MODELING THE TRANSIENT TWO-DIMENSIONAL TEMPERATURE RESPONSE OF CYLINDRICAL GEOMETRY FOR THE ENHANCEMENT OF LEARNING HEAT TRANSFER

A Thesis

Submitted to the Faculty

of

Purdue University

by

Jackson M. Jaworski

In Partial Fulfillment of the

Requirements for the Degree

of

Master of Science in Engineering

May 2020

Purdue University

Fort Wayne, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF THESIS APPROVAL

Dr. Donald Mueller, Co-Chair

    Department of Civil and Mechanical Engineering

Dr. Hosni Abu-Mulaweh, Co-Chair

    Department of Civil and Mechanical Engineering

Dr. Dan Coroian

    Department of Mathematical Sciences

Approved by:

    Dr. Hosni Abu-Mulaweh

        Director of the Graduate Program

I want to thank my mother, Jane, and father, Mark, along with my sister Sarah for all their love and support that went into making me the person I am today. Words can not express how grateful I am to have them in my life. They have been there through it all and supported me in every endeavor. I would lastly like to thank my friends for listening to me drone on about my research when they probably didn't want to and for always pushing me to aspire for better.

ACKNOWLEDGMENTS

I want to thank Donald Mueller, P.E., Ph.D. and Hosni Abu-Mulaweh, Ph.D. for their role as advisers of this thesis. They provided valuable incite and guidance to the project when it seemed like I was heading down a dead-end and always made sure I was on track with the objectives. Their leadership and knowledge was a guiding light in pursuit of finishing this thesis.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

T	Temperature
T_o	Ambient Temperature
T_i	Initial Cylinder Temperature
∇T	Laplacian of Heat Equation
Θ	Dimensionless Temperature
z	Length Direction
r	Radius Direction
θ	Angular Direction
Δr	Increment in r-direction
Δz	Increment in z-direction
$\Delta \theta$	Increment in θ-direciton
r_i	Radius at Given Length
L	Length of Entire Cylinder
r_0	Radius of Cylinder
L_c	Characteristic Length
x	x-direction
$\hat{z}$	Dimensionless Length
$\hat{r}$	Dimensionless Radius
m	Node in z-direction
n	Node in r-direction
M	Total Nodes in z-direction
N	Total Nodes in r-direction
h	Heat Transfer Coefficient
k	Thermal Conductivity
ρ	Density

c_p Specific Heat

α Thermal Diffusivity

t Time

p Current Time Iteration

Bi   Biot Number

Fo   Fourier Number

$\mathcal{O}$ Order of Accuracy

A Area

Q Total Heat Transfer

$\dot{Q}$ Heat Transfer Rate

q'' Heat Flux

$\dot{E}_{in}$ Energy In

$\dot{E}_{out}$ Energy Out

$\dot{E}_{st}$ Energy Stored

C_n Coefficient Values

J Bessel Function

ζ Root of Transcendental Equation

ABSTRACT

Jaworski, Jackson M. M.S.E., Purdue University, May 2020. Modeling the Transient Two-Dimensional Temperature Response of Cylindrical Geometry for the Enhancement of Learning Heat Transfer. Major Professors: Hosni I. Abu-Mulaweh and Donald W. Mueller, Jr, School of Mechanical Engineering.

This thesis considers the topics of heat transfer education, a modern approach to learning, and understanding one-dimensional verses multidimensional problems. The physical problem considered is transient conduction in a short cylinder immersed in an isothermal fluid. Many aspects can be modified, such as the material properties, but the length and radius are of primary concern. The thesis introduces the concepts required to develop a numerical method for solving the temperature gradient within the cylinder. This method is programmed in MATLAB with a graphical user interface allowing for interactive learning by performing iterative tests to discover various concepts; which can have a significant impact on learning. Many published research articles detail the effectiveness of incorporating hands-on computer programs into the heat transfer curriculum. The interaction effects from the inputs are also analyzed using a design of experiments full factorial method to determine which inputs are the most significant concerning the error between the one- and two-dimensional solutions. The main effects are the length and radius by a significant amount followed by the time, material, initial temperature, and convection coefficient. Finally, the program is used to develop a chart which given the geometry, material properties, and Fourier number can tell the user precisely when the one-dimensional assumption for both a slab and infinite cylinder breaks down.

1. INTRODUCTION

Engineering is a daunting subject that includes multiple complex sub-disciplines. One of the more challenging sub-disciplines for mechanical engineering students is heat transfer which describes how energy is transferred through or between objects due to a temperature difference. Students are typically introduced to the transient, or time-varying, conduction heat transfer problems with simple one-dimensional geometries. While these examples are adequate to introduce concepts, in practice, an engineer should know when a one-dimensional analysis is applicable or when a multi-dimensional analysis is required.

One-dimensional models introduced in classes are used to avoid the complex nature of multi-dimensional geometries which often require the development of a numerical method simulation to obtain a solution. Even an analytical solution of a one-dimensional cylindrical geometry involves complex terms such as Bessel functions that require computer solutions to quantify the behavior. Having students develop these types of programs on their own, while useful, would only take time away from another equally important topic in an introductory heat transfer course.

1.1 Thesis Motivation

The primary motivation of this thesis is to help students grasp engineering topics in the field of heat transfer and develop better classroom methods and experiments to aid in their learning. Most topics not only in the field of heat transfer but engineering in general are taught with a deductive approach. The deductive method of learning is starting with a general principle or theorem then working out the specifics and applying the theories to experiments and data. Research suggests that a deductive teaching style is more effective for short-term learning; however, it gives the learner

a sense of never being able to come up with the general formula themselves. This causes a shallow understanding of the material and a lack of confidence since it makes the material seem straightforward to develop [1].

Another method for learning would be through an inductive approach. An inductive approach involves starting with observations and data from experiments and then developing the principles. This method works inversely from the deductive method and is more in line with how the original principles were developed in the first place and how most research is conducted. Typically this type of learning gives the student a hands-on experience with the material allowing for longer retention and deeper understanding.

Specifically relating to heat transfer, when being introduced to the heat transfer in cylinders, students are taught of a clear boundary between when to use a particular type of solution based on the geometry of the cylinder. While this assumption holds generally, there are more factors which should be considered that determine whether a one- or two-dimensional solution should be applied; one of the most significant being the duration of the study. For every case there will be a small amount of time at the beginning of the study where the results are similar since they will start at ideally the same conditions. The two studies will quickly differ but depending on all the considering factors this could be a significant amount of time and useful for practical applications in large systems.

This leads into the secondary motivation of this thesis which is to develop a method either graphically or though a computer simulation that can calculate the exact Fourier number at which a one-dimensional analysis becomes inaccurate and a two-dimensional analysis is needed. The typical general assumption for conduction in a cylinder in a heat transfer textbook will list that if the length is at least ten times larger than the radius, a one-dimensional analysis can be performed [2]. This is due to the heat transfer through the ends of the cylinder having a negligible effect on the center point temperature. While being a useful assumption, there is a large amount

of uncertainty between at what the exact point is where a two-dimensional analysis must be utilized to reduce the error the assumption will cause.

1.2   Thesis Objectives

This thesis aims to develop an interactive, flexible, and intuitive program for engineering students to study transient, multi-dimensional conduction heat transfer problems in cylindrical geometry. This interactivity allows for a better understanding of fundamental concepts; specifically, to aid as a demonstration that under certain conditions, a plane wall or an infinite cylinder can approximate a finite cylinder. The thesis will focus on the particular case of cylinders initially at a uniform temperature as shown in Figure 1.1. The primary boundary condition will consist of convection from a large isothermal fluid body while neglecting buoyancy effects. The development of this program will be discussed to formulate a general process to follow when developing complex programs, specifically when trying to create an interface. This interface is critical to help provide a medium that is approachable for students as to not get overwhelmed.
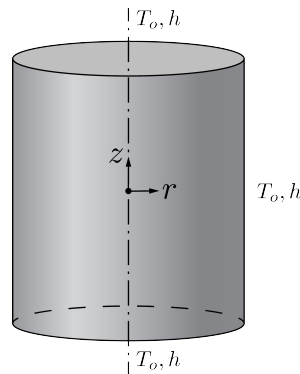
Figure 1.1.   Geometry of a short cylinder subjected to convection boundary conditions of a uniform temperature.

The secondary objective of the thesis is to develop a method of determining when a physical situation requires a two-dimensional simulation. In other words, when does

the one-dimensional assumption break down for a particular geometry and develop a clear boundary between the two cases. A graphical element would be the easiest form to comprehend this data since many charts and graphs currently exist in the field to determine other properties. This would allow the user to calculate some dimensionless properties and determine at what length of time a one-dimensional study exceeded a particular error threshold, i.e., no longer valid.

This thesis will attempt to accomplish the following to satisfy the previously mentioned objectives:

1. Develop a mathematical model for a cylinder at a uniform initial temperature subjected to a large fluid body of uniform temperature.

2. Propose a finite difference solution method capable of changing each variable.

3. Develop a user interface that allows the user to learn by changing variables.

4. Utilize the MATLAB software package to implement the proposed solution and user interface.

5. Develop a single or multiple graphical element(s) that can be used to determine if a particular cylinder requires a one- or two-dimensional analysis.

1.3   Thesis Organization

The thesis will be organized into a total of six chapters which will cover the background information on conduction heat transfer, the theory of the developed model, the simulation and GUI development, the results drawn from the work, and conclusion of the work with future recommendations. A detailed breakdown is structured into chapters as follows:

1. Introduction. Outline the basic concepts and motivation behind the work described in the following chapters.

2. Background. Existing literature is reviewed to provide all the necessary background information required to understand the work of this thesis.

3. Theory. Introducing a mathematical model for the problem including the stability analysis.

4. Model Development. The outline for a computer simulation is presented based on the mathematical model introduced in the prior chapter. Additionally, introducing general information about GUI development, along with the development of the specific GUI for the thesis.

5. Results. The theory from chapter three is analyzed using the model and GUI from chapter four to conclude one- and two-dimensional assumptions to a finite cylinder.

6. Conclusion. All work from the previous chapter is brought together, discussing the significance, limitations of results, and suggesting future work.

2. BACKGROUND

2.1 Aims

This chapter covers the relevant background information referred to in the rest of the thesis. The basics of conduction heat transfer will be introduced along with a review of existing literature on education techniques in heat transfer curriculum.

2.2 Conduction Heat Transfer

Heat transfer is the science which studies how energy is transferred through or between objects due to a temperature difference. This broad definition is an umbrella term which covers three modes in which energy, in the form of heat, can be transferred. This thesis focuses exclusively on conduction heat transfer, which is the transfer of energy from more energetic particles of a substance to adjacent less energetic particles due to interactions between these particles [2]. Thus, in the presence of a temperature gradient, the higher energy particles transfer energy to lower energy particles or in the direction of decreasing temperature. This type of net transfer of energy between particles is defined as diffusion, shown visually in Fig. 2.1, which describes the random molecular motion [3].

Heat conduction is classified as one-dimensional when conduction is significant in one direction only and negligible in the others, two-dimensional when conduction is significant in two directions and negligible in the last, and three-dimensional when conduction is significant in all directions. Most introductory courses in heat transfer deal primarily with one-dimensional heat problems. If a two-dimensional problem is proposed, it typically can be reduced into a more straightforward one-dimensional case.
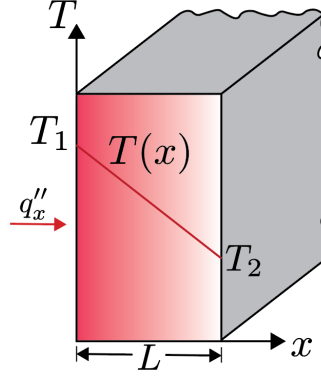
Figure 2.1. One-dimensional heat transfer by conduction as outlined by Fourier's law, Eq. 2.1, describing the diffusion of energy.

Since heat transfer deals with the transfer of energy between objects, the energy transfer must occur over some segment of time. A transient or time-varying analysis describes temperature as a function of time, while an analysis is described as steady-state if the temperature does not depend on time. This thesis looks at the center point temperature between one- and two-dimensional models over a period of time; thus, classifying the work as a transient heat transfer problem.

A fundamental equation that applies to conduction is Fourier's law, i.e.,

$$\dot{Q}_x = -kA\frac{\partial T}{\partial x} \tag{2.1}$$

which describes the heat rate by conduction through a particular area, in this case, the x-direction. Dividing by the cross-sectional area is the heat flux

$$q''_x = -k\frac{\partial T}{\partial x} \tag{2.2}$$

which is the heat transfer rate per area perpendicular to the direction under investigation. The minus sign in the equation is from the inherent nature of the energy moving in the direction of decreasing temperature.

2.3   Heat Conduction in Cylindrical Coordinates

One objective in a conduction analysis is to determine the temperature distribution for a given medium which can be done by developing a heat diffusion equation. The heat flux, Eq. 2.2, can be determined at a point or from a surface by the use of Fourier's law, Eq. 2.1. It is important to note that the temperature at any point is a scalar quantity while the heat flux is a vector quantity. When the temperature distribution within a medium is not uniform, this creates a heat flow where energy always moves from a region of higher temperature to a region of lower temperature.
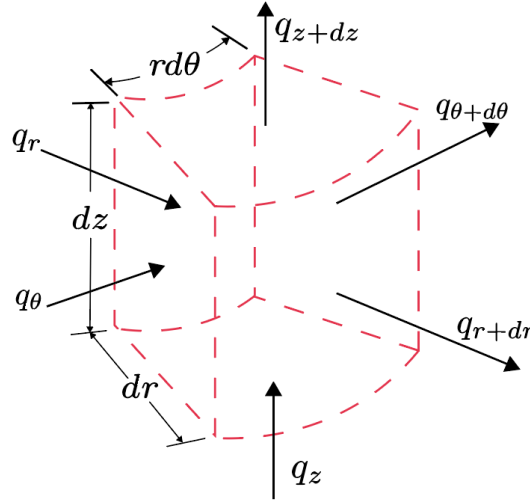
Figure 2.2.   Differential control volume for heat conduction in the cylindrical coordinate system (r, θ, z).

Since this thesis primarily is concerned with cylindrical geometry, the resulting equation well be in the cylindrical coordinate system. The first step in deriving the differential equation is starting with a stationary cylinder of constant material properties without heat generation. The process begins by defining a differential control volume over a medium in which is under investigation. There is energy entering the differential element from the radial, vertical, and angular directions which will then leave the element from the opposite respective surface, which can be seen in Fig. 2.2.

This relation can be written out mathematically by creating an energy balance over the element as follows:

$$\dot{E}_{in} - \dot{E}_{out} + \dot{E}_{gen} = \Delta \dot{E}_{st}. \tag{2.3}$$

Assuming that the medium is of constant material properties, no heat generation, and only in two-dimensions $T(r, z, t)$ the final heat diffusion equation is as follows:

$$\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial T}{\partial r}\right) + \frac{\partial^2 T}{\partial z^2} = \frac{1}{\alpha}\frac{\partial T}{\partial t}. \tag{2.4}$$

2.4 Discrete Approximation of Derivatives

Finite difference equations are developed around the Taylor series expansion about a point, called x, which includes the additional higher-order terms. Looking at the Fig. 2.3, there is a continuous function which contains points both in a positive (forward) direction and in a negative (backward) direction. There are three primary
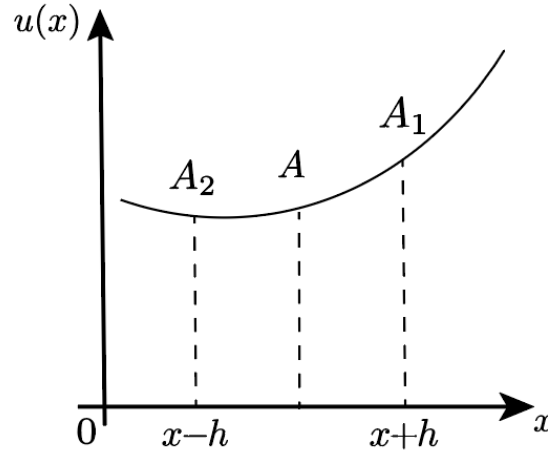
Figure 2.3. Derivative approximation of an arbitrary function $u(x)$.

ways of developing equations to approximate the derivatives to these points being a forward, backward, and central difference method. Each standard form of the Taylor series expansion utilizes two of these points to approximate a curve between; however, there are higher-order Taylor series expansions that utilize more points that will be

briefly covered at the end of this section. Using the Taylor series expansion in a forward direction yields

$$f(x + \Delta x) = f(x) + \Delta x \frac{df(x)}{dx}\bigg|_x + \frac{\Delta x^2}{2!} \frac{d^2 f(x)}{dx^2}\bigg|_x + \frac{\Delta x^3}{3!} \frac{d^3 f(x)}{dx^3}\bigg|_x + \cdots \qquad (2.5)$$

while in a backward direction yields

$$f(x - \Delta x) = f(x) - \Delta x \frac{df(x)}{dx}\bigg|_x + \frac{\Delta x^2}{2!} \frac{d^2 f(x)}{dx^2}\bigg|_x - \frac{\Delta x^3}{3!} \frac{d^3 f(x)}{dx^3}\bigg|_x + \cdots . \qquad (2.6)$$

These two equations are the basic building blocks for developing finite difference approximations for the first derivative about a point. The downside to using the Taylor series expansion is that it is an infinite series by definition. Due to the infinite nature of a Taylor series there will always be truncated terms left out when approximating a derivative. Looking at Eqs. 2.5 and 2.6, the first derivative to the function can be solved for while grouping all other higher order terms together. The forward direction results in

$$\frac{df(x)}{dx}\bigg|_x = \frac{f(x + \Delta x) - f(x)}{\Delta x} + \mathcal{O}(\Delta x) \qquad (2.7)$$

while the backward direction results in

$$\frac{df(x)}{dx}\bigg|_x = \frac{f(x) - f(x - \Delta x)}{\Delta x} + \mathcal{O}(\Delta x) \qquad (2.8)$$

where $\mathcal{O}(\Delta x)$ is the grouped higher order terms or the truncation error associated with the finite difference approximation [4]. This error is the difference between the derivative and the finite difference form which for Eq. 2.7 is represented by

$$\mathcal{O}(\Delta x) \equiv -\frac{\Delta x}{2!} \frac{d^2 f(x)}{dx^2}\bigg|_x - \frac{\Delta x^2}{3!} \frac{d^3 f(x)}{dx^3}\bigg|_x - \cdots . \qquad (2.9)$$

2.5 Errors Associated with Finite Differences

While developing a set of algebraic equations to approximate derivatives, errors begin to propagate in the solution. There are two main types of errors being truncation error and round-off error. Additionally, there can be stability issues with utilizing

an explicit scheme causing the resulting solution to be unstable if the step size is not within a specific bound.

Explicit finite difference solutions are subject to stability issues due to the forward difference in the time dimension. If the time step is not less then or equal to the associated terms, it causes the solution to oscillate between arbitrary values. This topic is discussed in more specific details in chapters 3 and 4 relating to the stability criteria. The stability issue can overall be avoided by utilizing a difference scheme such as the Crank-Nicolson or the DuFort-Frankel methods [5].

2.6 Total Heat Transfer

The total heat transfer from a body is a particularly interesting quantity to study and is useful in many engineering applications in the field of HVAC. The total heat transfer is usually calculated by running a steady-state analysis on the object in question; however, it can be useful to determine the instantaneous heat transfer at a given time or the total transfer up until a particular point. There are two primary ways of calculating this value, it can be calculated by looking at the heat flux between the boundary nodes of the object over time, or through analytical calculations if the geometry is of relative simplicity such as a cube or cylinder. In this thesis, the nature of cylindrical geometry will allow for use of the later method for simplicity and will be discussed further in chapter 4.

2.7 Average Temperature

A useful temperature to know within a solid body is the average temperature of the body. This temperature is difficult to calculate in cylindrical and spherical geometries since the small differential elements that make up a mesh have different volumes and must be weighted differently. A small piece on the inside should not have as large as an effect as a large piece on the outside. In order to calculate the

mean temperature, the average of the local instantaneous temperatures is taken for an infinite slab

$$\frac{T}{T_i} = \frac{1}{x} \int_0^x T dx \qquad (2.10)$$

and for an infinite cylinder

$$\frac{T}{T_i} = \frac{1}{\pi r_1^2} \int_0^r 2\pi r T dr \qquad (2.11)$$

where T is the current temperature and T_i is the initial temperature [6]. So for the infinite cylinder and infinite slab, each differential piece is averaged for its corresponding amount of length. The first and last pieces in the lines are special because they may involve a differential piece half the size since the temperatures are at the center-line and edge of the body. These will have a slightly smaller length compared to other pieces, and the center node will need to be located off-axis slightly to avoid a zero in the calculation for the location.

For the two-dimensional study where there are both a radial and length directions, the weighted average was taken for both directions. Eq. 2.10 is used to calculate the average temperature in the length direction since the pieces are of equal volume while Eq. 2.11 is used to calculate the average temperature in the radial direction since the pieces increase in volume as there is an increase in r. These two average temperatures are then averaged to get the mean temperature of the two-dimensional cylinder.

2.8   Characteristic Length and Geometry Ratio

The characteristic length is defined as the primary direction in which heat travels. Most heat transfer textbooks suggest this as a ratio of the volume of the object to its area [2, 3, 7]. This method works fine when dealing with a single type of analysis being an infinite cylinder, infinite slab, or a two-dimensional study; however, since this thesis will be comparing the results from all three examples, a characteristic length must be chosen that represents all three analyses. Table 2.1 presents is the various ways, derived conservatively, the characteristic length can be represented.

Table 2.1. Different ways to represent characteristic length.

Type	Equation
Volume to Area Ratio:	$L_c = \dfrac{V}{A} = \dfrac{r_0}{2 + 4r_0/L}$
Volume Ratio:	$L_c = \sqrt[3]{V} = \sqrt[3]{\pi r_o^2 L/2}$
Area Ratio:	$L_c = \sqrt{A} = \sqrt{2\pi r_0(L/2 + r_0)}$
Magnitude:	$L_c = \sqrt{r_0^2 + (L/2)^2}$
Infinite Slab:	$L_c = L/2$
Infinite Cylinder:	$L_c = r_0$

A technical approach can be applied by scaling the coordinates of the Laplacian in the heat equation

$$
\begin{aligned}
\nabla^2 T &= \Delta T \left[\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial \Theta}{\partial r} \right) + \frac{\partial^2 \Theta}{\partial z^2} \right] \\
&= \Delta T \left[\frac{1}{\hat{r} r_0^2}\frac{\partial}{\partial \hat{r}}\left(\hat{r}\frac{\partial \Theta}{\partial \hat{r}} \right) + \frac{4}{L^2}\frac{\partial^2 \Theta}{\partial \hat{z}^2} \right]
\end{aligned}
\tag{2.12}
$$

where $\hat{r} = r/r_o$, $\hat{z} = z/(L/2)$, and $\Theta = (T(r,z,t) - T_{min})/\Delta T$ [8]. This results in two extreme possibilities for the specific problem. Either the radius is much less then the length of the cylinder, or the length is much less then the radius of the cylinder. Performing an order of magnitude analysis on the former results in

$$
\begin{aligned}
\nabla^2 T &= \left(\frac{\Delta T}{r_0^2} \right)\left[\underbrace{\frac{1}{\hat{r}}\frac{\partial}{\partial \hat{r}}\left(\hat{r}\frac{\partial \Theta}{\partial \hat{r}} \right)}_{\mathcal{O}(1)} + \underbrace{\frac{4r_0^2}{L^2}}_{\ll 1}\underbrace{\frac{\partial^2 \Theta}{\partial \hat{z}^2}}_{\mathcal{O}(1)} \right] \\
&\approx \left(\frac{\Delta T}{r_0^2} \right)\left[\frac{1}{\hat{r}}\frac{\partial}{\partial \hat{r}}\left(\hat{r}\frac{\partial \Theta}{\partial \hat{r}} \right) \right].
\end{aligned}
\tag{2.13}
$$

Thus the Laplacian is on the order of magnitude of $\Delta T/r_0^2$ and negligible in the length direction. A similar analysis can be performed on the length direction resulting in

$$\nabla^2 T = \left(\frac{4\Delta T}{L^2}\right)\left[\underbrace{\frac{L^2}{4\hat{r}r_0^2}}_{\ll 1}\underbrace{\frac{\partial}{\partial\hat{r}}\left(\hat{r}\frac{\partial\Theta}{\partial\hat{r}}\right)}_{\mathcal{O}(1)}+\underbrace{\frac{\partial^2\Theta}{\partial\hat{z}^2}}_{\mathcal{O}(1)}\right]$$
$$\approx \left(\frac{4\Delta T}{L^2}\right)\left[\frac{\partial^2\Theta}{\partial\hat{z}^2}\right]. \tag{2.14}$$

Thus the Laplacian is on the order of magnitude of $4\Delta T/L^2$ and negligible in the radius direction.

This type of analysis is useful to see the exact scaling along a particular axis. While useful, the goal is to develop a robust program which can work in both the one-dimensional cases as well as the two-dimensional case. The specific directional characteristic lengths also miss the heat flux at the corners of objects as the order of magnitude would be of the same order. As explained earlier in the section the typical way to define a characteristic length is through a volume to area ratio. This only provides a crude result which essentially is an average of the heat flux throughout the entire solid. From the review of literature and consideration for the specific problem the characteristic length in this thesis is chosen to be from the magnitude of the length and the radius of the cylinder. This is the most conservative with the corner error while converging to a specific direction if sufficiently larger then the other. This is an area of potential improvement and further research should be conducted on the matter.

A term used frequently in chapters 4 and 5 is the ratio between the length and radius. Since the program will primarily be working with center temperature and using a quarter of the cylinder, the length is divided in half resulting in

$$R = \frac{L}{2r_0}. \tag{2.15}$$

This ratio allows a direct comparison between the length and radius and is useful for defining the final chart generated in chapter 5.

2.9   Design of Experiments

With the rise of continuous improvement techniques in industry and manufacturing, much research has been focused into the six-sigma field of optimizing the processes to eliminate waste and boost productivity. A useful tool used to discover significant variables in a process has been developed from the field of statistics termed a design of experiments (DoE). It is a branch of applied statistics which looks at planning, conducting, analyzing, and interpreting experiments to evaluate all the factors of a parameter or group of parameters [9].

When performing tests in an engineering environment, it is often tempting to keep all variables constant while changing a single parameter and seeing how that parameter affects the simulation. While this is a useful test, it neglects the interaction effects of multiple parameters when varied together. Varying multiple inputs together at different levels allows the interaction effects between parameters to be calculated by forming a statistically based approach of developing tests to view interaction effects. There are two main types of DoE's, full factorial and fractional factorial in which the former is presented in chapter 5.

2.9.1   Full Factorial

There are four essential features when it comes to creating a DoE: the number of input parameters, number of levels for which to set the inputs, the output(s) of the system, and the number of experiments needed to investigate the all the interaction effects between the inputs. The reason this particular type of DoE is called a 'full factorial' design is that every response of every combination of inputs and input levels are studied. The results of the system can indicate which input parameters have a significant effect on the output and help the operator achieve optimized results.

2.9.2 Fractional Factorial

The other primary type of DoE is a fractional factorial experiment in which only a fraction of experiments is conducted instead of every possible combination. This method could be useful in an extensive system with many inputs where it is not feasible to look at every type of experiment. The fractional factorial design uses the same procedure as the full factorial, but only specific experiments are run in which the operator believes will generate significant results. While this type of analysis may be seen as inefficient, many interactions past three inputs tend to be negligible while the interact effects of a single input on the output are typically visible. This type of analysis reduces the number of experiments and tries to focus on tests which will generate significant results.

There are a few advantages of this type of experiment, the foremost being that most results tend to be significant. Since only a selected few studies are run in which significant results will most likely be generated, the interactions can be found right away without having to perform dozens if not hundreds of tests. This method can lead to an immediate impact on the system since the results can be implemented right away without the need for analyzing a large amount of data from a full factorial experiment. This method is less time consuming and will lead to less downtime on a particular system or machine. If there is only a certain amount of time dedicated to these types of studies in the workplace, a fractional factorial could easily be performed during a maintenance hour or break shift.

The distinct disadvantage of this type of experiment is that not every interaction is analyzed. Since the experiments are ultimately cherry-picked, meaningful interactions could be missed. With possible significant interactions missing, it could lead to the process or system not being as efficient as possible. With the experiments hand-chosen, there is always the possibility insignificant runs are chosen which do not produce interesting results wasting time spent on the study.

2.10   Literature Review

Analyzing associated error of applying a one-dimensional assumption to a finite geometry in transient conduction has not been researched in recent literature. The causes of this error are outlined in many textbooks and papers; however, there is a lack of a comprehensive model defining thresholds of error for specific conditions. There have been multitudes of papers published on engineering education and a surprising amount in the field of heat transfer education. The following is a historical perspective of published papers in engineering education.

2.10.1   Learning and Teaching Styles in Engineering Education

There are two primary ways of conveying information to students in a lecture setting: just giving the students the information they need through formulas and theorems, or having the students develop the fundamentals through observing phenomena and data. The former method referred to as a deductive method of teaching, while the latter is referred to as an inductive method of teaching. While both methods convey the same information, the "best" method below the graduate school level is induction [10]. Felder's paper is the earliest publication that outlines the teaching and learning styles of professors and students. While many styles are discussed and evaluated, the authors go on to outline teaching techniques to best address all learning styles, such as following the scientific method, applauding creative solutions, and using computer-assisted instruction.

2.10.2   Non-Use of Computers in Undergraduate Engineering

Published a decade after the Felder's paper regarding learning and teaching styles in engineering education, Jones takes an informal look at the lack of computer use in engineering classrooms [11]. While most of his data collection is preliminary, the commentary on professor hesitation to implement computer and simulations into the

classroom is repeatedly seen in most responses. Unfortunately, Jones never published a follow-up paper with scientifically rigorous methods applied to the data. The fundamental problem stems from teachers failing to implement new technology into lectures and continuing the way it was taught to themselves originally. While learning software packages and putting together new lectures is time-consuming, this can be simplified by utilizing what the teacher already knows how to use, such as excel, word, and some programming language to name a few. Computers are meant to enhance students learning and re-enforce fundamental principles.

2.10.3   An Inductive Approach to Teaching Heat and Mass Transfer

The first paper to take Felder's work and implement some of the specific styles to develop a curriculum for their heat transfer classes [1]. Farrell and Hesketh take the inductive learning approach by adding a studio component to their classes where students get to work and play with models, both physical and virtual, to learn new topics. They start with seeing what is happening, and then they get to infer the governing principles behind what is occurring. This method helps make the information more relative since the students help derive the information themselves. Many of these principles were discovered initially, making the formulas seem less abstract.

2.10.4   Application of the Studio Model to Teaching Heat Transfer

Ribando is another academic that has been developing models and creating a studio model for his heat transfer courses. In his first published paper on the issue, he discusses his approach to constructing the studio model, which is similar but slightly differs from Farrell and Hesketh [12]. Physical laboratory experiments are excellent for teaching students but can be expensive, hard to develop, and hard to implement. Ribando's theory is to replace many outdated techniques such as looking at Heisler charts and convection tables and turning them into lookup tables and spreadsheets on a computer. Even a simple computer solution such as this can make a topic much

more approachable and more comfortable to learn if one does not have to spend time learning the tools to begin to solve a problem. The approach is to design a specific program that looks at and solves a particular problem to avoid the learning curve of general simulation software.

2.10.5 New Roles for Engineering Faculty

While not directly related to engineering education, Smerdon takes a look at how engineering faculty skill-sets are utilized within engineering departments and how to better utilize facuulty to increase the education process of engineering students [13]. Smerdon's main remark is to increase the efficiency of teaching and learning to re-engineer the engineering degree to help students succeed. Many students drop in the freshmen and sophomore years due to the difficulty and lack of excitement. Students need some hook that gets them into the program and excited to jump into the field of engineering. With students excited about the field, when their courses get tough, they are more resilient and less likely to drop out of the program. Computers and hands-on experiments are fun and allow students to see what is going on in an engineering problem.

2.10.6 "Hands-on" Approach to Teaching Undergraduate Heat Transfer

Published three years after his first paper, Ribando [14] provides an update on the status of his studio component for heat transfer classes. The first paper was more of an outline of how Ribando and colleagues implemented the studio portion. The updated paper introduces how and why the program was implemented but also provided observations and assessments on how the program performaned. Visually, all program are simple with charts and graphs to view and a variety of options to change intuitively with buttons and sliders. From the feedback given during assessment, the programs were highly successful and helped students better understand the lecture

material, with the only complaint being that partners for the studio were randomly assigned.

3. THEORETICAL WORK

3.1   Aims

This chapter covers the development of a mathematical model governing the con-
duction heat transfer within a cylindrical geometry at a uniform initial temperature
subject to a fluid at a uniform temperature. Firstly, developing a mesh for the two-
dimensional simplification and then deriving the mathematical equations from the
mesh at the corresponding nodes. The mesh has four edges and four corners along
with interior nodes requiring a total of nine equations to define the mesh fully.

3.2   Mesh Development
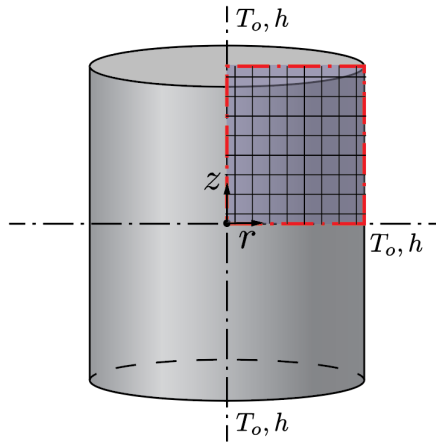
Figure 3.1.   Location of simplified mesh within cylindrical geometry.

A critical factor that the finite difference equations are derived from is the mesh
which governs the geometry. An advantage of cylindrical geometry under the given
boundary conditions is the symmetric nature created allowing for simplification of
the mesh as shown in Fig. 3.1. Since the top, side, and bottom boundary conditions

are assumed to be exactly the same, this creates an axis of symmetry about the mid-plane of the cylinder. The temperature distribution will be identical on the top and bottom halves, meaning that the bottom half can be ignored for simplification.
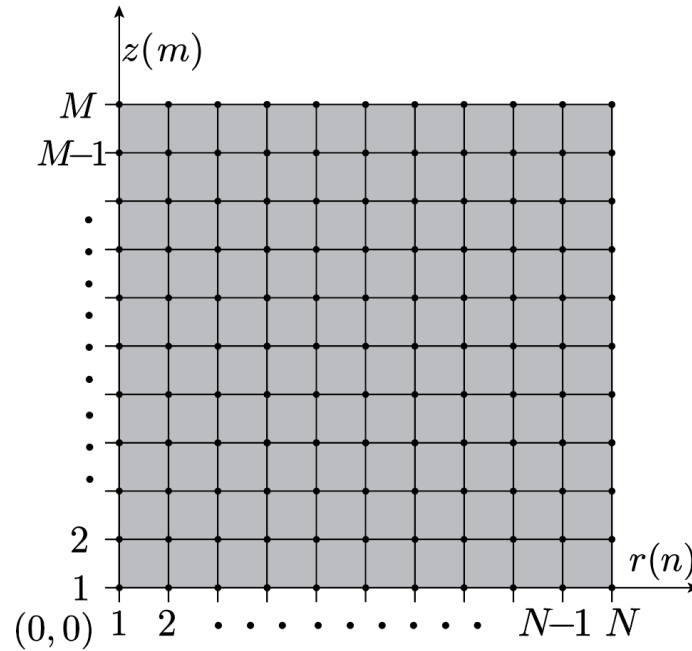
Figure 3.2.   Mesh developed which includes nodes on the centerline and outside surfaces for a quarter of the cylinder.

A similar boundary of symmetry can be developed by utilizing the adiabatic center of the cylinder. The geometry has an axis of symmetry down the center-line meaning that due to the boundary conditions, the radial direction will be identical for every cross section. This allows the mesh to only need to cover a quarter cross-section of the cylinder which will cover the entire temperature distribution of the cylinder. A visualization of the mesh can be seen in Fig. 3.2.

3.3   Revised General Equation

The two-dimensional general conduction heat equation in cylindrical coordinates developed in the previous chapter, Eq. 2.4, works well for analytical cases; how-

ever, this form of the heat equation is difficult to translate into a system model. The difficulty arises due to the radius being within a second order partial derivative with respect to radius causing the issue of not being able to apply any numerical method. Rearranging the equation using the product rule on the partial derivative of temperature with respect to radius results in

$$\frac{1}{r}\frac{\partial}{\partial r}\left(kr\frac{\partial T}{\partial r}\right) = \frac{1}{r}\left[r\frac{\partial k}{\partial r}\frac{\partial T}{\partial r} + k\frac{\partial}{\partial r}\left(r\frac{\partial T}{\partial r}\right)\right]. \tag{3.1}$$

This new form of the general heat equation is still not ideal for working with and can be further simplified by applying the product rule once more. Application of the product rule eliminates the radius from within the partial derivative yielding

$$\frac{1}{r}\frac{\partial}{\partial r}\left(kr\frac{\partial T}{\partial r}\right) = \frac{\partial k}{\partial r}\frac{\partial T}{\partial r} + \frac{k}{r}\left[\frac{\partial T}{\partial r} + r\frac{\partial^2 T}{\partial r^2}\right]$$
$$= \frac{\partial k}{\partial r}\frac{\partial T}{\partial r} + \frac{k}{r}\frac{\partial T}{\partial r} + k\frac{\partial^2 T}{\partial r^2}. \tag{3.2}$$

The result from Eq. 3.2 simplifies the partial radius term in the general heat equation which is much easier to translate into finite difference equations. Notice that the thermal conductivity is a function of at least the radius where the partial derivative with respect to the radius is applied. The thermal conductivity is treated as constant since this thesis deals with uniform material properties causing the partial derivative of thermal conductivity to be zero resulting in

$$\frac{1}{r}\frac{\partial}{\partial r}\left(kr\frac{\partial T}{\partial r}\right) = \frac{k}{r}\frac{\partial T}{\partial r} + k\frac{\partial^2 T}{\partial r^2}. \tag{3.3}$$

Eq. 3.3 can be substituted back into the general heat equation in cylindrical coordinates, Eq. 2.4, which results in a new revised equation:

$$\frac{k}{r}\frac{\partial T}{\partial r} + k\frac{\partial^2 T}{\partial r^2} + k\frac{\partial^2 T}{\partial z^2} = \rho c_p\frac{\partial T}{\partial t}. \tag{3.4}$$

3.4   Left Edge

In order to develop the finite difference equation at the center-line of a cylinder, the revised general equation needs to be modified [15]. The additional modification
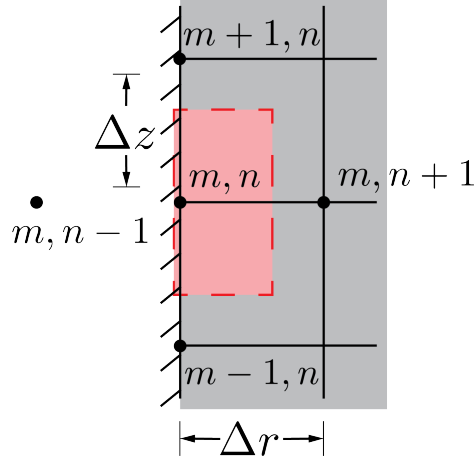
Figure 3.3. Element used for each node on the left edge defined by Eq. 3.8.

from the undefined case is a result from the zero radii at the center-line of the cylinder being in the denominator of the Eq. 3.4. The geometry of the left edge can be seen in Fig. 3.3. Solving Eq. 3.4 requires applying L'Hospital's rule on the first order partial derivative of temperature with respect to the radius and the associated multiplied terms shown in

$$\lim_{r \to 0} \left(\frac{1}{r} \frac{\partial T}{\partial r} \right) = \frac{\partial^2 T}{\partial r^2}. \tag{3.5}$$

The resulting equation is no longer subject to an undefined case and is substituted back into the revised general equation, Eq. 3.4, with the thermal conductivity constant divided through the result yielding

$$2 \frac{\partial^2 T}{\partial r^2} + \frac{\partial^2 T}{\partial z^2} = \frac{1}{\alpha} \frac{\partial T}{\partial t} \tag{3.6}$$

where α is the thermal diffusivity of the cylinder. Eq. 3.6 covers the limiting case where the radii are zero which allows the development of a finite difference form of the solution for the center-line temperatures. The finite difference form is derived using an explicit forward difference method to look at small increments in time and nodal spacing as follows:

$$\frac{1}{\alpha} \frac{T_{m,n}^{p+1} - T_{m,n}^p}{\Delta t} = 2 \left[\frac{T_{m,n+1}^p - 2T_{m,n}^p + T_{m,n-1}^p}{\Delta r^2} \right] + \left[\frac{T_{m+1,n}^p - 2T_{m,n}^p + T_{m-1,n}^p}{\Delta z^2} \right] \tag{3.7}$$

with a truncation error of order $\mathcal{O}(\Delta t) + \mathcal{O}(\Delta r^2 + \Delta z^2)$. Note that $T^p_{m,n-1} = T^p_{m,n+1}$ due to symmetry about the z-axis. This equation can be further simplified and put into standard form while neglecting the truncation error resulting in

$$T^{p+1}_{m,n} = \text{Fo}_r \left[4T^p_{m,n+1} + \frac{\Delta r^2}{\Delta z^2} \left(T^p_{m+1,n} + T^p_{m-1,n} \right) \right] + T^p_{m,n} \left[1 - 4\text{Fo}_r - 2\text{Fo}_z \right] \quad (3.8)$$

where $\text{Fo}_{(r,z)}$ is the nodal Fourier number.

3.5 Top Edge

The top edge of the cylinder is defined as the last row of the mesh spanning from the second column node to the $N - 1$ column node as shown in Fig. 3.4.

Figure 3.4. Element used for each node on the top edge defined by Eq. 3.14.

To develop the finite difference equation at the top edge of a cylinder, the revised general equation, Eq. 3.4, should be used. The top edge involves the case where $z = L/2$, noting that the cylinder is L long, and has a convection boundary condition such that

$$k \frac{\partial T}{\partial z} \bigg|_{z=\frac{L}{2}} = h(T_\infty - T^p_{m,n}) \quad (3.9)$$

where h is the heat transfer coefficient of the fluid. In order to apply the boundary condition to the revised general equation, the application of a backward Taylor series

expansion must be performed on the boundary condition to approximate the second partial derivative of temperature with respect to length yielding

$$k\frac{\partial T}{\partial z}\bigg|_{z=\frac{L}{2}-\Delta z} = k\frac{\partial T}{\partial z}\bigg|_{z=\frac{L}{2}} - \Delta z\left[\frac{\partial}{\partial z}\left(k\frac{\partial T}{\partial z}\right)\right]_{z=\frac{L}{2}} + \mathcal{O}(\Delta z^2) \qquad (3.10)$$

which can be simplified to

$$k\frac{\partial^2 T}{\partial z^2}\bigg|_{z=\frac{L}{2}} = \frac{k}{\Delta z}\frac{\partial T}{\partial z}\bigg|_{z=\frac{L}{2}} - \frac{k}{\Delta z}\frac{\partial T}{\partial z}\bigg|_{z=\frac{L}{2}-\Delta z} + \mathcal{O}(\Delta z^2) \qquad (3.11)$$

where Δz is the incremental spacing between nodes and $\mathcal{O}(\Delta z^2)$ is the error associated with truncating the rest of the expansion. The convection boundary condition, Eq. 3.9, and the Taylor series expansion, Eq. 3.11, are substituted into the revised general equation, Eq. 3.4, while dividing by the thermal conductivity to produce

$$\frac{1}{r_i}\frac{\partial T}{\partial r}\bigg|_{z=\frac{L}{2}} + \frac{\partial^2 T}{\partial r^2}\bigg|_{z=\frac{L}{2}} + \frac{h(T_\infty - T_{m,n}^p)}{k\Delta z} - \frac{1}{\Delta z}\frac{\partial T}{\partial z}\bigg|_{z=\frac{L}{2}-\Delta z} = \frac{1}{\alpha}\frac{\partial T}{\partial t}. \qquad (3.12)$$

The following equation can be converted to finite difference form using a backward difference technique as an approximation for first order partial derivative in the z-direction. A central difference technique approximates the first order partial derivative in the r-direction and the second order partial spacial derivative by application of the Taylor series expansion to each derivative. The evaluation of the first order partial time derivative uses a forward difference technique through another Taylor series to get the next temperature at an increased time-step resulting in

$$\frac{1}{\alpha}\frac{T_{m,n}^{p+1} - T_{m,n}^p}{\Delta t} = \frac{1}{r_i}\frac{T_{m,n+1}^p - T_{m,n-1}^p}{2\Delta r} + \frac{T_{m,n+1}^p - 2T_{m,n}^p + T_{m,n-1}^p}{\Delta r^2}$$
$$+ \frac{h}{k\Delta z}(T_o - T_{m,n}^p) - \frac{1}{\Delta z}\frac{3T_{m-1,n}^p - 4T_{m-2,n}^p + T_{m-3,n}^p}{2\Delta z} \qquad (3.13)$$

with a truncation error of order $\mathcal{O}(\Delta t)+\mathcal{O}(2\Delta r^2+\Delta z^2)$. This equation can be further simplified and put into standard form while neglecting the truncation error resulting in

$$
\begin{aligned}
T_{m,n}^{p+1} = \ & \mathrm{Fo}_r \left[T_{m,n+1}^{p}\left(1+\frac{\Delta r}{2r_i}\right) + T_{m,n-1}^{p}\left(1-\frac{\Delta r}{2r_i}\right) \right. \\
& \left. + \frac{\Delta r^2}{\Delta z^2}\left(-\frac{3}{2}T_{m-1,n}^{p} + 2T_{m-2,n}^{p} - \frac{1}{2}T_{m-3,n}^{p} + \mathrm{Bi}_z T_o \right) \right] \\
& + T_{m,n}^{p}\left[1 - 2\mathrm{Fo}_r - \mathrm{Fo}_z\mathrm{Bi}_z \right]
\end{aligned}
\tag{3.14}
$$

where Bi_z is the nodal Biot number in the z-direction.

3.6 Right Edge

The right edge of the cylinder is defined as the last column of the mesh spanning from the second row node to the $M-1$ row node as shown in Fig. 3.5.
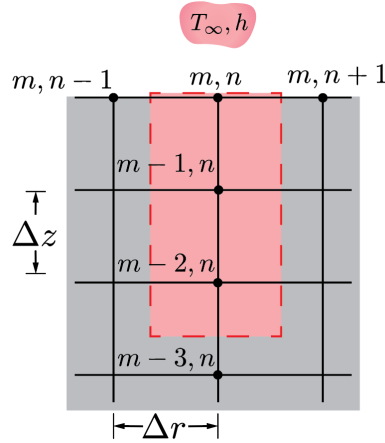

Figure 3.5. Element used for each node on the right edge defined by Eq. 3.20.

To develop the finite difference equation at the right edge of a cylinder, the revised general equation, Eq. 3.4, should be used. The right edge involves the case where $r = r_o$ and has a convection boundary condition such that

$$
k\frac{\partial T}{\partial r}\bigg|_{z=r_o} = h(T_\infty - T_{m,n}^{p}).
\tag{3.15}
$$

In order to apply the boundary condition to the revised general equation, the application of a backward Taylor series expansion must be performed on the boundary condition to approximate the second partial derivative of temperature with respect to radius yielding

$$k\frac{\partial T}{\partial r}\bigg|_{r=r_o-\Delta r} = k\frac{\partial T}{\partial r}\bigg|_{r=r_o} - \Delta r\left[\frac{\partial}{\partial r}\left(k\frac{\partial T}{\partial r}\right)\right]_{r=r_o} + \mathcal{O}(\Delta r^2) \qquad (3.16)$$

which can be simplified to

$$k\frac{\partial^2 T}{\partial r^2}\bigg|_{z=r_o} = \frac{k}{\Delta r}\frac{\partial T}{\partial r}\bigg|_{r=r_o} - \frac{k}{\Delta r}\frac{\partial T}{\partial r}\bigg|_{r=r_o-\Delta r} + \mathcal{O}(\Delta r^2) \qquad (3.17)$$

where Δr is the incremental spacing between nodes and $\mathcal{O}(\Delta r^2)$ is the error associated with truncating the rest of the expansion. The convection boundary condition, Eq. 3.15, and the Taylor series expansion, Eq. 3.17, are substituted into the revised general equation, Eq. 3.4, while dividing by the thermal conductivity to produce

$$\frac{h(T_\infty - T_{m,n}^p)}{kR} + \frac{h(T_\infty - T_{m,n}^p)}{k\Delta r} - \frac{1}{\Delta r}\frac{\partial T}{\partial r}\bigg|_{r=r_o-\Delta r} + \frac{\partial^2 T}{\partial z^2}\bigg|_{r=r_o} = \frac{1}{\alpha}\frac{\partial T}{\partial t}. \qquad (3.18)$$

The following equation can be converted to finite difference form using a backward difference technique using three nodes as an approximation for first order partial derivative in the r-direction. A central technique technique approximates for second order partial derivative in the z-direction. The evaluation of the first order partial time derivative uses a forward difference technique using another Taylor series to get the next temperature at an increased time-step resulting in

$$\frac{1}{\alpha}\frac{T_{m,n}^{p+1} - T_{m,n}^p}{\Delta t} = \frac{h}{kR}\left(T_o - T_{m,n}^p\right) + \frac{h}{k\Delta r}\left(T_o - T_{m,n}^p\right)$$
$$- \frac{1}{\Delta r}\frac{3T_{m,n-1}^p - 4T_{m,n-2}^p + T_{m,n-3}^p}{2\Delta r} + \frac{T_{m+1,n}^p - 2T_{m,n}^p + T_{m-1,n}^p}{\Delta z^2} \qquad (3.19)$$

with a truncation error of order $\mathcal{O}(\Delta t) + \mathcal{O}(\Delta r^2 + \Delta z^2)$. This equation can be further simplified and put into standard form while neglecting the truncation error resulting in

$$
\begin{aligned}
T_{m,n}^{p+1} = \mathrm{Fo}_r &\left[\mathrm{Bi}_r T_o \left(1 + \frac{\Delta r}{r_o} \right) - \frac{3}{2} T_{m,n-1}^p + 2 T_{m,n-2}^p - \frac{1}{2} T_{m,n-3}^p \right. \\
&\left. + \frac{\Delta r^2}{\Delta z^2} \left(T_{m+1,n}^p + T_{m-1,n}^p \right) \right] \\
&+ T_{m,n}^p \left[1 - \mathrm{Fo}_r \left[1 + \mathrm{Bi}_r \left(1 + \frac{\Delta r}{R} \right) \right] - 2\mathrm{Fo}_z \right]
\end{aligned}
\tag{3.20}
$$

where Bi_r is the nodal Biot number in the r-direction.

3.7   Bottom Edge

The bottom edge of the cylinder is defined as the first row of the mesh, spanning from the second column node to the $N - 1$ column node as shown in Fig. 3.6.
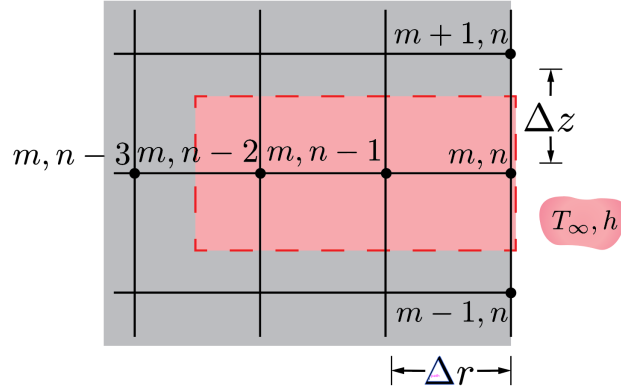

Figure 3.6.   Element used for each node on the bottom edge defined by Eq. 3.26.

To develop the finite difference equation at the bottom edge of the cylinder, the revised general equation, Eq. 3.4, should be used. The bottom edge involves the case where $z = 0$ and has and adiabatic boundary condition such that

$$
k \frac{\partial T}{\partial z} \bigg|_{z=0} = 0.
\tag{3.21}
$$

In order to apply the boundary condition to the revised general equation, the application of a backward Taylor series expansion must be performed on the boundary condition to approximate the second partial derivative of temperature with respect to length yielding

$$k\frac{\partial T}{\partial z}\bigg|_{z=0-\Delta z} = k\frac{\partial T}{\partial z}\bigg|_{z=0} - \Delta z\left[\frac{\partial}{\partial z}\left(k\frac{\partial T}{\partial z}\right)\right]_{z=0} + \mathcal{O}(\Delta z^2) \qquad (3.22)$$

which can be simplified to

$$k\frac{\partial^2 T}{\partial z^2}\bigg|_{z=0} = \frac{k}{\Delta z}\frac{\partial T}{\partial z}\bigg|_{z=0} - \frac{k}{\Delta z}\frac{\partial T}{\partial z}\bigg|_{z=0-\Delta z} + \mathcal{O}(\Delta z^2) \qquad (3.23)$$

where $\mathcal{O}(\Delta z^2)$ is the error associated with truncating the rest of the expansion. The adiabatic boundary condition, Eq. 3.21, and the Taylor series expansion, Eq. 3.23, are substituted into the revised general equation, Eq. 3.4, while dividing by the thermal conductivity to produce

$$\frac{1}{r_i}\frac{\partial T}{\partial r}\bigg|_{z=0} + \frac{\partial^2 T}{\partial r^2}\bigg|_{z=0} - \frac{1}{\Delta z}\frac{\partial T}{\partial z}\bigg|_{z=0-\Delta z} = \frac{1}{\alpha}\frac{\partial T}{\partial t}. \qquad (3.24)$$

The following equation can be converted to finite difference form using a central difference technique as an approximation for first order partial derivative in the r-direction. A central technique technique approximates the second order partial derivative in the z-direction. A forward difference technique using three nodes approximates the first order partial derivative in the z-direction. The evaluation of the first order partial time derivative uses a forward difference technique using another Taylor series to get the next temperature at an increased time-step resulting in

$$\frac{1}{\alpha}\frac{T_{m,n}^{p+1} - T_{m,n}^p}{\Delta t} = \frac{1}{r_i}\frac{T_{m,n+1}^p - T_{m,n-1}^p}{2\Delta r} + \frac{T_{m,n+1}^p - 2T_{m,n}^p + T_{m,n-1}^p}{\Delta r^2}$$
$$+ \frac{1}{\Delta z}\frac{3T_{m-1,n}^p - 4T_{m,n}^p + T_{m+1,n}^p}{2\Delta z} \qquad (3.25)$$

where $T^p_{m+1,n} = T^p_{m-1,n}$ due to symmetry and with a truncation error of order $\mathcal{O}(\Delta t)+$ $\mathcal{O}(2\Delta r^2 + \Delta z^2)$. This equation can be further simplified and put into standard form while neglecting the truncation error resulting in

$$T^{p+1}_{m,n} = \text{Fo}_r\left[T^p_{m,n+1}\left(1 + \frac{\Delta r}{2r_i}\right) + T^p_{m,n-1}\left(1 - \frac{\Delta r}{2r_i}\right) + \frac{\Delta r^2}{\Delta z^2}\left(2T^p_{m+1,n}\right)\right]$$
$$+ T^p_{m,n}\left[1 - 2\text{Fo}_r - 2\text{Fo}_z\right]. \tag{3.26}$$

3.8   Interior Nodes

The interior nodes of the cylinder are defined as the second through $M-1$ rows spanning the second through $N-1$ columns which can be seen in Fig. 3.7.
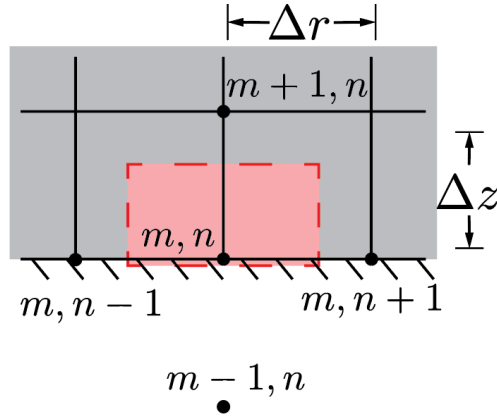

Figure 3.7.   Element used for each node on the left edge defined by Eq. 3.28.

To develop the finite difference equation for the interior of the cylinder, the revised general equation, Eq. 3.4, should be used. Application of the Taylor series expansion to the revised general equation while dividing by the thermal conductivity results in

$$\frac{1}{\alpha}\frac{T^{p+1}_{m,n} - T^p_{m,n}}{\Delta t} = \frac{1}{r_i}\frac{T^p_{m,n+1} - T^p_{m,n-1}}{2\Delta r} + \frac{T^p_{m,n+1} - 2T^p_{m,n} + T^p_{m,n-1}}{\Delta r^2}$$
$$+ \frac{T^p_{m+1,n} - 2T^p_{m,n} + T^p_{m-1,n}}{\Delta z^2} \tag{3.27}$$

with a truncation error of order $\mathcal{O}(\Delta t)+\mathcal{O}(\Delta 2r^2+\Delta z^2)$. This equation can be further simplified and put into standard form while neglecting the truncation error resulting in

$$
\begin{aligned}
T_{m,n}^{p+1} = \ \mathrm{Fo}_r &\left[T_{m,n+1}^p \left(1 + \frac{\Delta r}{2r_i} \right) + T_{m,n-1}^p \left(1 - \frac{\Delta r}{2r_i} \right) \right. \\
&\left. + \frac{\Delta r^2}{\Delta z^2} \left(T_{m+1,n}^p + T_{m-1,n}^p \right) \right] + T_{m,n}^p \left[1 - 2\mathrm{Fo}_r - 2\mathrm{Fo}_z \right].
\end{aligned}
\tag{3.28}
$$

3.9 Top Left Corner Node

The top left corner nodes of the cylinder is defined as the node within the first column in the last row or at the M row and at the $n = 1$) column as shown in Fig. 3.8.
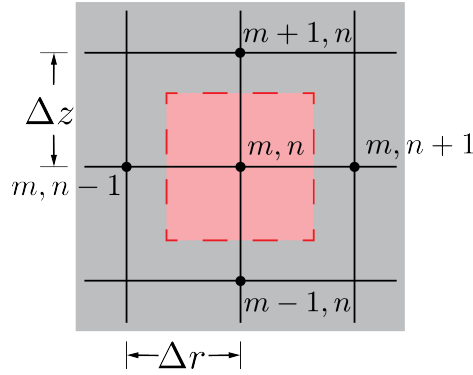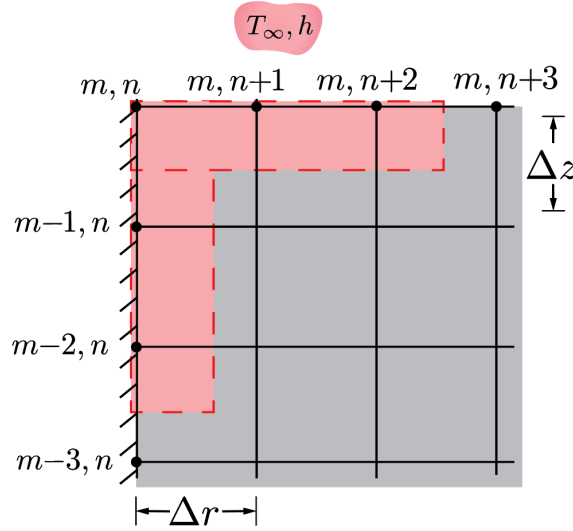

Figure 3.8. Element used for each node on the top left corner defined by Eq. 3.37.

To develop the finite difference equation at the top left corner node of the cylinder, the revised general equation, Eq. 3.4, should be used. The top left corner involves

the case where $r = 0$ and $z = L/2$ which involves an adiabatic boundary condition and convection boundary condition on the respective faces such that

$$k\frac{\partial T}{\partial z}\bigg|_{z=\frac{L}{2}} = h(T_\infty - T_{m,n}^p) \tag{3.29}$$

and

$$k\frac{\partial T}{\partial r}\bigg|_{r=0} = 0. \tag{3.30}$$

In order to apply the boundary conditions to the revised general equation, the application of a backward Taylor series expansion must be performed on each boundary condition to approximate both the second order partial derivatives of temperature with respect to r or z. Applying the expansion to Eq. 3.29 yields

$$k\frac{\partial T}{\partial z}\bigg|_{z=\frac{L}{2}-\Delta z} = k\frac{\partial T}{\partial z}\bigg|_{z=\frac{L}{2}} - \Delta z\left[\frac{\partial}{\partial z}\left(k\frac{\partial T}{\partial z}\right)\right]_{z=\frac{L}{2}} + \mathcal{O}(\Delta z^2) \tag{3.31}$$

which can be rewritten as

$$k\frac{\partial^2 T}{\partial z^2}\bigg|_{z=\frac{L}{2}} = \frac{k}{\Delta z}\frac{\partial T}{\partial z}\bigg|_{z=\frac{L}{2}} - \frac{k}{\Delta z}\frac{\partial T}{\partial z}\bigg|_{z=\frac{L}{2}-\Delta z} + \mathcal{O}(\Delta z^2) \tag{3.32}$$

where $\mathcal{O}(\Delta z^2)$ is the error associated with truncating the rest of the expansion. Applying the expansion to Eq. 3.30 yields

$$k\frac{\partial T}{\partial r}\bigg|_{r=0-\Delta r} = k\frac{\partial T}{\partial r}\bigg|_{r=0} - \Delta r\left[\frac{\partial}{\partial r}\left(k\frac{\partial T}{\partial r}\right)\right]_{r=0} + \mathcal{O}(\Delta r^2) \tag{3.33}$$

which can be rewritten as

$$k\frac{\partial^2 T}{\partial r^2}\bigg|_{r=0} = \frac{k}{\Delta r}\frac{\partial T}{\partial r}\bigg|_{r=0} - \frac{k}{\Delta r}\frac{\partial T}{\partial r}\bigg|_{r=0-\Delta r} + \mathcal{O}(\Delta r^2) \tag{3.34}$$

where $\mathcal{O}(\Delta r^2)$ is the error associated with truncating the rest of the expansion.

The convective boundary condition, Eq. 3.29, adiabatic boundary condition, Eq. 3.30, and both the respective Taylor series expansions, Eqs. 3.32 and 3.34 are substituted into the revised general equation, Eq. 3.4, while dividing by the thermal conductivity to produce

$$-\frac{2}{\Delta r}\frac{\partial T}{\partial r}\bigg|_{r=0-\Delta r} + \frac{h(T_\infty - T_{m,n}^p)}{k\Delta z} - \frac{1}{\Delta z}\frac{\partial T}{\partial z}\bigg|_{z=\frac{L}{2}-\Delta z} = \frac{1}{\alpha}\frac{\partial T}{\partial t}. \tag{3.35}$$

The following equation can be converted to finite difference form using a backward difference three node technique as an approximation for the first order partial derivative in the r-direction. Another backward difference three node technique approximates the first order partial derivative in the z-direction. The evaluation of the first order partial time derivative uses a forward difference technique using another Taylor series to get the next temperature at an increased time-step resulting in

$$\frac{1}{\alpha}\frac{Tm,n^{p+1}-T_{m,n}^p}{\Delta t} = -\frac{2}{\Delta r}\frac{3T_{m,n-1}^p-4T_{m,n-2}^p+T_{m,n-3}^p}{2\Delta r}+\frac{h}{k\Delta z}(T_o-T_{m,n}^p)$$
$$-\frac{1}{\Delta z}\frac{3Tm-1,n^p-4T_{m-2,n}^p+T_{m-3,n}^p}{2\Delta z} \tag{3.36}$$

where $T_{m,n-1}^p = T_{m,n+1}^p$, $T_{m,n-2}^p = T_{m,n+2}^p$, and $T_{m,n-1}^p = T_{m,n+1}^p$ due to symmetry and with a truncation error of order $\mathcal{O}(\Delta t)+\mathcal{O}(\Delta r^2 + \Delta z^2)$. This equation can be further simplified and put into standard form while neglecting the truncation error resulting in

$$T_{m,n}^{p+1} = \mathrm{Fo}_r\left[-3T_{m,n+1}^p + 4T_{m,n+2}^p - T_{m,n+3}^p\right.$$

$$+\frac{\Delta r^2}{\Delta z^2}\left(-\frac{3}{2}T_{m-1,n}^p + 2T_{m-1,n}^p - \frac{1}{2}T_{m-3,n}^p + \mathrm{Bi}_z T_o\right)\Bigg]$$

$$+T_{m,n}^p\left[1 - \mathrm{Fo}_z\mathrm{Bi}_z\right]. \tag{3.37}$$

3.10 Bottom Left Corner Node

The bottom left corner node of the cylinder is defined as the node within the first column $n = 1$ and the first row $m = 1$ as shown in Fig. 3.9. To develop the finite difference equation at the bottom left corner node of the cylinder, the revised general equation, Eq. 3.4, should be used. The bottom left corner involves the case where $r = 0$ and $z = 0$ which involves two adiabatic boundary conditions on the respective faces such that

$$k\frac{\partial T}{\partial z}\bigg|_{z=0} = 0 \tag{3.38}$$
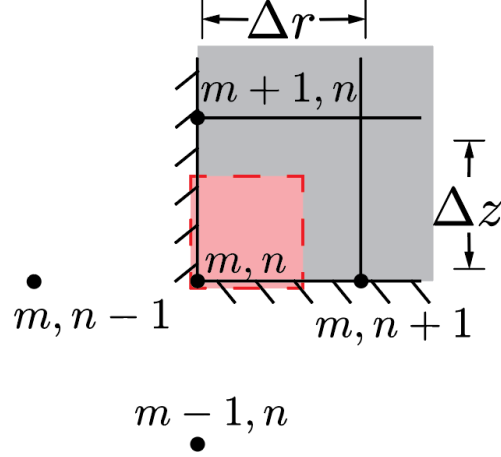
Figure 3.9.   Element used for each node on the bottom left corner defined by Eq. 3.46.

and

$$k\frac{\partial T}{\partial r}\bigg|_{r=0} = 0. \tag{3.39}$$

In order to apply the boundary conditions to the revised general equation, the application of a backward Taylor series expansion must be performed on each boundary condition to approximate both the second order partial derivatives of temperature with respect to r or z. Applying the expansion to Eq. 3.38 yields

$$k\frac{\partial T}{\partial z}\bigg|_{z=0-\Delta z} = k\frac{\partial T}{\partial z}\bigg|_{z=0} - \Delta z\left[\frac{\partial}{\partial z}\left(k\frac{\partial T}{\partial z}\right)\right]_{z=0} + \mathcal{O}(\Delta z^2) \tag{3.40}$$

which can be rewritten as

$$k\frac{\partial^2 T}{\partial z^2}\bigg|_{z=0} = \frac{k}{\Delta z}\frac{\partial T}{\partial z}\bigg|_{z=0} - \frac{k}{\Delta z}\frac{\partial T}{\partial z}\bigg|_{z=0-\Delta z} + \mathcal{O}(\Delta z^2). \tag{3.41}$$

where $\mathcal{O}(\Delta z^2)$ is the error associated with truncating the rest of the expansion. Applying the expansion to Eq. 3.30 yields

$$k\frac{\partial T}{\partial r}\bigg|_{r=0-\Delta r} = k\frac{\partial T}{\partial r}\bigg|_{r=0} - \Delta r\left[\frac{\partial}{\partial r}\left(k\frac{\partial T}{\partial r}\right)\right]_{r=0} + \mathcal{O}(\Delta r^2) \tag{3.42}$$

and

$$k\frac{\partial^2 T}{\partial r^2}\bigg|_{r=0} = \frac{k}{\Delta r}\frac{\partial T}{\partial r}\bigg|_{r=0} - \frac{k}{\Delta r}\frac{\partial T}{\partial r}\bigg|_{r=0-\Delta r} + \mathcal{O}(\Delta r^2) \tag{3.43}$$

where $\mathcal{O}(\Delta r^2)$ is the error associated with truncating the rest of the expansion.

Both adiabatic boundary conditions, Eqs. 3.38 and 3.39 along with both the respective Taylor series expansions, Eqs. 3.41 and 3.43 are substituted into the revised general equation, Eq. 3.4, while dividing by the thermal conductivity to produce

$$-\frac{2}{\Delta r}\frac{\partial T}{\partial r}\bigg|_{r=0-\Delta r} - \frac{1}{\Delta z}\frac{\partial T}{\partial z}\bigg|_{z=0-\Delta z} = \frac{1}{\alpha}\frac{\partial T}{\partial t} \tag{3.44}$$

The following equation can be converted to finite difference form using a forward difference three node technique as an approximation for the first order partial derivative in the r-direction. Another forward difference three node technique approximates the first order partial derivative in the z-direction. The evaluation of the first order partial time derivative uses a forward difference technique using another Taylor series to get the next temperature at an increased time-step resulting in

$$\frac{1}{\alpha}\frac{T_{m,n}^{p+1} - T_{m,n}^p}{\Delta t} = \frac{2}{\Delta r}\frac{3T_{m,n-1}^p - 4T_{m,n}^p + T_{m,n+1}^p}{2\Delta r}$$
$$+ \frac{1}{\Delta z}\frac{3T_{m-1,n}^p - 4Tm,n^p + T_{m+1,n}^p}{2\Delta z} \tag{3.45}$$

where $T_{m,n-1}^p = T_{m,n+1}^p$ and $T_{m-1,n}^p = T_{m+1,n}^p$ due to symmetry and with a truncation error of order $\mathcal{O}(\Delta t) + \mathcal{O}(\Delta r^2 + \Delta z^2)$. This equation can be further simplified and put into standard form while neglecting the truncation error resulting in

$$T_{m,n}^{p+1} = \text{Fo}_r\left[4T_{m,n-1}^p + \frac{\Delta r^2}{\Delta z^2}\left(2T_{m-1,n}^p\right)\right] + T_{m,n}^p\left[1 - 4\text{Fo}_r - 2\text{Fo}_z\right]. \tag{3.46}$$

3.11   Top Right Corner Node

The top right corner node of the cylinder is defined as the node within the last column $m = M$ and the last row or at $n = N$ as shown in Fig. 3.10. To develop the finite difference equation at the top right corner node of the cylinder, the revised general equation, Eq. 3.4, should be used. The top right corner involves the case where $r = 0$ and $z = 0$ which involves two convective boundary conditions on the respective faces such that
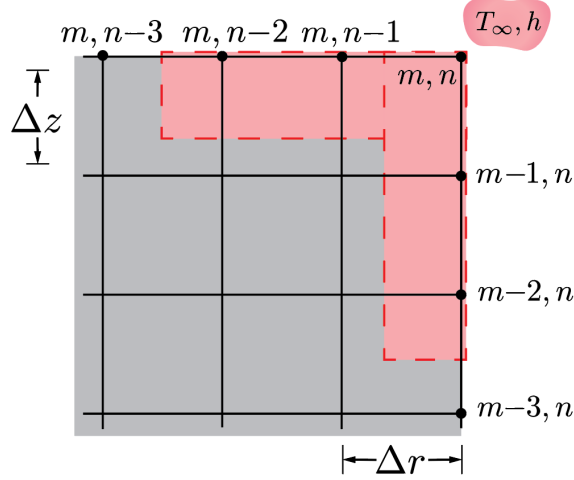
Figure 3.10.   Element used for each node on the top right corner defined by Eq. 3.55.

$$k\frac{\partial T}{\partial z}\bigg|_{z=\frac{L}{2}} = h(T_\infty - T_{m,n}^p) \tag{3.47}$$

and

$$k\frac{\partial T}{\partial r}\bigg|_{r=r_o} = h(T_\infty - T_{m,n}^p). \tag{3.48}$$

In order to apply the boundary conditions to the revised general equation, the application of a backward Taylor series expansion must be performed on each boundary condition to approximate both the second order partial derivatives of temperature with respect to r or z. Applying the expansion to Eq. 3.47 yields

$$k\frac{\partial T}{\partial z}\bigg|_{z=\frac{L}{2}-\Delta z} = k\frac{\partial T}{\partial z}\bigg|_{z=\frac{L}{2}} - \Delta z\left[\frac{\partial}{\partial z}\left(k\frac{\partial T}{\partial z}\right)\right]_{z=\frac{L}{2}} + \mathcal{O}(\Delta z^2) \tag{3.49}$$

which can be rewritten as

$$k\frac{\partial^2 T}{\partial z^2}\bigg|_{z=\frac{L}{2}} = \frac{k}{\Delta z}\frac{\partial T}{\partial z}\bigg|_{z=\frac{L}{2}} - \frac{k}{\Delta z}\frac{\partial T}{\partial z}\bigg|_{z=\frac{L}{2}-\Delta z} + \mathcal{O}(\Delta z^2) \tag{3.50}$$

where $\mathcal{O}(\Delta z^2)$ is the error associated with truncating the rest of the expansion. Applying the expansion to Eq. 3.48 yields

$$k\frac{\partial T}{\partial r}\bigg|_{r=r_o-\Delta r} = k\frac{\partial T}{\partial r}\bigg|_{r=r_o} - \Delta r\left[\frac{\partial}{\partial r}\left(k\frac{\partial T}{\partial r}\right)\right]_{r=r_o} + \mathcal{O}(\Delta r^2) \tag{3.51}$$

which can be rewritten as

$$k\frac{\partial^2 T}{\partial r^2}\bigg|_{r=r_o} = \frac{k}{\Delta r}\frac{\partial T}{\partial r}\bigg|_{r=r_o} - \frac{k}{\Delta r}\frac{\partial T}{\partial r}\bigg|_{r=r_o-\Delta r} + \mathcal{O}(\Delta r^2) \tag{3.52}$$

where $\mathcal{O}(\Delta r^2)$ is the error associated with truncating the rest of the expansion.

Both convective boundary conditions, Eqs. 3.47 and 3.48 along with both the respective Taylor series expansions, Eqs. 3.50 and 3.52 are substituted into the revised general equation, Eq. 3.4, while dividing by the thermal conductivity to produce

$$\frac{h(T_\infty - T_{m,n}^p)}{kr_o} + \frac{h(T_\infty - T_{m,n}^p)}{k\Delta r} - \frac{1}{\Delta r}\frac{\partial T}{\partial r}\bigg|_{r=r_o-\Delta r}$$

$$+ \frac{h(T_\infty - T_{m,n}^p)}{k\Delta z} - \frac{1}{\Delta z}\frac{\partial T}{\partial z}\bigg|_{z=\frac{L}{2}-\Delta z} = \frac{1}{\alpha}\frac{\partial T}{\partial t} \tag{3.53}$$

The following equation can be converted to finite difference form using a backward difference three node technique as an approximation for the first order partial derivative in the r-direction. Another backward difference three node technique approximates the first order partial derivative in the z-direction. The evaluation of the first order partial time derivative uses a forward difference technique using another Taylor series to get the next temperature at an increased time-step resulting in

$$\frac{1}{\alpha}\frac{T_{m,n}^{p+1} - T_{m,n}^p}{\Delta t} = \frac{h}{kr_o}\left(T_\infty - T_{m,n}^p\right) + \frac{h}{k\Delta r}\left(T_\infty - T_{m,n}^p\right)$$

$$- \frac{1}{\Delta r}\frac{3T_{m,n-1}^p - 4T_{m,n-2}^p + T_{m,n-3}^p}{2\Delta r} + \frac{h}{k\Delta z}\left(T_\infty - T_{m,n}^p\right)$$

$$- \frac{1}{\Delta z}\frac{3T_{m-1,n}^p - 4T_{m-2,n}^p + T_{m-3,n}^p}{2\Delta z} \tag{3.54}$$

with a truncation error of order $\mathcal{O}(\Delta t) + \mathcal{O}(\Delta r^2 + \Delta z^2)$. This equation can be further simplified and put into standard form while neglecting the truncation error resulting in

$$T_{m,n}^{p+1} = Fo_r\left[T_o\left(\text{Bi}_r\left(1 + \frac{\Delta r}{r_o}\right) + \frac{\Delta r^2}{\Delta z^2}\text{Bi}_z\right) - \frac{3}{2}T_{m,n-1}^p + 2T_{m,n-2}^p - \frac{1}{2}T_{m,n-3}^p\right.$$

$$+ \frac{\Delta r^2}{\Delta z^2}\left(-\frac{3}{2}T_{m-1,n}^p + 2T_{m-2,n}^p - \frac{1}{2}T_{m-3,n}^p\right)\Bigg]$$

$$+ T_{m,n}^p\left[1 - Fo_r\text{Bi}_r\left(1 + \frac{\Delta r}{r_o}\right) - Fo_z\text{Bi}_z\right]. \tag{3.55}$$

3.12 Bottom Right Corner Node

The bottom right corner node of the cylinder is defined as the node within the last column $n = N$ and the first row ($m = 1$) as shown in Fig. 3.11.
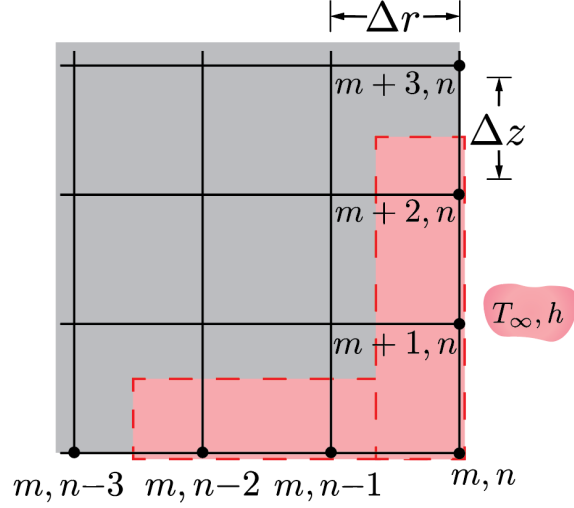
Figure 3.11. Element used for each node on the bottom right corner defined by Eq. 3.64.

To develop the finite difference equation at the top right corner node of the cylinder, the revised general equation, Eq. 3.4, should be used. The top right corner involves the case where $r = 0$ and $z = 0$ which involves an adiabatic boundary condition and a convective boundary condition on the respective faces such that

$$k\frac{\partial T}{\partial z}\bigg|_{z=0} = 0 \tag{3.56}$$

and

$$k\frac{\partial T}{\partial r}\bigg|_{r=r_o} = h(T_\infty - T_{m,n}^p). \tag{3.57}$$

In order to apply the boundary conditions to the revised general equation, the application of a backward Taylor series expansion must be performed on each boundary condition to approximate both the second order partial derivatives of temperature with respect to r or z. Applying the expansion to Eq. 3.56 yields

$$k\frac{\partial T}{\partial z}\bigg|_{z=0-\Delta z} = k\frac{\partial T}{\partial z}\bigg|_{z=0} - \Delta z\left[\frac{\partial}{\partial z}\left(k\frac{\partial T}{\partial z}\right)\right]_{z=0} + \mathcal{O}(\Delta z^2) \tag{3.58}$$

which can be rewritten as

$$k\frac{\partial^2 T}{\partial z^2}\bigg|_{z=0} = \frac{k}{\Delta z}\frac{\partial T}{\partial z}\bigg|_{z=0} - \frac{k}{\Delta z}\frac{\partial T}{\partial z}\bigg|_{z=0-\Delta z} + \mathcal{O}(\Delta z^2) \qquad (3.59)$$

where $\mathcal{O}(\Delta z^2)$ is the error associated with truncating the rest of the expansion. Applying the expansion to Eq. 3.57 yields

$$k\frac{\partial T}{\partial r}\bigg|_{r=r_o-\Delta r} = k\frac{\partial T}{\partial r}\bigg|_{r=r_o} - \Delta r\left[\frac{\partial}{\partial r}\left(k\frac{\partial T}{\partial r}\right)\right]_{r=r_o} + \mathcal{O}(\Delta r^2) \qquad (3.60)$$

which can be rewritten as

$$k\frac{\partial^2 T}{\partial r^2}\bigg|_{r=r_o} = \frac{k}{\Delta r}\frac{\partial T}{\partial r}\bigg|_{r=r_o} - \frac{k}{\Delta r}\frac{\partial T}{\partial r}\bigg|_{r=r_o-\Delta r} + \mathcal{O}(\Delta r^2) \qquad (3.61)$$

where $\mathcal{O}(\Delta r^2)$ is the error associated with truncating the rest of the expansion.

The adiabatic boundary condition, Eq. 3.56, convective boundary condition, Eq. 3.57, and both the respective Taylor series expansions, Eqs. 3.59 and 3.61 are substituted into the revised general equation, Eq. 3.4, while dividing by the thermal conductivity to produce

$$\frac{h(T_\infty - T_{m,n}^p)}{kR} + \frac{h(T_\infty - T_{m,n}^p)}{k\Delta r} - \frac{k}{\Delta r}\frac{\partial T}{\partial r}\bigg|_{r=r_o-\Delta r} - \frac{k}{\Delta z}\frac{\partial T}{\partial z}\bigg|_{z=0-\Delta z} = \frac{1}{\alpha}\frac{\partial T}{\partial t}. \quad (3.62)$$

The following equation can be converted to finite difference form using a backward difference three node technique as an approximation for the first order partial derivative in the r-direction. Another backward difference three node technique approximates the first order partial derivative in the z-direction. The evaluation of the first order partial time derivative uses a forward difference technique using another Taylor series to get the next temperature at an increased time-step resulting in

$$\frac{1}{\alpha}\frac{T_{m,n}^{p+1} - T_{m,n}^p}{\Delta t} = \frac{h}{kr_o}\left(T_o - T_{m,n}^p\right) + \frac{h}{k\Delta r}\left(T_o - T_{m,n}^p\right)$$
$$- \frac{1}{\Delta r}\frac{3T_{m,n-1}^p - 4T_{m,n-2}^p + T_{m,n-3}^p}{2\Delta r}$$
$$- \frac{1}{\Delta z}\frac{3T_{m-1,n}^p - 4T_{m-2,n}^p + T_{m-3,n}^p}{2\Delta z} \qquad (3.63)$$

with a truncation error of order $\mathcal{O}(\Delta t) + \mathcal{O}(\Delta r^2 + \Delta z^2)$. This equation can be further simplified and put into standard form while neglecting the truncation error resulting in

$$
\begin{aligned}
T_{m,n}^{p+1} = \mathrm{Fo}_r & \left[\mathrm{Bi}_r T_o \left(1 + \frac{\Delta r}{r_o} \right) - \frac{3}{2} T_{m,n-1}^p + 2 T_{m,n-2}^p - \frac{1}{2} T_{m,n-3}^p \right. \\
& \left. + \frac{\Delta r^2}{\Delta z^2} \left(-\frac{3}{2} T_{m-1,n}^p + 2 T_{m-2,n}^p - \frac{1}{2} T_{m-3,n}^p \right) \right] \\
& + T_{m,n}^p \left[1 - \mathrm{Fo}_r \mathrm{Bi}_r \left(1 + \frac{\Delta r}{r_o} \right) \right].
\end{aligned}
\tag{3.64}
$$

3.13 Stability Criteria

Since the technique used to develop the two-dimensional cylindrical solution involves an explicit method, there is a stability condition governing the time step in order for the solution to be numerically stable and converge to a given value. This condition requires that the secondary part of each of the prior derived equations involving the temperature at the current node be one or less as shown in 3.1. If the solution is greater than one, the solution is unstable and well oscillate between arbitrary values. For completeness, all derived equations are checked for stability by solving the time component of the equation greater than zero. The specific numerical implications of the stability criteria are discussed further in chapter 4.

Table 3.1. Summary of stability criteria equations.

Location	Equation	
Internal	$1 \geq 2\mathrm{Fo}_r + 2\mathrm{Fo}_z$	(3.65)
Left Edge	$1 \geq 4\mathrm{Fo}_r + 2\mathrm{Fo}_z$	(3.66)
Top Edge	$1 \geq 2\mathrm{Fo}_r + \mathrm{Fo}_z\mathrm{Bi}_z$	(3.67)
Right Edge	$1 \geq \mathrm{Fo}_r\mathrm{Bi}_r\left(1 + \dfrac{\Delta r}{r_o}\right) + 2\mathrm{Fo}_z$	(3.68)
Bottom Edge	$1 \geq 2\mathrm{Fo}_r - 2\mathrm{Fo}_z$	(3.69)
Top Left Corner	$1 \geq \mathrm{Fo}_z\mathrm{Bi}_z$	(3.70)
Bottom Left Corner	$1 \geq 4\mathrm{Fo}_r - 2\mathrm{Fo}_z$	(3.71)
Top Right Corner	$1 \geq \mathrm{Fo}_r\mathrm{Bi}_r\left(1 + \dfrac{\Delta r}{r_o}\right)$	(3.72)
Bottom Right Corner	$1 \geq \mathrm{Fo}_r\mathrm{Bi}_r\left(1 + \dfrac{\Delta r}{r_o}\right)$	(3.73)

4. MODEL DEVELOPMENT

4.1 Aims

This chapter covers the development of a computer simulation based on the equations developed in chapter 3. The simulation was developed in Matlab for its convenience and multitude of built-in functions. The coding structure is introduced along with the development of the GUI for users to interface with the program.

4.2 Program Overview

The goal of the program is to take a variety of inputs from the user and show the resulting center point temperature, average temperature, and heat transfer over time. It is meant to be used as a teaching aid in an introductory college-level heat transfer course to show the difference between a one- and two-dimensional solution. The ultimate goal is to see at what Fourier number a two-dimensional solution must be used. A one-dimensional solution can be used for small Fourier numbers up to a particular point in which this work intends to develop a method for finding.

The inputs to the program include initial cylinder temperature, fluid temperature, duration of the simulation, convection coefficient, material selection, radius, and length. More advanced options are available for experienced users such as the nodes used in both the radial and length directions, the number of roots used to solve the one-dimensional solutions for a cylinder and slab, along with adjusting the time axis in terms of seconds or Fourier number.

Another significant consideration when designing the program is the speed at which a solution is developed. Since the program is intended to be used as a teaching tool, it should run quickly without sacrificing accuracy to provide the user(s) with solutions to compare in a relatively short time. The program has been developed

with this in mind and can complete most calculations within thirty seconds. The only situation the program takes a significant amount of time is when the additional settings are increased to include more nodes in the mesh or more roots for the one-dimensional solutions. The use of some built-in Matlab functions are also minimized due to long processing time, such as the Bessel function.

The main graphical outputs of the program are the center temperature, average temperature, and heat transfer with the respective errors from the one-dimensional solutions. The program additionally supports a feature to print all calculated values to an excel spreadsheet to allow the user to save particular data sets.

4.3   Control Panel

The master function of the program is the control panel which takes in all the user inputs, calculates important parameters used for analysis, and runs subroutines to calculate the outputs. The program runs linearly, as shown in Fig. 4.1 to calculate the required outputs. Through the utilization of subroutines, the speed is dramatically increased due to Matlab prioritizing them. Subroutines are used for the following calculations:

1. Stability criteria

2. 1D cylindrical center point temperature

3. 1D slab center point temperature

4. 2D cylindrical center point temperature.

The control panel was originally designed as the main backbone of the program. It would gather inputs from the user, run the subroutines required to calculate all the necessary components, and then generate an output. The GUI was added to the program to allow the user to easily interface with the program without needing a strong background in Matlab to operate. Since the addition of the GUI interface file,
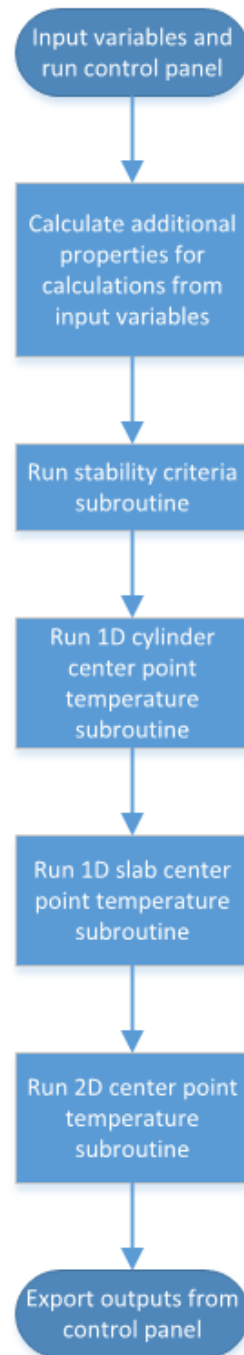
Figure 4.1. Linear progression of the control panel program showing the order of operation for each subroutine.

the control panel has become a redundant function that could be integrated directly to the GUI file. The separation of the GUI file and control panel file still exists mainly for Matlab's priority of subroutines and to keep the GUI file clutter-free only containing the data handling structures and graphing.

4.4 Stability Criteria

Since the technique used to develop the two-dimensional cylindrical solution involves an explicit method, there is a stability condition governing the time step in order for the solution to be numerically stable and converge to a given value. For completeness, each individual node is checked for stability by solving the time component of the equation greater than zero. Checking each node has minimal impact on the program performance and is used primarily for debugging. The value for the time step is then floored to two significant figures which always ensures stability is met. The last step of the program is to divide the specified duration of the study by the time step to find the total number of time steps required to run the simulation. This step is essential as it gives the other subroutines a stopping point at which they need to stop summing individual time steps.

If an implicit numerical method is used, then checking for stability criteria is not required. Due to the nature of an implicit method looking at the current step and the last step to update, rather than the explicit method of using the current step and next step to update is why a stability criterion is not required. Since the problem is two-dimensional, the Fourier and Biot number become direction-dependent. To specify which direction is under consideration, each dimensionless parameter is given a subscript to the corresponding direction.

4.5 1D Cylindrical Center Point Temperature

In order to compare a one-dimensional analysis to the two-dimensional, all the values must first be calculated using a one-dimensional analysis. Since the geometry

is in the cylindrical coordinate system, the use of Bessel functions are needed in order to solve the differential equation. Matlab has a built-in Bessel function which outputs the value for a given order and input. The subroutine has four main components which calculate the various outputs of the function:

1. Solve cylindrical transcendental roots

2. Solve coefficient values for the 1D solution

3. Solve 1D solution using roots and coefficients

4. Calculate average temperature and heat transfer.

The transcendental roots are solved by using

$$0 = \zeta_n J_1(\zeta_n) - \mathrm{Bi} J_0(\zeta_n) \tag{4.1}$$

where the Bessel functions are of the first kind. A root is found every increment of π, so solving for the root requires the use of Newton's method through the built-in fzero function in Matlab which solves for a zero within a given range. The roots are continuously solved until the number of roots specified by the user have been solved. The Bessel function is generated by using the Bessel function in Matlab. The use of this particular function is minimized due to the long processing time accumulated over the execution of the subroutine.

The coefficient values are calculated independently to reduce processing time in the one-dimensional solution loop. Calculating these separately additionally allows the user to double-check the accuracy of the roots to published data. While programming this subroutine, it was useful to have these values stored to double-check if needed for debugging proposes. The coefficient values are calculated using the following equation

$$C_n = \frac{2}{\zeta_n} \frac{J_1(\zeta_n)}{J_0^2(\zeta_n) + J_1^2(\zeta_n)} \tag{4.2}$$

where the Bessel functions are of the first kind.

Now that the roots have been solved for and the coefficient values of the equation have been solved, it is possible to calculate the temperature distribution within the cylinder. The dimensionless temperature can be found by using the following equation

$$\Theta = \sum_{n=1}^{\infty} C_n \, \exp\,(-\zeta_n^2 \mathrm{Fo}) J_0(\zeta_n r^*) \tag{4.3}$$

where r^* is the dimensionless location from 0 to 1 within the cylinder [2]. Eq. 4.3 sums the solutions from all the roots that were solved for in Eq. 4.1 for iterative steps of Fourier number until the desired time set by the user is reached. Additionally, the heat transfer from the center point is being calculated within this loop to save processing time.

The average temperature is calculated by multiplying the temperature at each node by a weighting factor based on the geometry. Since the geometry of the body is a cylinder, the weighting factors change with each differential volume element in the radial direction. The first and the last volume elements are modified slightly since their lengths are only half. Additionally, these nodes need to be placed off-axis slightly due to their half-size; therefore, a smaller half-length element is taken at half the node spacing away from the centerline and edge of the cylinder at the center of the node. This method does not affect the overall calculation since the width of the end element extends to the full radius of the cylinder.

The last component of the subroutine is to calculate the heat transfer from or to the cylinder. To calculate the heat transfer, look at how much energy enters or leaves a particular point, in this case, the center of the cylinder. The dimensionless total heat transfer can be found by using the following equation

$$\frac{Q}{Q_0} = 1 - \frac{2\Theta_0}{\zeta_1} J_1(\zeta_1) \tag{4.4}$$

where Θ_0 is the dimensionless centerline temperature. By taking a summation of equation Eq. 4.4, it results in the instantaneous heat transfer at the particular point in time for the cylinder.

All these values are stored to large matrices that account for the center temperature, average temperature, and heat transfer for each time step. The raw values are

available from the GUI through the print button. This technique can be a useful tool not only for debugging a program but as an in-depth analysis while studying these equations.

4.6   1D Slab Center Point Temperature

The analysis for the slab center point temperature is nearly identical to the one-dimensional cylinder beside the equations. While the geometry is in the cylindrical coordinate system, it makes more sense to think of a cross-section similar to a plane wall since most of the heat transfer is in the vertical direction if the slab is laying down. The subroutine has four main components which calculate the various outputs of the function:

1. Solve slab transcendental roots

2. Solve coefficient values for the 1D solution

3. Solve 1D solution using roots and coefficients

4. Calculate average temperature and heat transfer

The transcendental roots are solved by using

$$0 = \zeta_n \tan \ \zeta_n - \mathrm{Bi}. \tag{4.5}$$

A root is found every increment of $\pi/2$, so solving for the root requires the use of Newton's method through the built-in fzero function in Matlab which solves for a zero within a given range. The roots are continuously solved until the number of roots specified by the user have been solved.

The coefficient values are calculated independently to reduce processing time in the one-dimensional solution loop. Calculating these separately additionally allows the user to double-check the accuracy of the roots to published data. While programming

this subroutine, it was useful to have these values stored to double-check if needed for debugging proposes. The coefficient values are calculated using the following equation

$$C_n = \frac{4 \sin \ \zeta_n}{2\zeta_n + \sin(2\zeta_n)}. \tag{4.6}$$

With the roots and the coefficient values solved, it is possible to calculate the temperature distribution within the slab. The dimensionless temperature can be found by using the following equation

$$\Theta = \sum_{n=1}^{\infty} C_n \ \exp \ (-\zeta_n^2 \mathrm{Fo}) \cos(\zeta_n x^*) \tag{4.7}$$

where x^* is the dimensionless location from 0 to 1 within the wall [2]. Eq. 4.7 sums the solutions from all the roots that were solved for in equation 4.5 for iterative steps of Fourier number until the desired time set by the user is reached. Additionally, the heat transfer from the center point is being calculated within this loop to save processing time.

Summarily to the average temperature of the one-dimensional cylinder, the average slab temperature is calculated by multiplying the temperature at each node by a weighting factor based on the geometry. The geometry of the body is a slab; therefore, the weighting factors do not change with each differential volume element. It is best, for consistency, to calculate the average temperature for each node. The first and the last volume elements are modified slightly since their lengths are only half of what the typical node spacing will be. Additionally, these nodes need to be placed off-axis slightly due to their half-size; therefore, a smaller half-length element is taken at half the node spacing away from the centerline and edge of the cylinder at the center of the node. This process does not affect the overall calculation since the width of the end element extends to the full radius of the cylinder.

The last component of the subroutine is to calculate the heat transfer from or to the wall. This method can be done by looking at how much energy in the form

of heat enters or leaves a particular point; in this case, the center of the slab. The dimensionless total heat transfer can be found by using the following equation

$$\frac{Q}{Q_0} = 1 - \frac{\sin \zeta_1}{\zeta_1} \Theta_0 \tag{4.8}$$

where Θ_0 is the dimensionless centerline temperature. By taking a summation of equation Eq. 4.8, it results in the instantaneous heat transfer at the particular point in time for the slab.

All these values are stored to large matrices that account for the center temperature, average temperature, and heat transfer for each time step. The raw values are available from the GUI through the print button. This method can be a useful tool not only for debugging a program but as an in-depth analysis while studying these equations.

4.7 2D Cylindrical Center Point Temperature

The main equations for this subroutine are derived in detail in chapter 3. The function begins by allocating memory to matrices to store values, mainly the temperature distribution field. The function is broken up into a total of three main components being:

1. two-dimensional temperature distribution

2. average temperature distribution

3. and heat transfer from the solid.

The first objective is to make a temperature field consisting of the initial temperature of the solid. Once completed the function updates each edge of the field with the corresponding equation. The corners of the field require separate equations due to having an additional boundary condition and are handled after the edge nodes. Finally, the internal nodes are evaluated to create an updated temperature field. This matrix is then stored as the previous time matrix, and the process continues for the

next time step. The center point temperature is saved to a file since that is the main focus of the thesis.

The heat transfer is calculated by looking at the energy flow at the boundaries of the cylinder. This value can easily be calculated with Newton's law of cooling/heating

$$\frac{d\dot{Q}}{dt} = h\Delta T \frac{\partial A}{\partial r} \tag{4.9}$$

where ΔT is the change in temperature between the boundary nodes and the ambient fluid temperature.
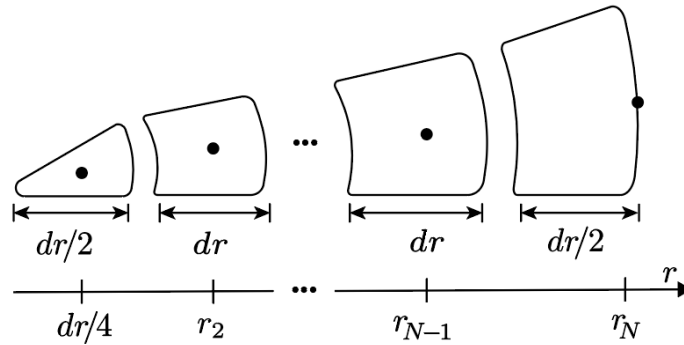
Figure 4.2. Differential elements representing the heat loss from the nodes along the top surface of the cylinder.

Note that in Fig. 4.2, the area in the r-direction is not constant. To compensate, the radius from the centerline to the center of each node is used along with the dr spacing to calculate the area of the piece. Additionally, the first node needs to be slightly modified since it is on the centerline. It has half the radial length of a common element being $dr/2$. The radius is taken at the centroid of the piece since taking the radius at the actual position will cause the piece not to be counted due to a zero radius..

The heat transfer along the surface edge is slightly easier to calculate since the area will be the same for each piece besides the top and bottom elements, which can be seen in Fig. 4.3. Summarily to the radial heat transfer, heat transfer along the side of the cylinder requires that the element is of half the size on the top and bottom. These pieces are taken at the overall radius of the cylinder.
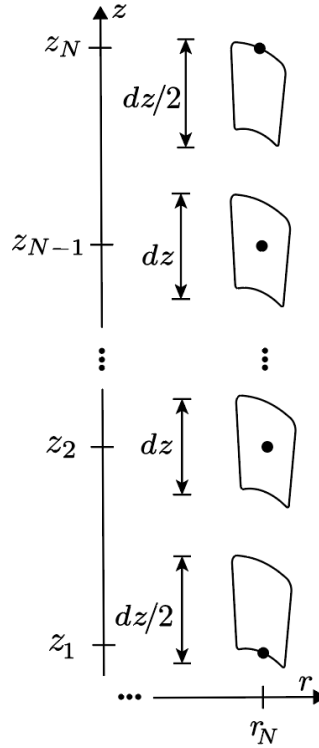
Figure 4.3. Differential elements representing the heat loss from the nodes along the side surface of the cylinder.

While Newton's law of cooling/heat will work for this application, the form of the equation is not particularly useful since the solution is being solved numerically. This equation can be rewritten into the form

$$\Delta Q = hA\Delta T\Delta t \tag{4.10}$$

by replacing the derivative terms with small changes and A is the area of the differential piece.

The total heat transfer can be found by summing the differential pieces along the top and multiplying by 2π

$$Q_{Top} = \sum_{n=1}^{N} 2\pi\Delta Q. \tag{4.11}$$

While the total heat transfer from the side can be found by summing the differential pieces along the side and multiplying by 4π as seen in

$$Q_{Side} = \sum_{m=1}^{M} 4\pi \Delta Q. \tag{4.12}$$

That allows for the total heat along the top and the bottom of the cylinder to be calculated; however, the total heat transfer from the cylinder would then be

$$Q_{Total} = 2Q_{Top} + Q_{Side} \tag{4.13}$$

due to there being both a top and a bottom to the cylinder plus the outside connecting the top and bottom caps.

The average temperature for the two-dimensional model is done by calculating the average temperature among the length of slab directions and the radial or cylindrical directions. In two-dimensional space there will be multiple rows and columns of nodes in each direction, the average temperature for each of these rows and columns is then averaged among there the corresponding method. These two values are then averaged one more time to provide the final average temperature of the entire cylinder itself.

4.8   GUI Development

The development of an interface is critical for ease of access to the program allowing it to be utilized with very little knowledge of programming. The software package Matlab, which was used to code all the equations, has a built-in GUI development kit. While this add-in is robust and works fantastic for quickly creating a GUI, it lacks intuitive instructions and online tutorials on how to develop a functioning interface. A quick overview will be provided of how the package works followed by a more detailed view of the GUI created for the heat transfer program.

When creating an interface, the design is straightforward. Drag and drop components one wishes to have into the editing area. Upon double-clicking components, a sub-menu pops up called the inspector that displays the control options for the specific component. The most useful is the name or tag component and the string which

displays words on the component. The tag of the component is used to reference the element while the string can be modified and set to a default value.

Once components have been placed into the interface editing area, the figure can be saved which creates a Matlab file. This .m file contains all the code necessary to run the program. The user interface grabs inputs from the user and saves them to the handles structure in Matlab, allowing them to be accessed outside of various functions. Typically, the program should have some push-button allowing it to start. This is where all the operating code would be stored. Upon completion of the operating code, the information can then be posted to the interface through text or a graphical figure.

The last important feature of the GUI development package is the use of callbacks which govern if a function is run while in the GUI. In order to have a button work, it must have a callback specified in the .m file. When a button is pressed or command activated, the function for the callback is run. By default, the GUI file creates many callback functions even if the buttons have no purpose, they can be deleted to save pre-processing time.

4.8.1   Application of GUI to Heat Transfer Code

Since the goal of the program is to be easily accessible for students to pick up and use, it is necessary to create an easy way to interface with the code. There are several ways to interface with a sophisticated program such as creating an input file, the function command line, or through some GUI to name a few examples. Each has advantages and disadvantages which will be discussed in detail.

So which form of interfacing would be best for this particular heat transfer application? Since most engineering students enrolled in a heat transfer course should have at least a minimum working knowledge of computer languages, that means any method discussed prior should be adequate. However, the depth at to which this knowledge extends should not be a limiting factor when operating the program. Due

to this, using the Matlab command line to input variables is clunky and should be avoided. An excel file would be an excellent option, but it would treat the Matlab program as a middle man who would add to computation time to read in specific values from a file. The GUI element is the ideal choice since it is created within Matlab itself, looks better than a standard excel sheet, and can export graphical elements directly to the interface. Additionally, the data from the program can be stored at the end of computation and printed to a file if the user wishes to interface with the data.

Before creating a GUI element within Matlab, it is recommended already have a working program or function within Matlab or a complex project. The GUIDE feature in Matlab can be challenging to debug and add functionality to elements. Dealing with the difficulties of the GUI tool and having to write a sophisticated program will add unnecessary stress to creating these types of features. Once a program has been developed, ensure it is easy to input the variables into the function and receive the outputs. A primary backbone function is easiest to run from the GUI which branches off into others instead of working in the GUI file directly.

Creating the Interface

Note that the GUI was created using the Matlab R2015b build. Utilizing other builds may cause various steps and processes to be different in the future or prior builds of the software. For this particular version of Matlab, the easiest way to begin a GUI is through the GUIDE figure creator which can be accessed by typing 'guide' in the command window. This process will open the GUIDE quick start menu shown in Fig. 4.4, which will have a few templates and allows the user to open any existing GUI figures to edit. The GUI created for heat transfer program is highly specific in operation, so a custom GUI was created from the blank template.

Once open, an untitled figure window appears that can be edited with various options for adding to the GUI window shown in Fig. 4.5. The left side of the window
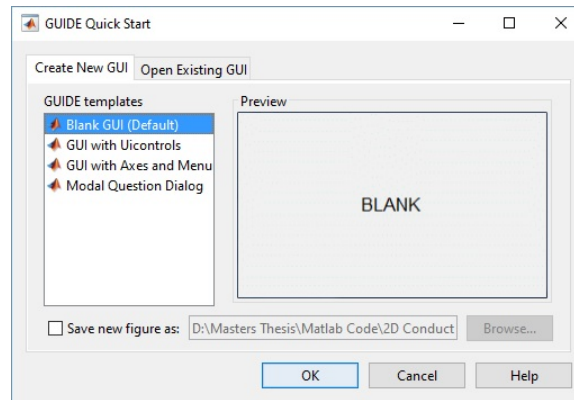
Figure 4.4.   GUIDE quick start menu in Matlab R2015b for the creation of GUI and figure elements.

has the buttons, menus, figures, text boxes, and so on that can be dragged and dropped into the window. It is recommended to put in panel elements first, then put the other options within the panel elements which have a couple of advantages. The panel elements allow for a clean and organized look of the final GUI window allow specific elements to be addressed for manipulation later, such as a hidden options menu. The specific panels can be named, allowing for buttons or options to be grouped together under a specific category.

The size of the GUI window can be adjusted by moving the black square in the bottom right of the window. The grid spacing and snap to the grid can be adjusted by using the Grid and Rulers button under the Tools menu option. The last important feature of the figure window itself is the green play button which will save the current configuration of the window and generate the corresponding GUI figure. Pressing this button the first time will additionally generate the .m file that will be manipulated in the next section to grab the inputs from the user.

When an element is dragged into the figure area, upon double-clicking, the property inspector window is displayed, as shown in Fig. 4.6. Many options can be adjusted from this menu from the appearance, position, alignment, and so on. The most important options in this menu are the tag, string, and callback components.
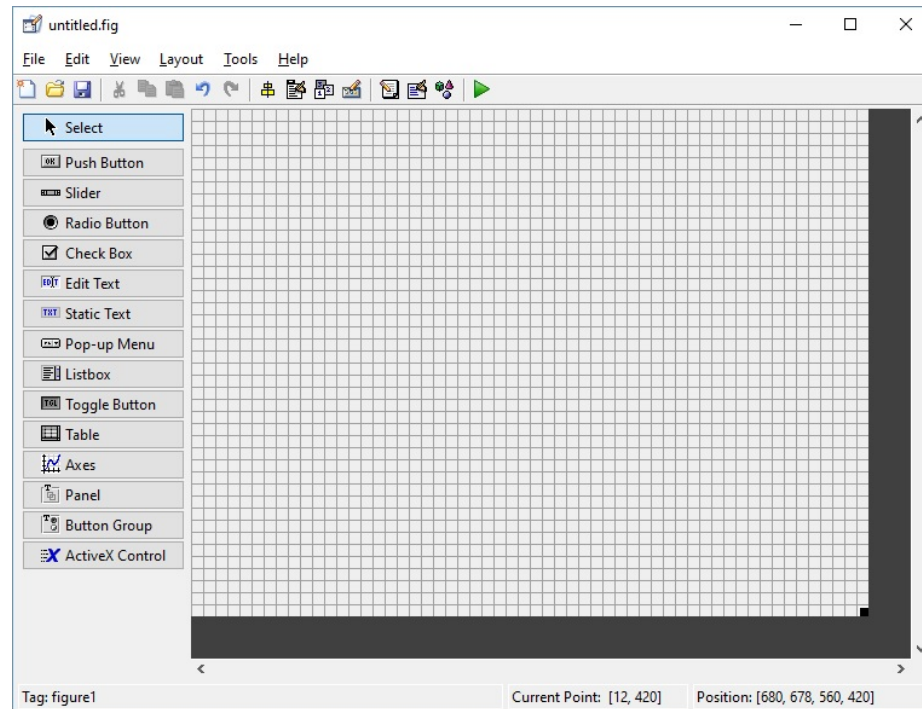
Figure 4.5. Blank GUI template from the GUIDE quick start menu.

Tag is the name of the element and is how to reference the element within the .m file to get the GUI element to perform an action. The string is text that will appear on the element in the window and can be converted to a value for calculations within Matlab. The callback is how a function for the element is created within the .m file, which will run when the element experiences a change of state.

The development of the GUI for the heat transfer program required the use of ten edit text elements, a drop-down list, two buttons, a checkbox, a radio button, six graphical panel elements, and static text which updates with live information from the program which can be seen in Fig. 4.7. The GUI is broken up into seven main components being the initial and boundary conditions panel, the material selection panel, the geometry panel, static program panel, control panel, advanced settings panel, and output graphical elements section. Each component is made up of edit boxes, text boxes, and graphical elements. The advanced settings menu is hidden
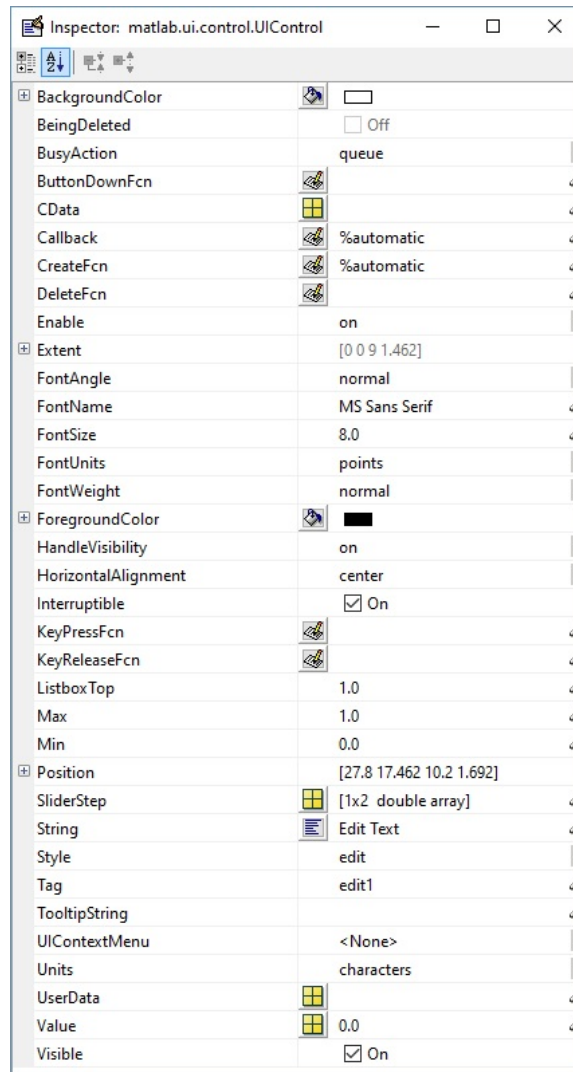
Figure 4.6.   Property inspector window where details of an element of the GUI can be manipulated.   This specific window shows the properties for the Edit Text menu option.

upon launch, and once the advanced settings box has been check, displays the panel for more advanced users to tweak the settings.

An important note is to include the default units the programmer wishes the users to work with while inputting data. This process could be more robust by including a menu to change the unit system, but for simplicity, the units are kept in a single standard. The default setting for each input has two main advantages: allowing
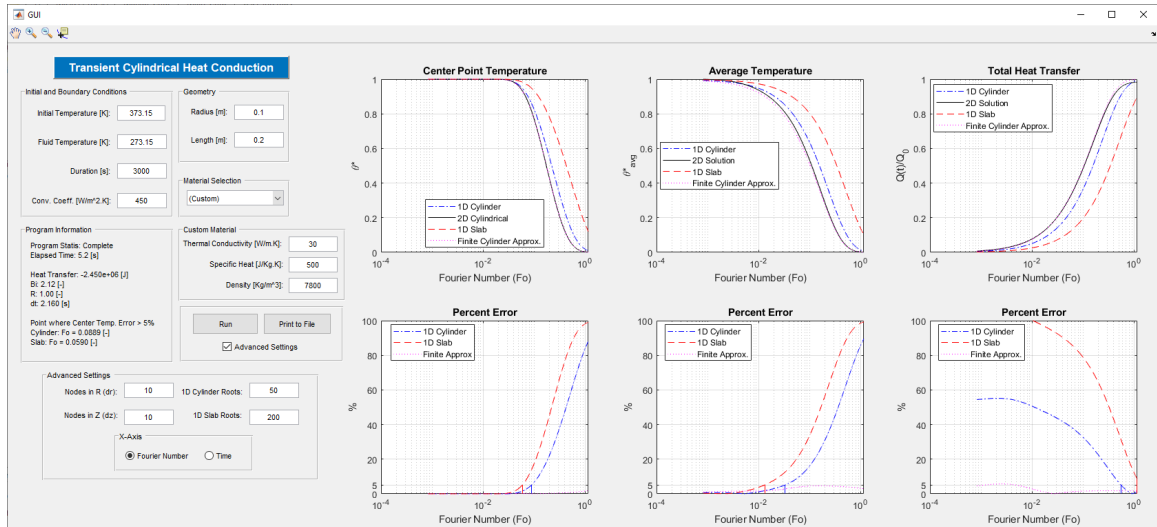
Figure 4.7. Complete GUI for the transient cylindrical heat conduction program with default run case pre-programmed into text boxes.

the programmer to quickly run a series of tests for debugging and giving the user a baseline for beginning to use the interface with changing variables and settings.

Getting Input Data

A function should be created which will convert the string data stored in the text fields into a double using the *str2double* command which can then be stored in the handles structure within Matlab. Creating a substructure within the handles structure is useful to categorize the values which can be done by putting a period after the word handles then a name such as data then another period and the name one wishes to give the stored data. Each element that was put into the GUI interface has a tag stored in the handles structure. If the tag is referenced, then any of the interface options can then be accessed. An example of how to grab the data might look like *"handles.data.Radius = str2doulble(get(handles.Radius,'String'));"*. This process is then be repeated as many times as there are inputs and will be run when

the getpara(handles) function is run, which will update the handles structure with the input data.

This data does not exist in the programs memory until the function is run to acquire the data. To achieve this, when the run button is pressed, the function to grab the input data is run then is passed into a separate function which runs the background calculations for the program. Once completed, the program exits the background calculations function back to the run button press function and saves all the output data to the handles structure for if the user wishes to print the data.

Running the interface

The user will first input data and select any settings or choose to go with the default options. When the run button is pressed, the function which reads the input data is run and stored to the handles structure. The conduction program is then run, which will first update the program into the running state so the user can see the program in operation. The GUI then passes all the input data and runs through the control panel procedure outlined in Fig. 4.1. Once completed, the output data is then passed to the graphical elements for a display to the user. Finally, the data is stored into the handles structure in case the user wishes to print the raw data values later.

Output Data

The output data is stored into the handles structure but not in a straightforward way. There is not an easy way to store data from a function once the run function has completed. This has to do with how MATLAB was coded and the scope of functions. Getting around this problem involves storing the data into the UserData tag of the print button itself acting as temporary storage of the data until the user decides to print or run another case. This method allows the data to be passed out

of a completed function into the handles function without losing the information due to leaving the function scope.

If the user decides to print the data, the results will be printed into an Excel file with the data columns labeled for easy identification along with the simulation time and date for archival purposes. Matlab has a variety of functions for writing data to an Excel file which were utilized to print the data. If a file already exists it is deleted so if the previous data set already run is valuable it should be renamed or moved to another location.

5. RESULTS

5.1 Aims

This chapter covers the potential impact of the program to be used as an educational tool, the design of experiments performed on the main problem inputs, and develops graphs to determine the exact point at which a two-dimensional analysis must be performed. The program has a graphical user interface that could be integrated into an introductory heat transfer course or lab component where students discover the general principles by changing various inputs of the system. An overview of all the main variables is analyzed using a design of experiments method to determine which of the parameters and there interactions have the largest impact on the overall simulation. Lastly, some graphs are developed to help determine when exactly a two-dimensional analysis must be applied for a specific geometry and Fourier number.

5.2 Potential Education Applications

Think back to any classroom environment during an advanced engineering course. The class was most likely taught with an old book containing information derived many years ago. The students learn the basics of the subject moving from general theorem to more specific details and guided examples; then homework is assigned over the material to challenge the students and extrapolate on the material to solve problems [16]. While this method is adequate, there is a large gap between practicing guided examples and solving complex problems; moreover, this method follows a linear pattern with little outside thought required of the students besides the problem sets. The information from another section is then taught often only relatively related

to the prior section, where the students lose the ability to perform those types of problems until reviewed again.

This entire process has been used in the engineering curriculum for ages and has not changed with advancing technology and computer access. This form of deductive learning causes students to learn information while it is needed then move onto the next section. General principles are introduced and memorized but not looked at in detail or derived in an intuitive way for the learner. With the rise of technology and the ability to perform more educational opportunities through the use of a computer, there are better ways to introduce the material to students. Additionally, a large amount of industry outside of academia uses computers and simulations to run tests and develop new products. In order to provide a well-rounded education for engineering students, it has become necessary to introduce computers and simulations into the classroom environment to better prepare students for a career.

While having students develop computer simulations would be a fantastic way to deepen detailed learning, get hands-on experience, and familiarity with computer software, it would be impractical for a core engineering course to focus so heavily on computer implementation. Many students may struggle with coding and lack the necessary background to develop robust simulations. This type of activity would be an excellent activity for a semester group project. Instead, introducing students to already existing software or pre-written programs would only require an introductory knowledge of the software or program which can be developed as part of the freshmen engineering curriculum or quickly introduced at the start of the lecture. These guided labs would focus on a particular aspect or general concept within the course, but students would have to discover the general principles from looking at data and graphs. This inductive form of learning has been shown to have a deeper level of understanding, and students can recall the information for a much more extended period when compared to a deductive learning style [1], [14].

While hands-on experience is fantastic at preparing students for homework sets and industry, the theoretical topics should not be ignored. The inductive style should

supplement the instruction of students breaking up a tedious classroom environment lecture.

5.2.1 Applications to Heat Transfer Education

In order to provide an example of what this type of learning environment could look like in a specific classroom setting, a program has been developed to be used in a heat transfer course or laboratory setting. The program covers the topic of conduction and looks at the differences between using a one- and two-dimensional transient analysis on a cylinder or slab which is placed in an isothermal fluid.

The instructor would need to introduce the concepts of a one-dimensional analysis and explain that the assumption is valid since the error will be insignificant when compared to a two-dimensional or three-dimensional analysis. The students can then utilize the program to run a few sample simulations to see how closely the one- and two-dimensional simulations match when the length is of significant magnitude greater than the radius.

Once students become comfortable with the program and the basic concepts at work, they should be encouraged to experiment with changing parameters and see how that affects the center point temperature, average temperature, heat transfer, and the associated errors between the one- and two-dimensional outputs. Students should write down what type of experiments they wish to perform and note any observations on each run. Currently, the program runs using the general theories of conduction heat transfer and does not include any statistical variation that would occur in a real laboratory setting; therefore, if two groups run the same experiment, they should see the same results from the simulation.

When completed, the students should share any notable observations from various simulations and try to develop the background theory and potential applications. From this activity, the instructor can then elaborate in further detail on the one-dimensional assumption and explain the usefulness. The most substantial benefit

being the computation time for the computer to simulate if it is known the error is small.

The experiment has the potential to be used in a more advanced heat transfer course by utilizing the advanced settings option menu. Students that are more familiar with programming, two-dimensional simulations, and numerical analysis would benefit from changing settings that affect the computational time and accuracy of the simulations. The simulations can be run with higher accuracy but at the cost of having to wait a significant amount of time longer for the computer to calculate to a much higher degree of accuracy. This process would lead to the topic of round-off vs. truncation error and how by just increasing the nodes does not necessarily mean the simulation is more accurate.

For a sample of an experiment that could be run for a laboratory class, please refer to Appendix (A).

5.3   Design of Experiments for Heat Transfer Program

Applying the design of experiments to the entirety of the developed heat transfer program would require a significant amount of time due to the sheer amount of inputs and variables; however, if the inputs are limited to the seven standard inputs as shown in Table 5.1, then the process becomes manageable. The standard inputs were developed and given high and low values of operation. Since the program is not statistically based, each run will produce similar results, so the order of the experiments does not matter. A complete list of the experiments run with the two investigated outputs being maximum cylinder error and maximum slab error is shown in Appendix B.

Once completed, the average values for each interaction were calculated for the cylinder error approximation. Interaction plots were developed for every combination of two inputs, as shown in Fig. 5.1. It is interesting to note that besides the interaction of the initial and fluid temperature, which produced identical lines, every pair of

Table 5.1.   High and low levels for the standard inputs of the heat transfer program.

Input	-1 Level	+1 Level
Initial Temperature (A):	373.15	423.15
Fluid Temperature (B):	273.15	323.15
Time (C):	3000	4000
Convection Coefficient (D):	350	450
Radius (E):	0.1	1.0
Length (F):	0.2	1.0
Material/Alpha (G):	Aluminum	Copper

inputs interact with one another to affect the maximum cylinder error output. It is hard to tell the significance of the interaction from this plot; however, due to the inconsistant scaling on the subplots to compare the interactions.

Figure 5.1.   All possible interaction effects between any two given inputs for cylinder error.

It becomes easier to see the effects when the interactions are ranked from highest to lowest and graphed in a Pareto chart, as seen in Fig. 5.2. Since there is a total of 127 different interaction effects on the maximum cylinder error result, only the significant effects, over 0.5, are shown. The significant visible effect is the length interaction on the error. Since the result was negative, it means that by increasing the length the error should be expected to go down, which matches both the theory and the test results. In order to more clearly see the other interaction effects, the graph is redrawn in Fig. 5.3 without the length interaction to view the other effects.
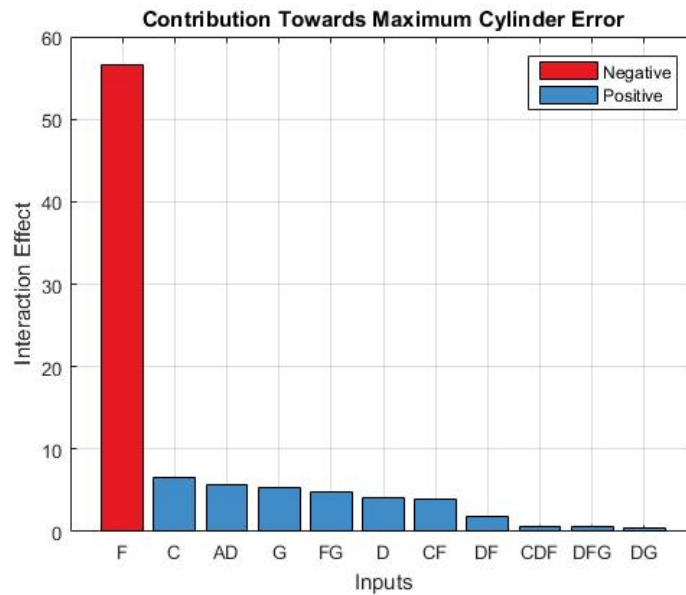
Figure 5.2. Pareto chart for the significant variables on the resulting cylinder error.

The second most important factor towards the maximum error is time. Under short time spans the error is small but rises quickly as the end effects of the cylinder begin to affect the center temperature. An interesting result is an effect on the initial temperature, and convection coefficient had on maximum error. It appears that as these values increase the maximum error will increase as well since the temperature gradient becomes large with an increased ability to transfer heat away from the cylinder.
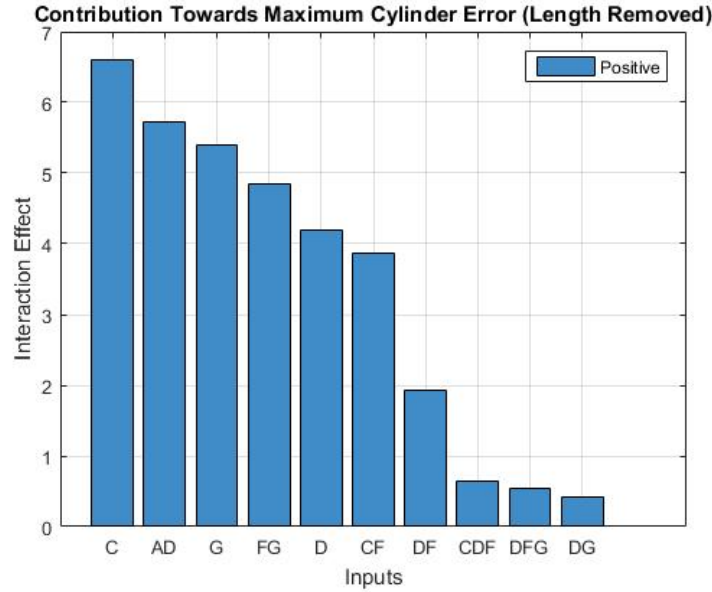
Figure 5.3.   Pareto chart, excluding length, for the significant variables on the resulting cylinder error.

The average values for each interaction were then calculated for the slab error approximation. Interaction plots were developed for every combination of two inputs, as shown in Fig. 5.4. It is interesting to note that besides the interaction of the initial and fluid temperature, which produced identical lines, every pair of inputs interact with one another to affect the maximum slab error output. This interaction can additionally be seen between the initial temperature and the length along with the fluid temperature and length. It is hard to tell the significance of the interaction from this plot; however, due to the inconsistent scaling on the subplots to compare the interactions.
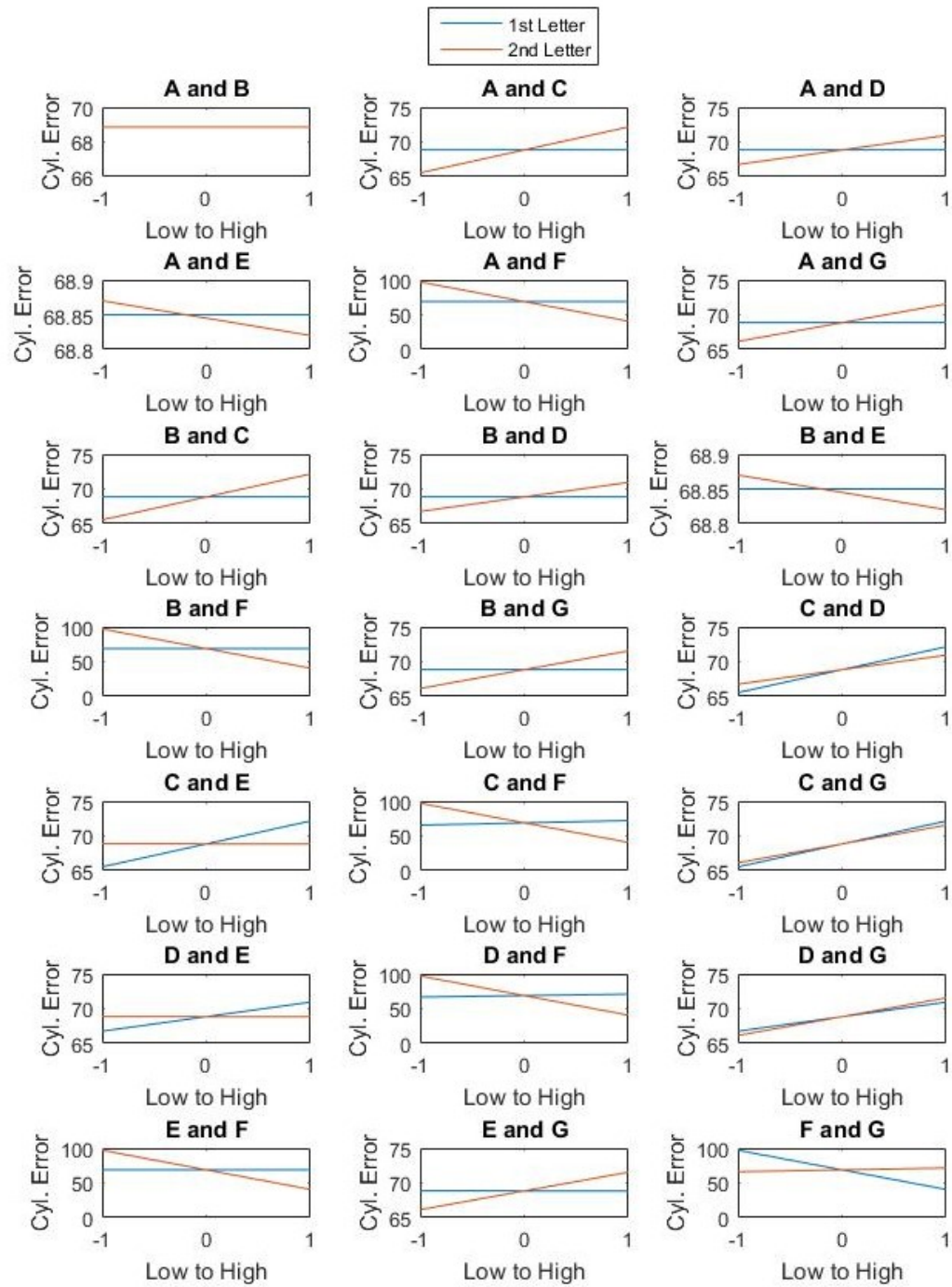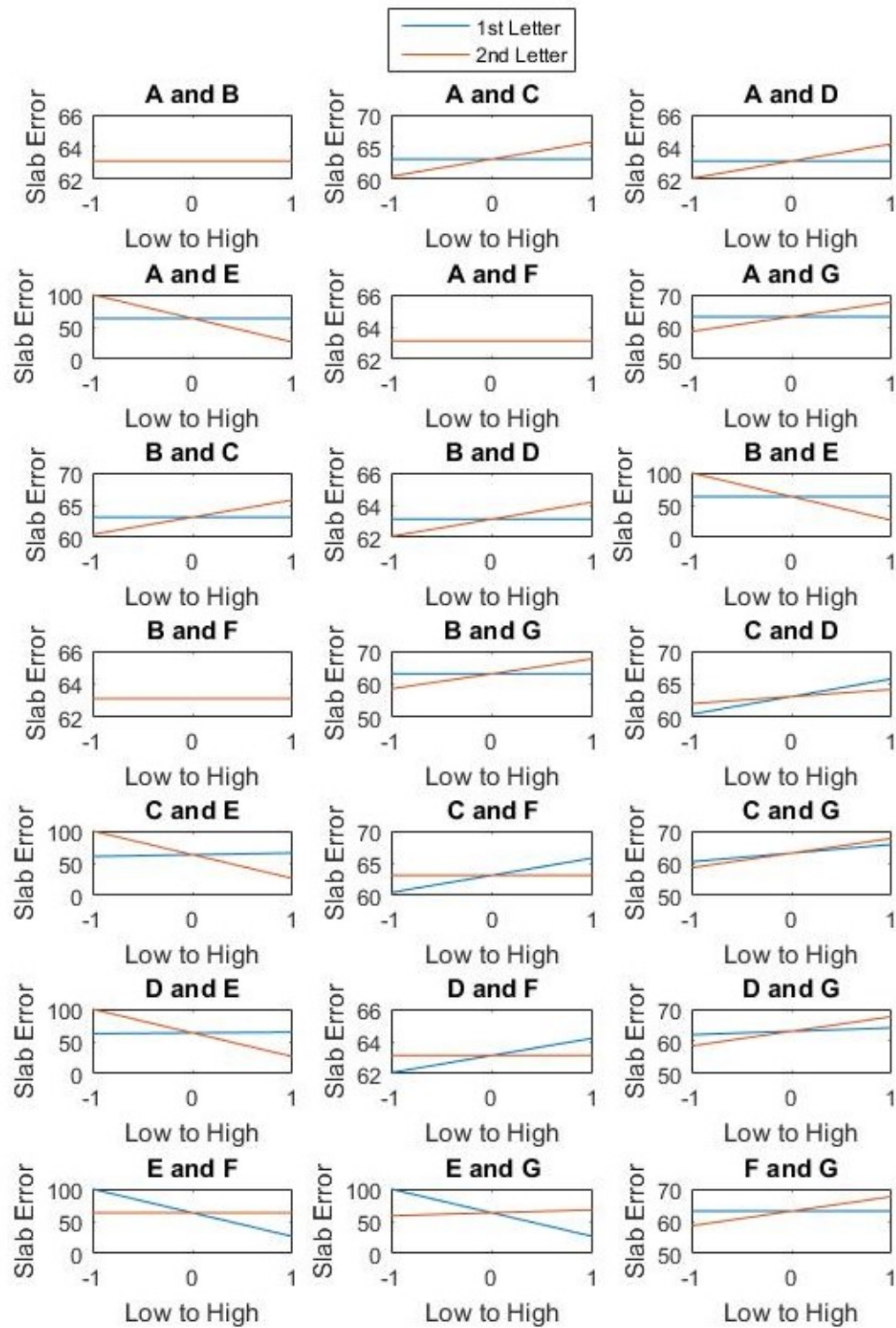
Figure 5.4. All possible interaction effects between any two given inputs for slab error.

Since there is a total of 127 different interaction effects on the maximum slab error result, only the significant effects, over 0.5, are shown in Fig. 5.4. The noticeable significant effect is the radius interaction on the error. Since the result was negative, it means that by increasing the length the error should be expected to go down, which matches both the theory and the test results. In order to more clearly see the other interaction effects, the graph is redrawn in Fig.5.6 without the length interaction to view the other effects.
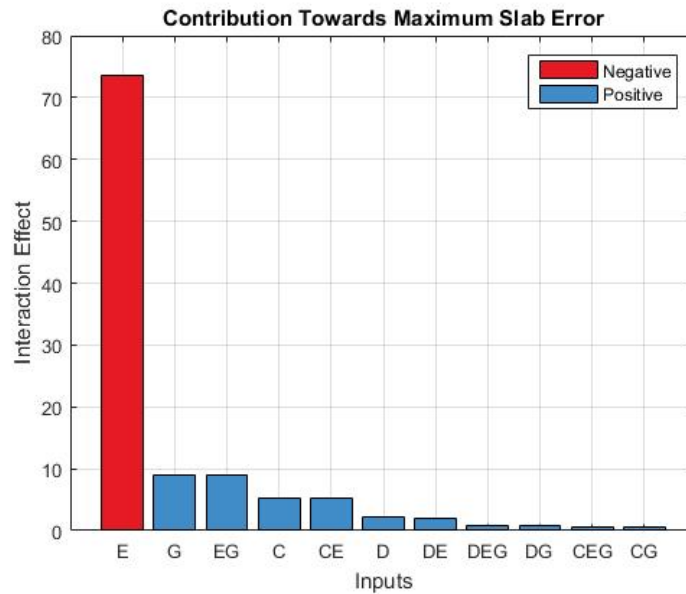
Figure 5.5.   Pareto chart for the significant variables on the resulting slab error.

In contrast to Fig. 5.3, Fig. 5.6 has a pairing of effects where with the addition of the radius causes an almost equal but slightly lesser effect. This effect can be seen with the second strongest interaction on the output being the material properties. Upon considering the interaction effect between the material properties with the radius, it produces an almost equivalent effect to just the material property itself. This is seemingly mirrored for each new variable. This effect needs to be studied in more detail to reveal the significance of this result fully. The main interaction effects involving the output of the slab error are the material property, the duration

of the study, and the convection coefficient along with the effects from the radius interaction.
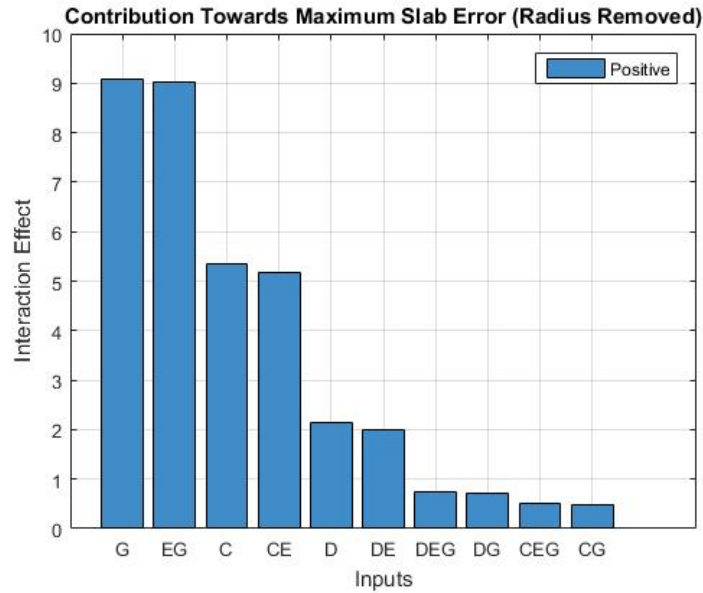
Figure 5.6. Pareto chart, excluding radius, for the significant variables on the resulting slab error.

5.4 Error Graphs

While the program developed is useful as a heat transfer education tool, the information provided from the interface could be utilized in other ways to conclude the temperature gradients within cylinders. One such application involved looking at the error between the center temperature of the two-dimensional simulation with the one-dimensional slab and cylinder results.

The idea is to create a graph which will tell the user when a specific cylinder or slab will require a two-dimensional analysis. Ideally, the graph should depend on the geometry, material, and length of the simulation, and the user can determine how long a one-dimensional analysis would suffice within a five percent error margin.

While the interface developed can display the results from a single simulation, it is not sufficient for storing large amounts data from multiple cases.

When initially investigating this idea, the main issue which arose was trying to limit the number of inputs required in order to simplify the graphic explicitly having the ratio between the cylinder length and radius on one axis while having to deal with the characteristic length from the Fourier number on the other axis. The workaround for this came from multiplying the Fourier number by the square of the Biot number; this eliminates the characteristic length from that axis while taking into account the material properties of the cylinder.

The other issue that arises is the volume of the cylinder under investigation. Cylinders with the same length to radius ratio can have widely differing errors at the same Fourier number due to the difference in volume. The fix for this is simple since the volume can be held constant. If the ratio is substituted into the volume of a cylinder equation, the radius can be calculated

$$r_0 = \left(\frac{V}{2\pi R} \right)^{\frac{1}{3}} \tag{5.1}$$

along with the length

$$L = 2r_0 R \tag{5.2}$$

meaning that only the volume and the ratio need to be inputs for the system. With these isolated, a script was written that cycles through ratios at a given volume and saves the error values. These error values can then be plotted into a contour plot as shown for the cylinder error in Fig. 5.7. This graph, while useful, is not easy to use since multiple would have to be created at different volumes and trying to interpolate between graphs would be inconvenient.

The answer is to choose an error ratio, such as five percent instead, then run another case and stack the contour plots on top of one another using the hold feature in MATLAB. The same thing can be done for the slab error with an inverse effect along the ratio axis. This final graph was split into two different entities, one for long time durations and another for a short time durations. In order to better understand
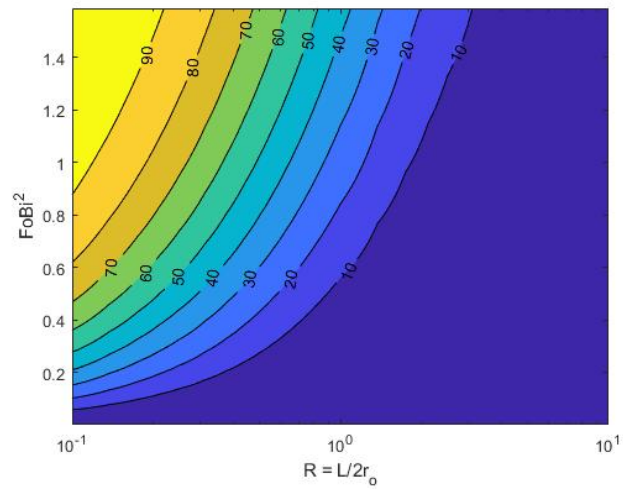
Figure 5.7. First attempt at creation of an error graph for the one-dimensional cylinder approximation.

what the final graphs represent a simple graph is constructed in Fig. 5.8 to explain how to utilize the graph.
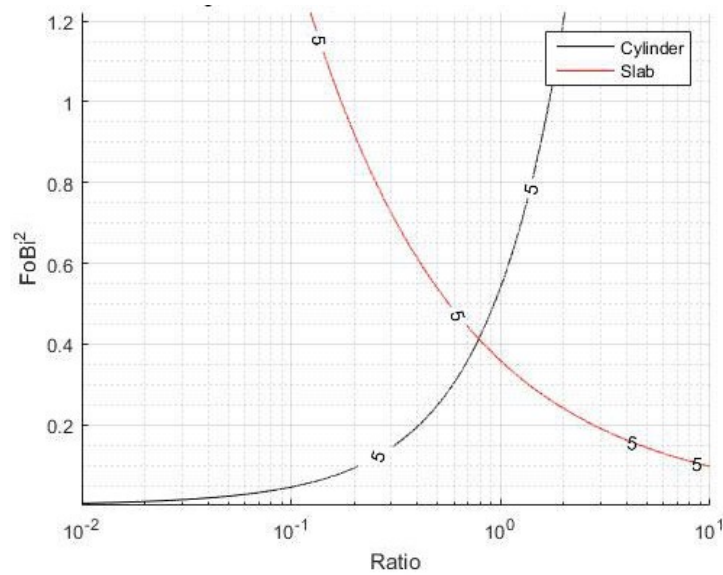
Figure 5.8. Example graph on how to read the contour plot.

Fig. 5.8 shows the five percent error thresholds for the cylinder and slab results for a volume of two meters squared. Anything below the cylinder error line represents where a one-dimensional cylinder analysis can be run. Anything below the slab error line represents where a one-dimensional slab analysis can be run. Anything below both the cylinder and slab error lines can be run with a cylinder or slab analysis. Finally, anything above both the cylinder and slab error lines must be run with a two-dimensional analysis.

Upon stacking the contour plots on top of one another, it generates the final graph in Fig. 5.9. Each line represents the respective cylinder or slab five percent error threshold where each line has an associated geometry volume. Any result above specific line for the analysis will require a two-dimensional solution to reduce the error from a one-dimensional approximation. This graph covers more extended duration tests from cylinders of larger volumes where the short duration tests from smaller volumes are shown in Fig. 5.10.
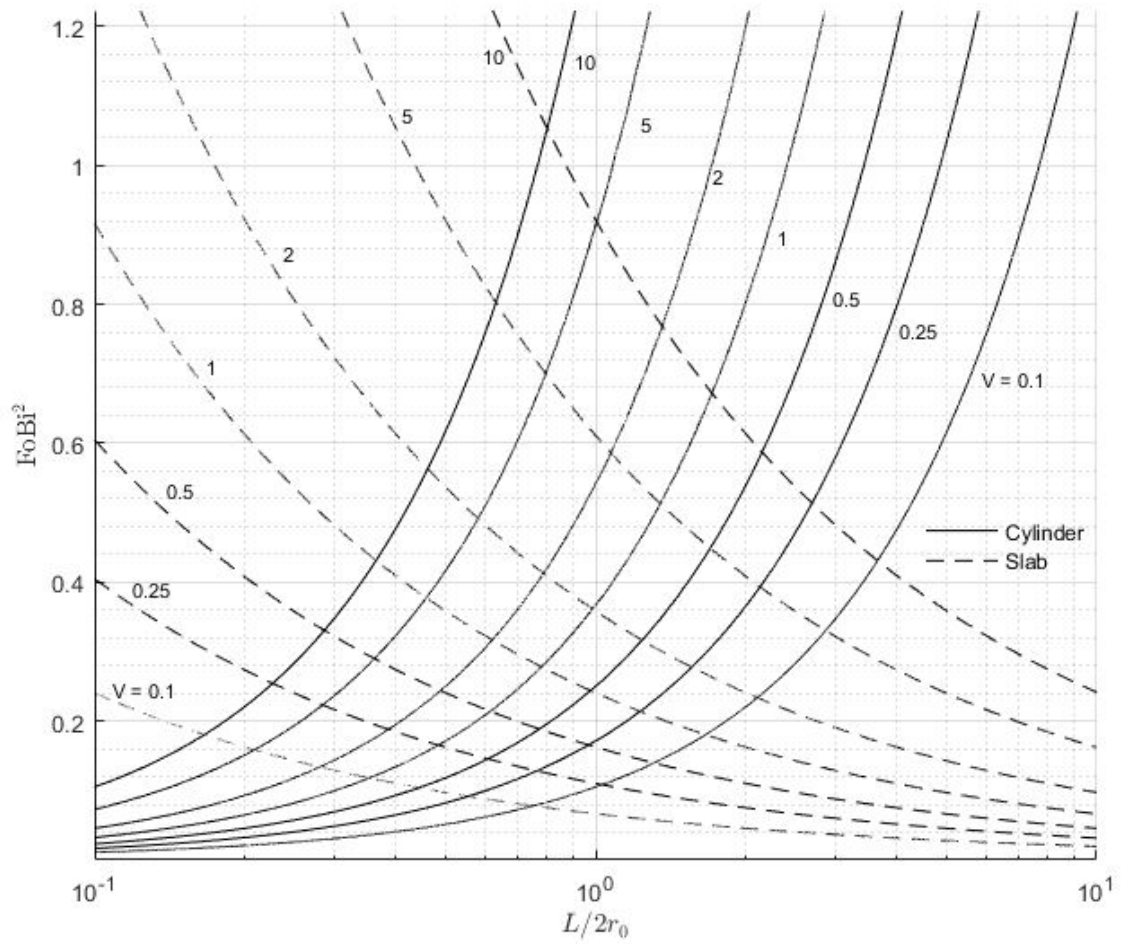
Figure 5.9. Long duration, large volume (in m^3) chart to determine when a one-dimensional approximation becomes inaccurate
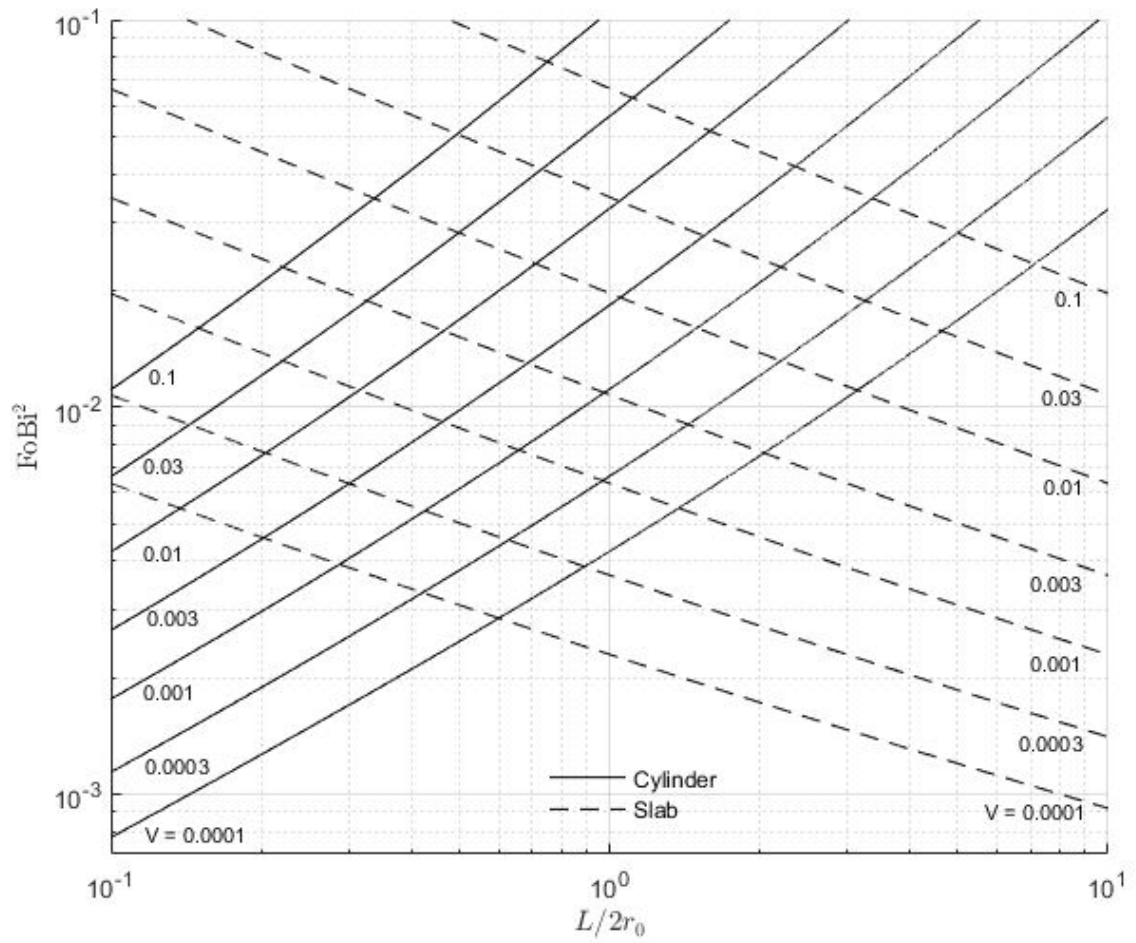
Figure 5.10. Short duration, small volume (in m^3) chart to determine when a one-dimensional approximation becomes inaccurate

6. CONCLUSIONS

6.1 Aims

This chapter covers the conclusions drawn from the research conducted throughout the thesis. All objectives are analyzed and summarized to describe all that was learned. Potential applications of the findings are discussed with recommendations for implementing into a classroom setting. Finally, some possible future directions of the work are discussed to continue the initial research into formal studies to gauge student interactions with the teaching style and implementation.

6.2 Accomplishment of Objectives

1. Develop a mathematical model for a cylinder at a uniform initial temperature subjected to a large fluid body of uniform temperature.

The model was developed using finite difference techniques, as shown in chapter 2 of this thesis. The mesh represents a cross-sectional view of a quarter slice within a cylinder which allows for much faster processing when computing the solution of the temperature field.

2. Propose a finite difference solution method capable of changing each variable.

A variety of inputs to the model can be manipulated including the initial temperature, ambient temperature, convection coefficient, material property, geometry, and length of simulation along with more complicated mesh and specific solution parameters.

3. Develop a user interface that allows the user to learn by changing variables.

A user interface was created utilizing the MATLAB software packing through the GUIDE add-in feature. This process allowed relatively easy creation of a GUI through

the built-in tools of the software program rather then programming a GUI from the ground up.

4. Utilize the MATLAB software package to implement the proposed solution and user interface.

The solution method has been coded in the MATLAB environment along with the GUI. The program is capable of being stored in a small file that can be run and saved to the program apps within MATLAB for a specific computer that will allow the user to run the program.

5. Develop a single or multiple graphical elements that can be used to determine if a particular cylinder requires a one- or two-dimensional analysis.

A graph has been developed that allows the user to determine whether a one- or two-dimensional study should be run based off of the geometry, length of time, and material properties. The user needs to find the corresponding curve on the graph and select anything below the curves for an error less than five percent.

6.3   Thesis Summary

In chapter 2, the basics of heat transfer were introduced, which are the primary features of the program in which a student would need to learn and discover from use of the program and through traditional lectures. These concepts are relatively simple taken one at a time, and most can be discovered at face value from the program after a few trials run. The basic concepts involving the design of experiments are additionally introduced, which are used later in the thesis to determine the interaction effects between the inputs to discover the significance of each on the center point temperature error.

The equations used to develop the model are derived in chapter 3. There are nine equations in total for the top, left, right, and bottom faces, top-left, top-right, bottom-left, bottom-right corners, and the interior nodes. The equations for the

stability criteria are also covered since the equations are derived with an explicit method. These equations will allow teachers and students to follow the methods used, which are scattered about many textbooks, and help aid in the creation of new models.

In chapter 4, the basic layout of the program developed in MATLAB is discussed explaining the various components which make up the model. The general structure consists of a GUI master function which leads to a control panel which computes the outputs from the given inputs. These inputs are passed into the control panel and are manipulated into general dimensionless parameters if possible for simple calculations. The stability criteria is determined, which allows the one-dimensional cylinder and slab approximations to the center point temperature to be calculated along with the average temperature and instantaneous heat transfer. Then the two-dimensional simulation is calculated for the center point temperature along with the average temperature and instantaneous heat transfer. These outputs are then exported back into the GUI where the error values are processed and plotted to the output interface in which the user can see the results.

In chapter 5, the merits of the program are discussed regarding education in a heat transfer classroom along with introducing a chart in which one can determine when a two-dimensional study must be run. The program has significant potential implications in a classroom or laboratory environment since students can discover the concepts of general problems themselves instead of through a traditional textbook or lecture. This inductive approach of learning is shown to have a significant impact on learning and topic retention, which can better prepare students for exams and real-life applications. The use of computer programs in the classroom allows students to work hands-on with models and simulations which make up a large portion of industry practices. The chart developed is significant in how the differences between a one- and two-dimensional analysis are handled. The chart is easy to use since the user only needs to know the geometry, material properties, and length of time of the simulation to use the graphic. Most sources state a rough approximation as the

length being ten times the radius as a significant threshold. While valid for most cases, from the graph, this can be seen to be potentially a high-end estimate for most cases depending on the length of time required for an experiment.

6.4 Limitations

The conclusions regarding the effectiveness in the classroom are based entirely on research showing the benefits of inductive learning being incorporated in the classroom. The program has been tested with mostly colleagues and professors instead of students learning the material for the first time. Further, since there has been minimal feedback regarding the function of the program, there are most likely improvements needed for ease of use regarding the program and intuitiveness of the GUI interface.

6.5 Suggestions for future work

While the program developed has potential significant applications in the field of engineering education, it would be interesting to see data conducted in a study seeing the effectiveness of a program such as this in a classroom or laboratory setting. Additionally, the program is static at the moment and will produce identical results given the same inputs. Developing some statistical variety through a Monte Carlo simulation involving the material properties or with the surrounding temperature would be interesting to incorporate to mimic a laboratory experiment.

REFERENCES

REFERENCES

[1] Stephanie Farrell and Robert P Hesketh. An inductive approach to teaching heat and mass transfer. *age*, 5:1, 2000.

[2] Frank P Incropera, Adrienne S Lavine, Theodore L Bergman, and David P DeWitt. *Fundamentals of heat and mass transfer*. Wiley, 2007.

[3] Yunus A. Çengel. *Heat transfer: a practical approach*. McGraw-Hill, New York, 2003.

[4] M Necati Ozisik. *Boundary value problems of heat conduction*. Dover Publications, Inc., 2013.

[5] WJ Minkowycz, Ephraim M Sparrow, GE Schneider, and RH Pletcher. Handbook of numerical heat transfer. *New York, Wiley-Interscience, 1988, 1035 p. No individual items are abstracted in this volume.*, 1988.

[6] Paul J Schneider. *Conduction heat transfer*. Addison-Wesley Pub. Co., 1955.

[7] Anthony F. Mills. *Basic heat and mass transfer*. Irwin, 1995.

[8] Leon R. Glicksman and John H. Lienhard V. *Modeling and Approximation in Heat Transfer*. Cambridge University Press, 2016.

[9] Keith M Bower. What is design of experiments (doe)?

[10] Richard M Felder, Linda K Silverman, et al. Learning and teaching styles in engineering education. *Engineering education*, 78(7):674–681, 1988.

[11] JB Jones. The non-use of computers in undergraduate engineering science courses. *Journal of Engineering Education*, 87(1):11–14, 1998.

[12] Robert J Ribando, Timothy C Scott, and Gerald W OLeary. Application of the studio model to teaching heat transfer. *age*, 6:1, 2001.

[13] Ernest T Smerdon. New roles for engineering faculty. In *Proceedings of the ICEE*, 2001.

[14] Robert J Ribando, Larry G Richards, and Gerald W OLeary. A hands-on approach to teaching undergraduate heat transfer. In *ASME 2004 International Mechanical Engineering Congress and Exposition*, pages 413–422. American Society of Mechanical Engineers Digital Collection, 2004.

[15] Rozhgar Othman. Steady state and transient analysis of heat conduction in nuclear fuel elements. *masters degree project, KTH-stockholm sweden*, 2004.

[16] Richard M Felder and Rebecca Brent. Is technology a friend or foe of learning? *Chemical Engineering Education*, 34(4):326–337, 2000.

APPENDICES

A. SAMPLE LABORATORY EXPERIMENT

Transient Heat Conduction: One- vs Two-dimensional Error Analysis

Objective: The objective of this experiment is to examine the error between performing a one-dimensional analysis and a two-dimensional analysis on a cylinder or slab by utilizing a computer program.

Procedure:

1. Log-in to the computer and open the Matlab software environment.

2. Download the Matlab program from the course website. (The instructor will provide a link).

3. Open the file entitled Transient_Conduction and a GUI should appear.

4. Run the simulation using the default settings and ensure you get the same output as shown by the instructor to verify the program is working.

5. Develop a test to run by manipulating the various parameters in the program. Be sure to write down the parameters used for each test and make notes and observations for each.

6. If the test is particularly interesting, save the file for review later using the print button.

7. Repeat steps 5 and 6 until at least ten different tests have been run for analysis.

Observations:

a) Record parameters for each test.

b) Record all observations noticed on the data sheet during the experiment.

c) Save test files and interesting graphs.

B. DOE: DESIGN MATRIX

Table B.1. : The design of experiment design matrix featuring a 2k factorial design with seven inputs.

	A	B	C	D	E	F	G	Maximum	Maximum
Run	Initial Temp.	Fluid Temp.	Time	Conv. Coeff.	Radius	Length	Alpha	Cylinder Error	Slab Error
1	373.15	273.15	3000	350	0.1	0.2	Alm.	93.74	99.65
2	373.15	273.15	3000	350	0.1	0.2	Copper	94.63	99.73
3	373.15	273.15	3000	350	0.1	1.0	Alm.	28.01	99.64
4	373.15	273.15	3000	350	0.1	1.0	Copper	37.13	99.73
5	373.15	273.15	3000	350	1.0	0.2	Alm.	93.72	11.51
6	373.15	273.15	3000	350	1.0	0.2	Copper	94.62	27.14
7	373.15	273.15	3000	350	1.0	1.0	Alm.	27.90	11.47
8	373.15	273.15	3000	350	1.0	1.0	Copper	37.12	27.19
9	373.15	273.15	3000	450	0.1	0.2	Alm.	96.97	99.92
10	373.15	273.15	3000	450	0.1	0.2	Copper	97.60	99.95
11	373.15	273.15	3000	450	0.1	1.0	Alm.	32.71	99.92
12	373.15	273.15	3000	450	0.1	1.0	Copper	43.78	99.95
13	373.15	273.15	3000	450	1.0	0.2	Alm.	96.96	13.61
14	373.15	273.15	3000	450	1.0	0.2	Copper	97.59	32.13

Table B.1 continued

	A	B	C	D	E	F	G	Maximum	Maximum
Run	Initial Temp.	Fluid Temp.	Time	Conv. Coeff.	Radius	Length	Alpha	Cylinder Error	Slab Error
15	373.15	273.15	3000	450	1.0	1.0	Alm.	32.52	13.54
16	373.15	273.15	3000	450	1.0	1.0	Copper	43.75	32.18
17	373.15	273.15	4000	350	0.1	0.2	Alm.	97.54	99.95
18	373.15	273.15	4000	350	0.1	0.2	Copper	97.98	99.96
19	373.15	273.15	4000	350	0.1	1.0	Alm.	37.90	99.95
20	373.15	273.15	4000	350	0.1	1.0	Copper	47.25	99.96
21	373.15	273.15	4000	350	1.0	0.2	Alm.	97.53	20.45
22	373.15	273.15	4000	350	1.0	0.2	Copper	97.98	38.04
23	373.15	273.15	4000	350	1.0	1.0	Alm.	37.76	20.43
24	373.15	273.15	4000	350	1.0	1.0	Copper	47.23	38.08
25	373.15	273.15	4000	450	0.1	0.2	Alm.	99.07	99.99
26	373.15	273.15	4000	450	0.1	0.2	Copper	99.31	100.00
27	373.15	273.15	4000	450	0.1	1.0	Alm.	43.60	99.99
28	373.15	273.15	4000	450	0.1	1.0	Copper	54.77	100.00
29	373.15	273.15	4000	450	1.0	0.2	Alm.	99.06	23.75
30	373.15	273.15	4000	450	1.0	0.2	Copper	99.31	44.26
31	373.15	273.15	4000	450	1.0	1.0	Alm.	43.37	23.69
32	373.15	273.15	4000	450	1.0	1.0	Copper	54.72	44.31
33	373.15	323.15	3000	350	0.1	0.2	Alm.	93.74	99.65
34	373.15	323.15	3000	350	0.1	0.2	Copper	94.63	99.73
35	373.15	323.15	3000	350	0.1	1.0	Alm.	28.01	99.64

Table B.1 continued

Run	A Initial Temp.	B Fluid Temp.	C Time	D Conv. Coeff.	E Radius	F Length	G Alpha	Maximum Cylinder Error	Maximum Slab Error
36	373.15	323.15	3000	350	0.1	1.0	Copper	37.13	99.93
37	373.15	323.15	3000	350	1.0	0.2	Alm.	93.72	11.51
38	373.15	323.15	3000	350	1.0	0.2	Copper	94.62	27.14
39	373.15	323.15	3000	350	1.0	1.0	Alm.	27.90	11.47
40	373.15	323.15	3000	350	1.0	1.0	Copper	37.12	27.19
41	373.15	323.15	3000	450	0.1	0.2	Alm.	96.97	99.92
42	373.15	323.15	3000	450	0.1	0.2	Copper	97.60	99.95
43	373.15	323.15	3000	450	0.1	1.0	Alm.	32.71	99.92
44	373.15	323.15	3000	450	0.1	1.0	Copper	43.78	99.95
45	373.15	323.15	3000	450	1.0	0.2	Alm.	96.96	13.61
46	373.15	323.15	3000	450	1.0	0.2	Copper	97.59	32.13
47	373.15	323.15	3000	450	1.0	1.0	Alm.	32.52	13.54
48	373.15	323.15	3000	450	1.0	1.0	Copper	43.75	32.18
49	373.15	323.15	4000	350	0.1	0.2	Alm.	97.54	99.95
50	373.15	323.15	4000	350	0.1	0.2	Copper	97.98	99.96
51	373.15	323.15	4000	350	0.1	1.0	Alm.	37.90	99.95
52	373.15	323.15	4000	350	0.1	1.0	Copper	47.25	99.96
53	373.15	323.15	4000	350	1.0	0.2	Alm.	97.53	20.45
54	373.15	323.15	4000	350	1.0	0.2	Copper	97.98	38.04
55	373.15	323.15	4000	350	1.0	1.0	Alm.	37.76	20.43
56	373.15	323.15	4000	350	1.0	1.0	Copper	47.23	38.08

Table B.1 continued

Run	A Initial Temp.	B Fluid Temp.	C Time	D Conv. Coeff.	E Radius	F Length	G Alpha	Maximum Cylinder Error	Maximum Slab Error
57	373.15	323.15	4000	450	0.1	0.2	Alm.	99.07	99.99
58	373.15	323.15	4000	450	0.1	0.2	Copper	99.31	100.00
59	373.15	323.15	4000	450	0.1	1.0	Alm.	43.60	99.99
60	373.15	323.15	4000	450	0.1	1.0	Copper	54.77	100.00
61	373.15	323.15	4000	450	1.0	0.2	Alm.	99.06	23.75
62	373.15	323.15	4000	450	1.0	0.2	Copper	99.31	44.26
63	373.15	323.15	4000	450	1.0	1.0	Alm.	43.37	23.69
64	373.15	323.15	4000	450	1.0	1.0	Copper	54.72	44.31
65	423.15	273.15	3000	350	0.1	0.2	Alm.	93.74	99.65
66	423.15	273.15	3000	350	0.1	0.2	Copper	94.63	99.73
67	423.15	273.15	3000	350	0.1	1.0	Alm.	28.01	99.64
68	423.15	273.15	3000	350	0.1	1.0	Copper	37.13	99.73
69	423.15	273.15	3000	350	1.0	0.2	Alm.	93.72	11.51
70	423.15	273.15	3000	350	1.0	0.2	Copper	94.62	27.14
71	423.15	273.15	3000	350	1.0	1.0	Alm.	27.90	11.47
72	423.15	273.15	3000	350	1.0	1.0	Copper	37.12	27.19
73	423.15	273.15	3000	450	0.1	0.2	Alm.	96.97	99.92
74	423.15	273.15	3000	450	0.1	0.2	Copper	97.60	99.95
75	423.15	273.15	3000	450	0.1	1.0	Alm.	32.71	99.92
76	423.15	273.15	3000	450	0.1	1.0	Copper	43.78	99.95
77	423.15	273.15	3000	450	1.0	0.2	Alm.	96.96	13.61

Table B.1 continued

Run	A Initial Temp.	B Fluid Temp.	C Time	D Conv. Coeff.	E Radius	F Length	G Alpha	Maximum Cylinder Error	Maximum Slab Error
78	423.15	273.15	3000	450	1.0	0.2	Copper	97.59	32.13
79	423.15	273.15	3000	450	1.0	1.0	Alm.	32.52	13.54
80	423.15	273.15	3000	450	1.0	1.0	Copper	43.75	32.18
81	423.15	273.15	4000	350	0.1	0.2	Alm.	97.54	99.95
82	423.15	273.15	4000	350	0.1	0.2	Copper	97.98	99.96
83	423.15	273.15	4000	350	0.1	1.0	Alm.	37.90	99.95
84	423.15	273.15	4000	350	0.1	1.0	Copper	47.25	99.96
85	423.15	273.15	4000	350	1.0	0.2	Alm.	97.53	20.45
86	423.15	273.15	4000	350	1.0	0.2	Copper	97.98	38.04
87	423.15	273.15	4000	350	1.0	1.0	Alm.	37.76	20.43
88	423.15	273.15	4000	350	1.0	1.0	Copper	47.23	38.08
89	423.15	273.15	4000	450	0.1	0.2	Alm.	99.07	99.99
90	423.15	273.15	4000	450	0.1	0.2	Copper	99.31	100.00
91	423.15	273.15	4000	450	0.1	1.0	Alm.	43.60	99.99
92	423.15	273.15	4000	450	0.1	1.0	Copper	54.77	100.00
93	423.15	273.15	4000	450	1.0	0.2	Alm.	99.06	23.75
94	423.15	273.15	4000	450	1.0	0.2	Copper	99.34	44.26
95	423.15	273.15	4000	450	1.0	1.0	Alm.	43.37	23.69
96	423.15	273.15	4000	450	1.0	1.0	Copper	54.72	44.31
97	423.15	323.15	3000	350	0.1	0.2	Alm.	93.74	99.65
98	423.15	323.15	3000	350	0.1	0.2	Copper	94.63	99.73

Table B.1 continued

	A	B	C	D	E	F	G	Maximum	Maximum
Run	Initial Temp.	Fluid Temp.	Time	Conv. Coeff.	Radius	Length	Alpha	Cylinder Error	Slab Error
99	423.15	323.15	3000	350	0.1	1.0	Alm.	28.01	99.64
100	423.15	323.15	3000	350	0.1	1.0	Copper	37.13	99.73
101	423.15	323.15	3000	350	1.0	0.2	Alm.	93.72	11.51
102	423.15	323.15	3000	350	1.0	0.2	Copper	94.62	27.14
103	423.15	323.15	3000	350	1.0	1.0	Alm.	27.90	11.47
104	423.15	323.15	3000	350	1.0	1.0	Copper	37.12	27.19
105	423.15	323.15	3000	450	0.1	0.2	Alm.	96.97	99.92
106	423.15	323.15	3000	450	0.1	0.2	Copper	97.60	99.95
107	423.15	323.15	3000	450	0.1	1.0	Alm.	32.71	99.92
108	423.15	323.15	3000	450	0.1	1.0	Copper	43.78	99.95
109	423.15	323.15	3000	450	1.0	0.2	Alm.	96.96	13.61
110	423.15	323.15	3000	450	1.0	0.2	Copper	97.59	32.13
111	423.15	323.15	3000	450	1.0	1.0	Alm.	32.52	13.54
112	423.15	323.15	3000	450	1.0	1.0	Copper	43.75	32.18
113	423.15	323.15	4000	350	0.1	0.2	Alm.	97.54	99.95
114	423.15	323.15	4000	350	0.1	0.2	Copper	97.98	99.96
115	423.15	323.15	4000	350	0.1	1.0	Alm.	37.90	99.95
116	423.15	323.15	4000	350	0.1	1.0	Copper	47.25	99.96
117	423.15	323.15	4000	350	1.0	0.2	Alm.	97.53	20.45
118	423.15	323.15	4000	350	1.0	0.2	Copper	97.98	38.04
119	423.15	323.15	4000	350	1.0	1.0	Alm.	37.76	20.43

Table B.1 continued

Run	A Initial Temp.	B Fluid Temp.	C Time	D Conv. Coeff.	E Radius	F Length	G Alpha	Maximum Cylinder Error	Maximum Slab Error
120	423.15	323.15	4000	350	1.0	1.0	Copper	47.23	38.08
121	423.15	323.15	4000	450	0.1	0.2	Alm.	99.07	99.99
122	423.15	323.15	4000	450	0.1	0.2	Copper	99.31	100.00
123	423.15	323.15	4000	450	0.1	1.0	Alm.	43.60	99.99
124	423.15	323.15	4000	450	0.1	1.0	Copper	54.77	100.00
125	423.15	323.15	4000	450	1.0	0.2	Alm.	99.06	23.75
126	423.15	323.15	4000	450	1.0	0.2	Copper	99.31	44.26
127	423.15	323.15	4000	450	1.0	1.0	Alm.	43.37	23.69
128	423.15	323.15	4000	450	1.0	1.0	Copper	54.72	44.31