University of Colorado Boulder

ASEN 3113: Thermodynamics and Heat Transfer

Lab 2: Heat Conduction Lab

Author:
Johnathan Abrams[1]

Author:
Rishab Pally[2]

Author:
Jack Ryan[3]

Author:
Blair Schulze[4]

Professor: Dr.Xinlin Li

Due: Oct 31, 2022

[1]SID: 109510078
[2]SID: 109519936
[3]SID: 109790916
[4]SID: 109605353

Smead Aerospace
UNIVERSITY OF COLORADO **BOULDER**

Contents

This paper provides an analysis on transient and steady state heat conduction using three different solid material rods which include Aluminum, Brass and Stainless Steel. Each rod is housed inside an insulated casing, while one side is being heated up the other is being chilled. Then data is collected using the DAQ(National Instruments USB Data Acquisition) hardware and LabView software to analyze the temperature readings from the evenly spaced thermocouples along the rods. This aids the user in better understanding the fundamentals of heat conduction through constant mediums. Using partial derivatives and Matalab modeling, we were able to determine an analytical steady state slope H_{an} of 98.7 and 114 C°/m, an experimental steady state slope H_{exp} of 54.3 and 66.8 C°/m, an initial state slope M_{exp} of -2.72 and -.375 C°/m, a thermal diffusivity α of $4.82 * 10^{-5}$ m^2/s and an adjusted thermal diffusivity α_{adj} of $2.32 * 10^{-5}$ m^2/s for Aluminium at 26 V and 28 V respectively. For Brass at 26 V and 29 V, we determined a H_{an} of 109 and 136 C°/m, a H_{exp} of 114 and 140 C°/m, a M_{exp} of 7.03 and 5.81 C°/m, a α of $3.56 * 10^{-5}$ m^2/s and an α_{adj} of $1.56 * 10^{-5}$ m^2/s respectively. Lastly for Stainless Steel we determined a H_{an} of 491 C°/m, a H_{exp} of 277 C°/m, a M_{exp} of 34.6 C°/m, a α of $4.05 * 10^{-6}$ m^2/s and an α_{adj} of $4.55 * 10^{-6}$ m^2/s. Overall this lab aids with better understanding heat conduction and how heat transfer varies between different types of metals.

Nomenclature

ρ	=	density
Q	=	rate of energy into the system
k	=	thermal conductivity
α	=	thermal diffusivity
α_{adj}	=	adjusted thermal diffusivity
A	=	area of the rod
c_p	=	specific heat capacity
T_0	=	initial temperature
L	=	length of Rod
n	=	value of term
t	=	time duration
H_{an}	=	analytical steady state slope
H_{exp}	=	experimental steady state slope
M_{exp}	=	initial state slope
Fo	=	Fourier Number

1 Introduction

There are 3 main ways heat is transferred from a body with high temperature to a body with a low temperature. These include convection, radiation and conduction(the heat transfer occurring in this lab). Heat conduction in a metal rod can be simplified to one end of the metal rod being heated up resulting in the cooler end of the rod also starting to heat up as well. This occurs due to the kinetic energy of the molecules increasing. As molecules from one side of the rod continue to move, they constantly collide with other molecules nearby, increasing the kinetic energy and heating up the rod gradually. The collision of the neighboring molecules is where the transfer of energy occurs.

When analyzing the metal rod, thermocouples were placed on the length of the rod. We were able to track the temperature and determine the rate at which the heat transfer is occurring at. To understand the relations between different types of materials and heat transfer, we accounted for variables such as density, specific heat capacity and thermal diffusivity. To do so, we used the Fourier's Number which is the ratio of thermal diffusivity times the duration of time over the length of material or object where heat conduction is taking place. The Fourier Number overall aids in characterizing the transient heat conduction occuring. Using the steady state solution of the heat equation, we are able to derive a new equation which accounts for the temperature of the rod at any given point of time. Note that boundary conditions and assumptions such as the rod having axial dimension, the initial temperature of the rod staying constant and the rod being perfectly insulated need to be accounted for as well.

2 Experimental Procedure

1. Initialize by setting the chilled water temperature to $10^{\circ}C$.
2. Using the USB cable, connect the thermocouple box to the computer.
3. Plug each of the thermocouples from the heat conduction apparatus or rod being tested to the thermocouple box receptacles.
4. On the computer, open the Thermocouple Testbench.vi application which is located under $[H : AES - Software - VIs - General - ThermocoupleTestbench]$.
5. Run the VI application by clicking the white arrow on the left hand corner of the toolbar.
6. Click the [Load Lab] preset and select the [ASEN 3113 - Heat Conduction Lab] preset.
7. Monitor the VI temperature to check whether the initial conditions are met while waiting for the temperature of the rod to be cooled to approximately $10^{\circ}C$.
8. Stop the VI application by clicking the [STOP and SAVE] button when the rod has reached a uniform initial temperature. Note: do not save your data at this step.
9. Check whether the power supply and current are set within the boundary requirements. Voltages should be between 15V and 30V while current should be approximately 400 mA.
10. Start running the VI application again by clicking the white arrow on the left hand corner of the toolbar.
11. Switch the power supply on by pressing the [OFF/ON] button.
12. Wait for the rod to be heated to a steady state distribution.
13. Keep monitoring the VI application to confirm whether the rod has reached an acceptable steady state condition.
14. Once the rod has reached the acceptable steady state condition, click the [STOP and SAVE] button and collect the data.
15. A prompt will require you to name the file of the most recent data collected. Save it under a name with material, voltage and current for easier reference later.
16. Repeat these steps with each of the different material rods given.

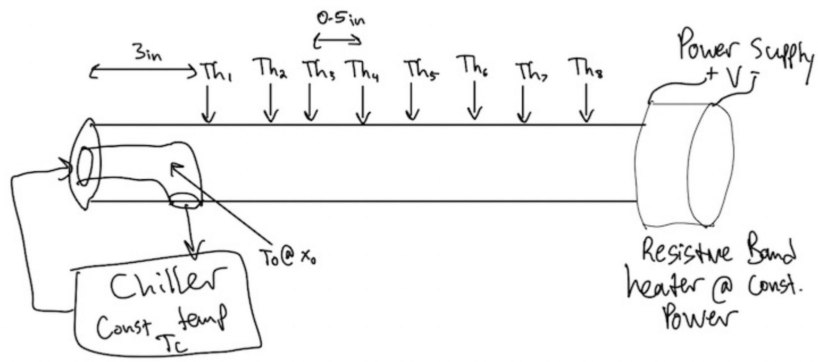

Figure 1: Diagram of Experimental Setup

3 Results

3.1 Steady State Distributions

The following values for analytical steady state temperature distribution slope were calculated using the $H_{an} = \frac{\dot{Q}}{kA}$ formula, with k being the thermal conductivity of each material, A being the cross-sectional area of the rod, and $\dot{Q}$ being the rate of heat transfer into the material, which was calculated by multiplying the voltage of the power supply by the current running through the material. Next, H_{exp} was calculated by using the MATLAB function polyfit on the experimental steady state temperature for each thermocouple against the position of each thermocouple on the metal rod. In addition, a first order polyfit (linear model) was conducted to fit the data. This yielded a line with slope of H_{exp} and y-intercept of T_0. Then, the steady state temperature distributions were plotted by using the steady state distribution slopes and initial temperatures to create linear models. The analytically derived model slopes and experimentally derived model slopes were plotted against each other and the experimental data temperature distribution lines.

Table 1: Initial Temperature and Temperature Distribution Slopes

Material	Aluminum 26V	Aluminum 28V	Brass 26V	Brass 29V	Steele 21V
H_{an}	98.7 C/m	114 C/m	109 C/m	136 C/m	491 C/m
H_{exp}	54.3 C/m	66.8 C/m	114 C/m	140 C/m	277 C/m
T_0	12.1 C	12.5 C	12.1 C	12.2 C	11.4 C

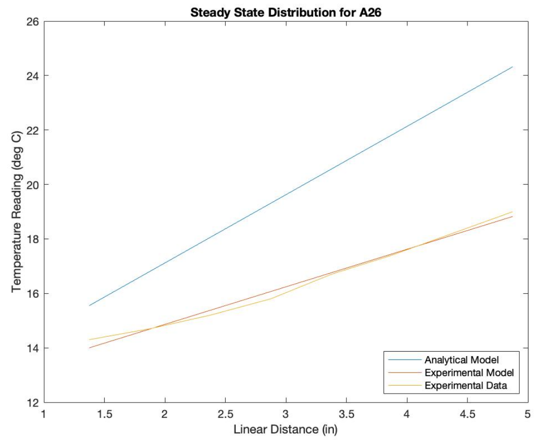

Figure 2: Steady State Distribution for Aluminum 26V

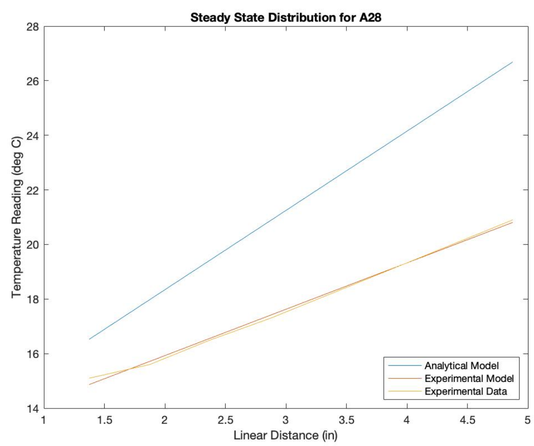

Figure 3: Steady State Distribution for Aluminum 28V

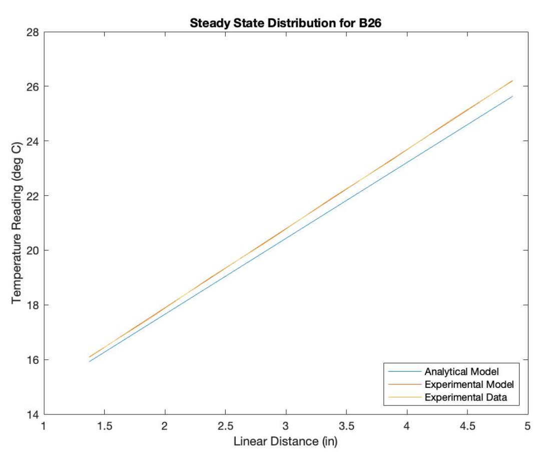

Figure 4: Steady State Distribution for Brass 26V

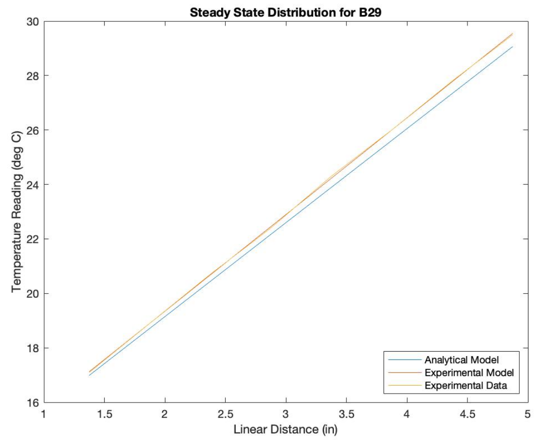

Figure 5: Steady State Distribution for Brass 29V

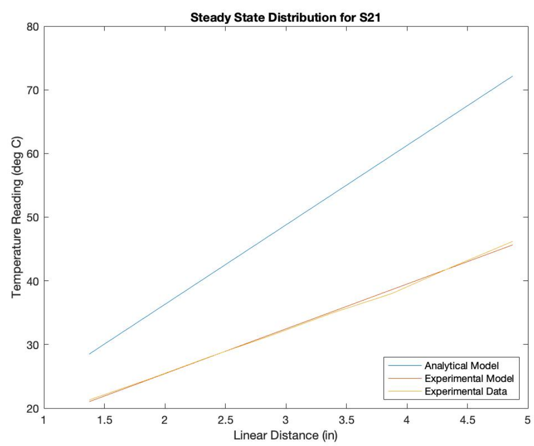

Figure 6: Steady State Distribution for Steele 21V

3.2 Time-Dependent Temperature Profiles

To generate the model 1A and model 1B time-dependent Temperature Profiles, the function $u(x,t) = T_0 + Hx + \sum_{n=1}^{10} b_n * sin(\lambda_n * x) * e^{-(\lambda_n)^2 * \alpha * t}$ was used to determine a model for the temperature of each thermocouple based on their position and the amount of time elapsed. Ten terms of the summation were used because we believed this to be a sufficient number based off of the high Fourier number from the prelab ($F_0 > 2$). These temperature profiles were first generated using the analytically derived steady state distribution slope (Model 1A), but then this model was refined by using the experimentally derived slopes (Model 1B). The plots of both models against the experimental data can be seen below.

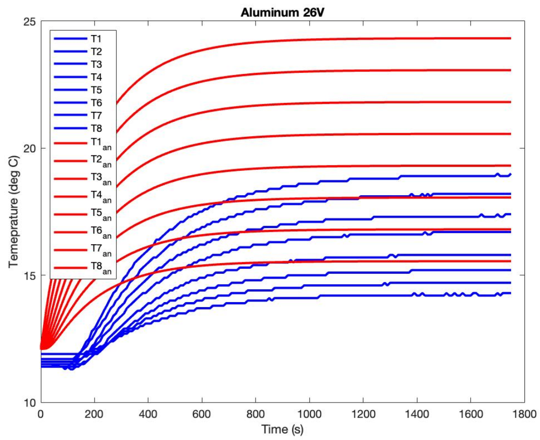

Figure 7: Model 1A for Aluminum 26V

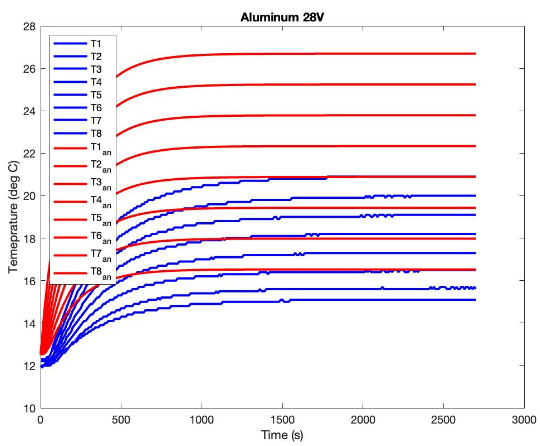

Figure 8: Model 1A for Aluminum 28V

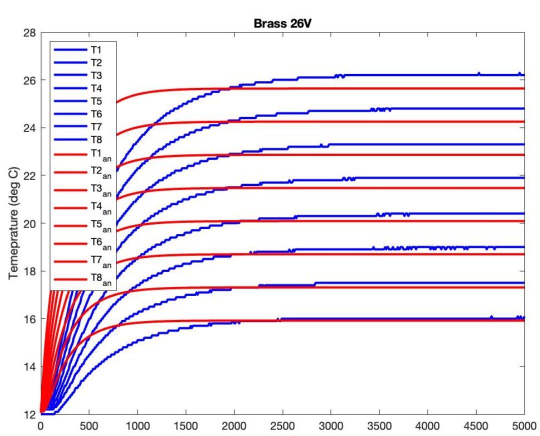

Figure 9: Model 1A for Brass 26V

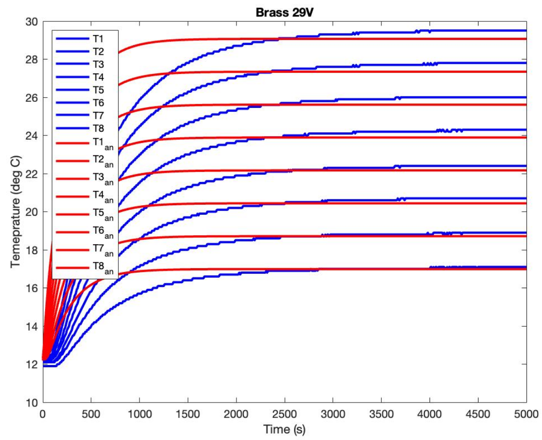

Figure 10: Model 1A for Brass 29V

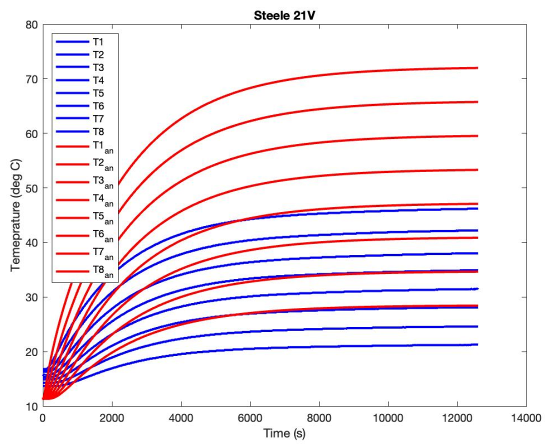

Figure 11: Model 1A for Steele 21V

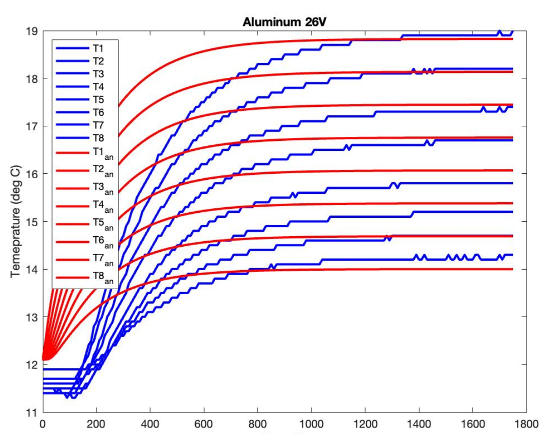

Figure 12: Model 1B for Aluminum 26V

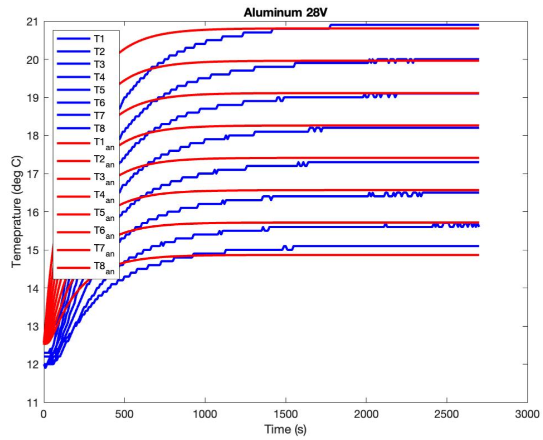

Figure 13: Model 1B for Aluminum 28V

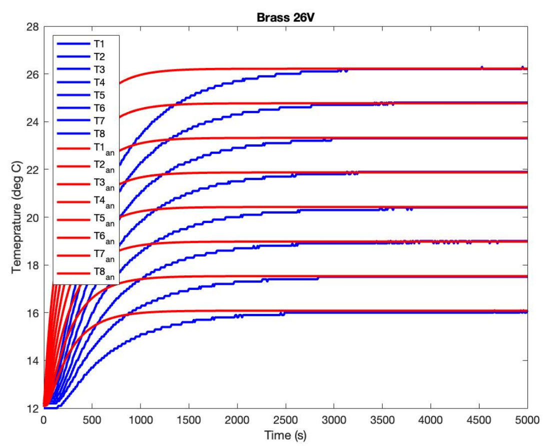

Figure 14: Model 1B for Brass 26V

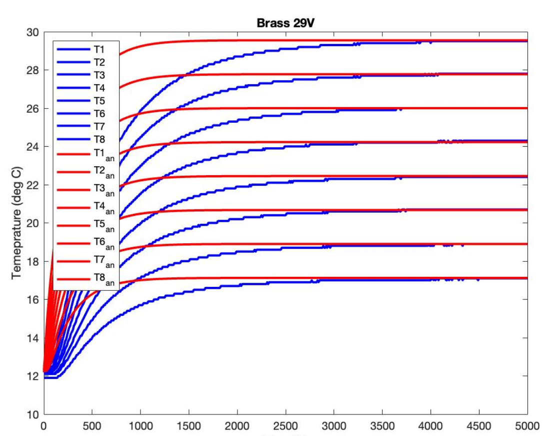

Figure 15: Model 1B for Brass 29V

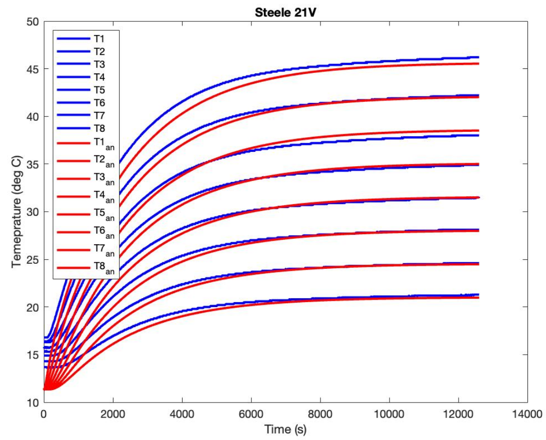

Figure 16: Model 1B for Steele 21V

3.3 Initial State Distributions

To determine the initial state slope M_{exp}, another linear fit was performed using the polyfit function with the initial temperatures of each thermocouple in the rod. This was performed against the positions of each thermocouple. For most materials, the initial temperature distribution had a non-negligible slope, and thus the initial temperature assumption (that the initial temperature accross the rod is equal) is invalid. To correct for this, the initial state slope was incorporated into the $u(x,t)$ formula (especially in the b_n term) in an attempt to more accurately describe the initial temperature distribution of the rod and the changes of temperature over time due to this non-even initial temperature distribution. This model (model 2) also used the experimentally determined steady state temperature distribution slope.

Table 2: Initial Temperature Distribution Slopes

Material	Aluminum 26V	Aluminum 28V	Brass 26V	Brass 29V	Steele 21V
M_{exp}	-2.72 C/m	-0.375 C/m	7.03 C/m	5.81 C/m	34.6 C/m

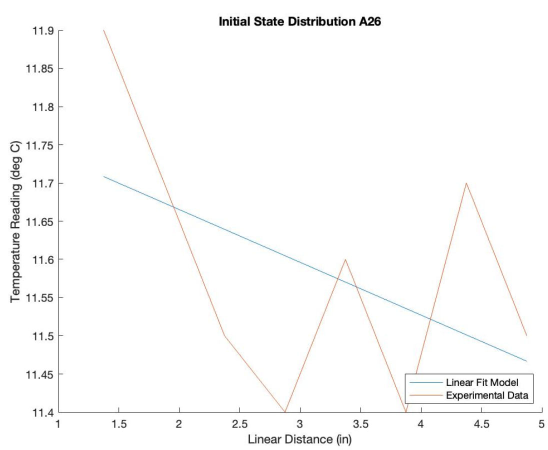

Figure 17: Initial State Distribution for Aluminum 26V

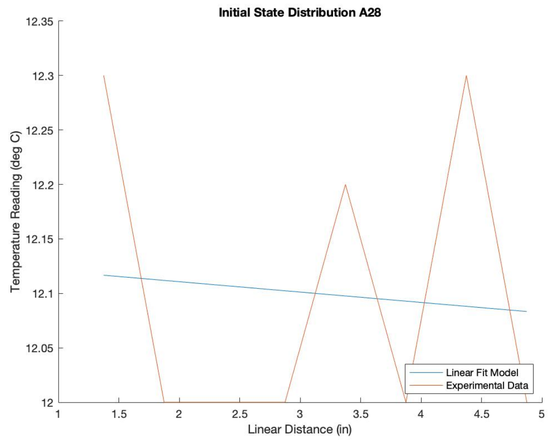

Figure 18: Initial State Distribution for Aluminum 28V

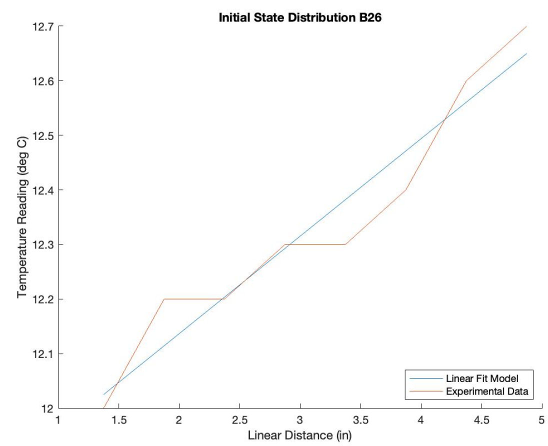

Figure 19: Intial State Distribution for Brass 26V

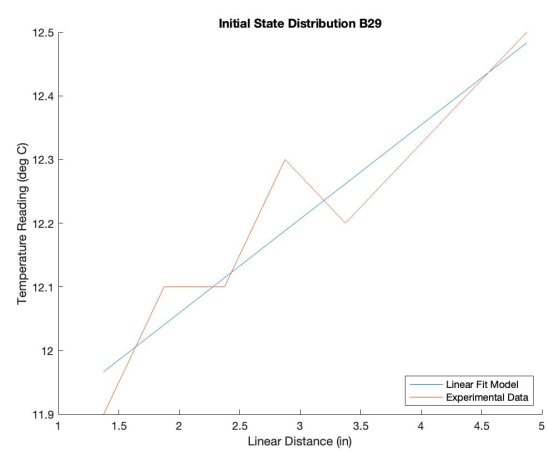

Figure 20: Initial State Distribution for Brass 29V

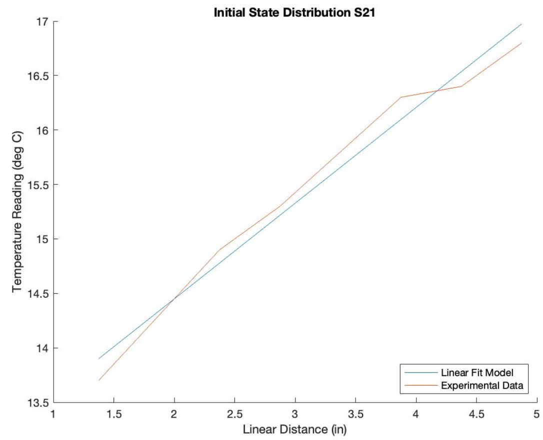

Figure 21: Initial State Distribution for Steele 21V

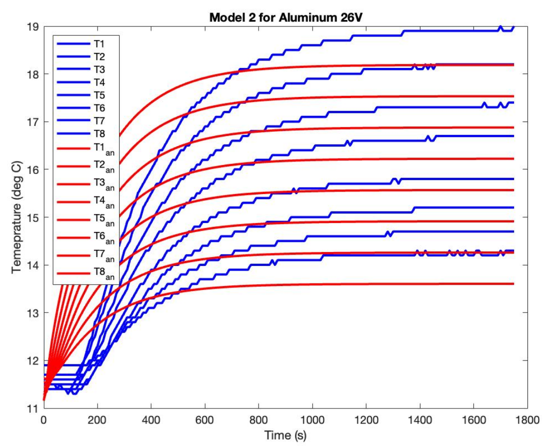

Figure 22: Model 2 for Aluminum 26V

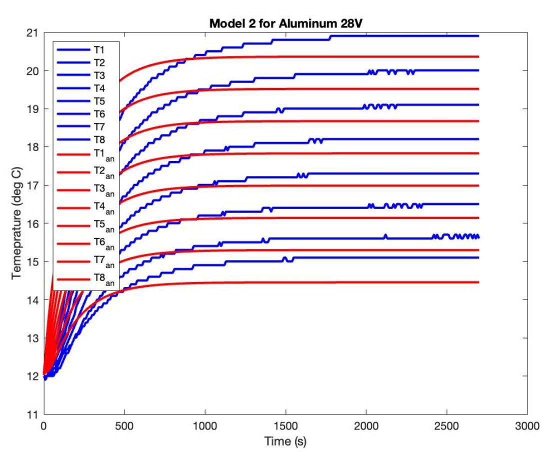

Figure 23: Model 2 for Aluminum 28V

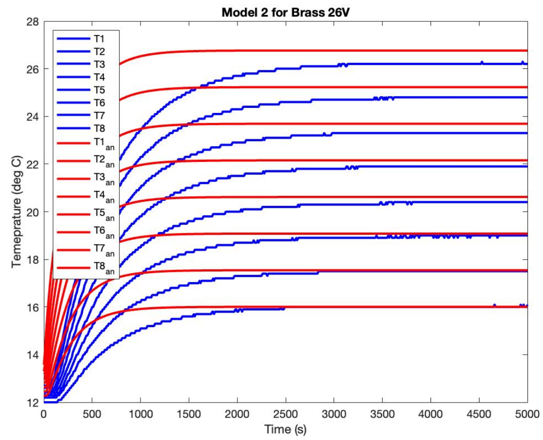

Figure 24: Model 2 for Brass 26V

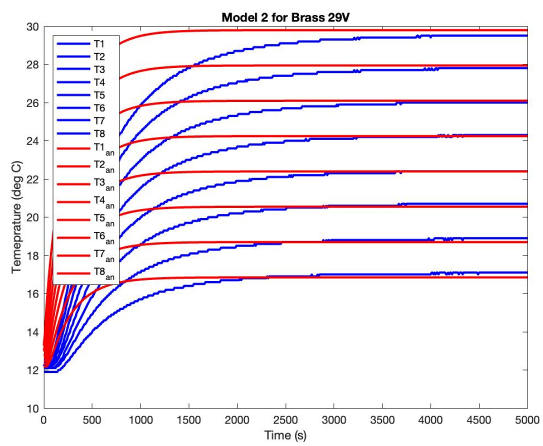

Figure 25: Model 2 for Brass 29V

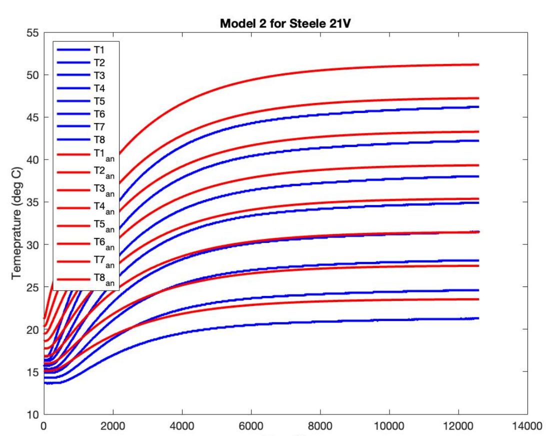

Figure 26: Model 2 for Steele 21V

3.4 Variance in Thermal Diffusivities

To further improve the transient phase of temperature profile model 1B, the thermal diffusivities of each material were adjusted until their respective models matched the transient portion of the experimental data more closely than in previous models. For aluminum and brass, this involved decreasing the thermal diffusivity, since they consistently overshot the experimental data in the transient phase. For steele, this involved increasing the thermal diffusivity slightly, since it had undershot the experimental data in the transient phase. The resulting models plotted against the exprimental data can be viewed below, along with a table containing the original and adjusted thermal diffusivity values.

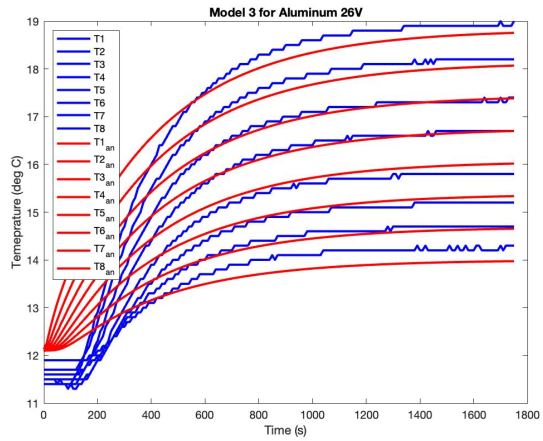

Figure 27: Model 3 for Aluminum 26V

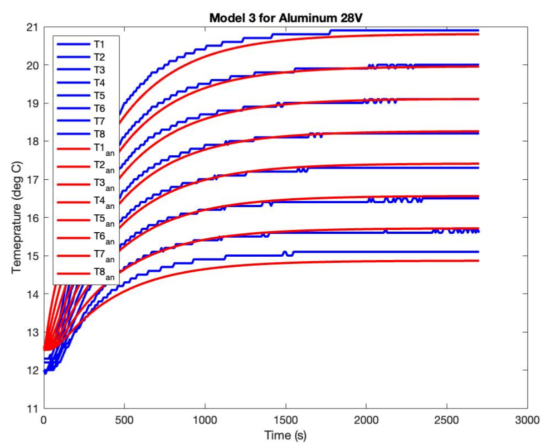

Figure 28: Model 3 for Aluminum 28V

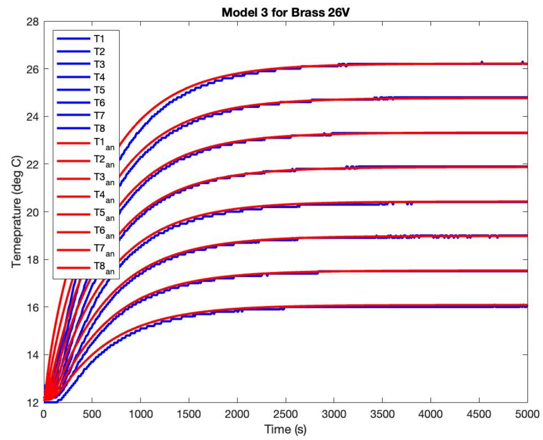

Figure 29: Model 3 for Brass 26V

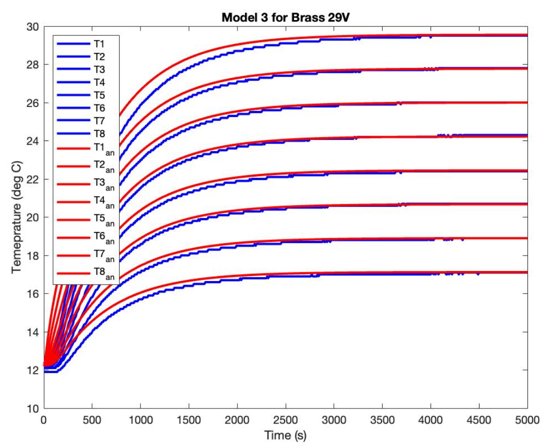

Figure 30: Model 3 for Brass 29V

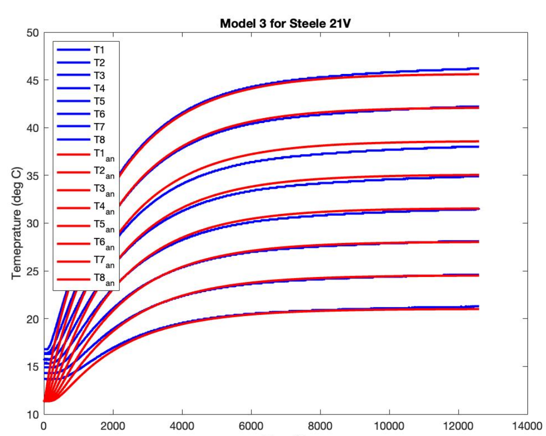

Figure 31: Model 3 for Steel 21V

Table 3: Fourier numbers and times to Steady State

Material	Aluminum 26V	Aluminum 28V	Brass 26V	Brass 29V	Steele 21V
α	$4.82 * 10^{-5}$ m$^2/s$	$4.82 * 10^{-5}$ m$^2/s$	$3.56 * 10^{-5}$ m$^2/s$	$3.56 * 10^{-5}$ m$^2/s$	$4.05 * 10^{-6}$ m$^2/s$
α_{adj}	$2.32 * 10^{-5}$ m$^2/s$	$2.32 * 10^{-5}$ m$^2/s$	$1.56 * 10^{-5}$ m$^2/s$	$1.56 * 10^{-5}$ m$^2/s$	$4.55 * 10^{-6}$ m$^2/s$

3.5 Time to Steady State

The times to steady state were determined via visual analysis of the experimental data temperature profiles. These values were than used along with the adjusted thermal diffusivity values and the lengths of the rods to calculate the Fourier numbers for each experimental data set. These results can be seen in the table below.

Table 4: Fourier numbers and times to Steady State

Material	Aluminum 26V	Aluminum 28V	Brass 26V	Brass 29V	Steele 21V
t_{ss}	1000 s	1500 s	2500 s	2500 s	8000 s
F_O	1.04	1.56	1.75	1.75	1.63

4 Analysis

4.1 Comparison of experimental and analytical Steady-State Slopes

Given set thermocouple positions and corresponding temperature measurements for this rod, we were able to use a linear extrapolation process to determine the steady state temperature distribution slope, H_{exp}, and the temperature at the cold end of the rod, T_0. The analytical steady state temperature distribution slope, H_{an}, for each trial was found using the rate of energy into the system divided by thermal conductivity and cross-sectional area. For aluminum and steel rod trials, we found that the analytical model slope was about twice as large as the experimentally found slope. However, our results for the brass rods showed that the analytical slope values were nearly identical to the experimentally found values. The discrepancies found in our steel and aluminum rod tests may be due to a conductivity factor. Brass is the most conductive metal of our three materials, which makes it more likely for the experimental temperature distribution to even out more similarly to the analytical derivation.

4.2 Justification of Initial Temperature Assumption

The initial temperature of our rod is assumed to be constant for each data set for the first part of this lab. Since the material composition throughout the rod is not guaranteed to be perfectly uniform, there may be some errors in this assumption. However, for the purposes of this experiment, the assumption was useful for finding the steady state temperature distribution slopes of each of our materials. Using a linear fit on the initial temperature distribution of our rod, we were able to find a new initial state slope M_{exp}. Comparing this new model to previous models that used the initial temperature assumption, we saw that aluminum and steel did not follow the linear fits as well as brass. This implies that while our initial temperature assumption may have been valid for our brass rod tests, we cannot continue with this assumption for aluminum and steel. An altered initial temperature profile for these materials would require initial experimental temperature data instead of assuming the initial temperature of the entire rod was constant and equal to T_0. This change would cause our steady slope-intercept function to have position dependence, and the initial condition for the transient solution would need to also depend on M_{exp}. This variable initial temperature profile should not change the steady state distribution of temperature across our rods since the time between the initial and steady state allows the temperature to reach equilibrium across the rod.

4.3 Varying Thermal Diffusivities

The third model created used the variation of Thermal diversities to more closely match the data. This model is the most accurate model but still has limitations when it comes to comparing the results of the real world experiment. Similar to the previous models developed Aluminum 26V shows the greatest variance from the model and is therefore easiest to exemplify the shortcomings of model 3. The starting temperatures in model 3 much like model 1 are all the same and although this is an assumption that seems reasonable it does not reflect the realities of the physical system. There is also a difference shown in the transient phase between the relative slopes of the model and the experiment this was adjusted for diffusivity but the performance still only match in certain sections. The strengths of this model partially rely on the comparison to data after the fact so it can be referred to as quasi-analytical. Model 3 shows why it is necessary to test models and to adjust to fit real world systems. Especially for the lower diffusivity materials it has exceptional accuracy in the transient phases most essential to understand in our model. It is also evident in Table 3 that the limitations of the experiment make the variance between universally accepted diffusivity values and the model 3 adjusted diffusivity values fairly high. The adjusted thermal diffusivity values are almost half of the accepted value for every metal tested except for Steele 21V which corroborates the idea that the limitations of the experiments thermal insulation are valid. Since, Steele has the lowest thermal diffusivity we expect it to lose less heat to the surroundings than the other metals tested. Model 3 shows the most accurate comparison to the real data for every metal so although we would want a better model or experiment for metals with high thermal diffusivities it provides a good understanding of the physical system especially one that is limited by its ability to insulate.

4.4 Fourier Number

The physical significance of the Fourier Number is it shows the relation between the rate of heat conduction through the rod and rate of heat being stored in the rod. While there is a lower rate of heat conduction through the rod, we can see a direct correlation with lower Fourier Number values whereas for higher rates of heat conduction, the values for the Fourier Number are greater. For metals with higher thermal diffusivities along with longer times to reach steady state, the Fourier Number is greater in magnitude. Note that the length of the rod also has a direct effect on the Fourier Number as a longer rod will result in a lower Fourier Number value(though in this lab our rod length is constant). As seen in Table 3, there is a variation in Fourier Number values across the different materials that were tested. Brass had the highest Fourier Number at 1.75 at both 26 V and 29 V inputs. Stainless Steel had the second highest Fourier value at 1.63 with a 21 V input and Aluminium had the lowest Fourier Number values at 1.04 and 1.56 at 26 V and 28 V inputs respectively. With the calculated values, we also need to consider the time to steady state. Aluminium took between 1000 to 1500 seconds, Brass took 2500 seconds and Stainless Steel took the longest at 8000 seconds to reach steady state. These values directly effect the magnitude of the Fourier Number.

4.5 Time to Steady State

Table 4 shows the time to steady state for all 5 different metals. The results of this analysis are interesting since the Brass 26v and 29V experiments show the exact same time to steady state even though analytically it could be assumed that Brass 29V would take a longer time to reach equilibrium. In this case Aluminum being the only other material tested at two different voltages actually provides the best example of the expected results of this experiment since it takes longer to reach the equilibrium temperature in the 28V experiment. Overall the data and the expected behavior of these metals follows the trends that can be found analytically. It is important to note that this procedure for finding time to steady state held a lot of subjectivity's and induced human error in a perfect world the way of determining time to steady state would be more objective although this proves difficult because of the realities of a real world experiment and the data associated with it.

4.6 Comparison of Models 2 and 3

We hypothesize that Model 2 would fit the initial temperature profile of the data best, since it adjusts for the slope of the initial temperature distribution. On the other hand, Model 3 will likely fit the transient portion of the data best, since alpha was adjusted to fit this segment of data in particular. Additionally, we predict that Model 3 will fit the Steady State portion of the data best since it employs the experimental steady state slope in its calculation of the temperature profile. Assuming that the only error present was the 2C error in the thermocouple measurements, plots of the experimental temperature data with error bars were plotted side by side with Model 2 and Model 3 for thermocouple 8. The results can be seen in the plots below.

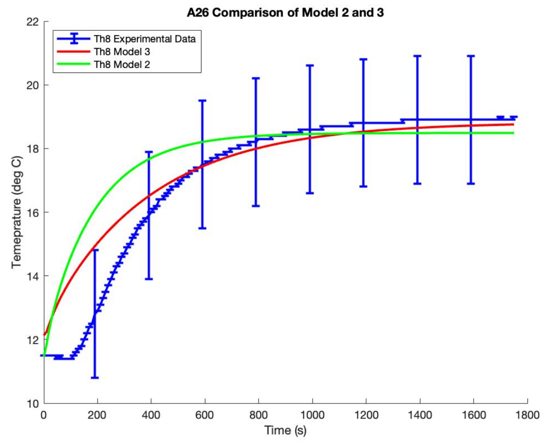

Figure 32: Model Comparison for Aluminum 26V

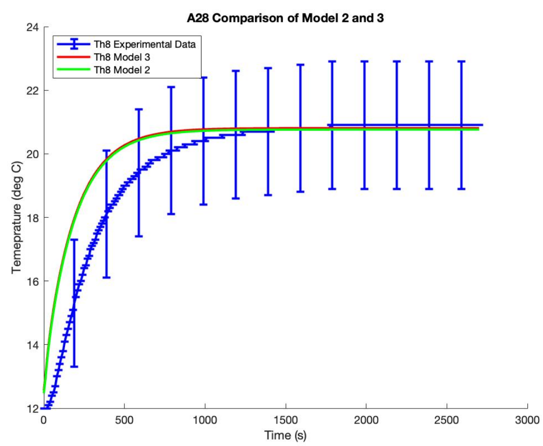

Figure 33: Model Comparison for Aluminum 28V

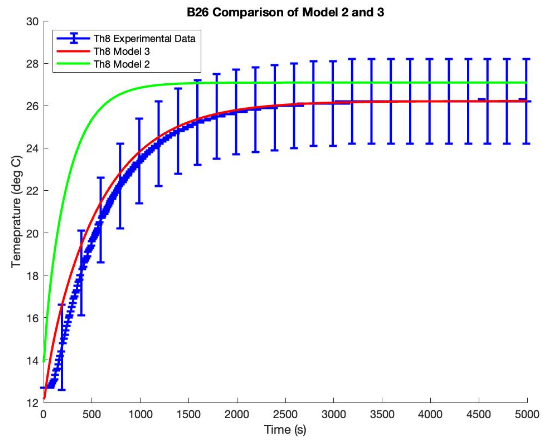

Figure 34: Model Comparison for Brass 26V

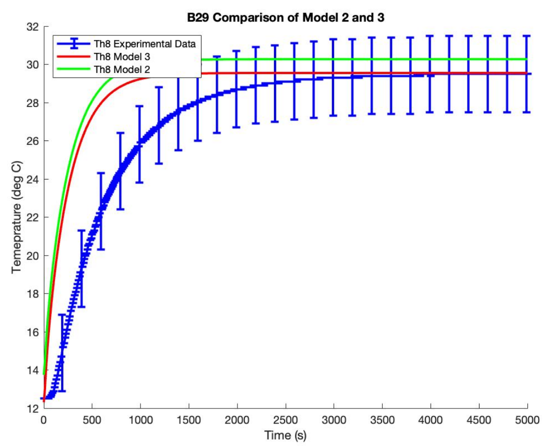

Figure 35: Model Comparison for Brass 29V

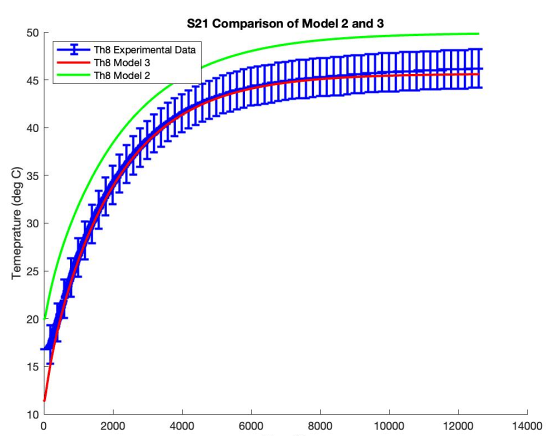

Figure 36: Model Comparison for Steel 21V

According to the plots above, it is evident that Model 3 fits the experimental data more closely for the steady state and transient sections of the data for almost all materials. For example, Model 3 fits the data of Steele and Brass 26V very well, while providing a much closer approximation than Model 2 for Aluminum 26V. For Aluminum 28V both models performed almost identically, and for Brass 28V, Model 3 only slightly out performed Model 2 in the steady state region, but did not outperform it in the transient phase. Model 2 seems to more closely match the initial temperature profile for most of the materials, but for some it performs about the same as Model 3. Thus, it seems that Model 3 overall performs better in the steady state and transient regions of the data, while Model 2 more closely approximates the initial temperature, just as was hypothesized.

4.7 Most useful Model to Predict Duration of Transient Phase

Model 3 would be most useful for an analysis of the transient phase. When comparing the different models used in this lab, model 3 has the most accurate transient phase prediction when compared to the experimental data during the transient phase. Models 1A and 1B align with the experimental data loosely at the steady state portion of the experiment and are significantly worse during the transient phase. By no longer assuming a constant initial temperature of the rod, Model 2 has a much more accurate time range for the transient phase than the previous two models. However, while Model 2 is a little closer to the overall shape of the transient solution, it still falls short of the curve and may not be the most useful model for this time-dependent analysis. The adjustment of thermal diffusivity for Model 3 allows the results to be tailored for each material more finely since there always exist inevitable imperfections in the materials that can significantly affect the predicted temperature distribution on the rod over time. Model 3 has a strong correlation with the experimental data steady state distribution and the transient phase curve, so to predict the duration of the transient phase in an experiment, Model 3 would provide the closest representation of temperature distribution.

5 References

[1] "The Online Materials Information Resource." MatWeb,
https://www.matweb.com/search/DataSheet.aspx?MatGUID=4f19a42be94546b686bbf43f79c51b7dckck=1

[2] "The Online Materials Information Resource." MatWeb,
https://www.matweb.com/search/QuickText.aspx?SearchText=Stainless+Steel+T-303+Annealed.

[3] "The Online Materials Information Resource." MatWeb,
https://www.matweb.com/search/QuickText.aspx?SearchText=Brass

[4] Edge, Engineers. "Thermal Diffusivity Table." Engineers Edge - Engineering, Design and Manufacturing Solutions.
https://www.engineersedge.com/heat$_{t}ransfer/thermal_{d}iffusivity_{t}able_{1}$3953.$htm$.

6 Appendix

6.1 Analytical Transient Derivation Work:

$$b_n = \frac{-2H}{L}\int_0^L x\sin(\lambda_n x)\,dx$$

Integration by Parts

$$= \frac{-2H}{L}\left(\frac{-x\cos(\lambda_n x)}{\lambda_n} - \int -\frac{\cos(\lambda_n x)}{\lambda_n}\right)$$

$$= \frac{-2H}{L}\left(\frac{-x\cos(\lambda_n x)}{\lambda_n} + \frac{\sin(\lambda_n x)}{\lambda_n^2}\Big]_0^L\right)$$

$$= \frac{-2H}{L}\left(\frac{-L\cos(\lambda_n L)}{\lambda_n} + \frac{\sin(\lambda_n L)}{\lambda_n^2} - 0\right)$$

$$= \frac{-2H}{L}\left(0 + \frac{(-1)^n}{(2\pi n-\pi)^2/4L^2}\right)$$

$$= \frac{-2H}{L}\left(\frac{4L^2(-1)^{n-1}}{(2\pi n-\pi)^2}\right)$$

$$= \frac{-8HL(-1)^{n-1}}{(2\pi n-\pi)^2}$$

$$= \frac{8HL(-1)^n}{(2n-1)^2\pi^2}$$

$$\boxed{b_n = \frac{8HL(-1)^n}{\pi^2(2n-1)^2}}$$

$$\int u\,dv = uv - \int v\,du$$

$$u = x \qquad du = 1\,dx$$

$$dv = \sin(\lambda_n x)$$

$$v = -\frac{\cos(\lambda_n x)}{\lambda_n}$$

$$\lambda_n = \frac{(2n-1)\pi}{2L}$$

$$\cos\left(\frac{(2n-1)\pi}{2L}L\right) = \cos\left(\pi n - \frac{\pi}{2}\right)$$

$$\sin\left(\frac{(2n-1)\pi}{2L}L\right) = \sin(\pi n - \pi/2)$$

$$\sin(\alpha-\beta) = \cos(\beta)\sin(\alpha) - \cos(\alpha)\sin($$

$$\cos(\alpha-\beta) = \cos(\alpha)\cos(\beta) + \sin(\alpha)\sin($$

$$\cos(\pi n - \pi/2) = \sin(\pi n) = 0$$

$$\sin(\pi n - \pi/2) = -\cos(\pi n) = (-1)^{n-1}$$

$$\lambda_n^2 = \left(\frac{(2n-1)\pi}{2L}\right)^2$$

$$= \frac{((2n-1)\pi)^2}{4L^2}$$

$$= \frac{(2\pi n-\pi)^2}{4L^2}$$

6.2 Matlab Derivation:

```
clear;clc;close all;

% Loading in Data
A_26 = load('Aluminum_26V_250mA');
A_28 = load('Aluminum_28V_269mA');
B_26 = load('Brass_26V_245mA');
```

```
B_29 = load('Brass_29V_273mA');
S_21 = load('Steel_21V_192mA');

% Prelab Data
T = [17.6,21.61,25.13,29.22,34.92,38.10,45.21,47.01];
x_L = [1.375,1.875,2.375,2.875,3.375,3.875,4.375,4.875];

%% Steady State Distributions

A_28(:,2) = [];
B_26(:,2) = [];
B_29(:,2) = [];
S_21(:,2) = [];


% Steady State temperatures
T_a26 = A_26(end,2:end);
T_a28 = A_28(end,2:end);
T_b26 = B_26(end,2:end);
T_b29 = B_29(end,2:end);
T_s21 = S_21(end,2:end);


%Time vectors
time_a26 = A_26(:, 1);
time_a28 = A_28(:, 1);
time_b26 = B_26(:, 1);
time_b29 = B_29(:, 1);
time_s21 = S_21(:, 1);


% thermal conductivity values
ka = 130; %W/mK
kb = 115; %W/mk
ks = 16.2; %W/mK

A = pi*(0.5*0.0254)^2;

% Analytical temperature distribution slopes
H_an_a26 = (26*0.250*0.0254)/(ka*A); %C/in
H_an_a28 = (28*0.269*0.0254)/(ka*A); %C/in
H_an_b26 = (26*0.245*0.0254)/(kb*A); %C/in
H_an_b29 = (29*0.273*0.0254)/(kb*A); %C/in
H_an_s21 = (21*0.192*0.0254)/(ks*A); %C/in


H_an_a26_m = ((26*0.250*0.0254)/(ka*A))/0.0254; %C/m
H_an_a28_m = ((28*0.269*0.0254)/(ka*A))/0.0254; %C/m
H_an_b26_m = ((26*0.245*0.0254)/(kb*A))/0.0254; %C/m
H_an_b29_m = ((29*0.273*0.0254)/(kb*A))/0.0254; %C/m
H_an_s21_m = ((21*0.192*0.0254)/(ks*A))/0.0254; %C/m


% Linear fit of experimental data to determine Experimental temp distribution slope
```

```
p1 = polyfit(x_L,T_a26,1);
H_a26 = p1(1);
T0_a26 = p1(2);
v_a26 = H_a26*x_L + T0_a26;

p2 = polyfit(x_L,T_a28,1);
H_a28 = p2(1);
T0_a28 = p2(2);
v_a28 = H_a28*x_L + T0_a28;


p3 = polyfit(x_L,T_b26,1);
H_b26 = p3(1);
T0_b26 = p3(2);
v_b26 = H_b26*x_L + T0_b26;


p4 = polyfit(x_L,T_b29,1);
H_b29 = p4(1);
T0_b29 = p4(2);
v_b29 = H_b29*x_L + T0_b29;



p5 = polyfit(x_L,T_s21,1);
H_s21 = p5(1);
T0_s21 = p5(2);
v_s21 = H_s21*x_L + T0_s21;

H_a26_m = H_a26/0.0254; %C/m
H_a28_m = H_a28/0.0254; %C/m
H_b26_m = H_b26/0.0254; %C/m
H_b29_m = H_b29/0.0254; %C/m
H_s21_m = H_s21/0.0254; %C/m


%% Steady State Distributions


% analytical temperature distributions
u_an_a26 = H_an_a26*x_L + T0_a26;
u_an_a28 = H_an_a28*x_L + T0_a28;
u_an_b26 = H_an_b26*x_L + T0_b26;
u_an_b29 = H_an_b29*x_L + T0_b29;
u_an_s21 = H_an_s21*x_L + T0_s21;


%Steady State Distributions
figure()
plot(x_L,u_an_a26)
hold on
plot(x_L,v_a26)
plot(x_L,T_a26)
```

```
xlabel('Linear Distance (in)')
ylabel('Temperature Reading (deg C)')
legend('Analytical Model','Experimental Model','Experimental Data','Location','southeast')
title('Steady State Distribution for A26')
hold off

figure()
plot(x_L,u_an_a28)
hold on
plot(x_L,v_a28)
plot(x_L,T_a28)
xlabel('Linear Distance (in)')
ylabel('Temperature Reading (deg C)')
legend('Analytical Model','Experimental Model','Experimental Data','Location','southeast')
title('Steady State Distribution for A28')
hold off

figure()
plot(x_L,u_an_b26)
hold on
plot(x_L,v_b26)
plot(x_L,T_b26)
xlabel('Linear Distance (in)')
ylabel('Temperature Reading (deg C)')
legend('Analytical Model','Experimental Model','Experimental Data','Location','southeast')
title('Steady State Distribution for B26')
hold off

figure()
plot(x_L,u_an_b29)
hold on
plot(x_L,v_b29)
plot(x_L,T_b29)
xlabel('Linear Distance (in)')
ylabel('Temperature Reading (deg C)')
legend('Analytical Model','Experimental Model','Experimental Data','Location','southeast')
title('Steady State Distribution for B29')
hold off

figure()
plot(x_L,u_an_s21)
hold on
plot(x_L,v_s21)
plot(x_L,T_s21)
xlabel('Linear Distance (in)')
ylabel('Temperature Reading (deg C)')
legend('Analytical Model','Experimental Model','Experimental Data','Location','southeast')
title('Steady State Distribution for S21')
hold off



%% Time-Dependent Temperature Profiles
```

```
% Model 1A


alpha_A26 = 4.82*10^(-5); %m^2/s
alpha_A28 = alpha_A26;
alpha_B26 = 3.56*10^(-5);
alpha_B29 = alpha_B26;
alpha_S21 = 4.05*10^(-6);


%Model 1A

time_dependent_temp(H_an_a26,T0_a26, A_26, time_a26, alpha_A26, 100, 'Aluminum 26V')
time_dependent_temp(H_an_a28,T0_a28, A_28, time_a28, alpha_A28, 200, 'Aluminum 28V')
time_dependent_temp(H_an_b26,T0_b26, B_26, time_b26, alpha_B26, 300, 'Brass 26V')
time_dependent_temp(H_an_b29,T0_b29, B_29, time_b29, alpha_B29, 400, 'Brass 29V')
time_dependent_temp(H_an_s21,T0_s21, S_21, time_s21, alpha_S21, 500, 'Steele 21V')



%Model 1B

time_dependent_temp(H_a26,T0_a26, A_26, time_a26, alpha_A26, 600, 'Aluminum 26V')
time_dependent_temp(H_a28,T0_a28, A_28, time_a28, alpha_A28, 700, 'Aluminum 28V')
time_dependent_temp(H_b26,T0_b26, B_26, time_b26, alpha_B26, 800, 'Brass 26V')
time_dependent_temp(H_b29,T0_b29, B_29, time_b29, alpha_B29, 900, 'Brass 29V')
time_dependent_temp(H_s21,T0_s21, S_21, time_s21, alpha_S21, 1000, 'Steele 21V')

%% Initial State Distributions (Model 2)

% Initial temperatures
T_in_a26 = A_26(1,2:end);
T_in_a28 = A_28(1,2:end);
T_in_b26 = B_26(1,2:end);
T_in_b29 = B_29(1,2:end);
T_in_s21 = S_21(1,2:end);


% Linear fit of experimental data to determine Initial state slope

p1_in = polyfit(x_L,T_in_a26,1);
M_exp_a26 = p1_in(1);
T0_exp_a26 = p1_in(2);
v_exp_a26 = M_exp_a26*x_L + T0_exp_a26;

p2_in = polyfit(x_L,T_in_a28,1);
M_exp_a28 = p2_in(1);
T0_exp_a28 = p2_in(2);
v_exp_a28 = M_exp_a28*x_L + T0_exp_a28;

p3_in = polyfit(x_L,T_in_b26,1);
M_exp_b26 = p3_in(1);
T0_exp_b26 = p3_in(2);
```

```
v_exp_b26 = M_exp_b26*x_L + T0_exp_b26;

p4_in = polyfit(x_L,T_in_b29,1);
M_exp_b29 = p4_in(1);
T0_exp_b29 = p4_in(2);
v_exp_b29 = M_exp_b29*x_L + T0_exp_b29;

p5_in = polyfit(x_L,T_in_s21,1);
M_exp_s21 = p5_in(1);
T0_exp_s21 = p5_in(2);
v_exp_s21 = M_exp_s21*x_L + T0_exp_s21;


M_exp_a26_m = M_exp_a26/0.0254;
M_exp_a28_m = M_exp_a28/0.0254;
M_exp_b26_m = M_exp_b26/0.0254;
M_exp_b29_m = M_exp_b29/0.0254;
M_exp_s21_m = M_exp_s21/0.0254;


%Steady State Distributions
figure()
hold on
plot(x_L,v_exp_a26)
plot(x_L,T_in_a26)
xlabel('Linear Distance (in)')
ylabel('Temperature Reading (deg C)')
legend('Linear Fit Model','Experimental Data','Location','southeast')
title('Initial State Distribution A26')
hold off

figure()
hold on
plot(x_L,v_exp_a28)
plot(x_L,T_in_a28)
xlabel('Linear Distance (in)')
ylabel('Temperature Reading (deg C)')
legend('Linear Fit Model','Experimental Data','Location','southeast')
title('Initial State Distribution A28')
hold off

figure()
hold on
plot(x_L,v_exp_b26)
plot(x_L,T_in_b26)
xlabel('Linear Distance (in)')
ylabel('Temperature Reading (deg C)')
legend('Linear Fit Model','Experimental Data','Location','southeast')
title('Initial State Distribution B26')
hold off

figure()
hold on
plot(x_L,v_exp_b29)
```

```
plot(x_L,T_in_b29)
xlabel('Linear Distance (in)')
ylabel('Temperature Reading (deg C)')
legend('Linear Fit Model','Experimental Data','Location','southeast')
title('Initial State Distribution B29')
hold off

figure()
hold on
plot(x_L,v_exp_s21)
plot(x_L,T_in_s21)
xlabel('Linear Distance (in)')
ylabel('Temperature Reading (deg C)')
legend('Linear Fit Model','Experimental Data','Location','southeast')
title('Initial State Distribution S21')
hold off


%Model 2 Plots


model_2(H_a26, M_exp_a26, T0_exp_a26, A_26, time_a26, alpha_A26, 1100, 'Model 2 for Aluminu
model_2(H_a28, M_exp_a28, T0_exp_a28, A_28, time_a28, alpha_A28, 1200, 'Model 2 for Aluminu
model_2(H_b26, M_exp_b26, T0_exp_b26, B_26, time_b26, alpha_B26, 1300, 'Model 2 for Brass 2
model_2(H_b29, M_exp_b29, T0_exp_b29, B_29, time_b29, alpha_B29, 1400, 'Model 2 for Brass 2
model_2(H_s21, M_exp_s21, T0_exp_s21, S_21, time_s21, alpha_S21, 1500, 'Model 2 for Steele


%% Variane in Thermal Diffusivities (Model 3)

alpha_A26_adj = 4.82*10^(-5) - 2.5*10^(-5); %m^2/s
alpha_A28_adj = alpha_A26;
alpha_B26_adj = 3.56*10^(-5) - 2*10^(-5);
alpha_B29_adj = alpha_B26;
alpha_S21_adj = 4.05*10^(-6) + 0.5*10^(-6);

time_dependent_temp(H_a26, T0_a26, A_26, time_a26, alpha_A26_adj, 1600, 'Model 3 for Alumin
time_dependent_temp(H_a28, T0_a28, A_28, time_a28, alpha_A28_adj, 1700, 'Model 3 for Alumin
time_dependent_temp(H_b26, T0_b26, B_26, time_b26, alpha_B26_adj, 1800, 'Model 3 for Brass
time_dependent_temp(H_b29, T0_b29, B_29, time_b29, alpha_B29_adj, 1900, 'Model 3 for Brass
time_dependent_temp(H_s21, T0_s21, S_21, time_s21, alpha_S21_adj, 2000, 'Model 3 for Steele

%% Time to Steady State

tss_A26 = 1000;
tss_A28 = 1500;
tss_B26 = 2500;
tss_B29 = 2500;
tss_S21 = 8000;

L = 5.875*0.0254; %m

Fo_A26 = (alpha_A26_adj*tss_A26)/L^2;
Fo_A28 = (alpha_A28_adj*tss_A28)/L^2;
```

```
Fo_B26 = (alpha_B26_adj*tss_B26)/L^2;
Fo_B29 = (alpha_B29_adj*tss_B29)/L^2;
Fo_S21 = (alpha_S21_adj*tss_S21)/L^2;


%% Error of Models 2 and 3

error_bars_analysis(H_a26, M_exp_a26, T0_a26, A_26, time_a26,  alpha_A26, alpha_A26_adj, 3
error_bars_analysis(H_a28, M_exp_a28, T0_a28, A_28, time_a28,  alpha_A28, alpha_A28_adj, 3
error_bars_analysis(H_b26, M_exp_b26, T0_b26, B_26, time_b26,  alpha_B26, alpha_B26_adj, 3
error_bars_analysis(H_b29, M_exp_b29, T0_b29, B_29, time_b29,  alpha_B29, alpha_B29_adj, 3
error_bars_analysis(H_s21, M_exp_s21, T0_s21, S_21, time_s21,  alpha_S21, alpha_S21_adj, 3






function time_dependent_temp(H, T0, experimental_data, time, alpha, fig_numb, fig_title)

x = [1.375,1.875,2.375,2.875,3.375,3.875,4.375,4.875]*0.0254; %m

H_m = H/0.0254; %C/m
L = (4 + (7/8) + 1)*0.0254; %m

n_axis = [1:1:10];

figure(fig_numb)
plot(experimental_data(:,1),experimental_data(:,2),'-b', 'LineWidth', 2)
hold on
plot(experimental_data(:,1),experimental_data(:,3),'-b', 'LineWidth', 2)
plot(experimental_data(:,1),experimental_data(:,4),'-b', 'LineWidth', 2)
plot(experimental_data(:,1),experimental_data(:,5),'-b', 'LineWidth', 2)
plot(experimental_data(:,1),experimental_data(:,6),'-b','LineWidth', 2)
plot(experimental_data(:,1),experimental_data(:,7),'-b','LineWidth', 2)
plot(experimental_data(:,1),experimental_data(:,8),'-b','LineWidth', 2)
plot(experimental_data(:,1),experimental_data(:,9),'-b','Linewidth', 2)
ylabel('Temeprature (deg C)')
xlabel('Time (s)')
title(fig_title)

linestyles = {'-r', '-r', '-r', '-r', '-r', '-r', '-r', '-r'};

figure(fig_numb)
hold on
for j = 1:length(x)

        for i=1:length(time)

            for n = 1:10

                %n = 1;
                bn = (8*H_m*L*((-1)^n))/((pi^2)*((2*n - 1)^2));
```

```
            lambda = ((2*n - 1)*pi)/(2*L);

            sums1(n) = bn*(sin(lambda*x(j)))*exp(-(lambda^2)*alpha*time(i));
        end

        u1(j,i) = (H_m*x(j) + T0) + sum(sums1(1:n));

    end
    plot(time, u1(j,:), linestyles{j}, 'LineWidth', 2)

end

figure(fig_numb)
legend('T1','T2','T3','T4','T5','T6','T7','T8','T1_{an}','T2_{an}','T3_{an}','T4_{an}','T5


end




function model_2(H, M, T0, experimental_data, time, alpha, fig_numb, fig_title)

x = [1.375,1.875,2.375,2.875,3.375,3.875,4.375,4.875]*0.0254; %m

H_m = H/0.0254; %C/m
M_m = M/0.0254;
L = (4 + (7/8) + 1)*0.0254; %m

n_axis = [1:1:10];

figure(fig_numb)
plot(experimental_data(:,1),experimental_data(:,2),'-b', 'LineWidth', 2)
hold on
plot(experimental_data(:,1),experimental_data(:,3),'-b', 'LineWidth', 2)
plot(experimental_data(:,1),experimental_data(:,4),'-b', 'LineWidth', 2)
plot(experimental_data(:,1),experimental_data(:,5),'-b', 'LineWidth', 2)
plot(experimental_data(:,1),experimental_data(:,6),'-b','LineWidth', 2)
plot(experimental_data(:,1),experimental_data(:,7),'-b', 'LineWidth', 2)
plot(experimental_data(:,1),experimental_data(:,8),'-b', 'LineWidth', 2)
plot(experimental_data(:,1),experimental_data(:,9),'-b', 'LineWidth', 2)
ylabel('Temeprature (deg C)')
xlabel('Time (s)')
title(fig_title)

linestyles = {'-r', '-r', '-r', '-r', '-r', '-r', '-r', '-r'};



figure(fig_numb)
hold on
for j = 1:length(x)

    for i=1:length(time)
```

```
            for n = 1:10

                %n = 1;
                bn = -(8*(M_m - H_m)*L*((-1)^n))/((pi^2)*((2*n - 1)^2));
                lambda = ((2*n - 1)*pi)/(2*L);

                sums1(n) = bn*(sin(lambda*x(j)))*exp(-(lambda^2)*alpha*time(i));
            end

            w = H_m*x(j) + T0;
            u1(j,i) = (w) + sum(sums1(1:n)) + M_m*x(j);

        end
        plot(time, u1(j,:), linestyles{j}, 'LineWidth', 2)

    end

figure(fig_numb)
legend('T1','T2','T3','T4','T5','T6','T7','T8','T1_{an}','T2_{an}','T3_{an}','T4_{an}','T5



end




function error_bars_analysis(H, M, T0, experimental_data, time, alpha2, alpha3, fig_numb,

x = [1.375,1.875,2.375,2.875,3.375,3.875,4.375,4.875]*0.0254; %m

H_m = H/0.0254; %C/m
M_m = M/0.0254;
L = (4 + (7/8) + 1)*0.0254; %m

n_axis = [1:1:10];

err = 2*zeros(1, length(experimental_data));
for z = 1:(length(err)/20)
    err(z*20) = 2;
end

figure(fig_numb)
hold on
errorbar(experimental_data(:,1),experimental_data(:,9), err,'-b', 'LineWidth', 2)
ylabel('Temeprature (deg C)')
xlabel('Time (s)')
title(fig_title)

linestyles = {'-r', '-r', '-r', '-r', '-r', '-r', '-r', '-r'};

figure(fig_numb)
hold on
```

```
for j = 1:length(x)

        for i=1:length(time)

            for n = 1:10

                %n = 1;
                bn = (8*H_m*L*((-1)^n))/((pi^2)*((2*n - 1)^2));
                lambda = ((2*n - 1)*pi)/(2*L);

                sums1(n) = bn*(sin(lambda*x(j)))*exp(-(lambda^2)*alpha3*time(i));
            end

            u1(j,i) = (H_m*x(j) + T0) + sum(sums1(1:n));

        end

end

figure(fig_numb)
plot(time, u1(8,:), linestyles{j}, 'LineWidth', 2)

for j = 1:length(x)

        for i=1:length(time)

            for n = 1:10

                %n = 1;
                bn = -(8*(M_m - H_m)*L*((-1)^n))/((pi^2)*((2*n - 1)^2));
                lambda = ((2*n - 1)*pi)/(2*L);

                sums1(n) = bn*(sin(lambda*x(j)))*exp(-(lambda^2)*alpha2*time(i));
            end

            w = H_m*x(j) + T0;
            u1(j,i) = (w) + sum(sums1(1:n)) + M_m*x(j);

        end
    end

figure(fig_numb)
plot(time, u1(8,:), '-g', 'LineWidth', 2)
legend('Th8 Experimental Data','Th8 Model 3','Th8 Model 2','Location','northwest')


end
```