward differences (to keep the A matrix SPD). This approximation can be done using the rule  $\nabla T \cdot \hat{\mathbf{n}} = 0$ , where  $\hat{\mathbf{n}}$  is the normal unit vector at the node T and  $\nabla T$  is the directional gradient at T. The first order differences are then used to construct a finite difference equations for each node on the boundary.

We will provide 3 examples that show all the unique cases along the boundary A-E-D, namely on the border A-E (not including A or E), the border E-D (not including E or D) and the node E.

**Nodes A-E** On the A-E nodes, the normal unit vector will face directly west.



Therefore  $\hat{\mathbf{n}} = (-1, 0)^T$ . We can construct a first order forward difference approximation for  $T_W$ .

$$\left(\frac{\partial T}{\partial x}, \frac{\partial T}{\partial y}\right) \cdot (-1, 0)^T = 0$$

$$-\frac{\partial T}{\partial x} = 0$$

$$\frac{T - T_W}{0.01} = 0$$

$$T_W = T.$$

This value of  $T_W$  can be used in the stencil around the point T.

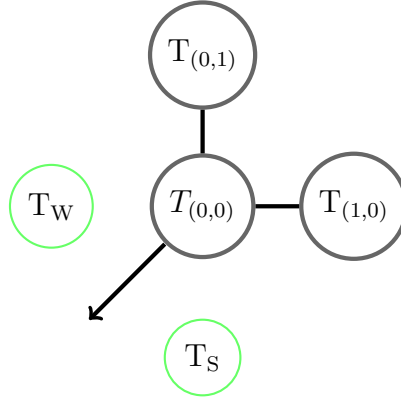


$$\begin{aligned}
\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} &= 0 \\
\frac{2T - T_W - T_E}{(0.01)^2} + \frac{2T - T_N - T_S}{(0.01)^2} &= 0 \\
2T - T_W - T_E + 2T - T_N - T_S &= 0 \\
4T - T - T_E - T_N - T_S &= 0 \\
3T - T_E - T_N - T_S &= 0.
\end{aligned}$$

As an example, let's look at node  $T_{(0,3)}$ .

$$3T_{(0,3)} - T_{(1,3)} - T_{(0,4)} - T_{(0,2)} = 0$$

**The Node E:** On the node E, or (0,0) the unit normal vector will be facing southwest.



As a normalised vector, it will be

$$\hat{\mathbf{n}} = \frac{1}{\sqrt{2}}(-1, -1)^T.$$

We can use this in our boundary condition to create a first order difference approximation for  $T_W + T_S$ .

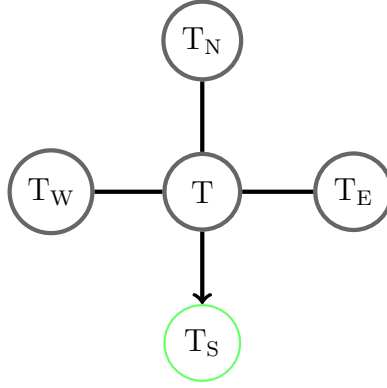
$$\begin{aligned}
\left(\frac{\partial T}{\partial x}, \frac{\partial T}{\partial y}\right) \cdot \frac{1}{\sqrt{2}}(-1, -1)^T &= 0 \\
\left(\frac{\partial T}{\partial x}, \frac{\partial T}{\partial y}\right) \cdot (-1, -1)^T &= 0
\end{aligned}$$

$$\begin{aligned}
\left(\frac{\partial T}{\partial x} + \frac{\partial T}{\partial y}\right) &= 0 \\
\frac{T - T_W}{0.01} + \frac{T - T_S}{0.01} &= 0 \\
T_W + T_S &= 2T.
\end{aligned}$$

Substituting this into our stencil, we can see that

$$\begin{aligned}
4T_{(0,0)} - T_{(0,1)} - T_{(1,0)} - T_W - T_S &= 0 \\
4T_{(0,0)} - T_{(0,1)} - T_{(1,0)} - 2T_{(0,0)} &= 0 \\
2T_{(0,0)} - T_{(0,1)} - T_{(1,0)} &= 0.
\end{aligned}$$

**Nodes E-D:** On the nodes E-D, the normal unit vector will face directly south.



This can be written as

$$\hat{\mathbf{n}} = (0, -1)^T.$$

Using the same boundary rule for the edge A-E-D (which contains the edge E-D) we can find a first order forward difference for  $T_S$ .

$$\begin{aligned}
\left(\frac{\partial T}{\partial x}, \frac{\partial T}{\partial y}\right) \cdot (0, -1)^T &= 0 \\
-\frac{\partial T}{\partial y} &= 0 \\
\frac{T - T_S}{0.01} &= 0 \\
T_S &= T
\end{aligned}$$

Using this in our finite difference equation, we get the following equation for a node along the boundary E-D.

$$4T - T_S - T_N - T_W - T_E = 0$$

$$4T - T - T_N - T_W - T_E = 0$$

$$3T - T_N - T_W - T_E = 0.$$

## 2.2 Node Ordering

One last thing we must do is construct a mapping from the 2D array of nodes to a 1D array of nodes. This is useful when creating the vectors  $x$  and  $b$  in the linear equation  $Ax = b$ .

Because we have a 6x6 array (ignoring the missing or null nodes in the top right corner) we can transform this into a 1D array with 36 values. We can use the following assumptions to construct the map

- Our index is  $(i,j)$  with  $i$  being the horizontal coordinate and  $j$  being the vertical coordinate
- $(0,0)$  is the bottom left node
- $(i+1,j)$  is to the right of  $(i,j)$  and  $(i,j+1)$  is above  $(i,j)$

From this we can deduce that every time we move across the 2D array, we are moving 1 unit across in the 1D array, and every time we move up in the 2D array, we are moving 6 across in the 1D array. Therefore,

$$(i, j) \rightarrow i + 6j.$$

## 2.3 Matrix Construction