To: David Willy

From: Martín Dorantes

Date: 10/10/20

Re: Computer Project 2

Problem Statement

The purpose of this project was to compute a simple 2D conduction problem within MATLAB. This project was split into parts A & B. Part A asked the student to plot the temperature profile through a 1m by 1m object using surf and heatmap plots and calculate heat flows at each boundary to ensure all surfaces add to zero. Part B asked the student to build off part A to change mesh sizes from 25, 36, 49, and 64. All plots and references used are shown within this report.

Approach

This project closely follows the 9-node problem from class, which was used as a reference. The given Figure 1 was used to depict where the node locations are within the 16-node system. The right and bottom boundaries are adiabatic, which is why the assumption of no heat flow was made. The left and top boundaries are theorized to yield zero, given that they produce heat flow. The main functions used were for loops and if statements to utilize a "live" code where the only changing variable is the number of nodes within the heatmap to fit the problem statement.

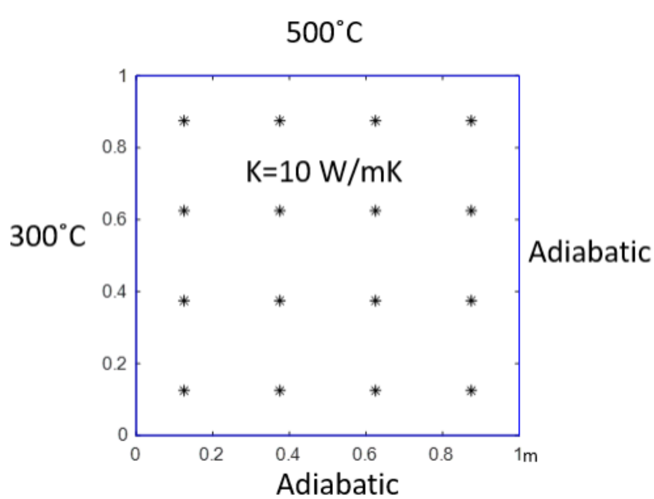

Figure 1: 16 node locations

Results

The Surf plot in Figure 2 shows the flux in heat flow through the system in both x and y directions as temperature transfers from the top boundary to the left boundary. The temperature change over the system is reasonable in the sense that the top boundary transfers more heat than the left, right, and bottom boundaries. Figure 3 represents the heatmap for the 16 nodes at each location from Figure 1.

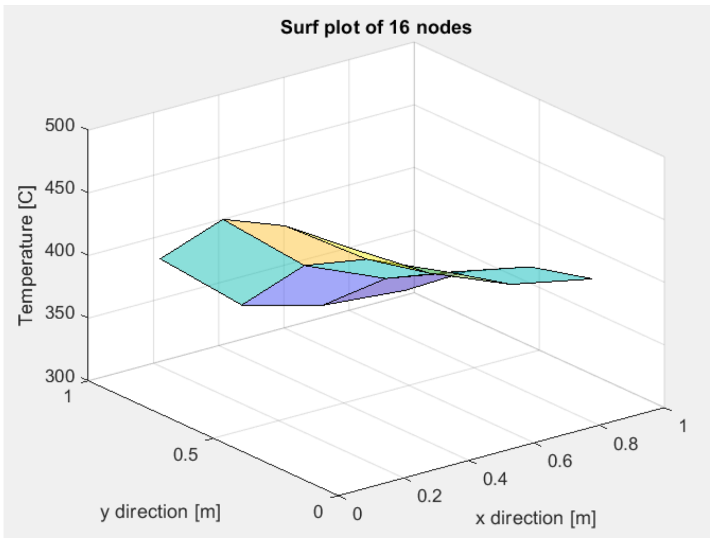

Figure 2: Surf Plot

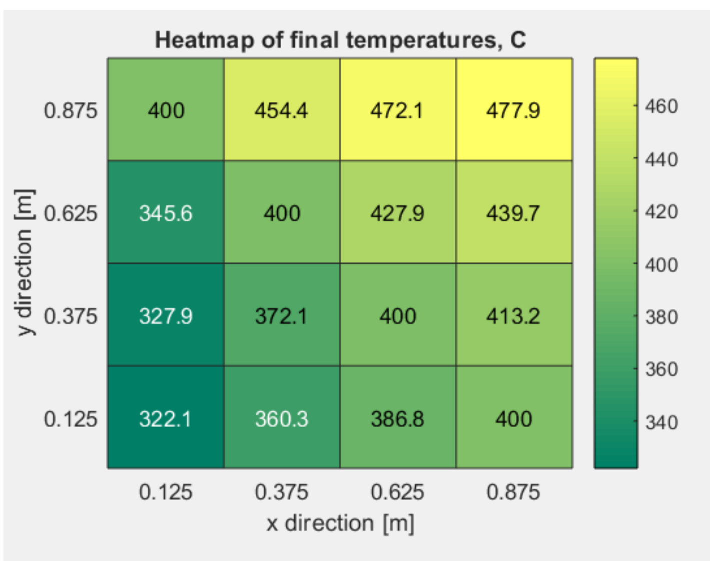

Figure 3: Heatmap for 16 nodes

For part 2 of part A, the task was to find total heat flow. Since the right and bottom boundaries are adiabatic, the assumption of zero heat flow was made. For the upper and left boundaries, the heat flows were almost absolute in each value, seen in Figure 4. The minuscule sum of the heat flow can be regarded as zero heat flow because the net change is so small to report as significant.

qleft	-3.9118e+03
qtop	3.9118e+03
qtot	-1.0004e-11

Figure 4: Total Heat Flow

In part B, the node mesh sizes changed to show the heatmap of 25, 36, 49, and 64 node mesh sizes. As the number of nodes within the system increases, the more accurate the temperature profile is for the system at each node location, shown in Figure 5. The code is a useful tool because it does not limit the user to a certain amount of node mesh sizes. The user does not have to hand-calculate any temperatures across the profile, only change the number of node area, and can be used for any amount of node mesh sizes (10, 20, 100, etc.).

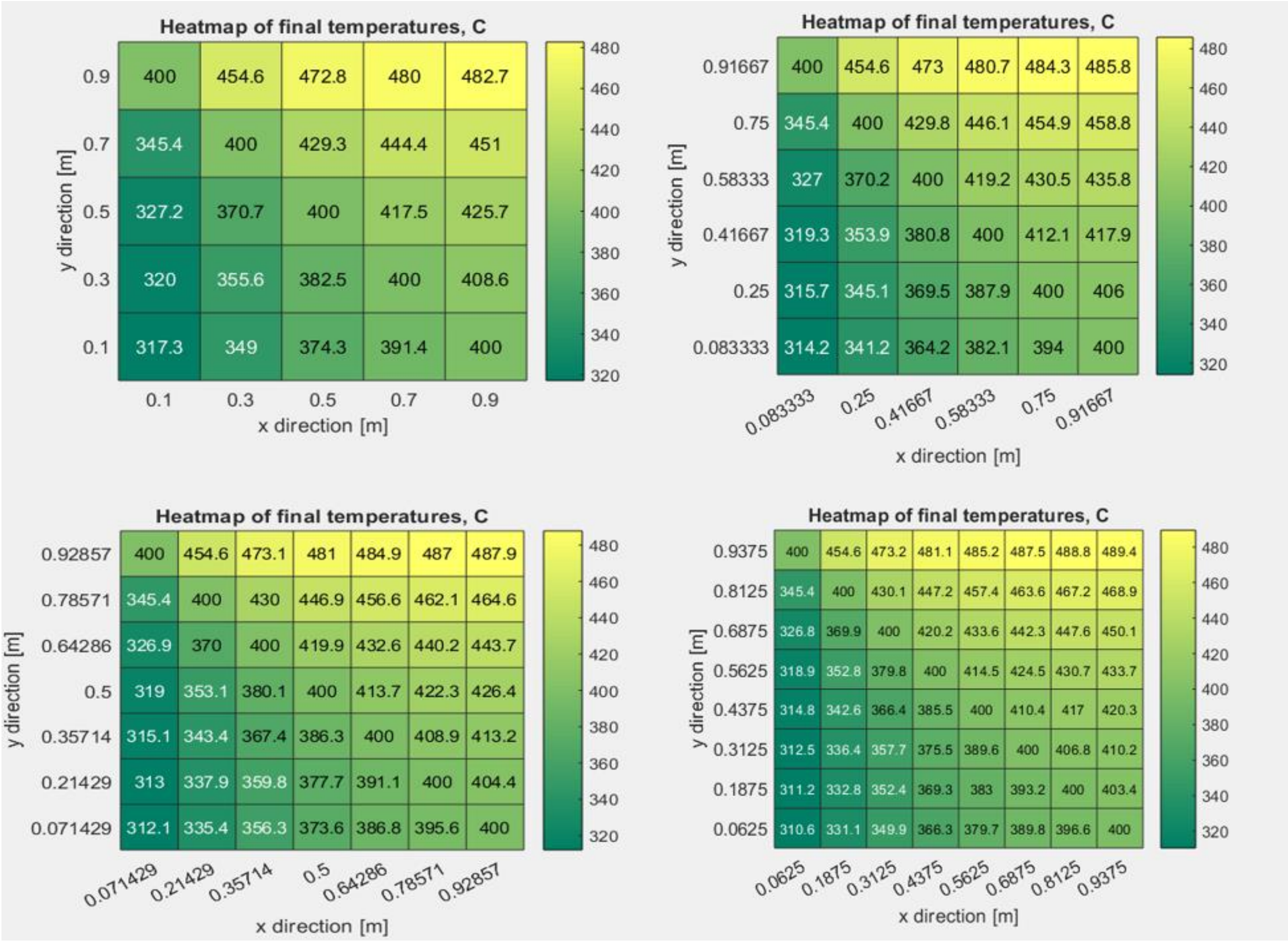

Figure 5: Heatmaps for 25, 36, 49, & 64 node mesh sizes

Conclusion

This project taught the importance of the heat flow from temperature boundaries and the map of how it flows through a system. It was interesting to see the different node mesh size heatmaps with different node sizes. The application of this project in the real world eliminates the unnecessary use of excess engineers on a team. If this project were for an engineering firm, there would be no need for an entire team to calculate the surf and heatmap plot due to programming skills taught in class.

Appendix

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%
%%Script:       CP2.m
%%
%%programmer:   Martin Dorantes
%%
%%date:         2020.10.10
%%
%%Returns:      16 node problem
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%

clear; clc;


k=10;                   %W/mK
Tbc1=500;               %C
Tbc2=300;               %C
n=4;                    % number of nodes in the x-direction
dx=1/n;
dy=1/n;
x=dx/2:dx:1-(dx/2);
y=1-(dy/2):-dy:dy/2;

[XX,YY]=meshgrid(x,y); % place nodes
X=XX';
Y=YY';

nodetype=zeros(n*n,1);
A=zeros(n,n);
b=zeros(n*n,1);

% figure(1)
% subplot(2,2,1)

for i=1:n*n
    if X(i)<dx && Y(i)>1-dy
        nodetype(i)=1;
        A(i,i)=-6;
        A(i,i+n)=1;
        A(i,i+1)=1;
```

```
        b(i)=-2*(Tbc1+Tbc2);
    elseif X(i)>dx && X(i)<1-dx && Y(i)>1-dy
        nodetype(i)=2;
        A(i,i)=-5;
        A(i,i-1)=1;
        A(i,i+n)=1;
        A(i,i+1)=1;
        b(i)=-2*Tbc1;
    elseif X(i)>1-dx && Y(i)>1-dy
        nodetype(i)=3;
        A(i,i)=-4;
        A(i,i-1)=1;
        A(i,i+n)=1;
        b(i)=-2*Tbc1;
    elseif X(i)<dx && Y(i)>dy && Y(i)<1-dy
        nodetype(i)=4;
        A(i,i)=-5;
        A(i,i-n)=1;
        A(i,i+n)=1;
        A(i,i+1)=1;
        b(i)=-2*Tbc2;
    elseif X(i)>1-dx && X(i)<1 && Y(i)>dy && Y(i)<1-dy
        nodetype(i)=6;
        A(i,i)=-3;
        A(i,i-n)=1;
        A(i,i+n)=1;
        A(i,i-1)=1;
        b(i)=0;
    elseif X(i)<dx && Y(i)<dy
        nodetype(i)=7;
        A(i,i)=-4;
        A(i,i-n)=1;
        A(i,i+1)=1;
        b(i)=-2*Tbc2;
    elseif X(i)>dx && X(i)<1-dx && Y(i)<dy
        nodetype(i)=8;
        A(i,i)=-3;
        A(i,i-n)=1;
        A(i,i-1)=1;
        A(i,i+1)=1;
        b(i)=0;
    elseif X(i)>1-dx && Y(i)<dy
        nodetype(i)=9;
        A(i,i)=-2;
```

```
        A(i,i-n)=1;
        A(i,i-1)=1;
        b(i)=0;
    else
        nodetype(i)=5;
        A(i,i)=-4;
        A(i,i-n)=1;
        A(i,i-1)=1;
        A(i,i+1)=1;
        A(i,i+n)=1;
        b(i)=0;
    end
end


T=A\b;
node_rs=reshape(nodetype,n,n)';
Treshape=reshape(T,n,n)';

%Heat map template, before temperature
% figure(2);
% h=heatmap(x,y,node_rs,'colormap',summer);
% title('Heatmap of final temperatures')
% xlabel('x direction [m]')
% ylabel('y direction [m]')

figure(3);
h=heatmap(x,y,Treshape,'colormap',summer);
title('Heatmap of final temperatures, C')
xlabel('x direction [m]')
ylabel('y direction [m]')

%Surf
figure(4);
T2=surf(X,Y,Treshape,'FaceAlpha',0.5);
title('Surf plot of 16 nodes')
xlabel('x direction [m]')
ylabel('y direction [m]')
zlabel('Temperature [C]')

%Heat flow
qtop=sum(k*2*(Tbc1-Treshape(1,:)));
qleft=sum(k*2*(Tbc2-Treshape(:,1)));
qtot=qtop+qleft;
```

```
disp(qtot);
```