PRELAB EXERCISE 1:
LINEAR HEAT CONDUCTION

ME 436 Heat Transfer

Introduction

The objective of this experiment is to explore *Fourier's Law* for linear conduction by studying steady-state heat transfer through various materials. Completion of this assignment will provide you with the necessary tools to properly study and explore your collected lab data.

Prerequisites

Before attempting this pre-lab assignment, it is imperative that you have:

- Reviewed the ***textbook sections:*** 2.1, 2.2 & 3.1-3.1.4,
- Reviewed the ***experiment procedures***,
- Watched the ***pre-lab videos*** on Blackboard (Bb), and
- Completed the ***pre-lab quiz*** on Bb.

Getting Started

First, be sure to download the starter code from Blackboard (Bb), and *extract* (unzip) its contents to the directory in which you wish to complete the exercise[1]. Before attempting, make sure that you have completed all of the *prerequisites* below.

Files included in this exercise:

Once you have unzipped the contents of the starter package, you should see the following files:

- `ex1.m`

[1]*Note: when using the Heat Transfer Lab computers, you must extract your code to the local hard drive:* `C:/temp/`*, NOT to the server (ie., Desktop, Group Folder, &, etc.)*

```
  – /lib
  – /data
[⋆] plotData.m
[⋆] calc_ks.m
[⋆] fouriers_law.m
[⋆] calc_contact_res.m
```

⋆ indicates files that you will need to complete.

Throughout this exercise, you will be using the script `ex1.m`, but will not be required to make any major modifications to it. You are only required to modify the functions *(often only 1-2 lines of code)* in the files specified above. Your submission will consist of a compilation (via word processor) of *'deliverables'*, specified throughout this document. Instructions for each deliverable will be highlighted in a blue box, similar to the following:

> **Deliverable 0.** ***Sample***s
> Follow the instructions in this blue box.

0. Warm-Up & System Check

Before starting, it's often a good idea understand the data by first visualizing it. This section will walk you through the essentials of doing so. *If you have any difficulties getting started, be sure to ask your TA for assistance.*

First, open the Matlab script `ex1.m` and read through the instructions. The section immediately following, clears the workspace and initializes our paths and raw data variable. Next, we get to a section titled *'Load Data'*. Here, we load a sample dataset from a text file and store it into the variable `M`:

```
% load tab separated data
M = load('data.txt');
```

Now, we have a matrix M of size 93×11 $(rows \times columns)$, or $M \in \mathbb{R}^{93\times 11}$. The columns in M correspond to:

1	2	3	...	9	10	11
$time(s)$	TC_1	TC_2	...	TC_8	$Voltage, V$	$Current, A$

In which TC_n refers to the thermocouple number, read in degrees Celsius. Next, you will see a line that looks similar to the following:

```
% COMMENT ME OUT!!
break_msg; dbstack; return;
```

This is what we refer to as a `return` command. You will need to *comment out* this line (place a % out front) in order to continue.

Next, let's separate out this data into more meaningful variables.[2]

```
t = M(:,1);         % get time vector, [s]
dat = M(:,2:9);     % get temps, [C]
V = M(:,10);        % Voltage, [V]
I = M(:,11);        % Current, [A]
```

Then we simply plot temperature with time:

```
figure;         % open new figure
h = plot(dat);  % plot the data
set(h, 'LineStyle', '—', 'LineWidth', 2)   % make sure the lines are readable
```

```
% don't forget to add labels!!
xlabel('Time [s]');
ylabel('Temperature [C]');
title('Transient Data, T/C','FontSize',16);
grid on
```

Now, if everything was done correctly, you should get something similar to Fig. 1 below.

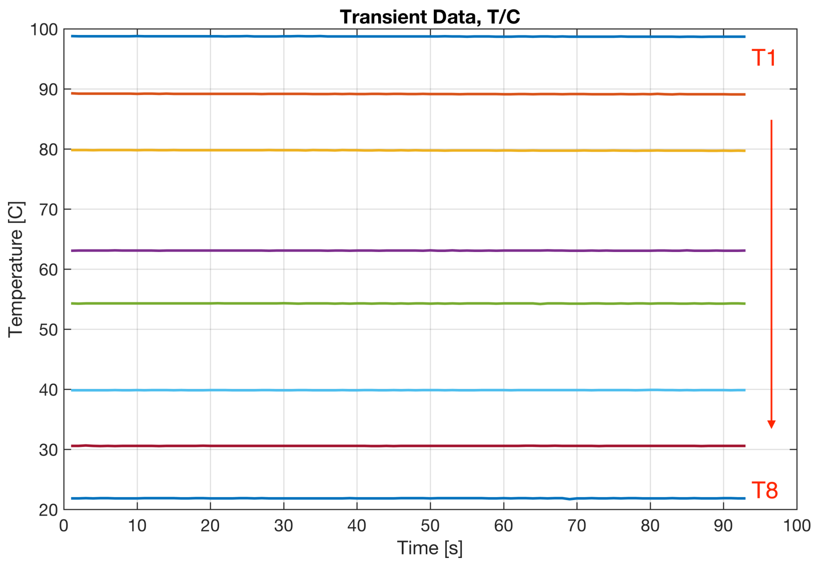

Figure 1: *Transient temperature data*

A few quick questions to ask:

[2]*Note: if you are not familiar with the syntax, check out: Matrix indexing in Matlab*

Question Set: 1. *Sanity Check.*

- Does your plot match Fig. 1 above?
- *Are you using the correct version of `MATLAB`?*
- Is all of our data at steady-state?

Checkpoint - If you had any difficulties obtaining Fig. 1, **stop here** and contact your Lab Instructor. You may run into more troubles ahead.

When you're ready to continue, comment out the following line:

```
% remove break
break_msg; dbstack; return;
```

---

1. Plot Steady-State Data

Now, let's plot the spatial temperature distribution. To do this, we simply average the transient data. This is easily done in `MATLAB` by the $mean()$ command:

```
% take an average of the last (~20) points
N = 20;
Tm = mean(dat(end-N:end,:));      % [C]
Vm = mean(V(end-N:end));          % [V]
Im = mean(I(end-N:end));          % [A]
```

Next, we call a `plotData.m` function to plot our data. As-it-is, this plot function *is incorrect* and will not run properly. Therefore, you will need to open `plotData.m` and adjust code accordingly in order for it to run properly. The sections that you will need to complete are outlined:

```
% plot first three T/Cs
plot(x(2:4),y(1:3),'ko-','MarkerSize', 8);

% plot middle two T/Cs
% ====================== YOUR CODE HERE ======================

% plot last three T/Cs
% ====================== YOUR CODE HERE ======================


% set labels & axis limits
% ====================== YOUR CODE HERE ======================
xlabel('');
ylabel('');
```

- *Note: Pay close attention to the instructions.*

When you're finished, you must ***go back*** to `ex0.m` before you can press `Run`. If you attempt to run the function `plotData.m`, you will receive an error. If your adjustments were correct, and `ex0.m` was properly executed, you should obtain Fig. 2.

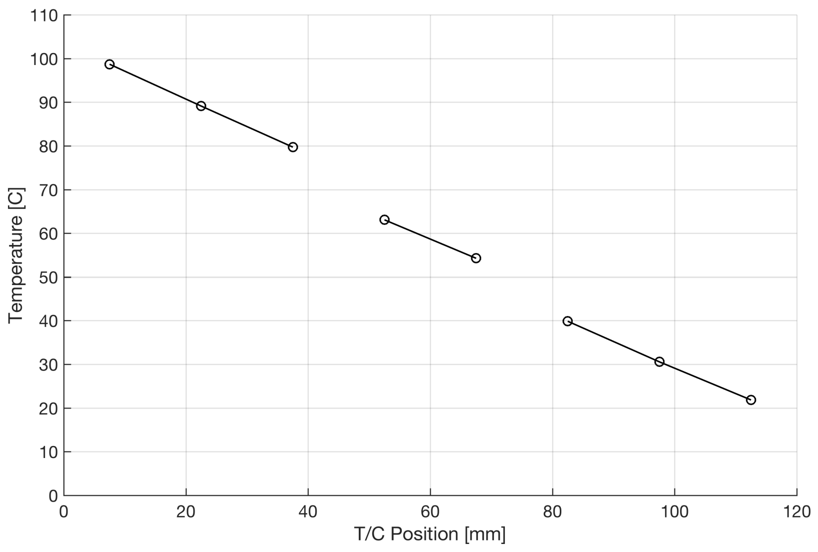

Figure 2: *Transient temperature data*

Then, if everything was done correctly in part 1a, you should arrive at a figure that resembles Fig. 3 below.

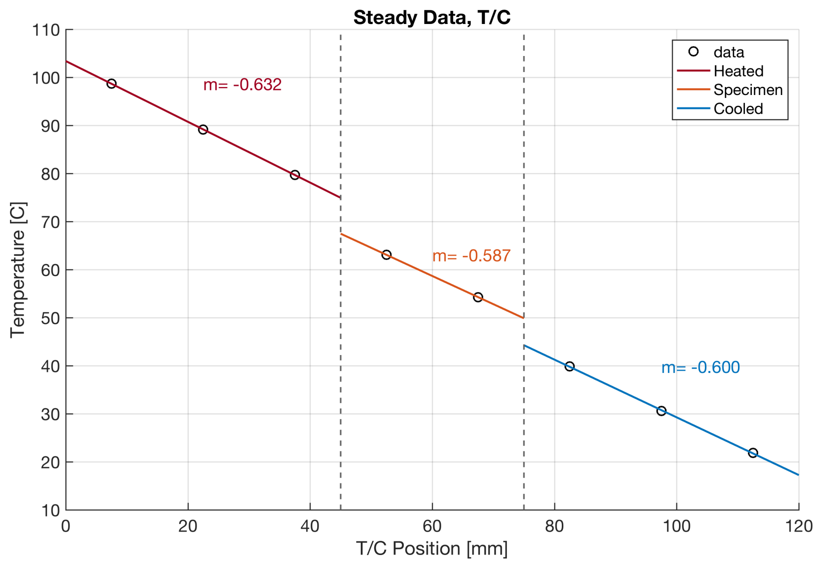

Figure 3: *Steady-state T/C data*

Here, we have added regression lines that extend slightly beyond their representative data points — until they reach a midpoint. This vertical intersection line dividing the two regression lines, is the *theoretical* point of contact between the two sections. Since we physically can't place a thermocouple at the very edge of each material, these *extrapolated* temperature values are the best we can do. We'll revisit these values when we discuss the contact resistance.

> **Deliverable 1.** ***Steady-state T/C plot*** (Fig. 3 above)
> Export your figure to an image: `File » Export Setup » Export`, and include it in your submission.

---

2. Calculate Sample Thermal Conductivity, k_s

In this exercise, you will calculate the thermal conductivity, k_s, of the unknown specimen. This computation is only made possible by one critical assumption: *we can safely assume that a constant heat flux passes through the entire linear conduction apparatus.* Let's quickly revisit how this is possible.

2.1. *Background*

Assume that we have a generic plane-wall, like that shown to the right. Recall from your lecture notes and p113 of the textbook, *one form of the heat equation* may be written as:

$$\frac{d}{dx}\left(k\frac{dT}{dx}\right) = 0 \qquad \textit{(Heat Equation)}$$

Rewriting:

$$k\frac{d^2T}{dx^2} = 0$$

Integrating twice, boundary conditions:

$$T(x) = (T_2 - T_1)\frac{x}{L} + T_1 \qquad (1)$$

with slope:

$$dT/dx = \frac{(T_2 - T_1)}{L}$$

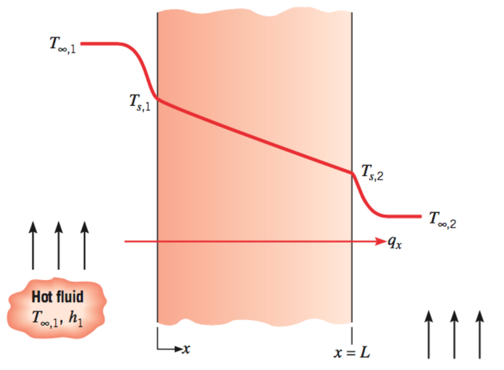

Figure 4: *1D Schematic*

Plugging into Fourier's Law:

$$\begin{aligned} q_x &= -kA\frac{dT}{dx} \\ &= \frac{kA}{L}(T_1 - T_2) \end{aligned}$$

Finally, the *heat flux* is

$$q_x'' = \frac{k}{L}(T_1 - T_2) \qquad (2)$$

From Eqns. (1) & (2) we've shown that the temperature profile is ***linear*** and the heat flux is ***constant***, independent of x.

Now, consider the simple composite wall below. Since the heat flux is constant throughout, we can say:

$$\boxed{\begin{aligned} q_1'' &= q_2'' \\ -k_1\left(\frac{dT}{dx}\right)_1 &= -k_2\left(\frac{dT}{dx}\right)_2 \end{aligned}}$$

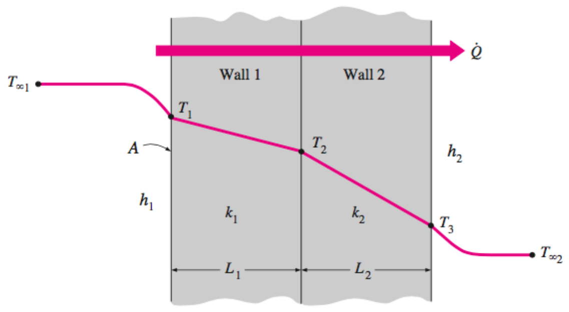

Figure 5: *Composite wall*

Now, using this information, derive an equation for the unknown k_s, before moving on to the next section.

2.2. *Implementation*

Next, we'll prepare a `MATLAB` function to solve for k_s. To do this, open the file named `calc_ks.m` and insert the k_s equation that you just derived. You will notice that the derivative terms have already been supplied.

Deliverable 2. ***(k_s)*** If done correctly, your estimated k_s value will be printed to the `MATLAB` console. Report your results (including units) and provide 1-2 sentences discussing whether your solution is reasonable or not.

3. Calculate Heat Rate

For this exercise, we calculate the heat rate, $q[W]$, for each section of the experiment apparatus. To do so, open up `fouriers_law.m` and insert the appropriate form of Fourier's Law:

$$q_x = -kA\frac{dT}{dx}. \quad \text{(Fourier's Law)}$$

Here, we are looking for the *heat rate*, not *heat flux* - so be careful with units. If your calculations were performed correctly, your `MATLAB` console should read:

```
PART 3: Calculate Heat Rate:

q[W]:

     Pin      qh       qs       qc
    _____    _____    _____    _____
    40.74    37.56    37.56    35.66
```

In which all units are in Watts $[W]$.

Deliverable 3. ***Heat Rate:*** Paste your `MATLAB` console output for this section into your report. Provide 1-2 sentences detailing differences and/or potential loss mechanisms between the P_{in} and your calculated values.

4. Calculate Contact Resistance

Open `calc_contact_res.m` and provide the necessary expressions (see text or lecture notes) for the contact resistance. Your values, if correct, should be on the order of $\approx 5 \times 10^{-5}\,(\mathrm{m^2C/W})$. Next, revisit your Fig. 3 produced above. Are these sharp discontinuities between materials realistic? Consider what factors may play a role in creating these 'jumps' and what could be used to minimize them.

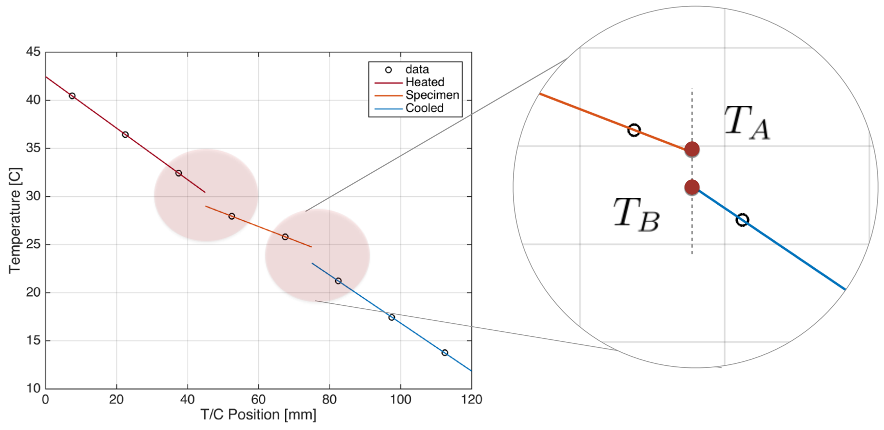

Figure 6: *Transient temperature data*

Deliverable 4. ***Contact Resistance*** Paste your `MATLAB` console output for this section into your report. Provide 1-2 sentences detailing differences and/or potential loss mechanisms between P_{in} and your calculated values.

5. Submission

Compile the above deliverables (blue boxes) into a single document. Be sure to include your name and section number. This assignment will be due when you arrive to class.