Modeling of Heat Transfer in 2D SLAB

[1]UKOBA, KINGSLEY; [2]ADGIDZI, D; [1]KOMOLAFE, O; [1]OLORUNFEMI, G

[1]Engineering Materials Development Institute, Akure, Nigeria

[2]Federal University of Technology, Minna, Nigeria

E-mail: ukobaking@yahoo.com

ABSTRACT

Heat transfer in slab is a very important factor especially in the Africa continent. This work modeled the heat transfer in a 2D Slab. It used the Finite Difference Method (FDM) technique. Lines of code were written in Octave and can also be executed in Mat Lab and graph generated.

KEYWORDS: finite difference method (FDM), heat, 2d slab, modeling

1.0 INTRODUCTION

A concrete slab is a common structural element of modern buildings. Horizontal slabs of steel reinforced concrete, typically between 100 and 500 millimeters thick, are most often used to construct floors and ceilings, while thinner slabs are also used for exterior paving. In many domestic and industrial buildings a thick concrete slab, supported on foundations or directly on the subsoil, is used to construct the ground floor of a building. These can either be "ground-bearing" or "suspended" slabs. In high rise buildings and skyscrapers, thinner, pre-cast concrete slabs are slung between the steel frames to form the floors and ceilings on each level (Wikipedia Concrete slab, 2013).

A continuum is a body of matter (solid, liquid or gas) or simply a region of space in which a particular phenomenon is occurring. The solution of linear and non – linear boundary value problems (BVPs) can be obtained through: direct integration exact solutions such as Separation of variables, Similarity solutions, Fourier and Laplace transforms and Approximate solutions such as Perturbation, Power series, Probability schemes, Method of weighted residuals, MWR, Ritz method, Finite difference method/technique (FDM), Finite element method (FEM).

Finite difference method (FDM) evaluates the values of field variables at the node points (pointwise approximation). The purpose of discretization is to obtain a problem that can be solved by a finite procedure.

Finite difference discretization: Basically, there are three (3) methods for discretizing a spatial domain, viz: Finite difference, Finite volumes and Finite elements.

Ashim, 2007 looked at similar work where he modeled 1D slab but used Comsol Multiphysic. He used specific dimensions for the 1D slab. In this work, the finite difference method (FDM) was used and coding was done in Octave and can also be run on MatLab software.

1.1 METHODOLOGY

Consider the heat transfer equation (2-D problem):

$$\frac{\partial T}{\partial t} = \alpha \nabla^2 T \qquad 1.0$$

Below is a slab initially at T_0 and having boundary conditions as shown below:

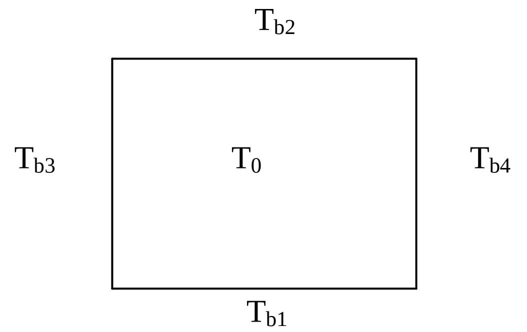


Figure 1: Heat transfer in slab

Discretization of slab:

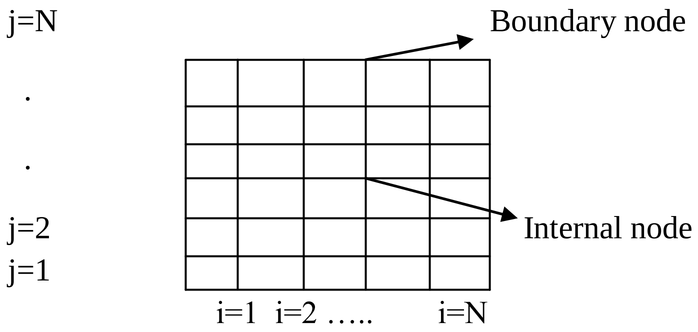


Figure 2: Discretized slab

Based on the discretization above, Equation 10 can be written as:

$$\frac{dT_{i,j}}{dt} = \alpha\left(\left.\frac{\partial^2 T}{\partial x^2}\right|_{i,j} + \left.\frac{\partial^2 T}{\partial y^2}\right|_{i,j}\right) \qquad 1.1$$

Graphically:

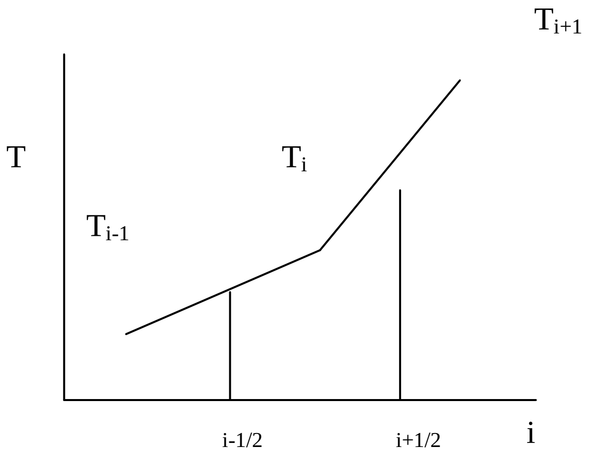


Figure 3: Graphical representation

For x-direction,

$$\frac{\partial^2 T}{\partial x^2} \cong \frac{\left.\frac{\partial T}{\partial x}\right|_{i+1/2} - \left.\frac{\partial T}{\partial x}\right|_{i-1/2}}{\Delta x}$$

$$\cong \frac{\frac{T_{i+1,j} - T_{i,j}}{\Delta x} - \frac{T_{i,j} - T_{i-1,j}}{\Delta x}}{\Delta x}$$

$$= \frac{T_{i+1,j} - 2T_{i,j} + T_{i-1,j}}{(\Delta x)^2} \qquad 1.2$$

Similarly, in the y-direction:

$$\frac{\partial^2 T}{\partial y^2} = \frac{T_{i,j+1} - 2T_{i,j} + T_{i,j-1}}{(\Delta y)^2} \qquad 1.3$$

Substituting Equations 1.2 and 1.3 into 1.1 gives:

$$\frac{dT_{i,j}}{dt} = \alpha\left(\frac{T_{i+1,j} - 2T_{i,j} + T_{i-1,j}}{(\Delta x)^2} + \frac{T_{i,j+1} - 2T_{i,j} + T_{i,j-1}}{(\Delta y)^2}\right) \qquad 1.4$$

Let $k = N_x(j-1) + i$ 1.5

The nodal temperatures can be represented in the mesh as follows:

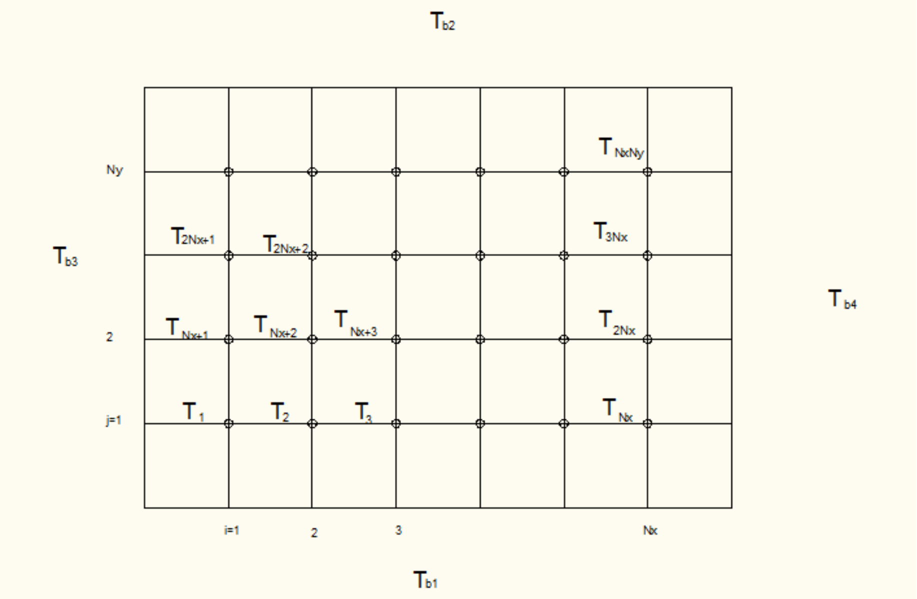


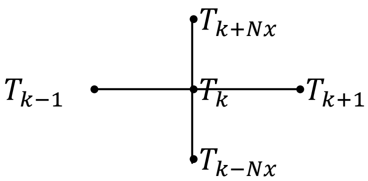
Figure 4: Nodal temperatures in discretized slab

Hence in terms of k:

$$\frac{dT_k}{dt} = \alpha\left(\frac{T_{k+1} - 2T_k + T_{k-1}}{(\Delta x)^2} + \frac{T_{k+Nx} - 2T_k + T_{k-Nx}}{(\Delta y)^2}\right) \qquad 1.6a$$

For $\Delta x = \Delta y$,

$$\frac{dT_k}{dt} = \frac{\alpha}{(\Delta x)^2}(T_{k+1} - 2T_k + T_{k-1} + T_{k+Nx} - 2T_k + T_{k-Nx}) \qquad 1.6b$$



Nodes not on the border are evaluated using Equation 16b. For boundary nodes:

Border 1: $T_{k-Nx} = T_{b1}$

Border 2: $T_{k+Nx} = T_{b2}$

Border 3: $T_{k-1} = T_{b3}$

Border 4: $T_{k+1} = T_{b4}$

If N_x=3 and N_y=3:

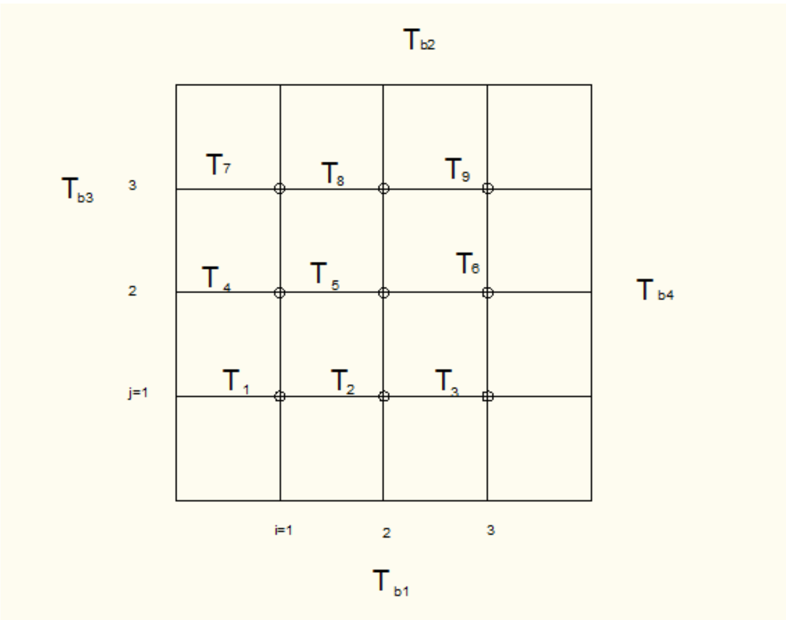


Figure 5: Nodal temperatures

The nodal temperatures are as follows:

$$\frac{dT_1}{dt} = \frac{\alpha}{(\Delta x)^2}(T_{b3} + T_{b1} + T_2 + T_4 - 4T_1)$$

$$OR \quad \frac{dT_1}{dt} = \frac{\alpha}{(\Delta x)^2}(-4T_1 + T_2 + T_4) + \frac{\alpha}{(\Delta x)^2}(T_{b1} + T_{b3})$$

$$\frac{dT_2}{dt} = \frac{\alpha}{(\Delta x)^2}(T_1 - 4T_2 + T_3 + T_5) + \frac{\alpha}{(\Delta x)^2}T_{b1}$$

$$\frac{dT_3}{dt} = \frac{\alpha}{(\Delta x)^2}(T_2 - 4T_3 + T_6) + \frac{\alpha}{(\Delta x)^2}(T_{b1} + T_{b4})$$

. . .

. . .

. . .

$$\frac{dT_9}{dt} = \frac{\alpha}{(\Delta x)^2}(T_6 + T_8 - 4T_9) + \frac{\alpha}{(\Delta x)^2}(T_{b2} + T_{b4})$$

Hence in matrix notation,

$$\frac{d\underline{T}}{dt} = \underline{A}\,\underline{T} + \underline{b} \qquad 1.7a$$

$$\frac{d}{dt}\begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \\ T_6 \\ T_7 \\ T_8 \\ T_9 \end{bmatrix} = \frac{\alpha}{(\Delta x)^2}\begin{bmatrix} -4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & -4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & -4 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & -4 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & -4 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & -4 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & -4 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & -4 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & -4 \end{bmatrix}\begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \\ T_6 \\ T_7 \\ T_8 \\ T_9 \end{bmatrix} + \frac{\alpha}{(\Delta x)^2}\begin{bmatrix} T_{b1} + T_{b3} \\ T_{b1} \\ T_{b1} + T_{b4} \\ T_{b3} \\ 0 \\ T_{b4} \\ T_{b2} + T_{b3} \\ T_{b2} \\ T_{b2} + T_{b4} \end{bmatrix} \qquad 1.7b$$

The following is the Octave codes for solving it:

1.1.1 The RHS code:

```
function f = RHS(t,T)

   % Calculates RHS of heat equation

C= 400 ;  % Specific heat cap J/kg K

rho = 7800;  % density   kg/m^3

K = 40;    % conductivity W/(m^2 K)

L = 1;

Nx=20;

Ny=20;

N = Nx * Ny ;
```

```
h = L/(Nx+1);
alpha = K/(C*rho);
beta1 = alpha /h^2;
Tb1=400.0;
Tb2=273.0;
Tb3=150.0;
Tb4=500.0;
for k = 1:N
   if k == 1  % bot left corner
   f(k)=beta1*(Tb3+T(k+1) -4.0*T(k) +T(k+Nx) +Tb1);
   elseif k == Nx  % bot rt corner
   f(k)=beta1 * (T(k-1)+Tb4 -4.0*T(k) +T(k+Nx) +Tb1);
   elseif k == N  % top rt corner
   f(k)=beta1 * (T(k-1)+Tb4 -4.0*T(k) +Tb2 +T(k-Nx));
   elseif k == N - Nx +1  % top left corner
   f(k)=beta1 * (Tb3+T(k+1) -4.0*T(k) +Tb2 +T(k-Nx));
   elseif k < Nx   % bot row
   f(k)=beta1 * (T(k-1)+T(k+1) -4.0*T(k) +T(k+Nx) +Tb1);
   elseif k > N - Nx  % top row
   f(k)=beta1 * (T(k-1)+T(k+1) -4.0*T(k) +Tb2 +T(k-Nx));
   elseif mod(k,Nx) == 0  % rightmost column
   f(k)=beta1 * (T(k-1)+Tb4 -4.0*T(k) +T(k+Nx) +T(k-Nx));
   elseif mod(k,Nx) ==1 % leftmost column
   f(k)=beta1 * (Tb3+T(k+1) -4.0*T(k) +T(k+Nx) +T(k-Nx));
   else
   f(k)=beta1 * (T(k-1)+T(k+1) -4.0*T(k) +T(k+Nx) +T(k-Nx));
   end % end if block
end  % end for loop
f = f' ;
end  % end whole function
```

The heat codes:

```
clc
clear all
Nx = 20;
Ny = 20;
T0 = 298.0*ones(1,Nx*Ny);
Ntimes= 11;
tspan = linspace(0,60,Ntimes);
[t, T] = ode45(@RHS, tspan, T0)
plot(t,T)
L=1.0;
dx=L/(Nx+1);
[X,Y] = meshgrid(dx:dx:(L-dx), (L-dx):-dx:dx);   % needed below to plot
for t=1:Ntimes
   for i=1:Nx
       for j =1:Ny
           k = Nx*(j-1) + i;
           U(j,i) = T(t,k);
       end
   end
contourf(X,Y,U)
pause(2)
end
```

Results:

The solution is obtained by running the heat code.

```
heat
t =

     0
     6
    12
    18
    24
    30
    36
    42
    48
    54
```

ISSN 2224-5804 (Paper) ISSN 2225-0522 (Online)
Vol.4, No.9, 2014

60

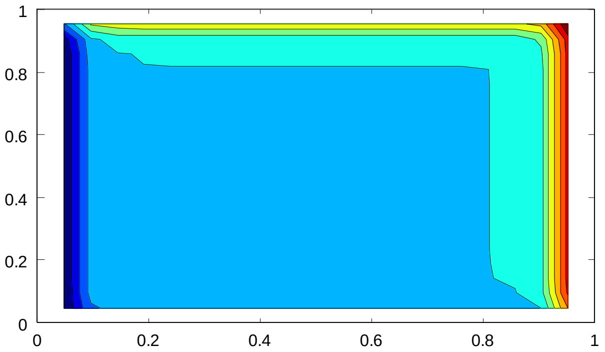


Figure 6: Graph generated

1.2 CONCLUSION

This work was able to model successfully 2D Slab using the Finite Difference method and coded in Octave and can also run in Matlab. This will give more information to in the area of slab and heat transfer.

REFERENCE

1. Ashim, D (2007). Modeling 1D slab in Comsol Multiphysics. Cornell University authored by Vineet Rakesh and Frank Kung.
2. Wikipedia concrete slab (2013). Retrieved from http://en.wikipedia.org/wiki/Concrete_slab on 4th September, 2013.