# Finite difference heat transfer analyses in Excel

In this first blog (after the hello world one) I’m tackling something I’ve wanted to do for some time now. That is setting up and solving a simple heat transfer problem using the finite difference (FDM) in MS Excel. The aim is to solve the steady-state temperature distribution through a rectangular body, by dividing it up into nodes and solving the necessary equations only in two dimensions. I’m going to illustrate a simple one-dimensional heat flow example, followed two-dimensional heat flow example, all programmed into Excel.

Finite difference analyses (FDA’s) are generally performed to predict the values of physical properties at discrete points throughout a body. In the case of a stationary body where heat transfer is primary phenomena, the temperature could be determined throughout as a function of heating or cooling on the boundaries, and the physical properties (heat transfer coefficient) of the material.

By definition, the FDM refers to a method for the numerical solution of differential equations. The solution to the solving the FDA as a whole that is described here is more methodological, relying rather on Excel’s ability to solve iterative loops, than performing matrix algebra.

## **Purpose**

Now why would I use Excel for FDM’s? There are primarily two reasons, and defines the purpose of this post:

1. Firstly, this helps to illustrates the basic concepts of the FDM, and someone following this should at the end have be able to apply the same principles in any other programming platform, and for problems other than simple heat transfer.

2. Secondly, certain aspects of some problems that extend beyond simple heat transfer are sometimes challenging to solve using available FDM or related software. An example of this could be a system where a material chemically reacts as a function of the composition and temperature in each node, and the chemical reactions need to be modeled using third party software. Although not the most efficient, Excel still remains a fairly flexible platform allowing for integration with many other software components.

Using Excel for to perform a finite difference analysis (FDA) does however have is its downsides. The system will be greatly simplified, and in the simplest form, the domain is only defined with fixed sized nodes being rectangular in shape.

**Definition**

To start off with the solution, the partial differential equation of the governing phenomena needs to be defined, in this case heat transfer. To arrive at the PDE, Fourier’s law is considered that defines that the negative of the gradient of temperature and area perpendicular to the direction heat is flowing. In one-dimensional form heat flux through a node is:

where q is the heat flux (W/m2),  T is the temperature (Kelvin) at a node through which heat is flowing, and k is the conductivity (W/mK) of the material through which heat is flowing.

**1D Heat Transfer**

The body for the temperature distribution which needs to be solved is broken up into a number of finite elements, or nodes, as illustrated below. Consider the node numbered x somewhere inside the body. The node to the left is numbered x-1, and to the right is x+1.

Heat is flowing through it in the x direction, and because we are only considering the steady state at the moment, the heat flux over the left boundary should be equal to the heat flux over the right boundary. Or in general, the heat flux into the node should be the heat flux out of the node.

The heat balance for the node x can be written as follows, discretized, and reduced to a single equation for the temperature of node x as function of the temperatures to the left and right:

**Boundary conditions**

The above sections covered nodes inside the body, but the nodes on the boundaries need different equations to be solved. For illustration, the problem was defined to have convective heat transfer (cooling) on the left side with a heat transfer coefficient of 5 W/m2K, and have a fixed temperature of 1600°C on the right side. The temperature equation for first node can be derived defining the heat flux into the first node as a function of the heat transfer coefficient:

**Application in Excel**

The next step is to program these equations into Excel and model the temperature profile in one dimension. The following two pictures show how to setup the solution, and then firstly how to enter the equations for node 0, and then the equations for nodes 1 to 8. Node 9 is simply given the value of 1600.

The above is then all that’s required to have Excel solve the heat transfer for our body. Ensure just that “iterative calculations” are enabled in the Excel options, and press F9 until the values in cells does not change anymore. To enhance the display of the results, the colors of the cells can be conditionally formatted to the value in the cells:

**2D Example**

The principles illustrated above in one dimension, can now simply be applied for two dimensions. The following illustrates our example domain. It is a square body, with a fixed temperature at the bottom, convective heat transfer at the top, no heat transfer in the x-direction on the right, and a heat loss value in the x-direction on the left.

**General internal node**

The domain is now divided into nodes in both the x- and y-directions, and the first step is to derive the temperature equation of a general internal node taking into account the heat fluxes in both directions. To follow is the equation that will be programmed for any node not on the boundary:

**Right boundary with no x-direction heat flux**

The temperature equations for the nodes on the right boundary is very similar to the internal nodes, except starting off that the qx is zero:

**Left boundary with heat flux**

The left boundary has a heat flux of -5000W in the x-direction, and the temperature equation can again be derived for all the nodes on the left boundary:

**Top boundary with convective heat transfer**

Finally the temperature equation is derived for nodes on the top boundary, including a term for convective heat transfer to the top:

**Application in Excel**

The above temperature equations define the temperature to be calculated for the boundary nodes, and the nodes on the inside, which could now be programmed into Excel. The equations are programmed for each cell as before, but part of an IF-statement linked to a cell with a TRUE or FALSE value. If the value is TRUE then the formula will be executed, else the value of the cell will be an arbitrary value (in this case 100). This is reset the iterative loops should an error occur and propagate to all cells.

The results above can be interpreted considering the colors provided by the conditional formatting. The warmer colors at the bottom show the body is hotter because of the high fixed temperature, and colder to the top due to the convective cooling. It is also cooler to the left due to the negative heat flux in the x-direction (heat loss to the left).