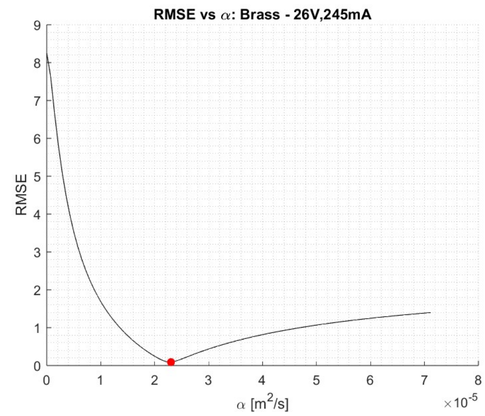
ASEN 3802 Lab 2 Part 3

Andrew Patella, Lauren Lajoie, Skylar Harris, Yaseen Mustapha

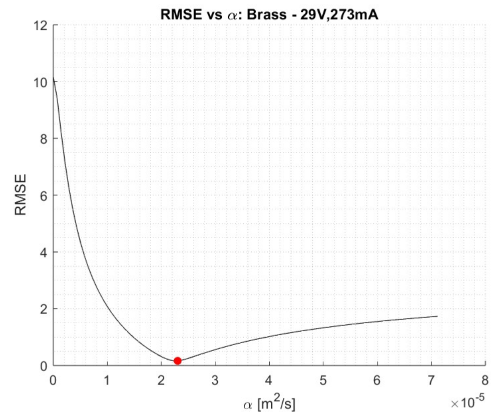
March 31, 2025

1 Task 1: Variance in Thermal Diffusivity

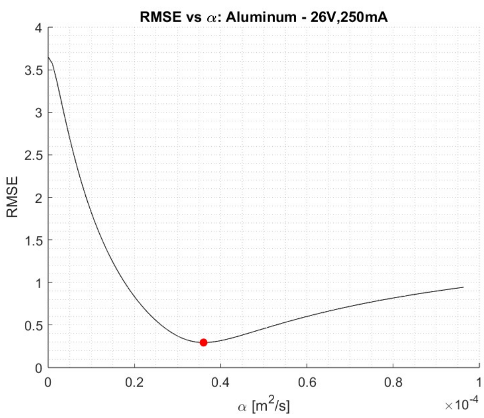
1.1 Plot and Table



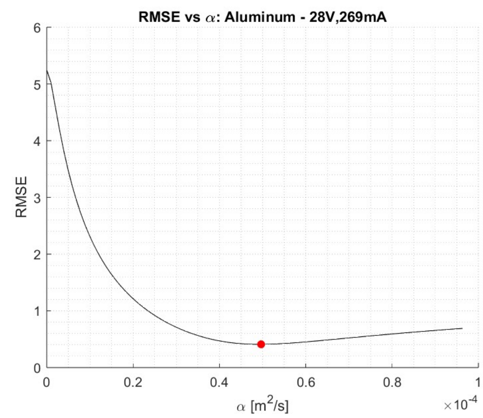
(a) Brass at 26V



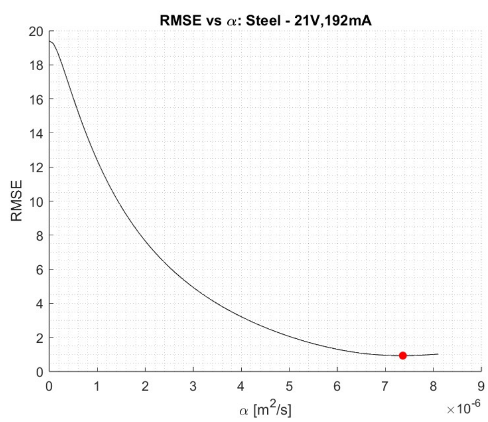
(b) Brass at 29V



(c) Aluminum at 26V



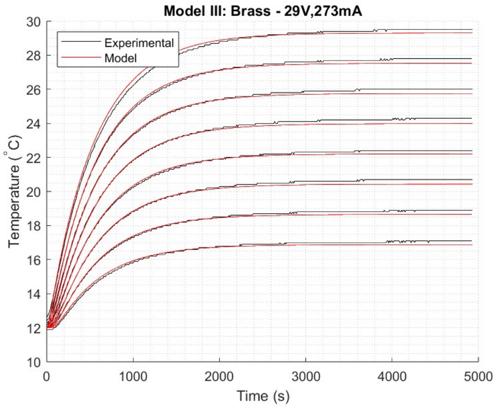
(d) Aluminum at 28V



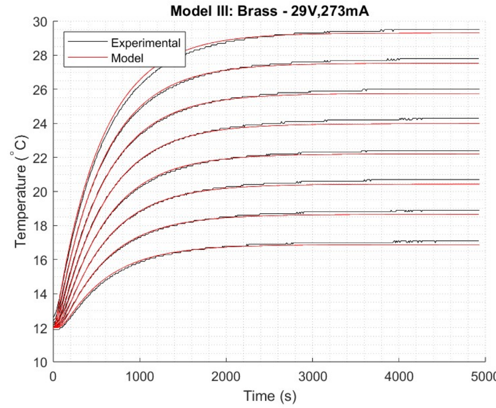
(e) Steel

Figure 1: Alpha Vs. Root Mean Square Error for the 5 different cases

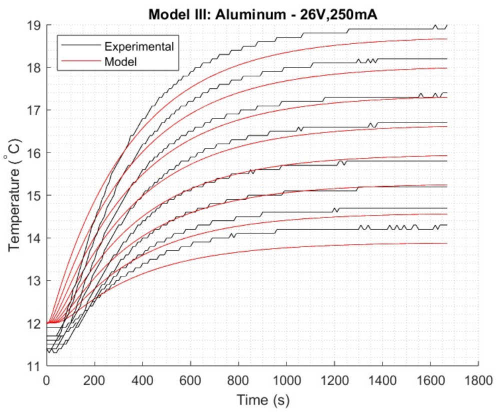
Material	α $[\frac{m^2}{s}]$	α_{adj} $[\frac{m^2}{s}]$
Steel 21V,192mA	4.05×10^{-6}	7.36×10^{-6}
Brass 26V,245mA	3.56×10^{-5}	2.30×10^{-5}
Brass 29V,273mA	3.56×10^{-5}	2.30×10^{-5}
Aluminum 26V,250mA	4.82×10^{-5}	3.60×10^{-5}
Aluminum 28V,269mA	4.82×10^{-5}	4.97×10^{-5}



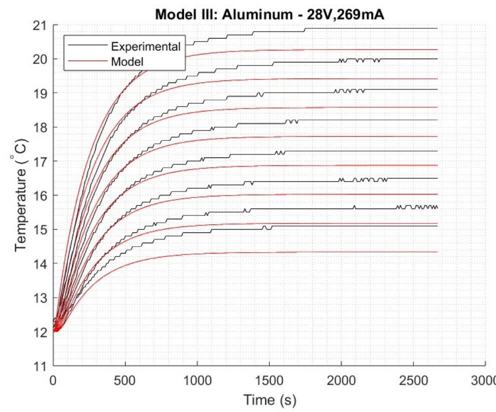
(a) Brass at 26V



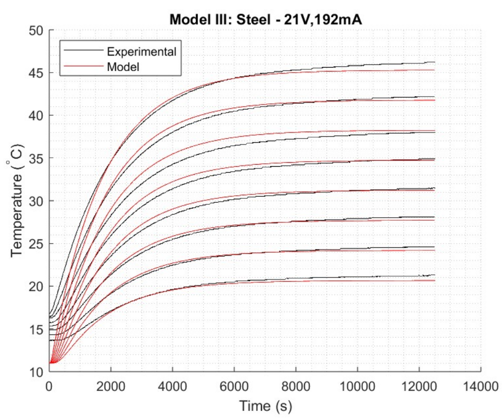
(b) Brass at 29V



(c) Aluminum at 26V



(d) Aluminum at 28V



(e) Steel

Figure 2: Plot of Model 3 against the experimental data

1.2 Discussion

To find our adjusted value for the thermal diffusivity, we utilized the root mean square error. This approach tries multiple values for α and determines which one decreases the error overall for all the channels at each tie. For each material, we found the root mean square error of the Temperature vs. Time curve for various values of thermal diffusivity. We found the error of 100 different thermal diffusivity coefficients from 0 to 2α. We chose the thermal diffusivity of each case that resulted in the smallest root mean square error because this value would result in the closest model of reality possible. Varying the thermal diffusivity changes the rate at which the model reaches steady state. A lower value of α causes a curve to reach steady state slower than a higher value of α. With our method, we were able to find α_{adj} that best matches the experimental data; as shown in Figure 2. Figure 1 illustrates our method for finding the minimum root mean square error in varying the thermal diffusivity. Figure 2 shows the application of the said thermal diffusivity compared to the experimental data. We used a root mean square for our model to take into account all thermocouples at all times. We wanted to minimize the error everywhere instead of just the error at a particular place on the rod or at a particular moment in time. This offers a more accurate overall model instead of focusing on just one thermocouple. The adjusted thermal diffusivity coefficients found differed for each case. Steel's thermal diffusivity was found to be nearly twice the given value. Brass's was identical for both cases and less than the given value. In the case of aluminum, we found that the thermal diffusivity in the 26V case was less than the given value, while the 28V case was nearly identical to the given value. Overall for each adjusted value we found for thermal diffusivity the value was in the same order of magnitude and seemed reasonable when compared to the values given in the table. We suspect that the deviation between the given and analytical thermal diffusivity coefficients is due to some of the simplifications we made to derive this model, such as the assumption of constant properties.

2 Task 2: Time to steady state

Material	t_{ss} $[s]$	F_o No adjustment	F_o With Adjustment
Steel 21V, 192mA	7,300	1.3281	2.4136
Brass 26V, 245mA	1,790	2.8629	1.8495
Brass 29V, 273mA	2,000	3.1988	2.0664
Aluminum 26V, 250mA	880	1.9051	1.4231
Aluminum 28V, 269mA	910	1.9700	2.0317

To find the approximate time to steady state, we used MATLAB to find where the temperature of the last thermocouple is within 2.5% of the final temperature in the experimental data. To verify our results for the steady state time we also used the MATLAB function grad to find where the change in temperature is zero as well as compared visually with our plots. We then used this steady-state time to find the Fourier numbers using $\frac{\alpha t_{ss}}{L^2}$. We found our Fourier number with the original thermal diffusivity and then the new Fourier number with adjusted thermal diffusivity. We then put them into the same table for easier comparison.

2.1 Discussion

The Fourier number is important because when it is greater than 0.2, a one-term approximation is appropriate. The Fourier number is physically significant because it represents the heat conduction versus the heat storage of a material. This means that the higher the Fourier number, the slower the heat transfer. This makes sense because a material with higher heat conduction will have more rapid heat transfer, while a material with higher heat storage will have a slower rate of heat transfer.

We saw the Fourier number change with the adjusted thermal diffusivity. When our adjusted thermal diffusivity increased, our Fourier number increased. This makes sense because the Fourier number is in direct relationship to the thermal diffusivity. For each case, we saw a different change in the Fourier number. We would expect aluminum to have the lowest Fourier number because it has the highest thermal conductivity and therefore time to steady state would be lower. We found that steel had the highest Fourier number. We think this propagates from our code to find the adjusted thermal diffusivity, where our main goal was just to match the experimental data. This process may have led to Fourier numbers that are not the most accurate representation of the physical materials. For our adjusted thermal diffusivity for steel we saw an increase in the Fourier number, which would physically represent more rapid heat transfer. For brass, we calculated the same adjusted thermal diffusivity for each case. The adjusted thermal diffusivity smaller than the original value and thus the Fourier number decreases for both cases. This physically represents a more slow heat transfer due to a decrease in thermal diffusivity. Finally, for aluminum, we found two different thermal diffusivity coefficients for the two cases. For the 26 volt case we found that the adjusted thermal diffusivity decreased from the original value. Therefore, the Fourier number also decreased. This physically represented that with the adjusted thermal diffusivity, the heat transfer would take longer. For the 28 volt case, we found that the adjusted thermal diffusivity increased, therefore the Fourier number also increased. Physically for the 28 volt case with the new thermal diffusivity the heat transfer would happen more rapidly.

Time to steady state is affected by thermophysical properties of the material as well as the geometry of the object. A longer length with cause a longer time for the furthest thermocouple to reach steady state because the heat transfer has to travel further. Time to steady state is also affected by the thermal diffusivity of the material. A higher diffusivity means a shorter time to steady state. Physically, this makes sense because the more the heat can spread (diffuse) the quicker the material can reach steady state. Finally, within the thermal diffusivity equation, specific heat, density and thermal conductivity also affect the time to steady state. To decrease time to steady state, from $\alpha = \frac{k}{c_p \rho}$ we can see how each property effects the thermal diffusivity. We found that increasing thermal conductivity, decreasing density and decreasing specific heat capacity all increase thermal diffusivity which will then decrease the time to steady state.

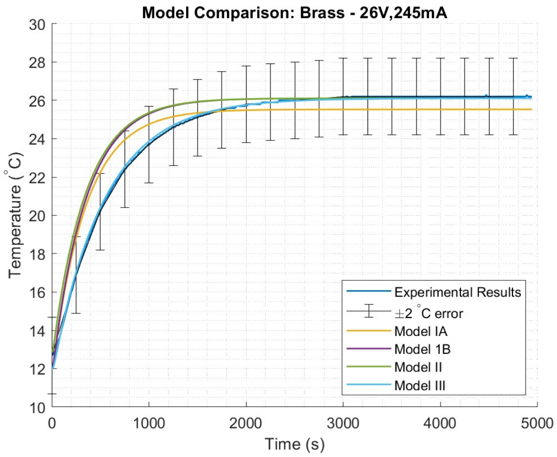
$$F_o = \frac{\alpha t_{ss}}{L^2} \Rightarrow t_{ss} = \frac{F_o L^2}{\alpha}$$

$$t_{ss} \propto \frac{L^2}{\alpha}$$

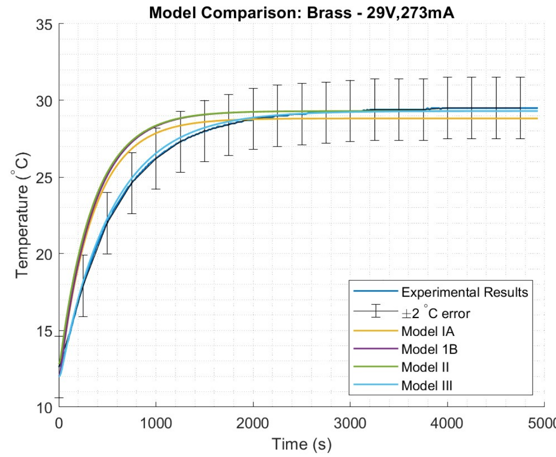
With the Fourier Number being held constant, increasing α will decrease the time to steady state at a linear rate. This is because the thermal diffusivity is a measure of how quickly heat can spread through a material, so if heat can spread quicker, then the time to steady state will decrease. Since they are linearly related, doubling α will result in half the time to steady state. The length of the bar has a proportional quadratic impact on the time to steady state. This makes sense as well. If the bar is longer, the heat will take longer to distribute along the bar. Due to the quadratic relationship, doubling the length of the bar will quadruple the time to steady state.

3 Task 3: Model Validation and Application

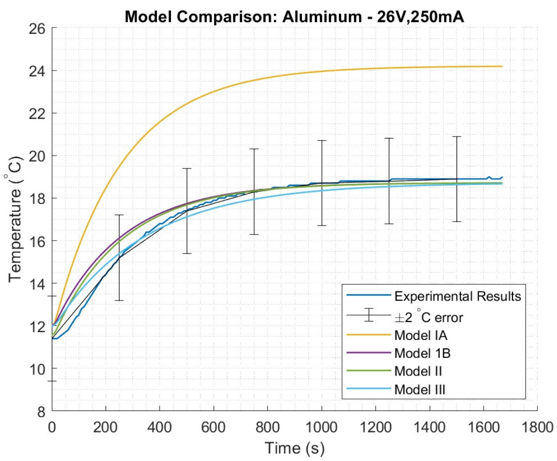
3.1 Plot



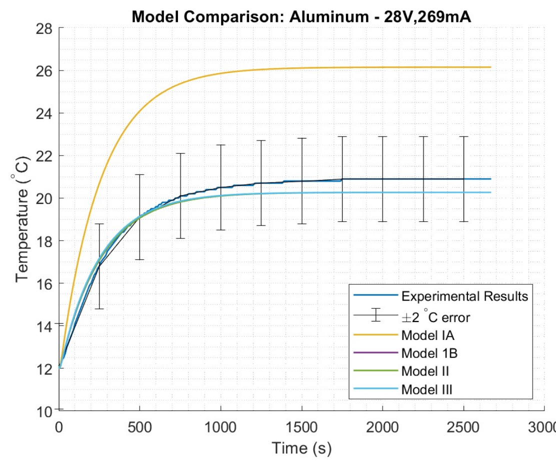
(a) Brass at 26V



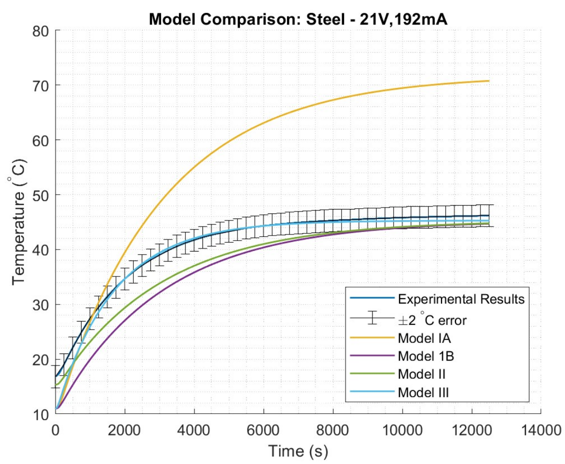
(b) Brass at 29V



(c) Aluminum at 26V



(d) Aluminum at 28V



(e) Steel

Figure 3: Alpha Vs. Root Mean Square Error for the 5 different cases

3.2 Discussion

There are three main phases of the experiment that need to be modeled: The beginning heat distribution and the beginning of the warming, the main heating phase where the most change in temperature occurred and the steady state at the end. Each of the models does a better job of predicting the different phases. The beginning steady state is best modeled by Model II, which takes into account the initial heat distribution. The model that does the best in the heating phase is the Model III, which takes into account the actual thermal properties for the metal and improves the estimate of α to match reality closer. All of the models perform about the same for the end steady state since the exponential dominates and the rate of change decreases significantly so there are fewer sources of error from other constants. The results from the plots match this hypothesis. When looking at the plots from varying our models we were able to see distinct improvements in the regions we were trying to correct, like model II correcting for the inital state condition The models all gave us the expected improvement in our plots.

The best model to use for modeling the time to steady state would be Model III. This model more accurately matches the experimental data with α because of how the model is derived. Since α matches more accurately, the Fourier number will match better. Since the time to steady state is most dependent on the Fourier number, this model will most accurately describe the time to steady state. It has some more error on the beginning by that is small error proportional to the change in time to steady state. The worst model to use is model IA, which neglects any improvements in slope, and initial condition. We can see this result in our plots of each model with the experimental data. We can see that our initial model, Model 1A was outside of the error bars of 2 degrees. But once we reached Model 3 every point was within the error bars.

4 Appendix

4.1 Contribution Report

Lauren Lajoie:
Part 1 - Tabled our results and wrote discussion.
Part 2 - Helped with code for tasks 1 and 2 and wrote discussion.
Part 3 - Found T_{ss} and F_0 for task 2 and wrote discussion.

Skylar Harris:
Part 1 - Helped with code, calculated the analytical slopes, and wrote discussion.
Part 2 - Wrote code for Tasks 1, 2, 3, and 4. Wrote discussion in the report.
Part 3 - Wrote code for Task 1 and wrote discussion in the report.

Yaseen Mustapha:
Part 1 - Worked on analysis and the report.
Part 2 - Did the derivation for task 1. Worked on analysis and the report.
Part 3 - Worked on analysis and the report.

Andrew Patella:
Part 1 - Calculated the steady-state slope from temperature data. Helped with analysis and data entry
Part 2 - Calculated the Fourier coefficients for the new model, did LaTeX typesetting, and wrote analysis. Helped with analysis and data entry.
Part 3 - Did error analysis graphs and LaTeX typesetting. Helped with analysis and data entry.

4.2 Code for Task 1

```
    %% ASEN 3802 Lab 2 Part 3 Task 1

clc;clear;close all;

al26 = readmatrix("Aluminum_26V_250mA");
al28 = readmatrix("Aluminum_28V_269mA");
br26 = readmatrix("Brass_26V_245mA");
br29 = readmatrix("Brass_29V_273mA");
st = readmatrix("Steel_21V_192mA");

dx = 0.0127; %m
x0 = 0.034925;
xpos = x0:dx:x0 + 0.1016;
% breaking data into channels
% cell row is which thermocouple
al26_c = cell(8,1);
al28_c = cell(8,1);
br26_c = cell(8,1);
br29_c = cell(8,1);
st_c = cell(8,1);
al26_t = al26(:,1);
al28_t = al28(:,1);
br26_t = br26(:,1);
br29_t = br29(:,1);
st_t = st(:,1);
for i = 1:8
al26_c{i} = al26(:,i+1);
al28_c{i} = al28(:,i+1);
br26_c{i} = br26(:,i+1);
```

```
br29_c{i} = br29(:,i+1);
st_c{i} = st(:,i+1);
end

al26_notime = al26(:,2:end);
al28_notime = al28(:,2:end);
br26_notime = br26(:,2:end);
br29_notime = br29(:,2:end);
st_notime = st(:,2:end);

x1 = 0.0349; % [m]
x2 = 0.0476; % [m]
x3 = 0.0603; % [m]
x4 = 0.073; % [m]
x5 = 0.0857; % [m]
x6 = 0.0984; % [m]
x7 = 0.1111; % [m]
x8 = 0.1238; % [m]
x = [x1,x2,x3,x4,x5,x6,x7,x8];

const_st.H_exp = 277; % [C/m]
const_st.T0 = 11; % [C]
const_st.k = 16.2; % [W/mK]
const_st.cp = 500; % [J/kgK]
const_st.rho = 8000; % [kg/m^3]
const_st.alpha = const_st.k/(const_st.rho*const_st.cp); % [m^2/s]
const_st.L = x8+0.0508; % [m]

const_br26.H_exp = 114; % [C/m]
const_br26.T0 = 12; % [C]
const_br26.k = 115; % [W/mK]
const_br26.cp = 380; % [J/kgK]
const_br26.rho = 8500; % [kg/m^3]
const_br26.alpha = const_br26.k/(const_br26.rho*const_br26.cp); % [m^2/s]
const_br26.L = x8+0.0508; % [m]

const_br29.H_exp = 139.8; % [C/m]
const_br29.T0 = 12; % [C]
const_br29.k = 115; % [W/mK]
const_br29.cp = 380; % [J/kgK]
const_br29.rho = 8500; % [kg/m^3]
const_br29.alpha = const_br29.k/(const_br29.rho*const_br29.cp); % [m^2/s]
const_br29.L = x8+0.0508; % [m]

const_al26.H_exp = 54.3; % [C/m]
const_al26.T0 = 12; % [C]
const_al26.k = 130; % [W/mK]
const_al26.cp = 960; % [J/kgK]
const_al26.rho = 2810; % [kg/m^3]
const_al26.alpha = const_al26.k/(const_al26.rho*const_al26.cp); % [m^2/s]
const_al26.L = x8+0.0508; % [m]

const_al28.H_exp = 66.8; % [C/m]
const_al28.T0 = 12; % [C]
const_al28.k = 130; % [W/mK]
const_al28.cp = 960; % [J/kgK]
const_al28.rho = 2810; % [kg/m^3]
const_al28.alpha = const_al28.k/(const_al28.rho*const_al28.cp); % [m^2/s]
```

```
const_al28.L = x8+0.0508; % [m]

alpha_br26 = linspace(0,2*const_br26.alpha,100);
alpha_br29 = linspace(0,2*const_br29.alpha,100);
alpha_al26 = linspace(0,2*const_al26.alpha,100);
alpha_al28 = linspace(0,2*const_al28.alpha,100);
alpha_st = linspace(0,2*const_st.alpha,100);

t_st = 0:10:12510; % [s]
t_al26 = 0:10:1670; % [s]
t_al28 = 0:10:2670; % [s]
t_br26 = 0:10:4940; % [s]
t_br29 = 0:10:4930; % [s]


for j = 1:length(alpha_st)
    const_st.alpha = alpha_st(j);
    for i = 1:length(t_st)
    u_st(i,:) = heatdistribution(t_st(i),x,const_st);
    end
    error_st(j) = mean(rmse(u_st, st_notime));
end
[~,idx] = min(error_st);
const_st.alpha = alpha_st(idx);

for j = 1:length(alpha_al26)
    const_al26.alpha = alpha_al26(j);
    for i = 1:length(t_al26)
    u_al26(i,:) = heatdistribution(t_al26(i),x,const_al26);
    end
    error_al26(j) = mean(rmse(u_al26, al26_notime));
end
[~,idx] = min(error_al26);
const_al26.alpha = alpha_al26(idx);

for j = 1:length(alpha_al28)
    const_al28.alpha = alpha_al28(j);
    for i = 1:length(t_al28)
    u_al28(i,:) = heatdistribution(t_al28(i),x,const_al28);
    end
    error_al28(j) = mean(rmse(u_al28, al28_notime));
end
[~,idx] = min(error_al28);
const_al28.alpha = alpha_al28(idx);

for j = 1:length(alpha_br26)
    const_br26.alpha = alpha_br26(j);
    for i = 1:length(t_br26)
    u_br26(i,:) = heatdistribution(t_br26(i),x,const_br26);
    end
    error_br26(j) = mean(rmse(u_br26, br26_notime));
end
[~,idx] = min(error_br26);
const_br26.alpha = alpha_br26(idx);

for j = 1:length(alpha_br29)
    const_br29.alpha = alpha_br29(j);
    for i = 1:length(t_br29)
```

```
    u_br29(i,:) = heatdistribution(t_br29(i),x,const_br29);
    end
    error_br29(j) = mean(rmse(u_br29, br29_notime));
end
[~,idx] = min(error_br29);
const_br29.alpha = alpha_br29(idx);

figure()
hold on
grid minor
plot(alpha_st,error_st,'k')
scatter(const_st.alpha,min(error_st),'filled','r')
xlabel('\alpha␣[m^2/s]')
ylabel("RMSE")
title('RMSE␣vs␣\alpha:␣Steel␣-␣21V,192mA')

figure()
hold on
grid minor
plot(alpha_al26,error_al26,'k')
scatter(const_al26.alpha,min(error_al26),'filled','r')
xlabel('\alpha␣[m^2/s]')
ylabel("RMSE")
title('RMSE␣vs␣\alpha:␣Aluminum␣-␣26V,250mA')

figure()
hold on
grid minor
plot(alpha_al28,error_al28,'k')
scatter(const_al28.alpha,min(error_al28),'filled','r')
xlabel('\alpha␣[m^2/s]')
ylabel("RMSE")
title('RMSE␣vs␣\alpha:␣Aluminum␣-␣28V,269mA')

figure()
hold on
grid minor
plot(alpha_br26,error_br26,'k')
scatter(const_br26.alpha,min(error_br26),'filled','r')
xlabel('\alpha␣[m^2/s]')
ylabel("RMSE")
title('RMSE␣vs␣\alpha:␣Brass␣-␣26V,245mA')

figure()
hold on
grid minor
plot(alpha_br29,error_br29,'k')
scatter(const_br29.alpha,min(error_br29),'filled','r')
xlabel('\alpha␣[m^2/s]')
ylabel("RMSE")
title('RMSE␣vs␣\alpha:␣Brass␣-␣29V,273mA')

for i = 1:length(t_st)
    u_st(i,:) = heatdistribution(t_st(i),x,const_st);
end
for i = 1:length(t_al26)
    u_al26(i,:) = heatdistribution(t_al26(i),x,const_al26);
end
```

```
for i = 1:length(t_al28)
    u_al28(i,:) = heatdistribution(t_al28(i),x,const_al28);
end
for i = 1:length(t_br26)
    u_br26(i,:) = heatdistribution(t_br26(i),x,const_br26);
end
for i = 1:length(t_br29)
    u_br29(i,:) = heatdistribution(t_br29(i),x,const_br29);
end

figure()
hold on
for i = 1:8
plot(al26_t,al26_c{i},'k')
plot(t_al26,u_al26(:,i),'r')
end
grid minor
xlabel("Time (s)")
ylabel('Temperature␣(^\circC)')
legend('Experimental','Model','','','','','','','','','','','','','','','Location','northwest')
title("Model III: Aluminum - 26V,250mA")

figure()
hold on
for i = 1:8
plot(al28_t,al28_c{i},'k')
plot(t_al28,u_al28(:,i),'r')
end
grid minor
xlabel("Time (s)")
ylabel('Temperature␣(^\circC)')
legend('Experimental','Model','','','','','','','','','','','','','','','Location','northwest')
title("Model III: Aluminum - 28V,269mA")

figure()
hold on
for i = 1:8
plot(br26_t,br26_c{i},'k')
plot(t_br26,u_br26(:,i),'r')
end
grid minor
xlabel("Time (s)")
ylabel('Temperature␣(^\circC)')
legend('Experimental','Model','','','','','','','','','','','','','','','Location','northwest')
title("Model III: Brass - 26V,245mA")

figure()
hold on
for i = 1:8
plot(br29_t,br29_c{i},'k')
plot(t_br29,u_br29(:,i),'r')
end
grid minor
xlabel("Time (s)")
ylabel('Temperature␣(^\circC)')
legend('Experimental','Model','','','','','','','','','','','','','','','Location','northwest')
title("Model III: Brass - 29V,273mA")
```

```
figure()
hold on
for i = 1:8
plot(st_t,st_c{i},'k')
plot(t_st,u_st(:,i),'r')
end
grid minor
xlabel("Time (s)")
ylabel('Temperature␣(^\circC)')
legend('Experimental','Model','','','','','','','','','','','','','','','Location','northwest')
title("Model III: Steel - 21V,192mA")


function u = heatdistribution(t,x,const)
    sum = [0,0,0,0,0,0,0,0];
    for n = 1:10
        b_n = (8*const.H_exp*const.L*(-1)^n)/((2*n-1)^2*pi^2);
        lambda_n = pi*(2*n-1)/(2*const.L);
        sum = sum + b_n.*sin(lambda_n.*x).*exp(-1*lambda_n^2*const.alpha*t);
    end
    u = const.T0+const.H_exp.*x + sum;
end
```

4.3 Code for Task 2

```
  close all;
clear;
clc;



%% Data
al26 = readmatrix('Aluminum_26V_250mA');
al28 = readmatrix('Aluminum_28V_269mA');
br26 = readmatrix('Brass_26V_245mA');
br29 = readmatrix('Brass_29V_273mA');
st = readmatrix('Steel_21V_192mA');

%% Constant
L = 0.1492;
const_st.k = 16.2; % [W/mK]
const_st.cp = 500; %J/kgK
const_st.rho = 8000; % [kg/m^3]
const_st.alpha = const_st.k/(const_st.rho*const_st.cp); % [m^2/s]
const_st.alphaadj = 7.36*10^-6;

const_br26.k = 115; % [W/mK]
const_br26.cp = 380; %J/kgK
const_br26.rho = 8500; % [kg/m^3]
const_br26.alpha = const_br26.k/(const_br26.rho*const_br26.cp); % [m^2/s]
const_br26.alphaadj = 2.30*10^-5;

const_br29.k = 115; % [W/mK]
const_br29.cp = 380; % [J/kgK]
const_br29.rho = 8500; % [kg/m^3]
const_br29.alpha = const_br29.k/(const_br29.rho*const_br29.cp); % [m^2/s]
const_br29.alphaadj = 2.3*10^-5;
```

```matlab

const_al26.k = 130; % [W/mK]
const_al26.cp = 960; % [J/kgK]
const_al26.rho = 2810; % [kg/m^3]
const_al26.alpha = const_al26.k/(const_al26.rho*const_al26.cp); % [m^2/s]
const_al26.alphaadj = 3.6*10^-5;

const_al28.k = 130; % [W/mK]
const_al28.cp = 960; % [J/kgK]
const_al28.rho = 2810; % [kg/m^3]
const_al28.alpha = const_al28.k/(const_al28.rho*const_al28.cp); % [m^2/s]
const_al28.alphaadj = 4.97*10^-5;

%% Original Alpha
final_st = 0.975*st(1252,9);
grad_st = gradient(st(:,9));
tss_st = st(find(st(:,9)>final_st,1));


final_al26 = 0.975*al26(168,9);
grad_al26 = gradient(al26(:,9));
tss_al26 = al26(find(al26(:,9)>final_al26,1));


final_al28 = 0.975*al28(268,9);
grad_al28 = gradient(al28(:,9));
tss_al28 = st(find(al28(:,9)>final_al28,1));


final_br26 = 0.975*br26(495,9);
grad_br26 = gradient(br26(:,9));
tss_br26 = br26(find(br26(:,9)>final_br26,1));


final_br29 = 0.975*br29(494,9);
grad_br29 = gradient(br29(:,9));
tss_br29 = br29(find(br29(:,9)>final_br29,1));


Fo_st = (const_st.alpha*tss_st)/L^2;
Fo_br26 = (const_br26.alpha*tss_br26)/L^2;
Fo_br29 = (const_br29.alpha*tss_br29)/L^2;
Fo_al26 = (const_al26.alpha*tss_al26)/L^2;
Fo_al28 = (const_al28.alpha*tss_al28)/L^2;

%% New Alpha

newFo_st = (const_st.alphaadj*tss_st)/L^2;
newFo_br26 = (const_br26.alphaadj*tss_br26)/L^2;
newFo_br29 = (const_br29.alphaadj*tss_br29)/L^2;
newFo_al26 = (const_al26.alphaadj*tss_al26)/L^2;
newFo_al28 = (const_al28.alphaadj*tss_al28)/L^2;
```

4.4 Code for Task 3

```matlab
    clc;clear;close all;

```

```matlab
al26 = readmatrix("Aluminum_26V_250mA");
al28 = readmatrix("Aluminum_28V_269mA");
br26 = readmatrix("Brass_26V_245mA");
br29 = readmatrix("Brass_29V_273mA");
st = readmatrix("Steel_21V_192mA");

%% Material and Geometric Properties
dx = 0.0127; %m
x0 = 0.034925;
xpos = x0:dx:x0 + 0.1016;
% breaking data into channels
% cell row is which thermocouple
al26_c = cell(8,1);
al28_c = cell(8,1);
br26_c = cell(8,1);
br29_c = cell(8,1);
st_c = cell(8,1);
al26_t = al26(:,1);
al28_t = al28(:,1);
br26_t = br26(:,1);
br29_t = br29(:,1);
st_t = st(:,1);
for i = 1:8
al26_c{i} = al26(:,i+1);
al28_c{i} = al28(:,i+1);
br26_c{i} = br26(:,i+1);
br29_c{i} = br29(:,i+1);
st_c{i} = st(:,i+1);
end

al26_notime = al26(:,2:end);
al28_notime = al28(:,2:end);
br26_notime = br26(:,2:end);
br29_notime = br29(:,2:end);
st_notime = st(:,2:end);

x1 = 0.0349; % [m]
x2 = 0.0476; % [m]
x3 = 0.0603; % [m]
x4 = 0.073; % [m]
x5 = 0.0857; % [m]
x6 = 0.0984; % [m]
x7 = 0.1111; % [m]
x8 = 0.1238; % [m]
x = [x1,x2,x3,x4,x5,x6,x7,x8];

const_st.H_exp = 277; % [C/m]
const_st.H_an = 491.2; % [C/m]
const_st.M_exp = 35.2; % [C/m]
const_st.T0 = 11; % [C]
const_st.k = 16.2; % [W/mK]
const_st.cp = 500; % [J/kgK]
const_st.rho = 8000; % [kg/m^3]
const_st.alpha = const_st.k/(const_st.rho*const_st.cp); % [m^2/s]
const_st.L = x8+0.0508; % [m]
const_st.alpha2 = 7.36e-6;

const_br26.H_exp = 114; % [C/m]
```

```
const_br26.H_an = 109.3; % [C/m]
const_br26.M_exp = 7; % [C/m]
const_br26.T0 = 12; % [C]
const_br26.k = 115; % [W/mK]
const_br26.cp = 380; % [J/kgK]
const_br26.rho = 8500; % [kg/m^3]
const_br26.alpha = const_br26.k/(const_br26.rho*const_br26.cp); % [m^2/s]
const_br26.L = x8+0.0508; % [m]
const_br26.alpha2 = 2.3e-5;


const_br29.H_exp = 139.8; % [C/m]
const_br29.H_an = 135.9; % [C/m]
const_br29.M_exp = 6.6; % [C/m]
const_br29.T0 = 12; % [C]
const_br29.k = 115; % [W/mK]
const_br29.cp = 380; % [J/kgK]
const_br29.rho = 8500; % [kg/m^3]
const_br29.alpha = const_br29.k/(const_br29.rho*const_br29.cp); % [m^2/s]
const_br29.L = x8+0.0508; % [m]
const_br29.alpha2 = 2.3e-5;

const_al26.H_exp = 54.3; % [C/m]
const_al26.H_an = 98.6; % [C/m]
const_al26.M_exp = -3.4; % [C/m]
const_al26.T0 = 12; % [C]
const_al26.k = 130; % [W/mK]
const_al26.cp = 960; % [J/kgK]
const_al26.rho = 2810; % [kg/m^3]
const_al26.alpha = const_al26.k/(const_al26.rho*const_al26.cp); % [m^2/s]
const_al26.L = x8+0.0508; % [m]
const_al26.alpha2 = 3.6e-5;

const_al28.H_exp = 66.8; % [C/m]
const_al28.H_an = 114.3; % [C/m]
const_al28.M_exp = 0.3; % [C/m]
const_al28.T0 = 12; % [C]
const_al28.k = 130; % [W/mK]
const_al28.cp = 960; % [J/kgK]
const_al28.rho = 2810; % [kg/m^3]
const_al28.alpha = const_al28.k/(const_al28.rho*const_al28.cp); % [m^2/s]
const_al28.L = x8+0.0508; % [m]
const_al28.alpha2 = 4.97e-5;

alpha_br26 = linspace(0,2*const_br26.alpha,100);
alpha_br29 = linspace(0,2*const_br29.alpha,100);
alpha_al26 = linspace(0,2*const_al26.alpha,100);
alpha_al28 = linspace(0,2*const_al28.alpha,100);
alpha_st = linspace(0,2*const_st.alpha,100);

t_st = 0:10:12510; % [s]
t_al26 = 0:10:1670; % [s]
t_al28 = 0:10:2670; % [s]
t_br26 = 0:10:4940; % [s]
t_br29 = 0:10:4930; % [s]

%% Finding Analytical Solutions

```

```
% Analytical solutions for model IA
for i = 1:length(t_st)
    u_stIA(i,:) = heatdistributionIA(t_st(i),x8,const_st);
end
for i = 1:length(t_al26)
    u_al26IA(i,:) = heatdistributionIA(t_al26(i),x8,const_al26);
end
for i = 1:length(t_al28)
    u_al28IA(i,:) = heatdistributionIA(t_al28(i),x8,const_al28);
end
for i = 1:length(t_br26)
    u_br26IA(i,:) = heatdistributionIA(t_br26(i),x8,const_br26);
end
for i = 1:length(t_br29)
    u_br29IA(i,:) = heatdistributionIA(t_br29(i),x8,const_br29);
end

% Analytical solutions for model IB
for i = 1:length(t_st)
    u_stIB(i,:) = heatdistributionIB(t_st(i),x8,const_st);
end
for i = 1:length(t_al26)
    u_al26IB(i,:) = heatdistributionIB(t_al26(i),x8,const_al26);
end
for i = 1:length(t_al28)
    u_al28IB(i,:) = heatdistributionIB(t_al28(i),x8,const_al28);
end
for i = 1:length(t_br26)
    u_br26IB(i,:) = heatdistributionIB(t_br26(i),x8,const_br26);
end
for i = 1:length(t_br29)
    u_br29IB(i,:) = heatdistributionIB(t_br29(i),x8,const_br29);
end

% Analytical Solutions for model II
for i = 1:length(t_st)
    u_stII(i,:) = heatdistributionII(t_st(i),x8,const_st);
end
for i = 1:length(t_al26)
    u_al26II(i,:) = heatdistributionII(t_al26(i),x8,const_al26);
end
for i = 1:length(t_al28)
    u_al28II(i,:) = heatdistributionII(t_al28(i),x8,const_al28);
end
for i = 1:length(t_br26)
    u_br26II(i,:) = heatdistributionII(t_br26(i),x8,const_br26);
end
for i = 1:length(t_br29)
    u_br29II(i,:) = heatdistributionII(t_br29(i),x8,const_br29);
end

% Analytical Solution for model III
for i = 1:length(t_st)
    u_stIII(i,:) = heatdistributionIII(t_st(i),x8,const_st);
end
for i = 1:length(t_al26)
    u_al26III(i,:) = heatdistributionIII(t_al26(i),x8,const_al26);
end
```

```
for i = 1:length(t_al28)
    u_al28III(i,:) = heatdistributionIII(t_al28(i),x8,const_al28);
end
for i = 1:length(t_br26)
    u_br26III(i,:) = heatdistributionIII(t_br26(i),x8,const_br26);
end
for i = 1:length(t_br29)
    u_br29III(i,:) = heatdistributionIII(t_br29(i),x8,const_br29);
end


%% Data analysis
dx = 0.0127; %m
x0 = 0.034925;
xpos = x0:dx:x0 + 0.1016;
% breaking data into channels
% cell row is which thermocouple
al26_c = cell(8,1);
al28_c = cell(8,1);
br26_c = cell(8,1);
br29_c = cell(8,1);
st_c = cell(8,1);
al26_t = al26(:,1);
al28_t = al28(:,1);
br26_t = br26(:,1);
br29_t = br29(:,1);
st_t = st(:,1);
for i = 1:8
al26_c{i} = al26(:,i+1);
al28_c{i} = al28(:,i+1);
br26_c{i} = br26(:,i+1);
br29_c{i} = br29(:,i+1);
st_c{i} = st(:,i+1);
end

%% Plotting Results\

% AL 26
plotvec = al26_t(1:25:end);
err = 2*ones(length(plotvec),1);

figure()
hold on
p1=plot(al26_t,al26_c{8},'LineWidth',1);
p2=errorbar(al26_t(1:25:end),al26_c{8}(1:25:end),err,'k');
p3=plot(t_al26,u_al26IA(:,8),'LineWidth',1);
p4=plot(t_al26,u_al26IB(:,8),'LineWidth',1);
p5=plot(t_al26,u_al26II(:,8),'LineWidth',1);
p6=plot(t_al26,u_al26III(:,8),'LineWidth',1);
grid minor
xlabel("Time (s)")
ylabel('Temperature (^\circC)')
legend([p1,p2,p3,p4,p5,p6],{'Experimental Results','\pm2 ^\circC error','Model IA','Model 1B','Model II','Model III'},'Location','southeast')
legend
title("Model Comparison: Aluminum - 26V,250mA")


```

```
% Steel
plotvec = st_t(1:25:end);
err = 2*ones(length(plotvec),1);

figure()
hold on
p1=plot(st_t,st_c{8},'LineWidth',1);
p2=errorbar(st_t(1:25:end),st_c{8}(1:25:end),err,'k');
p3=plot(t_st,u_stIA(:,8),'LineWidth',1);
p4=plot(t_st,u_stIB(:,8),'LineWidth',1);
p5=plot(t_st,u_stII(:,8),'LineWidth',1);
p6=plot(t_st,u_stIII(:,8),'LineWidth',1);
grid minor
xlabel("Time (s)")
ylabel('Temperature (^\circC)')
legend([p1,p2,p3,p4,p5,p6],{'Experimental Results','\pm2 ^\circC error','Model IA','Model 1B','Model II','Model III'},'Location','southeast')
legend
title("Model Comparison: Steel - 21V,192mA")


% AL 28
plotvec = al28_t(1:25:end);
err = 2*ones(length(plotvec),1);

figure()
hold on
p1=plot(al28_t,al28_c{8},'LineWidth',1);
p2=errorbar(al28_t(1:25:end),al28_c{8}(1:25:end),err,'k');
p3=plot(t_al28,u_al28IA(:,8),'LineWidth',1);
p4=plot(t_al28,u_al28IB(:,8),'LineWidth',1);
p5=plot(t_al28,u_al28II(:,8),'LineWidth',1);
p6=plot(t_al28,u_al28III(:,8),'LineWidth',1);
grid minor
xlabel("Time (s)")
ylabel('Temperature (^\circC)')
legend([p1,p2,p3,p4,p5,p6],{'Experimental Results','\pm2 ^\circC error','Model IA','Model 1B','Model II','Model III'},'Location','southeast')
legend
title("Model Comparison: Aluminum - 28V,269mA")

% Br 26
plotvec = br26_t(1:25:end);
err = 2*ones(length(plotvec),1);

figure()
hold on
p1=plot(br26_t,br26_c{8},'LineWidth',1);
p2=errorbar(br26_t(1:25:end),br26_c{8}(1:25:end),err,'k');
p3=plot(t_br26,u_br26IA(:,8),'LineWidth',1);
p4=plot(t_br26,u_br26IB(:,8),'LineWidth',1);
p5=plot(t_br26,u_br26II(:,8),'LineWidth',1);
p6=plot(t_br26,u_br26III(:,8),'LineWidth',1);
grid minor
xlabel("Time (s)")
ylabel('Temperature (^\circC)')
legend([p1,p2,p3,p4,p5,p6],{'Experimental Results','\pm2 ^\circC error','Model IA','Model 1B','Model II','Model III'},'Location','southeast')
```

```
legend
title("Model Comparison: Brass - 26V,245mA")

% Br 28
plotvec = br29_t(1:25:end);
err = 2*ones(length(plotvec),1);

figure()
hold on
p1=plot(br29_t,br29_c{8},'LineWidth',1);
p2=errorbar(br29_t(1:25:end),br29_c{8}(1:25:end),err,'k');
p3=plot(t_br29,u_br29IA(:,8),'LineWidth',1);
p4=plot(t_br29,u_br29IB(:,8),'LineWidth',1);
p5=plot(t_br29,u_br29II(:,8),'LineWidth',1);
p6=plot(t_br29,u_br29III(:,8),'LineWidth',1);
grid minor
xlabel("Time (s)")
ylabel('Temperature (^\circC)')
legend([p1,p2,p3,p4,p5,p6],{'Experimental Results','\pm2 ^\circC error','Model IA','Model 1B','Model II','Model III'},'Location','southeast')
legend
title("Model Comparison: Brass - 29V,273mA")

%% Functions
function u = heatdistributionIA(t,x,const)
    sum = [0,0,0,0,0,0,0,0];
    for n = 1:10
        b_n = (8*const.H_an*const.L*(-1)^n)/((2*n-1)^2*pi^2);
        lambda_n = pi*(2*n-1)/(2*const.L);
        sum = sum + b_n.*sin(lambda_n.*x).*exp(-1*lambda_n^2*const.alpha*t);
    end
    u = const.T0+const.H_an.*x + sum;
end

function u = heatdistributionIB(t,x,const)
    sum = [0,0,0,0,0,0,0,0];
    for n = 1:10
        b_n = (8*const.H_exp*const.L*(-1)^n)/((2*n-1)^2*pi^2);
        lambda_n = pi*(2*n-1)/(2*const.L);
        sum = sum + b_n.*sin(lambda_n.*x).*exp(-1*lambda_n^2*const.alpha*t);
    end
    u = const.T0+const.H_exp.*x + sum;
end

function u = heatdistributionII(t,x,const)
    sum = [0,0,0,0,0,0,0,0];
    for n = 1:10
        b_n = (8*(const.M_exp-const.H_exp)*const.L*(-1)^(n+1))/((2*n-1)^2*pi^2);
        lambda_n = pi*(2*n-1)/(2*const.L);
        sum = sum + b_n.*sin(lambda_n.*x).*exp(-1*lambda_n^2*const.alpha*t);
    end
    u = const.T0+const.H_exp.*x + sum;
end

function u = heatdistributionIII(t,x,const)
     sum = [0,0,0,0,0,0,0,0];
    for n = 1:10
        b_n = (8*const.H_exp*const.L*(-1)^n)/((2*n-1)^2*pi^2);
```

```
        lambda_n = pi*(2*n-1)/(2*const.L);
        sum = sum + b_n.*sin(lambda_n.*x).*exp(-1*lambda_n^2*const.alpha2*t);
    end
    u = const.T0+const.H_exp.*x + sum;
end
```