IMPROVED HEAT TRANSFER COEFFICIENT DATA FOR GAS TURBINE

COOLING CHANNELS USING INVERSE METHODS

By

Tommy Kurt Smith

A thesis

submitted in partial fulfillment

of the requirement for the degree of

Master of Science in Mechanical Engineering

Boise State University

December 2011

BOISE STATE UNIVERSITY GRADUATE COLLEGE

DEFENSE COMMITTEE AND FINAL READING APPROVALS

of the thesis submitted by

Tommy Kurt Smith

Thesis Title: Improved Heat Transfer Coefficient Data for Gas Turbine Cooling
Channels Using Inverse Methods

Date of Final Oral Examination: 10 October 2011

The following individuals read and discussed the thesis submitted by student Tommy
Kurt Smith, and they evaluated his presentation and response to questions during the final
oral examination. They found that the student passed the final oral examination.

James R. Ferguson, Ph.D.   Chair, Supervisory Committee

Donald Plumlee, Ph.D.   Member, Supervisory Committee

Inanc Senocak, Ph.D.   Member, Supervisory Committee

The final reading approval of the thesis was granted by James R. Ferguson, Ph.D., Chair
of the Supervisory Committee. The thesis was approved for the Graduate College by
John R. Pelton, Ph.D., Dean of the Graduate College.

ABSTRACT

Determination of heat transfer in channel flow is important in many fields, with particular interest to this research being the cooling channels in gas turbine engine blades. Validation of gas turbine engine design is an essential step in their development process. Accurate knowledge of heat transfer that occurs within turbine blades during their operation allows for reduction of thermal stresses, increasing blade life and energy efficiency. Measuring heat flux, q, directly is difficult, so it is often calculated based on Newton's Law of Cooling.

Use of thermochromic liquid crystals (TLCs) in determining heat transfer coefficients h is common, as they allow full-field temperature measurement by allowing the experimenter to measure surface temperatures in a non-invasive fashion. Direct measurement of bulk flow temperature T_∞ is difficult, with computation requiring detailed upstream information. T_∞ and h are known for established geometries, but become uncertain in complex geometries.

The goal of this study was to develop a technique using inverse methods to estimate h and T_∞ simultaneously using experimental transient TLC surface temperature data. To apply this method to complex geometries, it was first desired to develop it on a simple geometry. An experimental apparatus was designed, immersing a flat plate in a wind tunnel capable of varying fluid flow speed and temperature. The surface of the plate was coated with TLCs and recorded with a digital camera. The plate was subjected to a

sudden heating of the air flow, and the TLC response was recorded. The hue camera data was converted to temperature data, being validated by an array of thermocouples.

Analytical models were developed that related surface temperature to time, h, and T_∞, in which the profile of T_∞ in time was assumed first to be a step function, then a series of ramps. These surface convection formulations were used with a conjugate gradient inverse method to estimate h and T_∞ using TLC hue temperature data as the input.

The inverse method was tested with models and data of increasing complexity at three plate positions at various distances from the plate leading edge. First, a step change model was used to verify h and T_∞ could be estimated simultaneously. Then, experimental hue temperature data was used with the series of ramps model to estimate these parameters.

The lead position (Position 1) worked very well with the step function, producing T_∞ values within 4% of true values, and h values within accepted ranges. Positions 2 and 3 had relatively successful results, predicting T_∞ with 10% accuracy but with h values greater than accepted correlation ranges. Use of generated data with the series of ramps formulation predicted algorithm convergence with large error, which was corroborated with parameter estimation using experimental data. Experimental data produced large variances in initial T_∞ slopes, but was still able to minimize the objective function in a stable way. It was concluded that the method works but will require additional constraints for increased accuracy.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

xii

xiii

xiv

xvi

xviii

xix

NOMENCLATURE

Symbols

<u>Latin</u>

a confidence interval factor

A area (m^2)

b ramp y-axis intercept (K)

B time-varying boundary condition (K)

B blue

c_p specific heat capacity (J/kg-K)

C covariance

C M-m

D diameter (m)

E voltage (V)

G green

h heat transfer coefficient (W/m^2-K)

H Heaviside function

H hue

k thermal conductivity (W/m-K)

m temperature ramp slope (K/s)

m minimum of RGB

xx

M	maximum of RGB
p	pitch (m)
q	heat flux (W)
r	degrees of freedom
R	red
S	saturation
t	time (s)
T	temperature (K)
$\hat{T}$	offset temperature $T - T_0$ (K)
U	x velocity component (m/s)
v	local axial velocity (m/s)
v	kinematic viscocity (m^2/s)
V	average axial velocity (m/s)
V	value
x	stream-wise distance (m)
y	distance from a surface (m)

Greek

α	thermal diffusivity (m^2/s)
β	semi-infinite convenience variable $\frac{h\sqrt{\alpha t}}{k}$ (-)
γ	conjugation coefficient (-)
δ	boundary layer thickness (m)
δ	search step size (-)
θ	angle of reflection (rad)

xxi

λ wavelength (m)

η similarity variable (-)

ζ root of a transcendental equation

φ objective function

ρ density (kg/m^3)

τ time offset (s)

τ time constant (s)

σ^2 variance

σ standard Deviation

Matrices

d   data vector

d   search direction vector

e   error misfit vector

G   functional matrix

J   Jacobian sensitivity matrix

m   parameter vector

P   experimental parameter vector

∇S   gradient vector

Dimensionless Numbers

Bi   Biot number (-)

Fo   Fourier number (-)

Re   Reynolds number (-)

Pr   Prandtl number (-)

Nu   Nusselt number (-)

Subscripts

d   data

es   estimated value

f   fluid

f   fundamental

h   hydraulic

h   heat transfer coefficient

i   intermediary, or ith value

∞   infinity, denotes driving values

in   inlet

m   parameter

n   final, or nth value

s   surface

s   Seebeck

T   temperature

w   wall

0   initial or starting value

center centerline of a plate

x   based on *x*

emf   electromagnetic field

cjc   electronic cold junction

sys   total system

xxiii

Superscripts

*	Dimensionless
'	first derivative
"	per area
T	transpose
k	kth iteration

Abbreviations

CJC	cold junction compensator
CMY	cyan, magenta, yellow
CORR	correlation
COV	covariance
DAQ	data acquisition
EMF	electromagnetic field
HSV	hue, saturation, value
HTC	heat transfer coefficient
RGB	red, green, blue
SSE	sum of squares error
TLC	thermochromic liquid crystal
TSP	temperature-sensitive paint
VAR	variance

1 INTRODUCTION

The gas turbine engine has become an increasingly integral part of everyday life due to its many advantages over other engines. Their high thermal efficiency make stationary gas turbine engines a mainstay for generating electricity, and their high power-to-weight ratio make them ideal power plants for vehicles – especially aircraft. This makes their use in the aerospace industry pervasive. As research is conducted in miniaturization and increased efficiency, the use of gas turbine engines in industry is likely to grow larger for the indefinite future.

1.1 Temperature Measurement Problems in Complex Channels

Since the efficiency of a gas turbine increases with increasing combustion temperatures, cooling channels were added to gas turbine blades in the 1960s to maintain mechanical integrity [1]. Since then, efforts have continued to maximize the cooling efficiency of these blades through better channel design and, concurrent with this, a need developed for better techniques to validate designs. The complex geometry of these blades makes traditional measurement techniques difficult. The goal of this thesis is to present an alternative heat transfer measurement technique in gas turbine cooling channels, a technique that combines transient experimental techniques with inverse parameter estimation methods. This technique has the potential to be more accurate and easier to perform than traditional methods.

1.1.1   Gas Turbine Engine Design

Gas turbine engines operate on the thermodynamic Brayton cycle. Air is pulled into the engine through the inlet and compressed in a multiple stage compressor. The compressed air is then injected with fuel and ignited in the combustion chamber. The combustion gasses push through the turbine extracting energy and then is exhausted. The extracted energy can then subsequently be converted to electricity on a stationary gas turbine engine, or diverted for use in propulsion. Figure 1.1 shows these cycle steps on a typical gas turbine engine.
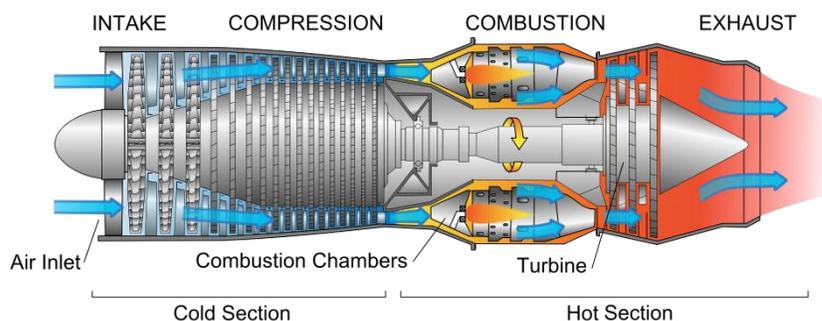
Figure 1.1 Turbine Stages: Gas turbine engines follow a Brayton cycle [2].

The efficiency of gas turbine engines is largely dependent on the inlet temperature to the turbine following the combustion chamber. Higher temperatures translate to more energy that can be extracted with the turbine blades. However, higher temperatures shorten the life of these blades. Servicing and replacing turbine blades is a major upkeep cost in the life of an engine. To extend blade life and reduce costs, the blades are cast with internal cooling channels. As can be seen in Figure 1.2, these channels have a complex geometry, with air flowing into them via the "root" at the base of the blade, making several passes along the length, and then exiting through both the leading and trailing edges. Additionally, turbulators are often added to the channels to increase heat transfer rates.
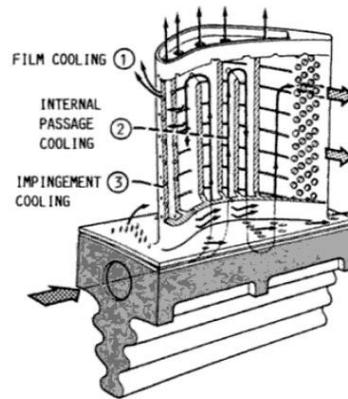
Figure 1.2 Turbine Blade Cross Section: This isometric view shows a cross section of a typical engine blade and its internal cooling passages. The arrow traces the path of cooling air and lists the different types of cooling effects [1].

The design challenge of these cooling channels is how to design the channels such that coolant flow is used effectively. The goal of the blade designer is to keep the blade at a uniform temperature below a temperature limit to lengthen blade life. To do this, the designer shapes the channel to direct coolant flow and maximize heat transfer rate at places where the blade is hottest, and reduce it at the places where the blade is cooler, all while minimizing channel pressure drop through the channel and maintaining uniform blade temperature.

Current testing techniques to analyze cooling channel heat transfer in these blades are involved, requiring extensive work and data analysis. An example technique is the use of an array of heat flux gauge thermopile sensors [3]. These measure heat transfer coefficients indirectly by calculating heat flux based on a temperature differential. They require an insertion point into the walls of the apparatus for each sensor, making the technique invasive. To obtain a temperature field, this and other techniques require approximations and interpolation of point temperature measurements. Because of this, heat transfer data gathered is at times inaccurate and uncertain [4].

With more accurate heat transfer values within the channel, hot spots on the walls of the cooling channels could be more readily and accurately identified, and more efficient cooling channels could be designed. This, in turn, leads to both reduced maintenance costs and greater engine efficiency. Since the channels cool more efficiently, combustion can be run at a higher temperature without adding to operating costs. Or, since less air is needed for cooling, more air can be used for combustion, increasing the power and efficiency of the engine. Both benefits result in a better overall product. Also, higher efficiency could be used for increased pressure used in fuel-air mixing. This would lead to a more complete combustion mixture, reducing NO_x emissions and enabling a cleaner engine.

1.1.2   Defining the Driving Temperature

The core of blade cooling channel design involves controlling heat transfer at the channel wall. To do this, having measurements that help the engineer calculate accurate heat flux, q'', values is vital. In nearly every heat transfer design scenario in which convection plays a role, it is desirable to also know the temperature difference between the fluid and the surface, $(T_\infty - T_s)$, which drives heat transfer, with accuracy. This leads to more accurate heat transfer coefficients, h, and surface heat fluxes, q''.

These three variables are interrelated by Newton's Law of Cooling:

$$q'' = h(T_\infty - T_s) \tag{1.1}$$

where T_s is the temperature of the surface through which the heat flux occurs. Heat transfer between a fluid and a surface is a complex process. Newton's Law of Cooling is an empirical formula that simplifies calculations for this process. It has been verified experimentally and has been found useful for engineering purposes. The driving

temperature T_∞ is an abstraction. One knows where to measure the surface temperature T_s; its location is implicit in its name. The location of measurement of T_∞, however, is not as simple. The combination of the temperature difference $(T_\infty - T_s)$ and h need to be evaluated such that the correct heat flux, q'', is calculated.

This evaluation is complicated by the definition of T_∞. T_∞ is the temperature outside the boundary layer for an external flow, and a *mixing cup* temperature at a flow cross section in a bounded flow. In heat transfer engineering, the mixing cup temperature is defined as the *mean temperature* of the stream in a duct. It is important to note that the mean temperature is not an average. Rather, it is derived from the First Law of Thermodynamics for a bulk flow:

$$T_\infty = \frac{1}{VA} \iint_A vT dA \tag{1.2}$$

in which V is the average axial velocity, v is the local axial velocity, and A is the cross-sectional area of the duct [5]. Locating T_∞ for this mixing cup temperature is difficult for all but the simplest of internal flow geometries.

The true driving force behind heat transfer is not a temperature difference at all, but rather a temperature gradient. The correct evaluation of heat flux in the fluid at a fluid-solid interface is found through Fourier's Law of Conduction:

$$q'' = -k_f \frac{\partial T_f}{\partial y}\Big|_{y=0} \tag{1.3}$$

in which the *f* subscript denotes properties for the fluid. Fourier's Law of Conduction is also an empirical formula like Newton's Law of Cooling. However, instead of containing a temperature difference, Fourier's Law contains a temperature differential at the interface. Also, the kinetic theory of gasses supports the relationship [6]. This definition of q'' is a more proper one, driven by the temperature gradient at the wall. Fourier's Law

of Conduction when combined with Newton's Law of Cooling leads to a more complete definition of h. When Equations 1.1 and 1.3 are combined and rearranged to define h, the result is:

$$h = \frac{-k\frac{\partial T_f}{\partial y}\Big|_{y=0}}{(T_\infty - T_s)}. \tag{1.4}$$

The heat transfer coefficient, h, is a measure of the ratio of the temperature gradient at the wall to the driving temperature difference. However, the problem of locating T_∞ remains. Figure 1.3 shows a visual representation of this gradient.
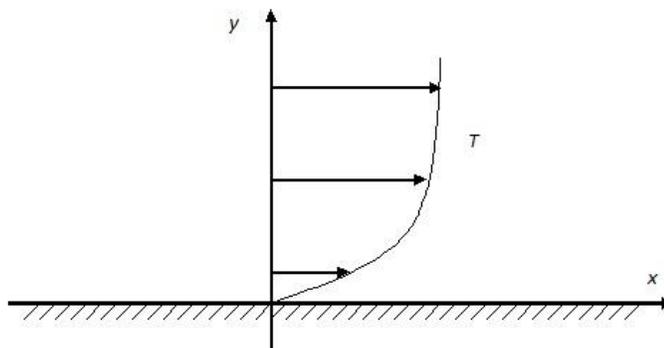
Figure 1.3 Temperature Gradient: The temperature profile T is uniform as it approaches the free stream, but has a gradient at the wall.

While this relationship is very useful for gaining an intuitive grasp of h, it is not very useful experimentally since the temperature gradient at the wall is difficult to measure. This is why Newton's Law of Cooling, despite its flaws, is useful. It contains only temperatures that need to be measured, and the difficulty lies in locating T_∞ for accurate h and q'' values.

These values have already been determined empirically for many common heat transfer problems. For example, when the free stream is uniform in external flow, simply measuring T_∞ a distance outside of the boundary layer is sufficient. Likewise, in a

straight uniform channel in which flow is symmetric, calculating the mixing cup temperature is simplified since the channel and flow symmetry make calculating averages a straightforward process. However, when seeking T_∞ in the more complex system of a gas turbine engine blade cooling channel, the problem becomes complex.

Such a complex geometry leads to a non-uniform flow field and a lack of symmetry. Defining the cross section to compute a mixing cup temperature is problematic. Further complications arise from the fact that, in channel flow, there is often not a single $\boldsymbol{T}_\infty$. As air flows through a channel, heat is constantly flowing through the walls, changing $\boldsymbol{T}_\infty$ in the stream-wise direction. This means that, for each cross section of the duct, there exists a unique $\boldsymbol{T}_\infty$. This $\boldsymbol{T}_\infty$ must be consistent with energy lost or gained through the upstream walls of the duct. Consequently, an accurate calculation of $\boldsymbol{T}_\infty$ requires knowing q'' over all upstream wall surfaces.

Figure 1.4 shows an example of how free stream temperature can change stream-wise in a channel for a constant wall temperature. Dimensionless distance x^* is the distance from the start of the duct divided by hydraulic diameter, $\frac{x}{D_H}$. Dimensionless temperature T^* is calculated from stream temperature T_∞, wall temperature T_s, and inlet temperature T_in, $T^* = \frac{T_\infty - T_{in}}{T_s - T_{in}}$. This further reinforces the idea that defining and determining T_∞ through measurement is problematic. For this figure, it was assumed wall boundary layers had crossed, and the entrance length was neglected. Also, h was held constant.
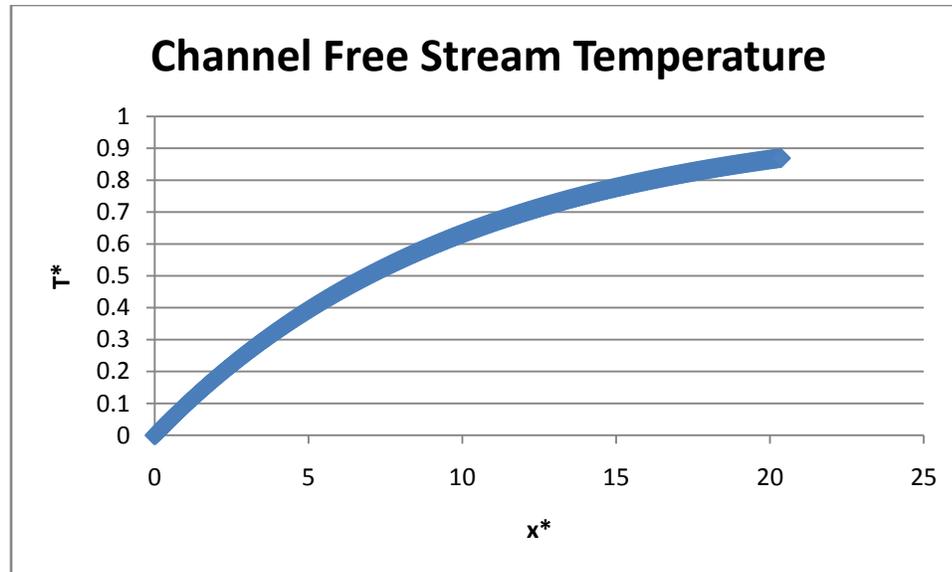
Figure 1.4 Channel Temperature: Each stream-wise location in the duct has a unique T_∞. The conditions for this example are constant wall temperature, circular duct.

Using the mathematical definition of mixing cup temperature (Equation 1.2) in an experimental context is prohibitive, since there are few non-invasive ways to determine T at multiple points in a flow without changing the velocity distribution. There are methods, however, for determining the temperature field on solid surfaces. The purpose of this research was to explore alternate methods of arriving at accurate T_∞ and h values that require fewer measurements and calculations of the free stream temperature. Combining the use of transient experimental techniques with inverse methods, simpler and more accurate estimates of h and T_∞ are possible in complex geometries, requiring only local surface temperature history.

1.2   Problem Description

The goal of this research was to develop a method to determine both the local heat transfer coefficients, h, and the corresponding bulk free stream temperatures, T_∞, on a flat plate in cross flow subjected to a transient near-step temperature change of the fluid,

given surface temperature response data. Instead of treating T_∞ as a value to experimentally measure to find the surface heat flux, it was treated as an additional parameter to *h* to determine from the experimental surface temperature history. In this way, the problems arising from Newton's Law of Cooling requiring experimental determination of T_∞ were no longer relevant. With accurate *h* and T_∞ values, q'' could be calculated. The new problem, however, was how to solve for this additional parameter in T_∞.

To determine *h* and T_∞, analytical models for forced convective heat transfer across the surface of a semi-infinite object were used. These models present surface temperature as a function of time, with *h*, T_∞, and material properties as parameters (Equation 1.5).

$$T_s = T_s(t; h, T_\infty, \dots) \tag{1.5}$$

To solve for *h* and T_∞, parameter estimation with inverse methods was implemented using numerical methods. The role of the inverse method was to determine the parameters that best fit, in a least squares sense, the surface temperature data using the analytical model. The particular inverse solution method used was the conjugate gradient method. This method is iterative, starting with an educated guess at the parameter vector and using a given model to calculate a new vector to compare with data. Based on how different the two vectors are, the method changes the parameters and recalculates, repeating this procedure until the difference between the measured data and calculated data is minimized within a predetermined threshold.

For purposes of calculation, it was simplified that T_∞ was a piecewise-linear function to approximate its actual shape, as observed in Figure 1.5. It can be seen that the

initial rise of the free stream temperature and the long, slow rise portions can be very accurately represented by a single ramp each. The transition between the two, however, involved rapid shape change, and thus more ramps are needed to represent the form with accuracy. Second, h was assumed to be constant. It can be seen why this is true from Equation 1.4. For most time, the rate of increase between the temperature gradient at the wall $\frac{\partial T}{\partial y}\Big|_{y=0}$ and the temperature difference $(T_\infty - T_s)$ is very close. This relationship breaks down for very small times at which the wall gradient approaches infinity. However, work done with FLUENT numerical models revealed this time to be on the order of a few milliseconds, which was considered small enough for the constant h assumption to hold [7]. For this assumption, the fluidic boundary layer had already been established.
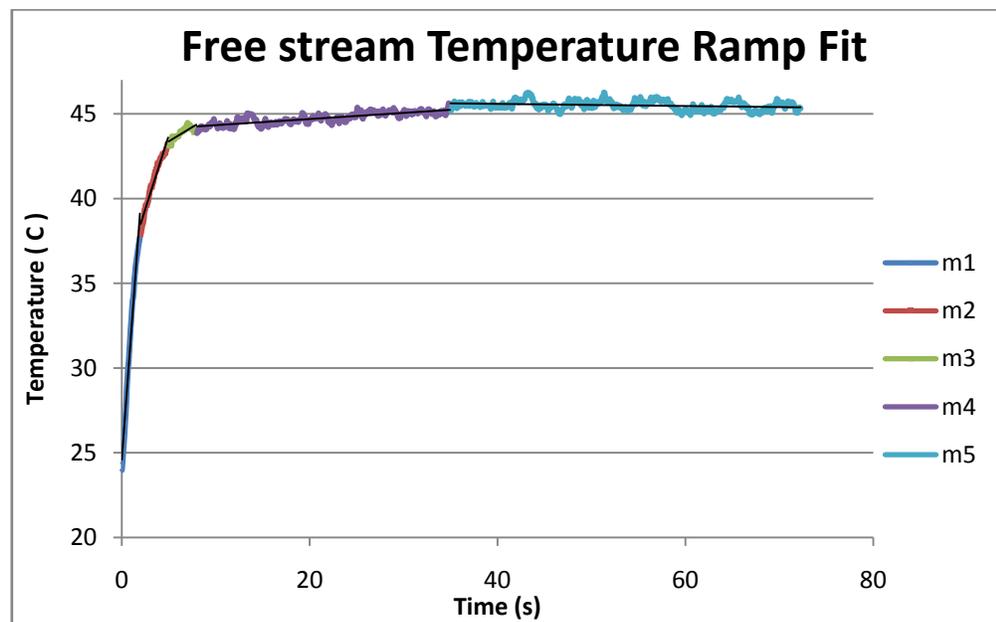
Figure 1.5 Temperature Ramp Fit: Typical free stream temperature rise of a wind tunnel experimental trial. The m_i values indicate different fit slopes.

It was desired to have a method that could accurately calculate parameters over an entire surface without interpolation. Therefore, the temperature history of the surface was

determined through use of thermochromic liquid crystals (TLC). These crystals are applied over a surface similar to paint and are engineered to display hues of the color spectrum over a certain temperature range, being clear outside of this range. TLCs can be manufactured to associate a hue value with a particular temperature. The crystals were calibrated *in situ* so that hue values could be converted to temperature values using a calibration relationship. This method gives accurate temperature values over the surface of an object [8].

The primary goal of this work is to verify that this method works on simple, well understood geometries so that it can be applied to more complex geometries, such as the cooling channels of a gas turbine blade. A flat plate was chosen as the model geometry because its thermal response is well known.

To verify the technique, the flat plate was painted with TLCs and placed in a heated airstream The hue change of the TLCs was monitored by a camera and converted to temperature history using a temperature/hue calibration. Finally, the conjugate gradient inverse method was applied to the temperature history data to calculate h and T_∞. The computed values of h and T_∞ were compared with generally accepted correlations for h over a flat plate and the measured T_∞ values.

2 BACKGROUND

The goal of the experiment was to find heat transfer coefficients h and free stream temperature T_∞ given the transient surface temperature response of a solid. The test object was a flat plate placed in a small-scale wind tunnel. The plate was subjected to a sudden rise in T_∞, and the surface temperature response T_s was measured by recording the color change (color play) of a coating of thermochromic liquid crystals (TLCs) painted on the surface. A semi-infinite solid analytical model was applied that related the parameters T_∞ and h to T_s, and an inverse method was used to solve for these parameters using the surface temperature data.

2.1 Literature Review

There were several notable research projects conducted in the last several years that lead up to and influence the motivation for this research. A summary of these projects follows.

While TLCs have been in use for decades, they are still a relative newcomer in the variety of ways to determine heat transfer coefficients. Before TLCs became common, heat transfer coefficients were determined indirectly by measuring heat flux. Childs et al. [9] generalized these techniques into four categories.

1. Heat flux measurement through differential temperature measurement involves a sensor comprised of two or more thermocouples (often a single thermopile) placed closely to one another. Knowledge of the thermal conductivity of the material between the thermocouples allows the heat flux

to be found through Fourier's Law. Because the sensors have to be installed inside the surface of the material in question, the material of the sensor needs to either have similar thermal properties of the surface, or the error associated with the thermal discontinuity needs to be addressed.

2. Calorimetric methods use heat balance equations applied to a temperature sensor taking temporal measurements. Within this category, there are many different ways of measuring energy storage. Slug calorimeters are sensors that measure the temperature change of a volume of material to determine the heat gained, and are common. Other methods include null point calorimeters, thin-film sensors, and whole-surface monitoring.

3. Energy supply or removal methods also use energy balance equations, but in conjunction with controlled energy that is either supplied or removed from the system. Methods to control energy transfer include electric heaters, convective passages, or the Peltier effect.

4. Finally, the mass transfer analogy uses measurements of mass transfer instead of heat transfer, and uses correlations to relate the two. An example of this would be the naphthalene sublimation technique in which naphthalene, a solid that sublimates in air at standard pressure, is applied to a surface of interest. The mass distribution of the naphthalene is measured before and after the experiment, and mass loss is correlated with heat flux.

Butler and Baughn [10] observed the long history of transient heat transfer coefficient measurements in their investigation on the differences boundary conditions have on measurements of h. The transient class of heat transfer measurement techniques

fall into category 3 of Childs et al. classification structure. Particularly, Butler and Baughn note the transition from point-sensor based measurements to surface-coating based measurements. Early transient experiments employed methods such as resistance thermometers placed on ceramic substrates immersed in shock tunnels in 1973. Thermal paint use is documented later in the 1970s by T.V. Jones [11]. A decade later, Butler and Baughn note the use of phase-change coatings and melting-point coatings.

Ireland and Jones [8] reviewed the research done with TLC use in heat transfer and shear stress experiments, including the advantages and challenges of using TLC measurements.

Ireland and Jones begin by explaining the physics and chemistry of TLCs and how their structure allows them to reflect changes in temperature with changes in color. Response time was discussed, which was reported to be on the order of 3 ms. Also of note was the discussion of TLC use in transient heat transfer experiments. Finally, Ireland and Jones covered the use of TLCs to solve for multiple heat transfer parameters.

Using inverse methods with TLC generated data, Ferguson [12] outlined a method of solving for the coefficients of the temperature/hue calibration of TLCs simultaneously with heat transfer coefficients. Ferguson discussed the problems associated with using liquid crystals, and how those problems can be accounted for by using the inverse method to solve for calibration coefficients. Froerer [13] and Jochem [14] used these methods to investigate heat transfer processes on a flat plate immersed in an air stream and a cylinder, respectively.

Maurer [15] investigated generic convective cooling schemes of gas turbine engines, recognizing the lack of experimental data at the operating conditions of such an

engine and explored common heat transfer enhancing geometries contained in those systems. Maurer used a test stand outfitted with turbulators coated in TLCs to collect temperature data in both transient and steady-state experiments. Also, computational fluid dynamics simulations were employed to augment interpretation of the experimental data. Analytical methods for correcting for geometry effects in the solutions to Fourier's heat equation were examined. The results of cooling enhancement factors and pressure losses were summarized for geometries such as "V" and "W" ribs, dimples, and hemispheres of various heights and aspect ratios.

Poser [4] investigated TLC data collection methods with a scaled gas turbine cooling channel with the goal of providing heat transfer coefficients that were both accurate and low in uncertainty. Poser discusses TLC application and data acquisition methods, including the advantages and disadvantages of using RGB and HSL color models. Much emphasis was given by Poser to data analysis and signal filtering techniques. RGB signal data was preprocessed using a wavelet filter to reduce noise, then normalized to minimize lighting effects. The data reduction process was largely automated by the use of artificial neural networks.

Anderson and Baughn [16] examined hysteresis effects in thermochromic liquid crystals. These authors found that TLCs exhibited different peak R, G, and B values at the same temperatures dependent on whether the TLCs were exposed to heating or cooling conditions. In addition, it was found that when TLCs were heated above a certain temperature, their R G B values were shifted up on subsequent experiments. Baughn and Anderson recommended calibrating in the same thermal direction as desired experimental trials as well as avoiding temperatures much higher than the range of the TLCs in use.

Research has also been done on gas turbine blade cooling in the past decade without the use of thermochromic liquid crystal thermography. Morris and Chang [17] measured heat transfer coefficients in a rotating geometry using channel wall heat transfer equations accounting for rotational and Coriolis effects. Measurements were done with point temperature measurements at leading and trailing channel edges combined with prescribed generated wall heat fluxes.

Frackowiak et al. [18] used inverse methods to determine heat sink power source values in a simply connected regions mimicking gas turbine blade channels, given temperature and heat flux distributions at the channel walls. A Laplace equation-based model was used, and inverse results were compared with published experimental data for the C3X-type blade, found to be in agreement with each other.

Heidrich et al. [19] used transient data combined with the Levenberg-Marquardt inverse method to determine heat transfer coefficients and rib turbulators positions in gas turbine blade cooling channels. The authors recognize that determination of rib location is difficult after blade casting but necessary as a check for design specifications. To measure surface temperature, the authors used infrared thermography to detect temperatures of the outside surface of a steel rectangular channel. The method developed located rib positions with 3% error, but did not report error values for *h*.

These authors also compared the use of TLCs and infrared cameras at estimating heat transfer coefficients [20]. They constructed a rectangular channel with turbulators; one side of the channel was steel, and its external temperature was measured with an infrared camera. The opposing tunnel wall was Plexiglas, and its internal temperature was measured with TLCs. The authors found infrared estimated *h* values 20% higher than

TLCs due to the effects of longer required data collection time coupled with the effects of lateral conduction.

2.2 Methods of Temperature Measurement

There are currently a variety of different ways to measure surface temperatures, each with their pros and cons. The four most widespread methods are thermocouples, temperature sensitive paint (TSP), infrared cameras, and thermochromic liquid crystals (TLC).

2.2.1 Thermocouples

One of the earliest developed and most used method of detecting experimental temperatures today are thermocouples. These thin wire analog devices use the Seebeck principle to generate a voltage proportional to the temperature at the junction of two dissimilar metals. They are reliable, can withstand high temperatures, and are simple to use. Additionally, with the use of modern day electronic cold junctions, thermocouples are simple to calibrate and use.

Thermocouples come with several drawbacks, however. Each thermocouple is capable of only a single point measurement. To obtain any kind of temperature field, multiple thermocouples are required. They are also an intrusive measurement device. To detect temperature with a thermocouple, the device needs to be both physically touching the surface (and immersed in the air stream) and physically connected to the data collection device. Therefore, thermocouple use potentially alters the experiment they're measuring. Minimizing this intrusiveness can be difficult in experiments that use thermocouples.

A thermocouple's response time is also a drawback. Even though thermocouple metals are heat conductive, they still require time to change temperature to that of their environment. This response time is often small, on the order of several milliseconds, but its presence cannot be ignored in experiments requiring accuracy. The response time can be minimized by choosing the smallest thermocouple possible for the application, minimizing its thermal capacitance.

2.2.2   Infrared Cameras

A more modern way of measuring temperature is with an infrared camera. All surfaces emit thermal radiation in the infrared spectrum proportional to their temperature. An infrared camera can measure this radiation and convert it to images with corresponding temperatures. These cameras offer a non-intrusive, full-field measurement of temperature.

Infrared cameras are highly expensive however, with quality experimental-grade cameras costing tens of thousands of dollars. They also require a direct view of the object they are measuring. Even objects such as quartz windows that are transparent to infrared radiation will interfere with an infrared camera's measurements, since the window itself emits radiation that is read by the camera. These disparate radiations cannot be separated by the camera, and thus skew the data. Also, use of infrared cameras introduces another unknown into the calculation: the surface emissivity of the TLC-coated surface. Therefore, accurately measuring an object inside a wind tunnel would require the lens be immersed in the tunnel. In small-scale experiments, the lens is comparatively large, and results in a highly intrusive measurement device.

2.2.3   Temperature Sensitive Paint

A temperature sensitive paint consists of a binder mixed with a luminescent molecule. It works on the basis of thermal quenching, in which the active molecules in the paint reflect light at different wavelengths as their temperature is increased. This measurement method, like the infrared camera, is both full-field and non-invasive. However, since the paint emits visible light, it is measured with a normal camera, and thus can detect accurate temperatures through clear materials such as glass or transparent plastics.

The major drawback of TSP is that the color change is irreversible. Once the luminescent molecules in the paint release their photons, recapturing them is difficult. Therefore, use of a TSP is only good for a single experimental trial, and would have to be reapplied with each subsequent trial. Furthermore, TSP only functions with temperature increases. To measure the temperatures of a hot surface cooling down, for example, a different method would be required.

2.2.4   Thermochromic Liquid Crystals

Thermochromic liquid crystals are similar to temperature sensitive paint in that they consist of a molecular structure and a binder that are painted onto a surface. However, rather than using a luminescent molecule, the binder is instead combined with micro-encapsulated cholesteric crystalline structures. Cholesteric structures align in planes, forming a layered helix. The structures need to be encapsulated as they are fragile and degrade quickly under stress. This can be an advantage in some cases, and un-encapsulated crystals can be used to measure sheer stress. The pitch of these helical structures changes based on temperature, causing the helix to tighten or unwind. These

pitch variations result in crystals polarized to different wavelengths, reflecting different colors of light. TLCs are engineered to reflect different light wavelengths at different temperatures, resulting in a coating that can measure temperature in a full-field, non-intrusive way but is completely reversible [8].

TLC measurements are not full-range, though. They can be manufactured to change color through temperature ranges that can be as narrow as $1°C$ for narrow band crystals, and up to tens of degrees Celsius for wide band TLCs [8]. This somewhat limits the crystals' versatility in an experimental setting, but allows them to be tailored to experimental conditions. Also, the color play of the crystals, which is the way in which the TLCs change color, is highly sensitive to different factors such as the camera viewing angle and lighting conditions, including incidence angle and spectral characteristics of the incident light. As a result, calibration of the TLCs is difficult. Movement of the test item or camera, or any change in lighting conditions can require a new calibration.

Several other factors can affect liquid crystal measurements as well. The way the surface is prepared, such as back coating, number of layers, and painting technique can all affect the homogeneity of the crystal surface as well as hue intensity. Varying air humidity can change a calibration from one day to the next. Also, TLCs have a hysteresis effect associated with them. When the crystals are heated beyond a certain temperature then cooled, their calibration curve will be different for subsequent experiments unless steps are taken to "reset" them.

TLCs were chosen as the temperature measurement tool for these experiments because they offer full-field temperature sensing, can perform repeated experiments with a single application, and are non-invasive. To account for the different factors that can lead to errors associated with TLCs, select thermocouples were used complementarily as a verification tool.

3 THEORY

For this research, a mathematical model was needed that related a surface's

transient temperature response with its material properties to a given convective

boundary condition. For solids with a low thermal diffusivity and small time scale, a

semi-infinite model is ideal. A closed-form analytical solution was developed for the

transient surface temperature response from basic principles and appropriate boundary

conditions. An inverse method, a mathematical method for determining parameters in

complex models, was used to solve for the heat transfer coefficient h and free stream

temperature T_∞ parameters, from transient surface temperature data.

3.1   Surface Convection Model

For a semi-infinite solid, the governing heat equation reduces to a second order,

homogeneous differential equation.

$$\frac{\partial^2 T}{\partial x^2} = \frac{1}{\alpha} \frac{\partial T}{\partial t} \tag{3.1}$$

With boundary and initial conditions

B.C. $$T(x \to \infty, t) = T_0 \tag{3.2}$$

I.C. $$T(x, t = 0) = T_0 \tag{3.3}$$

To solve this partial differential equation, a similarity solution can be used [21].

By selection of the similarity variable $\eta \equiv \frac{x}{(4\alpha t)^{1/2}}$ and substituting into Equation 3.1, the

PDE collapses into an ordinary differential equation with similarly transformed boundary

and initial conditions.

$$\frac{\partial^2 T}{\partial \eta^2} = \frac{1}{\alpha}\frac{\partial T}{\partial \eta} \tag{3.4}$$

B.C. $$T(\eta \to \infty) = T_0 \tag{3.5}$$

I.C. $$T(\eta \to \infty) = T_0 \tag{3.6}$$

The solution to Equation 3.4 varies depending on what is chosen as the second boundary condition on *x*. Of relevance to this research is surface convection.

$$-k\frac{\partial T}{\partial x}\bigg|_{x=0} = h[T_\infty - T(0,t)] \tag{3.7}$$

The solution to the Equation 3.4 further varies depending on the form T_∞ takes. For example, if T_∞ is constant (step change) the closed-form solution for the surface temperature T_s becomes [21]

$$T_s = T_0 + (T_\infty - T_0)[1 - \exp(\beta^2)\mathrm{erfc}(\beta)] \tag{3.8}$$

$$\beta = \frac{h\sqrt{\alpha t}}{k}$$

However, a step-step change was not the best representative of the conditions the wind tunnel heater was able to generate. The free stream temperature profile in the tunnel was characterized by a steep rise followed by a shallow rise to equilibrium. This profile can more accurately be modeled as a piecewise ramp function with number of ramps *n*.

$$T_\infty = \begin{cases} m_1 \cdot t + b_1, & 0 \leq t \leq t_1 \\ \quad\quad\vdots \\ m_i \cdot t + b_i, & t_i \leq t \leq t_{i+1} \\ \quad\quad\vdots \\ m_n \cdot t + b_n, & t_n \leq t \leq \infty \end{cases} \tag{3.9}$$

To solve the heat equation in which the surface convection boundary condition contains this form of T_∞, it is convenient to first solve the problem with a single ramp, then take advantage of the linearity of the PDE and apply superposition techniques. The

use of Duhamel's Theorem of superposition simplifies both the ramp and series of ramps

solutions in an elegant fashion.

For convenience, let

$$\hat{T} = T - T_0.$$

Duhamel's Theorem gives the temperature response of a system with a time-varying

boundary condition in which a *fundamental solution* to the problem is known [22]. The

fundamental solution is defined as the response of the system with a zero initial condition

to a single, constant non-homogeneous term with magnitude unity. This can be stated as

$$\hat{T}(x,t) = \int_{\tau=0}^{t} T_f(x, t-\tau) \frac{dB(t)}{dt}\Big|_{t=\tau} d\tau + B_{t=0} T_f(x,t) \tag{3.10}$$

where T_f is the fundamental solution, B is the time-varying boundary condition, and τ is

a dummy variable.

The first step to applying Duhamel's Theorem is finding the fundamental

solution. For this problem, the fundamental solution solves

$$\frac{\partial^2 T_f}{\partial x^2} = \frac{1}{\alpha} \frac{\partial T_f}{\partial t} \tag{3.11}$$

With boundary and initial conditions

$$T(x \to \infty, t) = 0 \tag{3.12}$$

$$\frac{\partial T_f}{\partial x}\Big|_{x=0} - \frac{h}{k} T_f = \frac{h}{k} \tag{3.13}$$

This is recognized as the problem posed by Equations 3.1-3.7 with $T_0 = 0$ and $T_\infty = 1$.

With this recognition, the solution to the fundamental problem is already known as

$$T_f(x,t) = \operatorname{erfc}\left(\frac{x}{2\sqrt{\alpha t}}\right) - \exp\left(\frac{hx}{k} + \beta^2\right) \operatorname{erfc}\left(\frac{x}{2\sqrt{\alpha t}} + \beta\right) \tag{3.14}$$

$$\beta = \frac{h\sqrt{\alpha t}}{k}$$

The time-varying term $B(t)$ is the ramp function that defines T_∞

$$B(t) = m \cdot t \tag{3.15}$$

When these are substituted into Duhamel's Theorem, the resulting integral

$$\hat{T}(0,t) = m \int_o^t T_f(\text{x}, t - \tau)\, d\tau \tag{3.16}$$

is easily solvable with analytical mathematical software such as Mathematica and yields

the single-ramp surface response solution.

$$\hat{T}(0,t) = mt \left[1 - \frac{2}{\sqrt{\pi}\beta} - \frac{1 - \exp(\beta^2)\text{erfc}(\beta)}{\beta^2} \right] \tag{3.17}$$

The multi-ramp solution can be achieved by breaking the integral in Duhamel's

Theorem into a series of integrals for each ramp [23].

$$\hat{T}(x,t) = \sum_{\text{i}=1}^{\text{n}} m_\text{i} \int_{\tau=\text{t}_{\text{i}-1}}^{t_i} T_f(x, t - \tau)\, d\tau \tag{3.18}$$

In Equation 3.18, the t_i terms indicate the beginning of each ramp, with $t_0 = 0$, $t_n = t$,

and the m_i terms representing each ramp function's slope, respectively. Each ith element

of the summation is relevant only for times after t_i. For example, even though four ramps

might ultimately be used, for times less than t_1, the surface temperature response still

only includes the first element of the summation, being equal to the single ramp solution.

See Appendix A for a detailed solution.

After three elements of the summation have been calculated, a pattern emerges.

$$\hat{T}(0,t) = m_1 t \cdot g(\beta_0) + \sum_{i=1}^{n-1} [H(t - t_i)m_{i+1}(t - t_i)g(\beta_i) - m_i(t - t_i)g(\beta_i)]$$

where
$$H(\tau) = \begin{cases} 0 & \tau < 0 \\ 1 & \tau \geq 0 \end{cases}$$

$$\beta_i = \frac{h}{k}\sqrt{\alpha(t - t_i)} \tag{3.19}$$

$$g(\beta_i) = 1 - \frac{2}{\sqrt{\pi}\beta_i} + \frac{1 - \exp(\beta_i^2) \cdot \text{erfc}(\beta_i)}{\beta_i^2}$$

This is the surface convection model used with the inverse method. With a step change free stream function, the unknown parameters were T_∞ and h, a two parameter problem. With this series of ramps free stream function, the number of parameters are increased according to the number of ramps that represent T_∞, and the parameters become m_i, t_i, and h. So, for example, a 3 ramp input temperature would have 6 parameters. To simplify the problem, the t_i parameters were chosen before the inverse method was applied based on thermocouple data of the shape of the free stream temperature profile. This would reduce the example 3 ramp problem to 4 parameters (h, m_1, m_2, m_3).

3.2   The Inverse Method

3.2.1   Forward Problem

A forward problem is a problem in which the parameters are known, and the effects are unknown. A parameter is defined as something that "may define a physical entity directly (e.g., density, voltage...), or may be coefficients or other constants in a functional relationship that describes a physical process" [24]. In the equation $\boldsymbol{Gm = d}$, **m** represents the vector of parameters, **d** represents the data (effects), and **G** is the functional matrix that relates the two.

An example problem is a cube of copper on a hotplate. The forward problem would be, given the temperature of the hotplate and the material properties of the copper (parameters, **m**), determine the temperature of a point in the copper with respect to time (effects, **d**). Forward problems can often be solved analytically, and thus the solutions are directly applicable. For this research, the heat transfer to a flat plate in cross flow was

examined. The forward problem was: given the bulk free stream temperature T_∞, the heat transfer coefficient *h,* and material properties, find the surface temperature response of the flat plate.

3.2.2 Inverse Problem

An inverse problem is any problem in which one or more of the parameters, **m**, are unknown, but the effects, **d**, are known. These problems occur often in many fields especially in instances where the dependent variable of the forward problem is measured. Suppose in the above example of a cube of metal on a hotplate, a thermocouple read the temperature of the center of the cube, and the metal was of an unknown material. In this case, the effects are known, and some of the parameters unknown. This problem is much less straightforward than the forward problem, and given the model applied to the system, requires different solution techniques.

3.2.2.2 Conjugate Gradient Method. Inverse methods can be further divided into two categories: *linear* and *non-linear*. Linear inverse methods are based upon linear system models, and their solution can be typically arrived at in one calculation, or one step, while non-linear inverse methods, based upon non-linear system models, typically require an iterative solution method to solve [24]. To understand the linearity and non-linearity of the problem, it is useful to examine the model again:

$$\hat{T}(0,t) = m_1 t \cdot g(\beta_0) + \sum_{i=1}^{n-1} [H(t-t_i)m_{i+1}(t-t_i)g(\beta_i) - m_i(t-t_i)g(\beta_i)]$$

where
$$H(\tau) = \begin{cases} 0 & \tau < 0 \\ 1 & \tau \geq 0 \end{cases}$$

$$\beta_i = \frac{h}{k}\sqrt{\alpha(t-t_i)}$$

$$g(\beta_i) = 1 - \frac{2}{\sqrt{\pi}\beta_i} + \frac{1 - \exp(\beta_i^2) \cdot \text{erfc}(\beta_i)}{\beta_i^2}$$

This system is linear with respect to some parameters, and non-linear with respect to others. In the case of the m_i parameters, it can be seen that they appear in the model simply multiplying other parameters and the data. All m_i parameters are raised to the first power, and none are found within other functions. Therefore, the system is linear with respect to m_i. The β_i parameter (which contains *h*), however, is found in the sub-function *g*. Here, β_i is found in both an exponential and a complementary error function, making the model non-linear with respect to *h*. So, even though there is linearity with respect to some parameters, the non-linearity of *h* requires a nonlinear solution method.

For this research, the conjugate gradient inverse method was chosen to solve for the parameters. The conjugate gradient method is popular for solving inverse heat transfer problems due to its robustness and the fact that no matrix inversion is necessary [25]. In the case of this research, the effects were known: the surface temperature of the flat plate with respect to time $\boldsymbol{T_s}$. A model **G** was assumed with multiple unknown parameters: the slopes, m_i, that make up T_∞, and *h*. For a non-linear problem such as this one, key relationships connecting the data and the parameters are given by the Jacobian matrix, **G**, defined as

$$\boldsymbol{G} = \boldsymbol{J} = \begin{bmatrix} \frac{\partial \mathbf{T_s}}{\partial h} & \frac{\partial \mathbf{T_s}}{\partial m_1} & \cdots & \frac{\partial \mathbf{T_s}}{\partial m_n} \end{bmatrix}. \tag{3.20}$$

Many heat transfer problems have an ill-posed Jacobian sensitivity matrix, that is, a Jacobian matrix whose determinant is zero or close to zero. The degree to which a matrix is ill-posed can be determined by its conditioning number. This can be thought of

as a measure of amplification of the system matrix G. In an ill-posed matrix, small perturbations in the data can lead to large changes in parameter values. Attempts to invert such matrices lead to failure of an algorithm without some form of weighting scheme. Also, since inverting large matrices is computationally demanding, the conjugate gradient method is typically less-computationally expensive per iteration than other algorithms.

The conjugate gradient method is not without its flaws, however. A good starting guess is needed to ensure convergence. Also, the algorithm tends to require more iterations for convergence than other least squares inverse methods. However, given that each iteration is likely to be quicker, this flaw is negated for large iteration count problems.

All non-linear inverse methods have the same basic solution method. First, the problem is linearized using the derivative to make it solvable using linear methods. This yields a matrix system containing the Jacobian, a matrix composed of partial derivatives with respect to the unknown parameters.

$$J \cdot \Delta m = \Delta d \tag{3.21}$$

A starting guess is then chosen for the parameters. An estimated solution vector, calculated using the guessed parameters, is then compared with the data vector. If the error between the two is not within a specified tolerance (error misfit), the algorithm updates the parameters in an attempt to minimize some objective function φ. This process is then iterated until a stopping criterion is reached.

The conjugate gradient method is an extension of the method of steepest descent [25], both of which attempt to minimize the objective function:

$$\varphi = e^T e \tag{3.22}$$

$$\mathbf{e} = (\boldsymbol{T_s} - \boldsymbol{T_{es}})$$

where **e** is the error misfit vector, found by taking the temperature vector data $\boldsymbol{T_s}$ and subtracting the temperature vector found calculated with estimated parameters $\boldsymbol{T_{es}}$. In this way, the objective function is minimized by varying the parameters that are used to calculate $\boldsymbol{T_{es}}$: h, $\boldsymbol{m_1}$,…, $\boldsymbol{m_n}$. For this inverse method and others, the objective function is identical to the sum of squares error (SSE) $\boldsymbol{S^2}$. In the method of steepest descent, the search direction is based directly on the gradient of the objective function at the current iteration step. In conjugate gradient, the search direction is determined from a linear combination of the gradient at the current location and the search direction at the previous iteration step. This is advantageous because the direction of steepest descent is not necessarily the quickest path to the answer.

Figure 3.1 illustrates an example comparison of the solution path of steepest descent (light) versus conjugate gradient (dark).
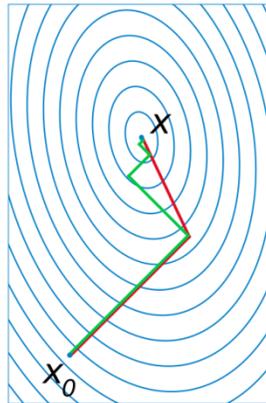
Figure 3.1 Iterative Method Comparison: The conjugate gradient path (darker line) arrives at the solution in two iterations, versus five for steepest descent [26].

In this figure, 'X_0' is the starting guess, and 'X' is the solution. The steepest descent arrives at the solution in five steps, and the conjugate gradient method takes two. Care must be taken, however, when choosing how to weight the current gradient with the

past gradient in the conjugate gradient algorithm. The weight given to each can significantly affect the speed of convergence of the algorithm. In some cases, a weighting scheme could lead to a slower convergence than the steepest descent algorithm.

    3.2.2.3 Conjugate Gradient Algorithm. The algorithm for the conjugate gradient method uses the essential steps found in Table 3.1 [25]:

Table 3.1 CG Algorithm: Steps in the conjugate gradient algorithm.

1	Form an initial guess of the parameters in vector form **P**.	$P = \begin{bmatrix} h \\ m_1 \\ \vdots \\ m_n \end{bmatrix}$	(3.23)
2	Solve the forward problem to obtain an estimated surface temperature vector T_{es}.	$T_s(P) = T_{es}$	(3.24)
3	Compute the sum of squares error. Compare this value to the stopping criteria. If the criteria aren't satisfied, continue.	$SSE = (T_s - T_{es})^T(T_s - T_{es})$	(3.25)
4	Compute the Jacobian sensitivity matrix **J**.	$J = \begin{bmatrix} \dfrac{\partial T_s}{\partial h} & \dfrac{\partial T_s}{\partial m_1} & \cdots & \dfrac{\partial T_s}{\partial m_n} \end{bmatrix}$	(3.26)
5	Compute the gradient vector.	$\nabla S = -2J^T[T_s - T_{es}]$	(3.27)
6	Compute the conjugation coefficient γ^k. There are several methods available. For simplicity, the Fletcher-Reeves formulation was used. *n* is the number of parameters, and $\gamma = 0$ for *k*=0.	$\gamma^k = \dfrac{\sum_{i=1}^{n}[\nabla S^k]_i^2}{\sum_{i=1}^{n}[\nabla S^k]_i^2}$	(3.28)
7	Compute the search direction d^k. Note that for *k*=0, the search direction is simply equal to the gradient.	$d^k = \nabla S + \gamma^k d^{k-1}$	(3.29)
8	Compute the search step size.	$\delta = \dfrac{[Jd]^T[T_s - T_{es}]}{[Jd]^T[Jd]}$	(3.30)
9	Compute the new estimated parameters vector **P**.	$P^{k+1} = P^k - \delta^k \cdot d^k$	(3.31)
10	Increase *k* by 1 and iterate back to Step 2.	$k = k + 1$	(3.32)

This algorithm was implemented in MATLAB, and can be found in Appendix B.

3.2.2.4 Challenges of the Inverse Method. The inverse method returns a set of parameters that minimizes the objective function. While the objective function may have a clear absolute minimum, it may also have several local minima. Depending on the starting parameters chosen, an inverse algorithm may converge at one of these local minima. In effect, the algorithm becomes "stuck," because the gradient at a local minimum is zero, and the update algorithm bases its step size on the magnitude of the gradient.

There are several ways to prevent this. A starting guess of parameters close to their true values will ensure convergence, or routines can be added to an algorithm to update the parameters based on different criteria if convergence isn't achieved after a certain number of iterations. Extra work can be avoided, though, if the shape of the objective function is examined. In doing this, one can avoid many pitfalls associated with inverse techniques, while gaining insight into the problem.

Figure 3.2 shows an objective function for the case of a step function fluid temperature rise, parameters similar to those used in other experimental trials, and generated surface temperature input data without noise. The inverse problem in this case is a simpler version of the series of ramps fluid temperature input version used to analyze actual data. In this case, with a single step T_∞, the inverse method reduces to finding two parameters, h and T_∞.

Objective function contour plot min at h=95 Tinf=333.15
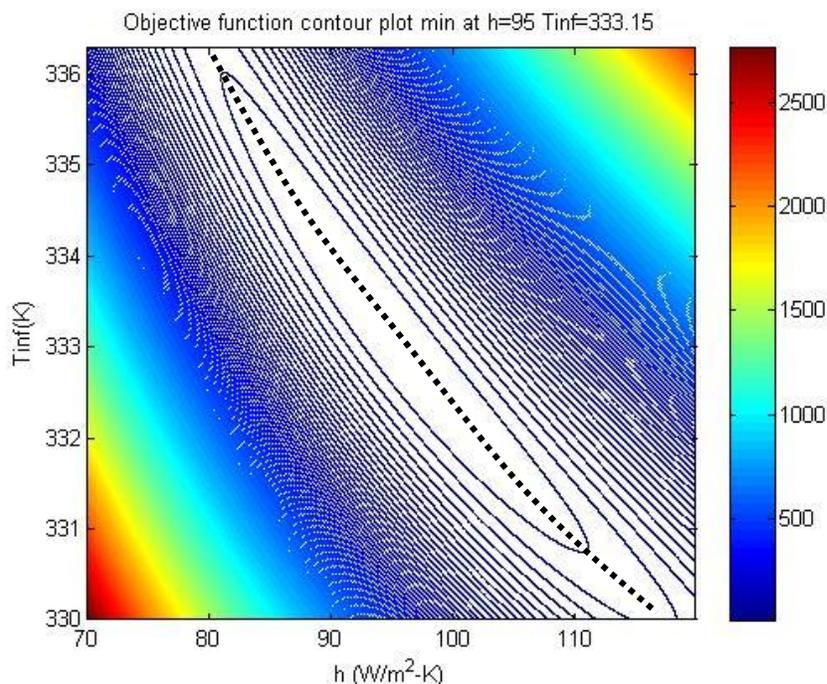
Figure 3.2 Contour Plot: An example objective function to visualize potential problems in the inverse method. The dotted line indicates a path along the minimum of the trough.

The scales of this plot were chosen to make the existence of the minimum clearer visually. In the plot, the trench the objective function forms appears diagonal. This trench indicated the problem is more sensitive to one parameter than the other. However, because of the different parameter scales, which parameter the objective function is more sensitive to is unclear visually. Examining the gradient near the minimum shows the sensitivity clearer. When the conjugate gradient algorithm for these two parameters is allowed to converge, the gradient one iteration before the point of convergence is -18.7 in the T_∞ direction and -0.29 in the *h* direction. This is a result of the ill-posed nature of the problem, indicative of a poorly conditioned Jacobian matrix. The much larger slope in the T_∞ direction reveals that small perturbations to T_∞ can produce large changes in the objective function; T_∞ is sensitive, and therefore easily found. But, large perturbations in *h* produce relatively small changes in the objective function. This parameter is not

sensitive, and has a higher degree of uncertainty associated with it. The existence of a minimum for *h* is made more apparent if the trench in the above figure is plotted, as seen in Figure 3.3.
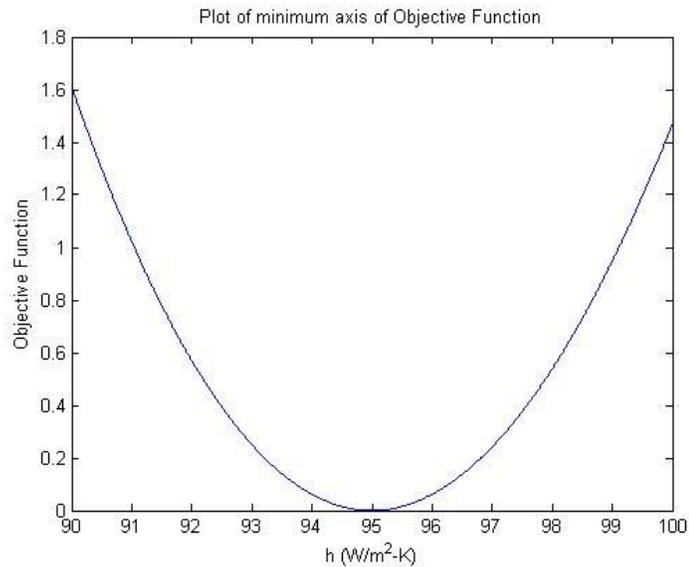
Figure 3.3 Contour Plot Slice: A plot of the minimum h values from Figure 3.2. The *h* axis is taken along the minimum of the trough, indicated by a dotted line in the above figure.

Uncertainty in the measured data also affects the accuracy and convergence of the algorithm. As large changes in *h* can produce small changes of residuals, the inverse is also true: small uncertainties in the data can result in large uncertainties in *h*, though T_∞ is less sensitive. It has been shown that realistic changes in uncertainty have little effect on the ability of the algorithm to converge, or required iterations for convergence [12].

3.3   Solution Appraisal

There are many statistical tools available to appraise how well a solution converges and how accurate the solution is. The solution evaluation benchmarks used in the conjugate gradient algorithm were the sum of squares error (SSE), gradient, correlation, chi-squared goodness of fit, and parameter confidence intervals. Each method

alone does not always accurately predict convergence. However, when used together, they offer the experimenter a powerful way to assimilate and evaluate data.

3.3.1 SSE

The average sum of squares error is one of the easiest benchmarks to calculate, and as such is one of the first to be considered when evaluating convergence. The normal SSE was already calculated in Step 3 of the conjugate gradient algorithm in Table 3.1.

$$SSE = (\boldsymbol{T_s} - \boldsymbol{T_{es}})^T (\boldsymbol{T_s} - \boldsymbol{T_{es}})$$

The average SSE is found from:

$$\overline{SSE} = \sqrt{\frac{(T_s - T_{es})^T (T_s - T_{es})}{n}} \tag{3.33}$$

This gives the average error misfit between any data point and the fitted curve. This value can be plotted vs. iteration to appraise solution convergence and to compare with the data standard deviation in the chi-squared goodness of fit benchmark.

3.3.2 Chi-Squared

The chi-squared goodness of fit compares the error misfit of the fitted curve to the variance of the data:

$$\chi^2 = \frac{SSE}{(n - r - 2)\sigma^2} \tag{3.34}$$

In this equation, r is the number of degrees of freedom, and σ^2 is the variance of the data.

In essence, the chi-squared benchmark compares how well the algorithm fitted the data to how well it can reasonably be expected to fit the data. A chi-squared of 1 is a "perfect" fit, meaning the model fit the data as well as the data accuracy predicts it would. A value greater than 1 indicates the model could do better. A value less than 1 indicates the model fit the data too well; the fit line lies inside the error bars of the data.

Examining this value can give information on the model used. A value too far away from one can indicate the model needs to be reexamined.

3.3.3 Gradient

A solution is reached when a global minimization of the objective function is achieved. At the minimum, the gradient of the function is zero. It was calculated in Step 5 of the conjugate gradient algorithm in Table 3.1.

$$\nabla S = -2J^T[T_s - T_{es}] \tag{3.35}$$

This value can be plotted versus iteration to examine convergence. Knowing how close the gradient of a solution is to zero is a good measure of convergence. Since a gradient exists for each parameter, the norm of the gradient vector ∇S was used.

There are problems with using the gradient alone to evaluate convergence, however. It was found that a solution would often arrive at one or more local minima before arriving at the global minimum. Therefore, an algorithm using the gradient as its stopping criteria would most likely stop the program too soon.

3.3.4 Correlation

The third appraisal technique used was the correlation between the error vector and time at each iteration step. It should be noted that, in the calculation of correlation, the error vector e and not SSE was used. The elements in e can be both positive and negative.

$$e = [T_s - T_{es}] \tag{3.36}$$

The correlation between two variables, in this case e and time t, can be thought of a measure of how much dependence there exists between the two variables. It can also be

thought of as a scaled covariance, in which zero is totally uncorrelated, and 1 is complete

correlation [24]. A correlation between *e* and *t* is calculated by:

$$CORR = \frac{Cov(e,t)}{\sqrt{Var(e)Var(t)}} = \frac{\sum(e-\bar{e})\sum(t-\bar{t})}{n\sqrt{Var(e)Var(t)}} \tag{3.37}$$

So, for example, if the correlation of *e* and *t* is 0, e can be thought of as random with

respect to *t*. Figure 3 shows examples of data sets with different correlations:
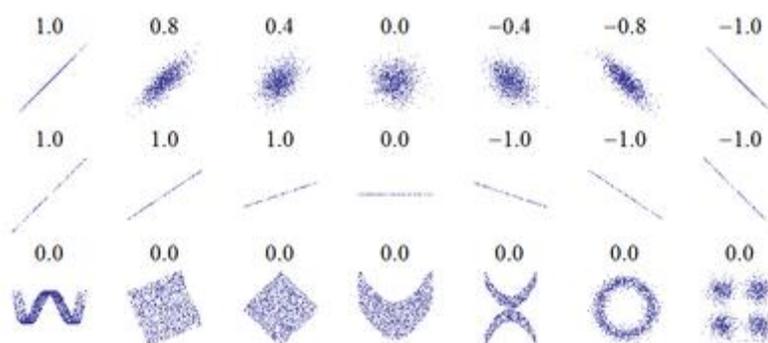
Figure 3.4 Correlation: These are visualizations of what values of correlation different data patterns would have [27].

Correlation is important because it gives a measure of the degree to which

solution error is due to noise in the data, and how much is due to a poor choice of model.

However, lack of correlation does not indicate a flawless model. Referring to the

previous example, the correlation of zero would have its error completely dependent

upon data noise, and the correlation of 1 would have its error completely dependent upon

a poor model.

3.3.5 Parameter Confidence Intervals

In this study, three inter-related evaluations of solution accuracy were used:

standard deviation of the parameters and 95% and 99% confidence intervals. The

parameter covariance matrix C_m is obtained from the data covariance matrix C_d. Assuming the data standard deviations are the same for each data point, the parameter covariance matrix is given by [25]:

$$\boldsymbol{C}_{m=}\sigma_d\boldsymbol{C}_d = \sigma_d\sqrt{[\boldsymbol{J^TJ}]^{-1}} \tag{3.38}$$

where the parameter standard deviations, σ_m, are given by the diagonal values of C_m.

The confidence interval for a parameter P is given by:

$$\bar{P} - a \cdot \sigma_m \leq P \leq \bar{P} + a \cdot \sigma_m \tag{3.39}$$

Where *a*=1.96 for 95% confidence, and *a*=2.576 for 99% confidence. These solution accuracy statistics are only valid, however, when information is known about the data variances. Unless the data variances are known or can be reasonably estimated, parameter accuracy cannot be determined. Data variances can be approximated by the uncertainty of the measurement technique.

3.4   Light and Color

The nature of light and color remains one of the most interesting and most studied subject in physics. This history can be traced back to the earliest experiments to discover the nature of light. While Sir Isaac Newton was not the first person to split light into its component colors with a prism, he was the first to conclude that light is the summation of all the colors of the spectrum. Newton also believed in the corpuscular view of light, that is, light as a particle that travels from the object to the eye. This theory would be disputed by Christian Huygens, who favored a wave theory of light. The theory would be supported later by Young's famous double slit experiment. With the advent of quantum theory, it is now thought that light behaves as both a particle and a wave [28].

3.4.1   The Visible Spectrum

Newton knew that light was made up of color. Light is indeed a combination of different wavelengths of the electromagnetic spectrum, each corresponding to a color. Figure 3.5 shows the range of wavelengths and their corresponding colors.
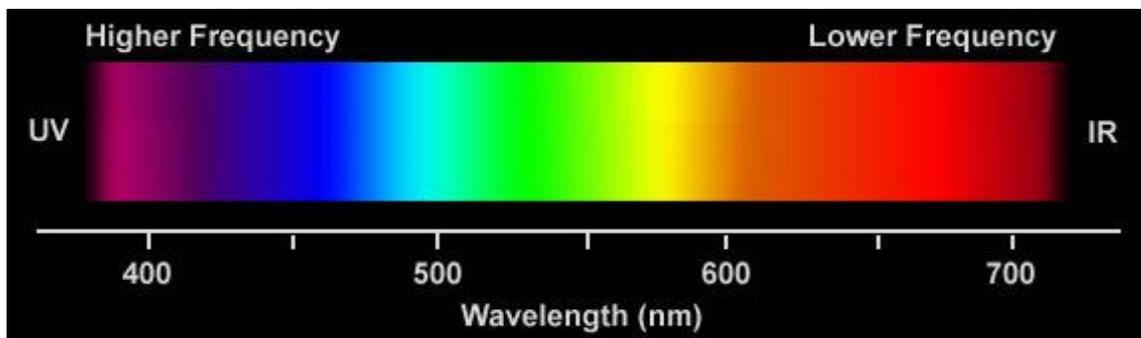
Figure 3.5 Color Spectrum: Light is made up of a combination of electromagnetic wavelengths [29].

An object's color primarily depends on what wavelengths the object absorbs, and which it reflects. Secondary effects of color include the intensity of light, angle of perception, and how diffuse the light reflections are. For example, an apple is red because the skin of the apple tends to reflect higher frequencies of color and absorb lower. The sky is blue because blue wavelengths tend to pass through the air unmolested, while other wavelengths become scattered.

3.4.2   Perception

Visible light makes up only a small range of the electromagnetic spectrum but constitutes all of the wavelengths that the human eye can perceive. Light is focused by the eye lens to the back of the retina, which is lined with photoreceptors [30]. Two types of photoreceptors exist: rods and cones. Figure 3.6 shows the location of eye photoreceptors.
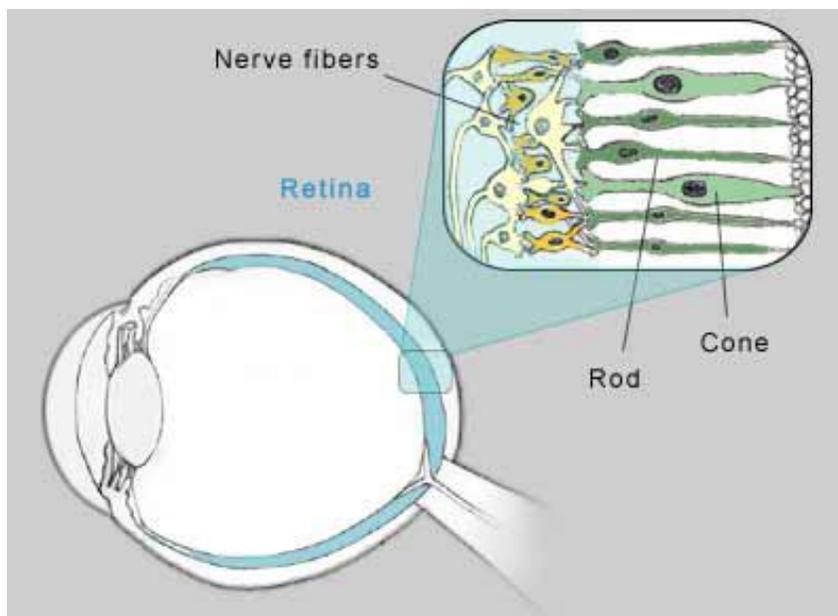
Figure 3.6 Rods and Cones: Photoreceptors at the back of the retina consist of rods and cones [31].

Of the two types, rods are used primarily to see in low-light conditions, and cones are used for color and daylight.

It is difficult to tell whether people perceive color in the same way from individual to individual. A "red" to one person could be different than a "red" perceived by another. There are also deficiencies in some individuals in the way their biology handles light and color, like color-blindness. It is known, however, that there are three kinds of cone cells in the eye: S, L, and M. These labels represent short, long, and middle-wavelength cone cells, referring to the wavelength of light they are sensitive to. This means the human eye is *trichromatic*, and synthesizes all colors out of the red, green, and blue colors the cones perceive. The eye has a particularly large number of M cones, which correspond to the green section of the spectrum.

3.4.3 Color Representation

Physical color is a complex mixture of different wavelengths and intensities of light. To store color as data and/or recreate color digitally, a color model (color space) is

needed. A color model is a mathematical way of representing color. There exists a wide array of color models, each with advantages that make it useful for a particular application or industry [32]. RGB (red, green, blue) is used by computer and television screens to represent color, since this most closely corresponds with the way the human eye perceives color. HSL/HSV (hue, saturation, lightness/value) is used in image processing tasks. CMY (cyan, magenta, yellow) is used by the print industry. For this project, the RGB and HSV color models were used.

3.4.3.1   RGB Color Model

The RGB color model is an additive color model, and recreates color by using different combinations of red, green, and blue [33]. The system can be visualized as a cube on a three-dimensional coordinate axis, with black at the origin and red, green, and blue on each of the axes, as seen in Figure 3.7. In the RGB system, values can range between 0 and 255.
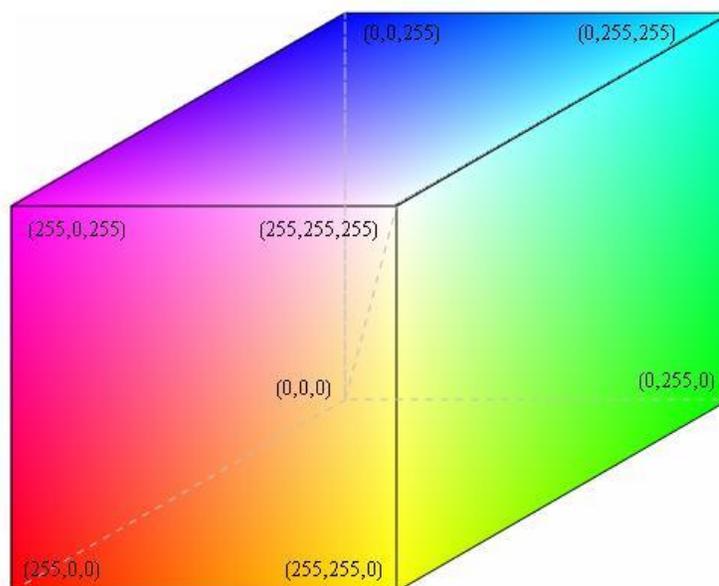
Figure 3.7 RGB Color Cube: The RGB color system can be represented by a cube, with red, green, and blue colors on the primary axis vertices, and cyan, magenta, and yellow, white, and black on the other vertices [34].

The RGB system is useful because it most closely mimics the response of the human eye, and is the most common color model used. In digital cameras, such as the one used in this research, it is used as the native method of recording color. Its disadvantage lies in the fact that all three values are needed to represent a color, making it a somewhat non-intuitive color system. It is difficult to look at a color and intuit how much red, green, and blue are in it. A more intuitive model is HSV.

3.4.3.2 HSV Color Model

HSV stands for hue, saturation and value, also called HSB (hue, saturation, brightness). Another closely related system is HSL (hue, saturation, lightness). Both are cylindrical-coordinate systems in contrast to the cubic system of RGB. The HSV color space was used in this research because of built-in conversion algorithms in the MATLAB computational software used.

HSV can be represented by a cone, as shown in Figure 3.8.
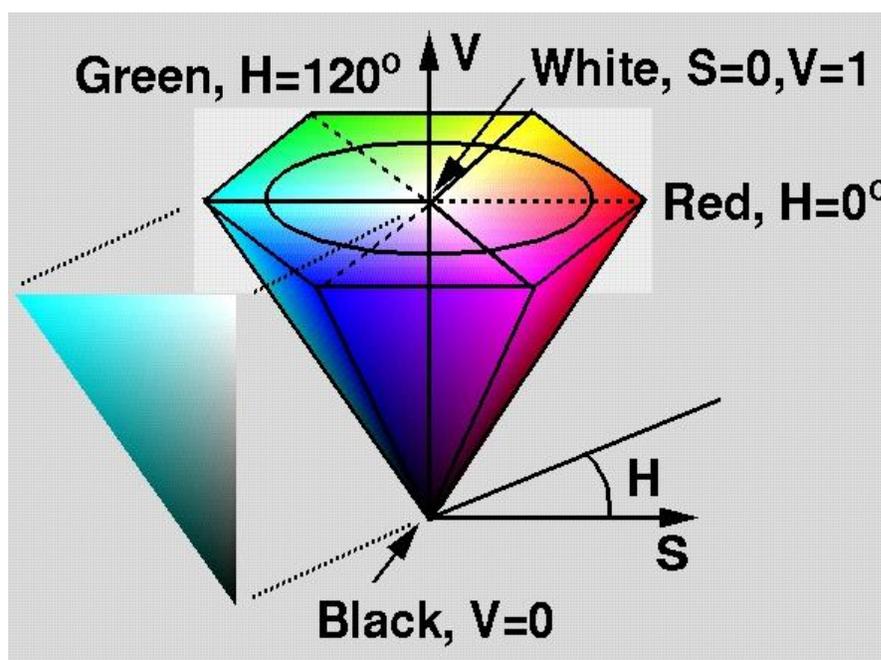
Figure 3.8 HSV Cone: The HSV system can be represented by a cone, with hue being the angular coordinate, saturation being the radial coordinate, and value being the height coordinate [34].

In the above figure, hue is represented by the angular coordinate, saturation by the radial coordinate, and value as the height coordinate. Physically, hue represents the primary wavelength of light contained in the signal. Saturation represents the amount of white contained in the color, and can be thought of as the color purity. Value represents the amplitude intensity of the light.

HSV is a more intuitive color system than RGB since hue represents the color of the object. The hue's range from 0 to 1 (scaled from the 0 to 360 degree angle) represents all the colors of the visible spectrum. HSV is useful from an image processing standpoint because the color signal can be represented by the hue value alone, versus needing each of the R, G, and B signals. For this project, a single color signal is much easier to calibrate to a temperature than three. However, since cameras record in the RGB color space, the image must be first converted to HSV before it is analyzed.

3.4.3.3 RGB to HSV Conversion

There are multiple ways to convert RGB to HSV. MATLAB uses a method that tilts the RGB cube on a vertex and projects it onto a hexagon. Figure 3.9 shows this projection.
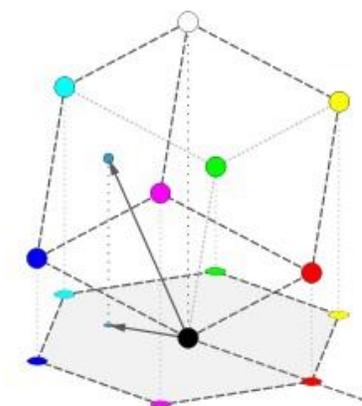
Figure 3.9 Projection: The RGB cube is tilted and projected onto a plane to calculate hue [35].

To calculate hue in this way, let M be the maximum value of R, G, and B, and m be the minimum value [35].

$$M = \text{Max}(R, G, B) \tag{3.40}$$

$$m = \text{Min}(R, G, B) \tag{3.41}$$

$$C = M - m \tag{3.42}$$

A hue value is then calculated from the RGB signal.

$$H' = \begin{cases} \text{undefined} & \text{if } C = 0 \\ \frac{G-B}{C} \bmod 6 & \text{if M = R} \\ \frac{B-R}{C} + 2 & \text{if M = G} \\ \frac{R-G}{C} + 4 & \text{if M = B} \end{cases} \tag{3.43}$$

Finally, the hue value is normalized.

$$H = \frac{H'}{6} \tag{3.44}$$

It should be noted that hue becomes undefined at values of C close to 0, which correspond to values where each of red, green, and blue are either close to 0 or close to maximum. This singularity exists because a hue value at $0°$ and a hue value at $360°$ correspond to the same color.

The value component of HSV is simply defined at the max value of R, G, or B.

$$V = M \tag{3.45}$$

Saturation is defined as the ratio of C to M.

$$S = \frac{C}{M} \tag{3.46}$$

4   EXPERIMENTAL METHODS

To examine the capability of the inverse method to find heat transfer coefficients and fluid temperature from experimental data as outlined in Section 1.2, the simple, well understood geometry of a flat plate was chosen as the test object. The plate was placed in a small-scale wind tunnel capable of heating the air flow. Data was collected with a camera/TLC and thermocouple system. The methodology of experimentation is detailed in the following sections.

4.1   Experimental Apparatus

Figure 4.1 shows the elements of the experimental apparatus, which includes a small-scale wind tunnel with blower and heating element, test plate, camera, and lights.
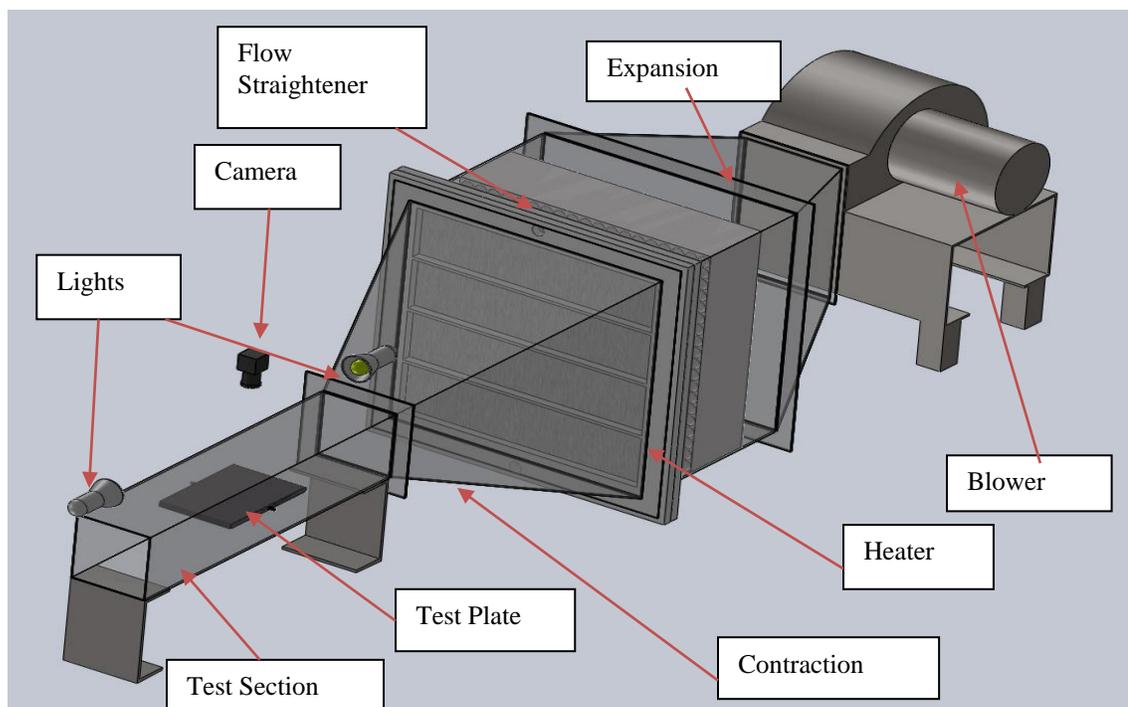
Figure 4.1 Experimental Apparatus: The experimental apparatus, including blower, heater, wind tunnel, test plate, camera, and lights.

Room temperature air started by being accelerated by the blower. The air decelerated through an expansion and then passed through a flow straightener to produce a uniform velocity profile. The air then flowed through a wire mesh heater in which its temperature rose. Then, the flow accelerated through a contraction in the wind tunnel before entering the test section. Finally, the air exited to ambient temperature and pressure at the end of the test section.

Elements of the apparatus:

- Blower: The air in the wind tunnel was accelerated by a 1.5 hp Dayton blower. A transformer received 230 V AC input power and supplied 115 V DC power to the blower. A Toshiba VF-57 controller varied the rotation frequency of the blower to achieve desired wind velocity. The wind tunnel was capable of speeds as low as 2.35 $\frac{m}{s}$ at 10% of the controller speed to 30 $\frac{m}{s}$ at 60% of the controller speed. Velocities above or below these values were not determined or desired.

- Expansion and contraction: The structure of the wind tunnel proper was built in English measurements and was comprised of 1/3" acrylic plastic fastened with a combination of metal and plastic bolts. The expansion section expanded the flow from the blower to a 21" square cross section. The flow then entered a bank of straight 0.25" diameter tubes to remove flow irregularities. After passing through the heater, flow then contracted to a 6" square cross section at the beginning of the test section.

- Heater: The heater element of the tunnel was placed between the tube bank and contraction portions. A fine stainless steel wire mesh was held between plates of polyethylene plastic, with additional ceramic fiber insulation to prevent melting.

Resistance heating was achieved by applying a voltage across ends of the wire mesh supplied by a 280 V DC max power supply. This heating setup was able to generate air temperatures up to 30° C above room temperature at a minimal air flow rate of $15\frac{m}{s}$.

- Test section: The test section was 30" long and 6" square internal cross section. Flow traveled 10" before reaching the front of the test plate. Small holes were drilled into the test section to allow mounting of the plate as well as access for thermocouples and a pitot tube. These holes were subsequently sealed with clear tape when not in use.

- Test plate: The ½" (1.27 cm) thick test plate was mounted in the center of the test section with wooden dowels. The dowels allowed for rotation of the plate to desired angles of attack.

- Camera: The camera used was a Hitachi VCC-151 CCD camera capable of collecting data at 30 frames a second at 752 x 480 pixel resolution in a native RGB color scheme. The camera was mounted 7.5" above the test section on a mounting rack (not shown). To prevent glare from overhead lights reflecting off of the acrylic, the space between the camera and tunnel wall used for viewing was shielded with black construction paper.

- Lights: Halogen lights were mounted at select spots around the tunnel to provide lighting in addition to the normal fluorescent lighting provided by the room.

4.2 Experimental Procedure

In this research, a polycarbonate flat plate painted with TLCs was mounted in the wind tunnel parallel to the flow. The leading edge of the plate was a semi-circle to reduce the separation "bubble" associated with a blunt leading edge, as shown in Figure 4.2.

An experimental trial consisted of the following steps:

1. The wind tunnel flow was activated with all test section elements being at the same initial temperature.

2. The video capture program XCAP was started. A camera positioned directly above the plate recorded the color change of the TLC-coated surface at 15 frames per second.

3. Simultaneously, thermocouple data collection was started on another computer running LabVIEW, also at 15 frames per second.

4. The heater was then turned on, subjecting the flow to a near step change in temperature to T_∞.

5. When the image buffer of the XCAP program was full, both data collection systems were stopped.

Thermocouple data and image data was synced using an LED positioned in the camera window. The LED's activation coincided with the activation of the heater. The first frame in which the LED was lit was assumed to coincide with the start of the temperature rise of the free stream thermocouple. This was used to synchronize data from the thermocouple and image data sets. This data, along with temperature data from both free stream thermocouples and surface-mounted thermocouples, was used for analysis.
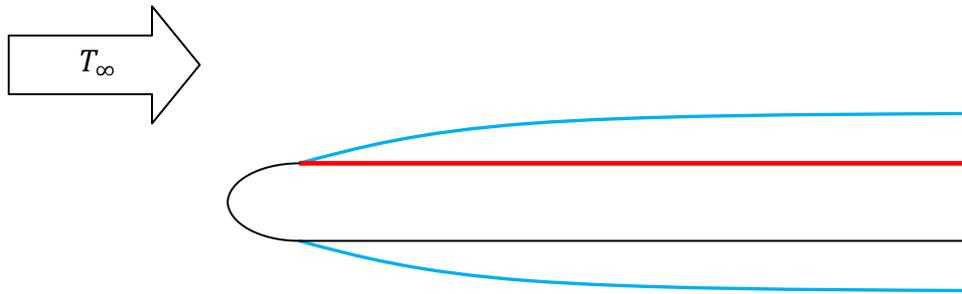
Figure 4.2 Flat Plate Boundary Layer: Side-view of the flat plate. The front was rounded to reduce negative flow effects, such as boundary layer separation. The red line indicates the coating of TLCs.

4.3   TLC Data Collection

4.3.1   Thermochromic Liquid Crystals

Thermochromic liquid crystals (TLCs) are crystalline organic molecules that are mixed with a binder to allow painting on surfaces. Their unique properties allow them to be engineered to change color over a certain temperature range. With proper calibration, they become a valuable experimental tool, being capable of giving full-field temperature measurements.

4.3.1.1   Principle of Operation

TLC's unique temperature sensing properties come from their phase transitions at certain temperatures and how these phases interact with light. The three basic states of matter are solid, liquid, and gas, which all substances can experience. Liquid crystals,

however, can experience additional phases between the solid and liquid phases. These are

called their mesophases [36]. These intermediate phases are anisotropic, meaning

direction-dependent, and contribute to their unique light properties as the directional

dependence of the molecules change. Figure 4.3 shows the phase transition steps of

TLCs.
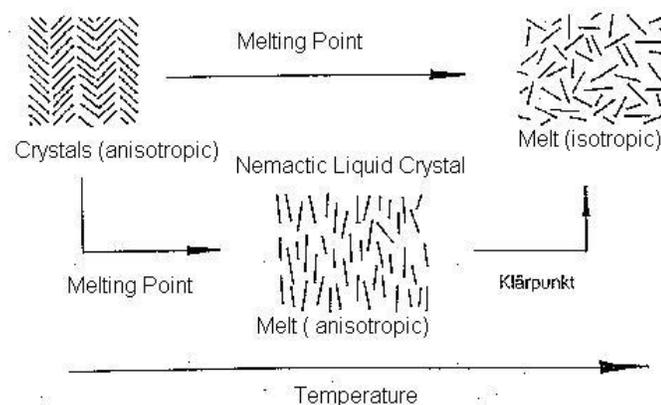
Figure 4.3 TLC Phase Transitions: TLCs have mesophases between their solid and liquid phases [37].
There are two main types of liquid crystal phases: nematic (with cholesteric being a special sub-type) and smectic. In the nematic phase, there is little positional order and the crystals are free to move in relation to each other. There is orientational order, however, as the crystals are roughly aligned as seen in

Figure 4.4. In the smectic phase, there is both orientational order and positional

order along one dimension.
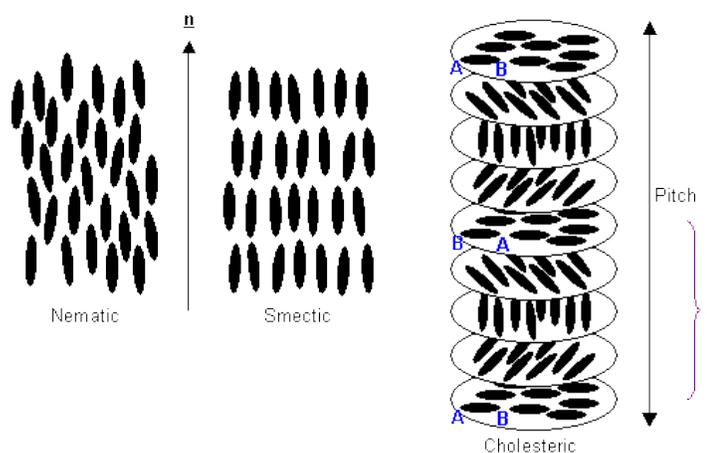
Figure 4.4 TLC Phases: TLCs have two main phases: nematic (of which cholesteric is a special type), and smectic [38].

A *chiral* structure is one that cannot be superimposed on its mirror image, and denotes a lack of symmetry. The particular chiral structure of relevance to liquid crystals is the helix. The chiral nematic, or cholesteric phase of liquid crystals is characterized by a helical arrangement of the crystal molecules with each plate itself having a nematic structure, as shown in Figure 4.5. This occurs when long-range intermolecular forces cause the planes to align [8].
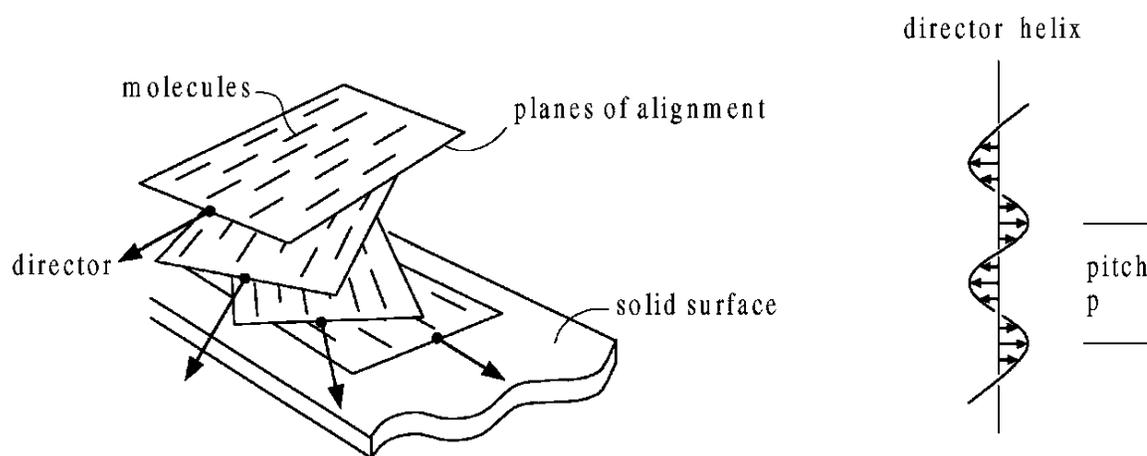
Figure 4.5 Cholesteric Pitch: In the cholesteric phase, TLCs align in twisted planes creating a helix [8].

The spectral properties of the cholesteric structure become relevant to temperature sensing when the pitch of the helix *p* is on the order of the wavelength of visible light. The wavelength of light reflected by the TLCs is given by

$$\lambda = n \cdot p \cdot \sin(\theta), \tag{4.1}$$

where *n* is the index of refraction of the material, and θ is the angle of reflection. The crystals start out aligned in planes, clear to visible light. They then begin to twist as they change from nematic to cholesteric. The pitch is large at first, and the crystals reflect the largest wavelengths of visible light (red). As the temperature rises and the helix tightens,

the crystals transition through the spectrum from red to green to blue. Finally, the crystals transition to an isotropic liquid, and become clear once again.

4.3.1.2   Experimental TLCs

The experimental TLCs used in this study were made by LCR Hallcrest. Hallcrest manufactures custom TLC mixtures in which the temperature range for color play is specified. Their TLCs are labeled as "RXXCXW" in which the numbers after the "R" designate the temperature at which red becomes visible, or the start temperature, in degrees Celsius. The number after the "C" indicates the temperature range. The TLCs used for this study were micro-encapsulated R25C15W, indicating their color change started at 25 °C and stopped at 40 °C. These are considered wide-band liquid crystals since they change color over a wide temperature range. Any crystals whose range was close to 1 °C would be considered narrow-band. Accuracy of the crystals is reliant on response time, which has been found to be on the order of 3 ms [8].

There are also many forms of TLCs that can be acquired. Hallcrest offers pre-painted TLCs that come in sheets, as well as paint designed for screen printing. The TLCs used in this research were micro-encapsulated SPN100, designed for spray-painting with an airbrush. The "100" denotes a water-soluble paint for easy removal and reapplication.

4.3.1.3   Thermocouple Mounting

To mount the thin-foil thermocouples, the surface of the flat plate was first washed of oil and residue. The desired positioning was marked in dry-erase marker. One side of each thermocouple was sprayed with 3M sprayable adhesive that was not water-soluble. Using tweezers, the thermocouples were carefully placed on the plate, with leads

overhanging. The leads were then tucked under the side-mount screws to ensure a tight fit with the thermocouple extension wire. Finally, the lead contact areas were cleaned of excess adhesive with acetone.

4.3.1.4   Painting Method

The top surface of the flat plate, with thin-foil thermocouples already mounted, was first washed with soap and water to remove any dirt and oil. Because the spray adhesive used to mount the thermocouples was not water soluble, this did not affect their mount integrity. However, care had to be taken to prevent tearing of the fragile thermocouples. After washing, care was taken to avoid skin contact with the top surface to avoid oil transfer. The plate was allowed to air-dry.

A Paasche Model H airbrush was used to coat the test plate with black background paint and TLCs. The various features of the chosen airbrush affected the painting methodology in various ways. The Model H is a single-action airbrush, meaning the spray button controls only the airflow (on or off), while paint flow is adjusted by twisting the nozzle. The single-action paintbrush was ideal for this application since a steady, repeatable flow of paint was desired. The airbrush was also a siphon-feed model, meaning it relied on the siphon action from the flow of air to pull paint into the stream, rather than gravity. A gravity feed allows for lower air pressures to be used and finer lines to be achieved with the air brush. However, since large, thin spray areas were desired for painting, a siphon feed type was adequate. Finally, the airbrush was an external-mix model, meaning the paint mixed with the air outside of the airbrush. This sometimes led to paint clogging the nozzle of the airbrush, so care had to be taken to keep the nozzle clean.

A paint booth was set up under a fume hood to provide ventilation. The plate was set on a stand to make it elbow-level, to aid in accurate painting. First, the plate was prepped with a black undercoating. Black was used as an undercoating to absorb incident light because it was desirable to only have the color of the TLCs reflected off of the surface, and not that of incident light or other light reflections. A matte black finish absorbs the most light and reflections, allowing the TLC color signal seen by the camera to be as pure as possible. A water-soluble black sprayable paint that was designed to be used as a TLC undercoat was used. The black undercoating was made by Hallcrest, catalog number BB-g1.

The painting method was designed to give as even a coat as possible. To this end, it was found that many thin coats provided the best surface finish. Even though the black paint was designed to be applied by airbrush, it was found that thinning with water was still needed. Water was added to the black paint in a roughly 1:3 ratio. The paint was added to the airbrush reservoir, and the air compressor was set to 25 psi (103 MPa). The nozzle was adjusted, intermittently spraying on a white surface, until the desired spray pattern was reached. A nozzle that was too open resulted in large blotches and splatters; a nozzle that was too closed resulted in too thin of a spray field or nozzle clogging. The nozzle was adjusted until a thin, but visible, even spray field was achieved at a distance of 16 cm between surface and nozzle.

A slight splatter occasionally occurred when the spray button is first depressed or released. Therefore, the spray button was always depressed and released when the nozzle was not pointed at the plate. The paint was applied in a moderate, back and forth motion, starting at the top of the plate and moving toward the bottom. Only a few seconds drying

time were required between coats. The black undercoat was thick enough when it was no

longer transparent and any minute surface imperfections were smoothed by paint, giving

the surface an opaque, visually flaw-free finish. Care was taken to avoid applying more

coats than were necessary, however, as too thick a coat could affect the thermal

properties of the surface. About twenty coats were sufficient to achieve a good surface

finish.

The black undercoat was allowed to dry until it was matte to the eye, but still

slightly tacky to the touch. The same painting method was applied to the thermochromic

liquid crystals. Catalog number SPN100R25C15W crystals were used. Again, they

needed to be thinned by water for application at a ratio of 2 parts water to 5 parts crystals.

The ideal thickness of a TLC coating is the one that shows the most vivid colors. It was

found that too thin of a coat introduced extra noise into the signal due to the variability of

thickness as well as poor hue signal in general. Samples of polycarbonate were painted

with thicknesses of 10, 25, and 40 coatings of TLCs. These were placed *in situ* under

experimental conditions, and the 40 coating sample yielded the highest hue signal to

noise ratio, although only marginally more vivid than the 25 coated sample. Using this

information, 40 coatings were applied to the test plate.

4.3.2   Video Capture

The video capture system used consisted of a computer with X Cap video capture

software and a CCD camera linked by coaxial cable.

The computer used was a Dell Dimension 9150 with 2 gigabytes of RAM, a 2.8

GHz dual-core processor, and a PIXCI SV 5 image capture card. The camera used in

these experiments was a Toshiba digital color CCD camera, capable of video capture at 30 frames per second in the RGB color space at a resolution of 752 by 480 pixels.

4.3.2.1   Camera

A CCD, or charge-coupled device, is a mechanism for transferring movement of electrons into an analog or digital signal [39]. In a camera, a grid of photosensitive sensors and capacitors are stimulated by light focused through a lens. The CCD transforms the electrons gathered by the capacitors into a unique digital signal for each capacitor proportional to the light detected by the photosensitive sensors. Each grid element forms a pixel in the unified image. The camera used for this experiment contained a 752 by 480 grid of sensors collecting light data.

For a color camera, a Bayer filter is placed over the grid of light-detecting sensors. A Bayer filter is a grid of red, green, and blue color filters that correspond to the pixel array on the camera [40]. Each element of the Bayer filter forms a repeating pattern of red, green, and blue that allows the camera to form a color picture from a synthesis the RGB color signals (see Section 3.4: Light and Color). Figure 4.6 shows the typical patterns of a Bayer filter. Because three colors are represented and the pixel grid occurs in sets of four, green is repeated twice, since the human eye is more sensitive to green than the other colors.
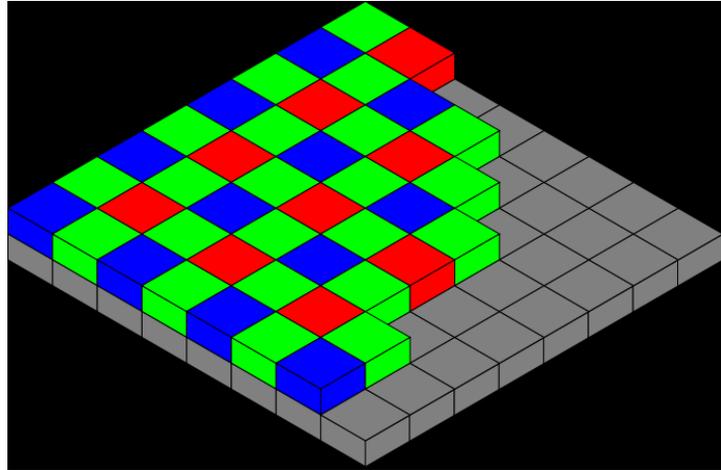
Figure 4.6 Bayer Filter: A typical Bayer filter pattern. Green is repeated twice per set of four pixels due to the human eye's particular sensitivity to this color [40].

During experimental trials, it was found that unexpected irregularities occurred in the blue signal of the TLC data. Figure 4.7 shows an example RGB signal from an experimental trial. It can be seen that while the red and green signals are relatively smooth, the blue signal has a "bump" around 180 frames. Also, the blue signal has noticeably more noise than the other two. This "bump" was known to be an artifact, since the blue signal peak value occurs last for TLC color play, but the "bump" occurred before the green signal reached its peak value. A flaw in the manufacturing of the TLCs was not the cause, as the bump happened regardless of TLC type or lighting conditions. Also, this bump occurred before an HSV color model was applied. It was therefore assumed that the irregularity of the signal was a peculiarity of the camera, more specifically the camera's Bayer filter. This irregularity carried over to the conversion to hue values appearing as a "bump." However, this flaw was able to be mitigated with proper care of the temperature-hue calibration.
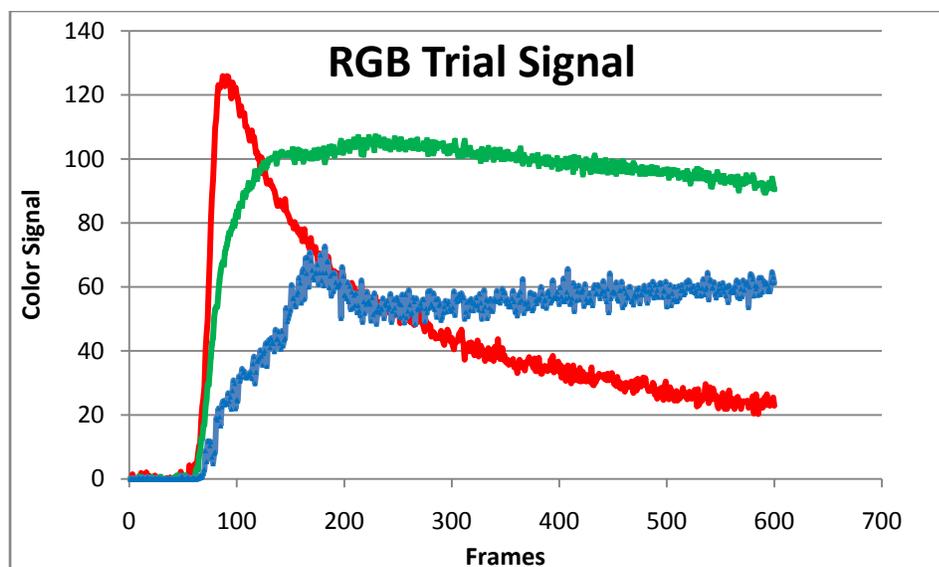
Figure 4.7 Blue Signal Bump: Example of Bayer filter flaw in trial data. An irregular "bump" occurred in the blue signal data around 180 frames.

4.3.2.2   XCap Software

XCap Limited is image processing software published by Epix Inc. [41]. Designed to interface with the PIXCI SV 5 video capture graphics card used in the computer, XCap uses a Java-based GUI and "C" based image processing algorithms. While XCap is capable of a wide array of image processing features, the only ones used in this research were the features relevant to collecting and storing series of RGB images from a camera.

XCap was capable of collecting images at 30 frames per second. To achieve this speed, images were stored into an image buffer during data collection, which exists in RAM memory. After data collection was complete, the images could then be saved to long-term memory. The maximum image series of 983 images corresponded to about 1 GB of data at native resolution and 65.5 seconds. This meant that the 2 GB of RAM in the computer was sufficient to store all data images, taking into account RAM used by normal computer processes.

Images were saved in a Tagged Image File Format (TIFF) image file format [42]. This format has advantages over others, such as JPEG or GIFF. TIFF is capable of storing images in a lossless format, meaning exact original data can be reconstructed from compressed data. This was desirable because compression losses could have potentially led to RGB data that was not accurate. Using loss-free image formats allow the experimenter to bypass this potential pitfall. Also, unlike other image formats that use loss-free compression, TIFF files may be edited and resaved with no loss of image quality. This was not necessary, however, as original images were never altered in this research.

4.3.3 <u>Lighting</u>

The goals in creating the lighting setup for these experimental trials were to:

1. create lighting strong enough to provide vivid hues to the camera

2. create lighting weak enough to keep any of the R, G, and B colors from reaching their max value of 255, indicating oversaturation of the CCD

3. reduce/eliminate glare caused by the acrylic walls of the tunnel

4. create uniform brightness values along the length of the plate

The first source of lighting available for these experiments was the laboratory fluorescent overhead lighting. However, these created a strong glare off of the top side of the wind tunnel, which conflicted with item 3 above. To solve this problem, a black box was created to shield the camera and top portion of the wind tunnel from the overhead lights (see Figure 4.8). While this eliminated glare, it also reduced the intensity of light able to reach the plate, as now only the sides of the wind tunnel were available to illuminate the plate.

To increase the RGB signal levels (item 1 above), additional lighting was required. Two FS GU10-C 35 watt Phillips indoor floodlights were used, secured to the top of the wind tunnel with flex railing to allow mobility. These were Edison-style lights, with a medium base and 2.5" diameter. The lights were positioned above the wind tunnel, 12" from the nearest edge of the plate and 7" above the top plane of the plate, centered laterally, shown in Figure 4.8.
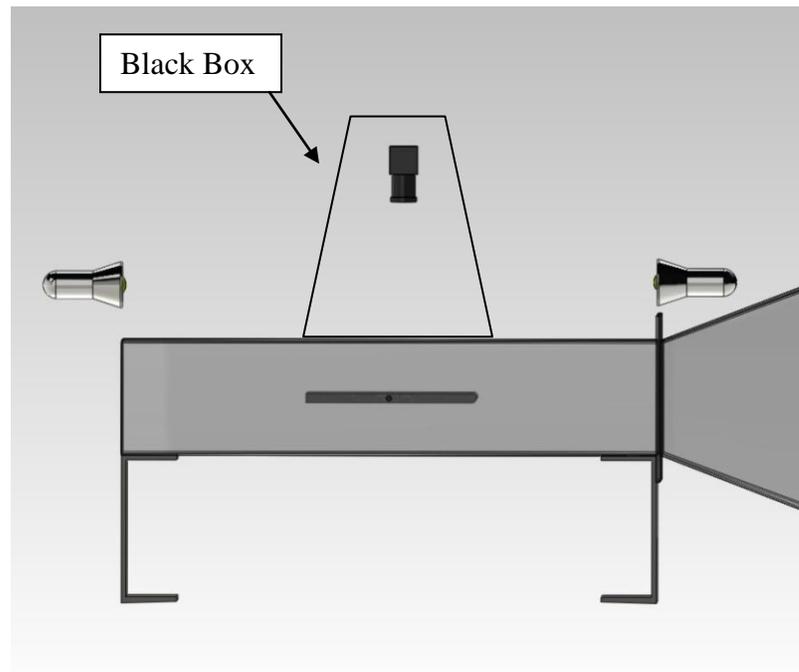
Figure 4.8 Lighting Side View: Side view of the wind tunnel, demonstrating black box and lighting locations.

To have uniform illumination as specified in item 4, the angle of the lights needed to be adjusted. The XCap software could track HSB values as a function of plate position, and this was used to evaluate brightness uniformity. In this way, it was found that small changes in angle could have large changes in brightness. This made it difficult to achieve good uniformity. Too steep of an angle, and the brightness at the edges of the plate was too high. Too shallow of an angle and the lights did not provide enough light to the

middle of the plate. It was found that an angle near zero degrees with the horizontal was ideal, though still not providing perfect uniformity. Figure 4.9 shows an X Cap plot of the final lighting solution, taken with a black plate.
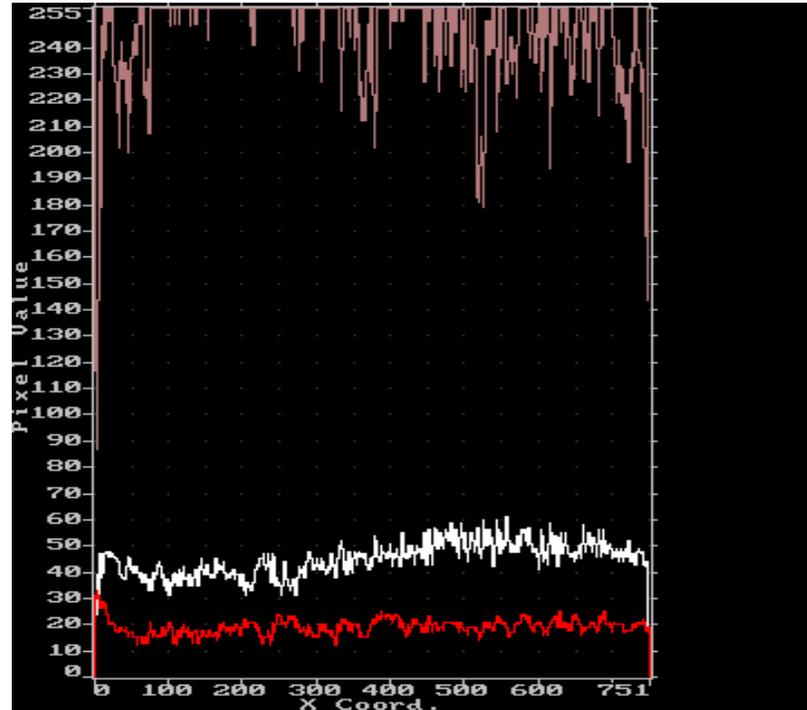
Figure 4.9 Light Level Reading: HSB plot of lighting before experimental trial. Red, pink, and white lines correspond to H, S, and B, respectively. The plate was black for this plot.

In this plot, the x coordinate refers to the stream-wise location of the plate. The red, pink, and white lines are hue, saturation, and brightness, respectively. The high saturation indicates a vivid image. It can be seen that the white brightness line is close to uniform, with values being slightly higher at the back end of the plate. This was not enough to affect the uniformity of the hue, however. Additionally, since each position on the test plate was calibrated individually, this deviation in brightness was negligible.

To satisfy item 2, the camera aperture was adjusted to restrict the amount of light allowed into the camera. It was found that the red signal was most prone to reaching max, so this was the signal monitored. Deciding aperture adjustment was done on a trial-and-

error basis, as the model of camera used did not have aperture adjustment marks. The aperture was initially left fully open, and an experimental trial was done. Subsequent trials were then performed, progressively tightening the aperture until the red signal was just short of reaching maximum value.

Another factor that needed to be taken into account in the lighting setup was the temperature of the light incident upon the black plate. *Color temperature* refers to the bias light shows to particular colors of the spectrum [43]. This value is measured in Kelvins (K). Figure 4.10 shows a chromaticity diagram showing how values of color temperature translate to colors.
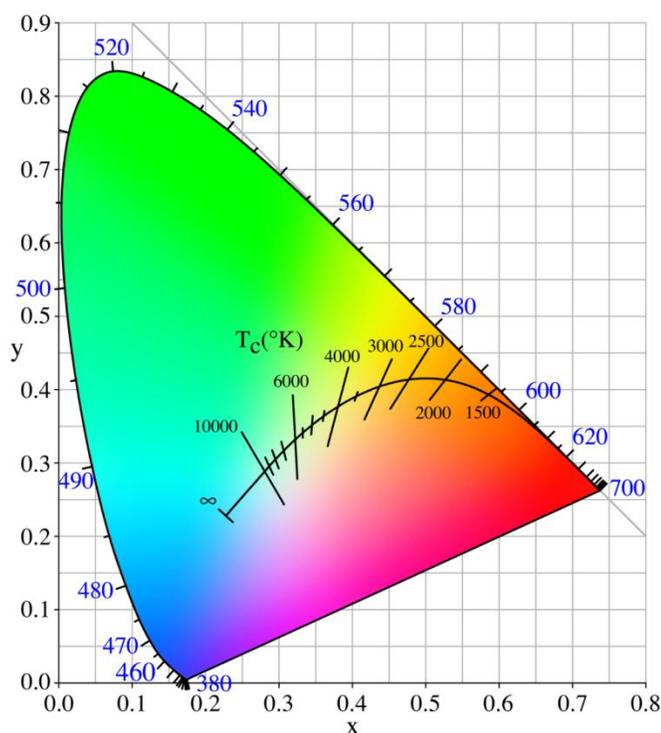
Figure 4.10 Chromaticity Diagram: Color temperatures are red at low values around 800 K, and blue at higher values around 8000 K [44].

It can be seen that low values around 800 K correspond to warm light (light that favors red), and high values, around 8000 K, correspond to cool light (light that favors

blue). Each light bulb or light source has a color temperature. For example, incandescent bulbs tend to emit warmer light, while fluorescent bulbs are cooler.

For an ideal black plate, the RGB color signal read by the camera would have zero values for each color value. Although the plate is nominally black, it is actually grey, reflecting low level RGB values. These low level values add to the hue values produced by the TLCs, skewing the resulting hue signal. To account for this, the color noise was measured from the RGB signal *in situ* and subtracted before conversion to HSB. Figure 4.11 shows a plot of the background color noise at the 1" location of the plate, taken while the TLCs were in their 'clear' region.
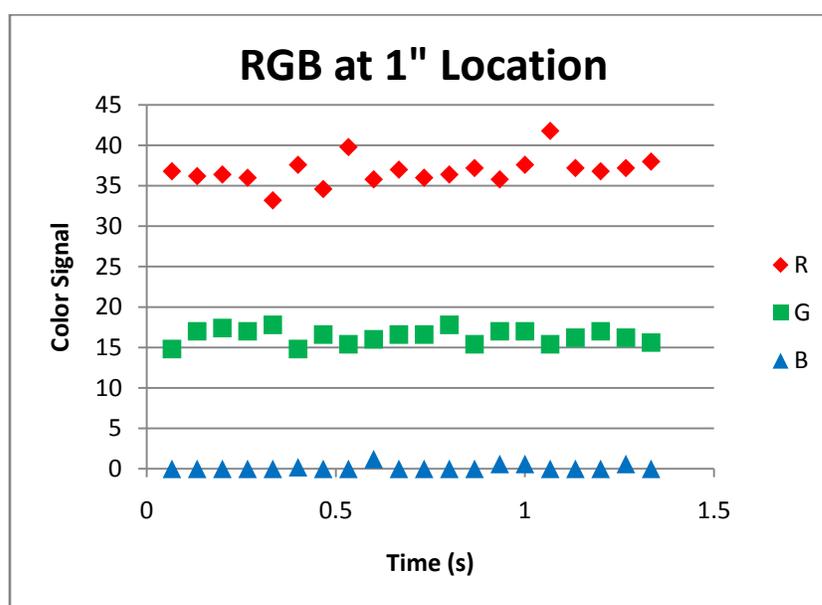
Figure 4.11 Color Noise: Light incident upon the plate caused a bias in data originating from the light's color temperature.

It can be seen that the background color noise was warm in nature, with red having the highest value, and almost no blue. This was somewhat surprising, since the primary light sources used in the experiment, halogen and fluorescent, tend toward cooler color temperatures. Additional variables that could contribute to the values in Figure 4.11

include color bias in the black plate undercoating, filtering effects of the acrylic test section, and light reflections from colored surfaces. Full details of the noise-subtracting process can be found in Section 4.5.7: Background Gray Subtraction.

4.4   Thermocouple Temperature Sensing

In the 1800s, Thomas Johann Seebeck discovered that by joining two dissimilar metals together and applying a temperature gradient across them, a magnetic field was created. This has subsequently been dubbed the Seebeck Effect. While Seebeck initially thought of its applications to magnetic fields, Leopoldo Nobili would eventually recognize the creation of a voltage through the Seebeck Effect and invent the thermocouple [45]. Today, thermocouples are used as an accurate and reliable means of measuring temperature. In this project, thermocouples were used as a means of calibrating the TLCs, checking TLC temperature data gathered during experimental trials, and confirming parameter values of T_∞.

4.4.1   Types

In this study, K-type thermocouples manufactured by Omega were used exclusively. K-type thermocouples are made from chromel and alumel for each respective lead. K-type thermocouples are one of the standard base-metal thermocouple types, the others being T, J, E, and N. Base-metal thermocouples are desirable for their low cost. However, they are inferior to rare-metal thermocouples such as B, R, and S types, due to the fact that they undergo oxidation easily. This makes it harder to remove impurities in the construction of the thermocouple, and lowers the accuracy. For class 2 thermocouples, the tolerances for uncalibrated rare-metal thermocouples are about $\pm$ 1 C,

versus $\pm$ 1.5 C for base-metal thermocouples [46]. Since the increase in accuracy was not enough to justify the tenfold increase of cost, base-metal thermocouples were used.

4.4.2   Principle of Operation

When a temperature gradient exists in a wire, dT, the electrons at the hot end of the wire move toward the colder end, and a differential Seebeck voltage dE_s is produced in proportion to the gradient:

$$dE_s = s(T)dT. \tag{4.2}$$

The proportionality constant $s(T)$ is called the Seebeck coefficient of the conductor, and is a property of the material [46].

To measure this voltage, another wire must be attached to the first. This, however, creates another Seebeck voltage. If the two wires have the same Seebeck coefficient, the two Seebeck voltages will be equal and opposite, creating a net zero voltage. When two dissimilar metals are used, such as in Figure 4.12, the net voltage is not zero, and is on the order of tens of millivolts.
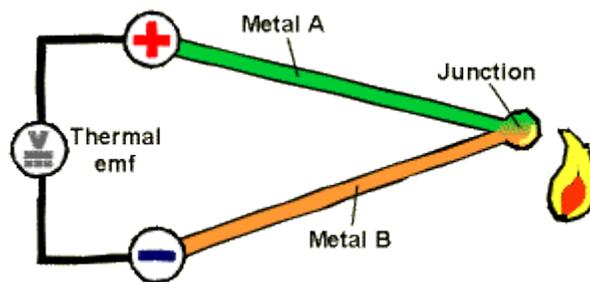
Figure 4.12 Seebeck Effect: Two dissimilar metals create a thermal emf [47].

This voltage can be measured and, when used with proper calibration and reference temperatures, can accurately and quickly measure temperature.

4.4.3   Reference Junction

Since thermocouples are a differential measurement device, a reference temperature must be known to determine the temperature of interest. Until recently, the traditional method of creating a reference was to add a second and third thermocouple to create a reference junction, then immerse the reference junction in a bath of melting ice water held at its freezing point, as seen in Figure 4.13 [45].
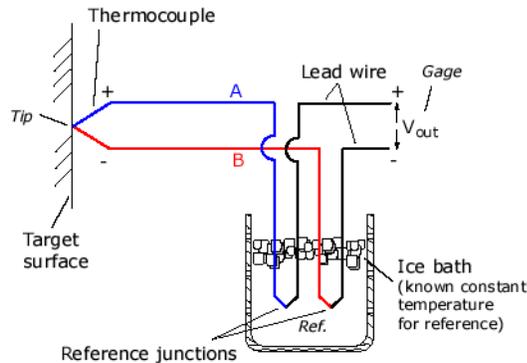
Figure 4.13: Creation of a thermocouple cold junction through use of an ice bath [48].

The advantages of this method were simplicity in temperature calculation, since $E(0\ C) = 0$ [46]. This method is cumbersome, however, since an ice bath has to be created for every experiment, and needs to be frequently refreshed and stirred.

Recently, advances in microelectronics have allowed for digital electronic cold-junction compesators (CJC). These circuits use a thermistor to measure the temperature of the reference junction and cancel the voltage at that point, electronically lowering the reference temperature to 0 °C. Such circuits contain both an amplifier and linearizer in one chip [45], greatly simplifying calculations and turning ice bath cold-junction methods into a specialized technique.

A disadvantage of the electronic cold junction is the sample rate time constant. A typical CJC's response rate limits sampling rate, reducing data collection to the order of a few hertz. A data acquisition board with higher response times are available but for a

higher monetary cost. For this research, a TCIC Series thermocouple interface card was used, supplying an electronic cold-junction, ADC conversion, and temperature conversion (see Section 4.4.6). The maximum sampling rate of the TCIC was 400 samples per second.

4.4.4 Thermocouple Layout

Experimental trials made use of four thermocouples. One 0.01" diameter insulated K-type Omega thermocouple was placed halfway between the top of the test plate and the wind tunnel test section wall to measure free stream temperature (Figure 4.15). Three 0.0005" thick K-type Omega surface-mount thermocouples were placed on the surface of the test plate, at positions shown in Figure 4.14.
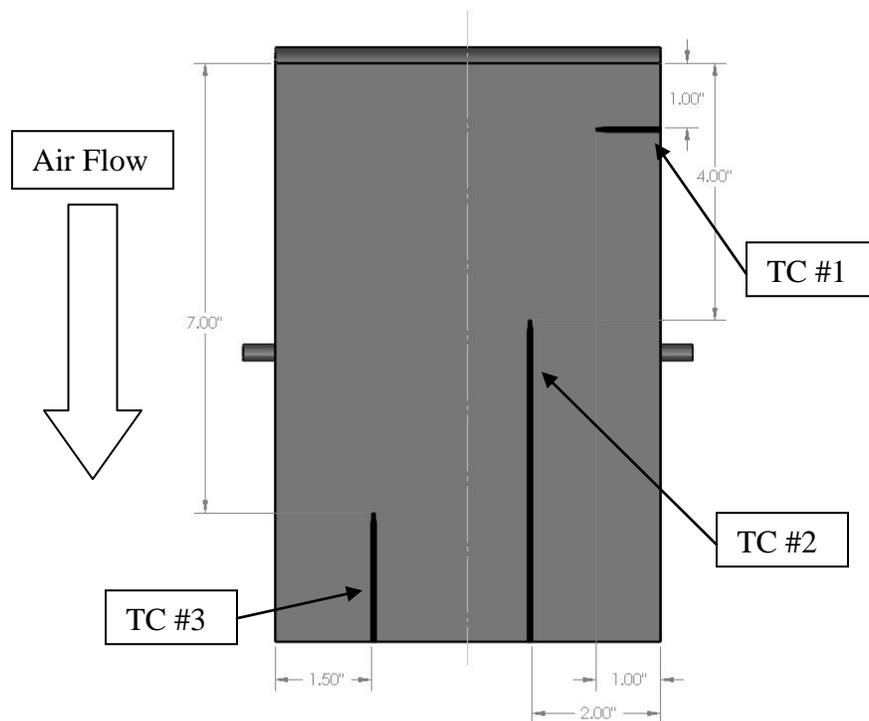
Figure 4.14 Thermocouple Layout: Positions of three surface-mount thermocouples.
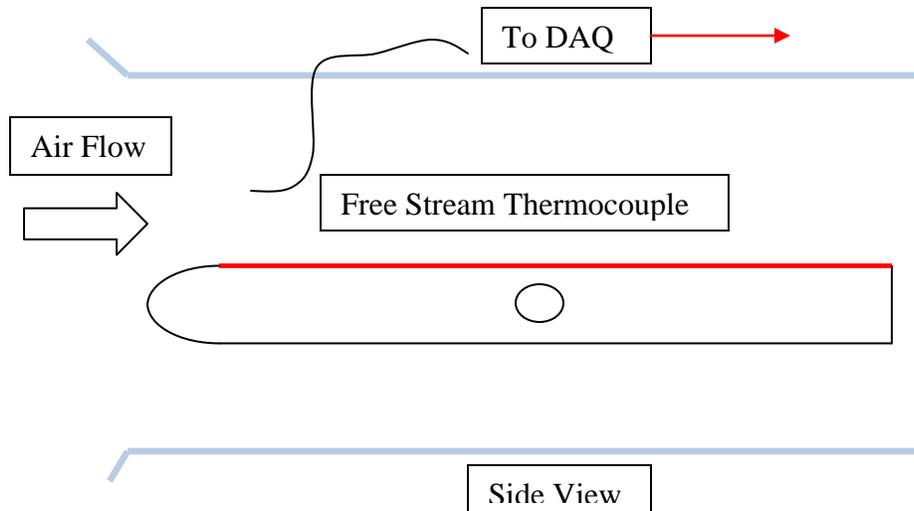
Figure 4.15 Free Stream T.C. Position: The free stream thermocouple was positioned near the leading edge, halfway between the test plate surface and the top of the test section.

These thermocouples were mounted with a spray adhesive directly to the polycarbonate surface, placed under all black and TLC paint. They terminated on the edges of the plate, where they were attached by K-type screws to insulated K-type thermocouple extension wire. This wire was run along the edges of the plate to where they exited the wind tunnel through small holes to the data acquisition unit.

The thermocouples were placed at stream-wise positions of 1", 4", and 7" from the plate leading edge, indicated in Figure 4.14. These positions allowed for multiple checks to the hue/temperature data obtained with the TLCs and allowed for interpolation between positions. The lateral positions of the thermocouples were chosen for multiple reasons. First, the thermocouple needed to be placed outside of the wall boundary layer effects. Section 4.5.2 demonstrated that the boundary layer grew no more than 0.8 cm (0.3 in). Therefore, the thermocouples were placed at sufficient distances from the test section wall. Next, it was desired that the middle stream-wise positions of the plate be

clear, as that corresponded with the most desirable TLC locations. So, they were placed at a distance of at least 1" from the centerline. The positions of the thermocouples were also constrained by the 6.5" length of the lead of the surface mount design as well as the need to have them terminate on the edges. Given these considerations, the thermocouples were placed as far apart as possible so that any small flow effects created by their minimal thickness would not influence the other two. The resulting lateral positions shown in Figure 4.14 were deemed most efficient to accomplish all of the positioning goals.

4.4.5 Calibration

It is always necessary to calibrate a thermocouple to verify that it is returning an accurate temperature. Problems such as wire resistance errors, interference errors, and reference junction errors can all cause a thermocouple to read incorrectly without a proper calibration.

In situ calibration is preferred because it ensures immersion conditions are the same as those used in calibration [46]. For this reason, *in situ* calibrations were used for all thermocouples in this research. The thermocouples were calibrated in two steps: the free- stream thermocouple was calibrated first. Next, the three surface-mount thermocouples were calibrated to the free stream thermocouple.

The free stream thermocouple was calibrated under experimental trial conditions without being detached from the data acquisition unit, only removed from the test section of the wind tunnel. Also, electrical conditions were kept the same as well: the blower, heater, and other electronics were running under experimental conditions during calibration.

The thermocouple was wrapped around an alcohol glass thermometer, and atmospheric conditions were noted (Table 4.1) with a digital barometer.

Table 4.1 Atmospheric Conditions: Conditions at the time of calibration.

Atmospheric conditions	
Humidity	23%
Room temp	23.5 $^{\circ}$C
Bar. Pressure	919 mb

Two calibration points were chosen: the boiling and freezing points of pure water. To achieve these two conditions, distilled water was poured into an insulated glass beaker and placed on a hot plate with stirrer. The thermocouple-wrapped alcohol thermometer was inserted through the insulation such that the termination points were in the center of the beaker, two inches from the bottom. With the stirrer and heater engaged, the water was allowed to heat up to boiling temperature. With the above atmospheric conditions, the boiling point of pure water was calculated to be 97.34 °C. However, the maximum temperature the alcohol thermometer reached was 95.2 °C. It was believed that this was due to the thermometer being too far away from the bottom of the beaker where actual boiling point temperatures were reached. In practice, the water closest to the hot plate was at boiling point, and a temperature gradient formed from that point to the surface. With this in mind, the temperature of the alcohol thermometer was reasonable, and treated as the true temperature. The values for this calibration point can be seen in the first column of Table 4.2.

The second calibration point was chosen to be the freezing point of water. Unlike with the boiling point, pure water's freezing point does not change significantly with elevation so the true freezing point was taken to be 0 °C. To achieve this, the stirrer of the

hot plate was engaged but the heating element left off. Crushed ice cubes made with

distilled water were added to the water in the beaker to create an ice bath. Melting point

equilibrium was achieved, with additional ice cubes added periodically. Again, with the

thermometer placed in the center of the beaker, 0 °C was not reached due to the

temperature gradient, but the thermometer was taken as true. The lowest temperature the

thermometer reached was 0.5 °C, seen in the second column of Table 4.2.

Table 4.2 Calibration Values: Alcohol thermometer and thermocouple calibration point temperatures.

Boiling Temperatures (°C)		Freezing Temperatures (°C)	
Thermometer	95.2	Thermometer	0.5
Thermocouple	94.34	Thermocouple	0.3

With these points, a linear calibration curve was generated for the free stream

thermocouple, seen in Figure 4.16.
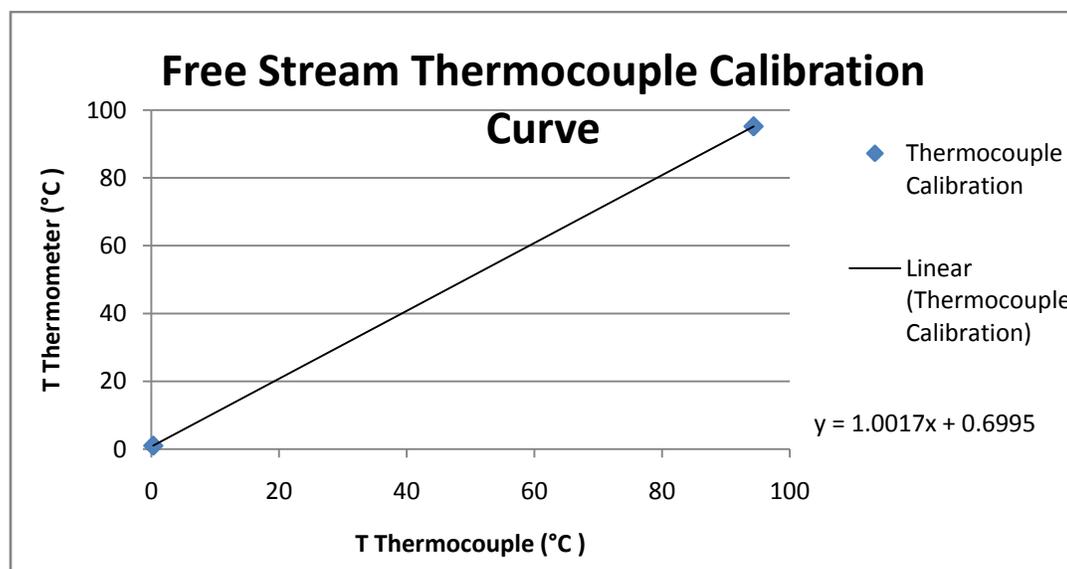
Figure 4.16 Calibration Graph: Calibration curve generated for the free stream thermocouple against an alcohol thermometer taken as "true."

To calibrate the three surface-mount thermocouples, the free stream thermocouple

was replaced in the wind tunnel test section. Using the heater and fan, the four

thermocouples were allowed to reach equilibrium at multiple temperatures and linear calibration curves were generated using the calibrated free stream thermocouple. Table 4.3 summarizes the calibration curves calculated from this procedure.

Table 4.3 Calibration Curves: Calibration curves for the surface mount thermocouples.

Thermocouple	Calibration Equation (°C)
1"	$y = 1.0198x + .1808$
4"	$y = 1.0567x + 1.1597$
7"	$y = 1.1403x + .9156$

4.4.6 <u>TCIC DAQ Board</u>

The Thermocouple Interface Card (TCIC) data acquisition unit was used as the interface between the thermocouples and computer, and incorporated many features that helped in data collection, including USB interface, on-board K-type conversion and built-in electronic cold junction.

Figure 4.17 TCIC DAQ Board: TCIC Interface Card with on-board electronic cold junction [49].

The main advantage of this interface card was its high sampling rate: 400 Hz total (50/Hz per channel with 8 channels). Since a maximum sampling rate of 15 Hz was used, the rate the card provided was more than adequate.

4.4.7   Accuracy

The total accuracy of the temperature measured is a combination of two factors: system accuracy and thermocouple accuracy [46].

The system accuracy is the sum of the thermal EMF error and CJC (cold junction compensation error).

$$E_{sys} = E_{EMF} + E_{CJC} \tag{4.3}$$

Thermal EMF error is caused by the junction of the leads of the thermocouple and the channel stations of the data acquisition board. These junctions of dissimilar metals create their own Seebeck effect and act as another thermocouple. However, as long as the temperatures around the DAQ board are steady, this error is small, on the order of 0.01 °C. Of greater significance is the error caused by the temperature measurement by the CJC. This error is caused by the thermistor, which provides the reference temperature not being accurate, and is on the order of 0.5 °C. The total system error is given by the TCIC DAQ board data sheet as 1.5 °C, taking all the above factors into account.

Thermocouple error is caused by manufacturing errors in the thermocouple and impurities in the metal leads. For a typical K-type thermocouple, the accuracy is 1.5 °C. This gives a total accuracy of 3.0 °C. However, it is important to note that this value is both an estimate and applies to an uncalibrated thermocouple. It can be seen from Table 4.2 that at boiling point temperatures, the uncalibrated thermocouple was accurate to within 1 °C, and 0.2 °C at freezing point temperatures. Since maximum experimental

temperatures were about 50 °C, and the thermocouples were calibrated, full system accuracy can be estimated to be about 0.2 °C.

4.4.8   Response Time

The response time of the thermocouple is the time difference between the change in environmental temperature and when the thermocouple reports this change. This lag in measurement is a function of both the heat transfer characteristics of the thermocouple and the response of the DAQ board CJC.

A thermocouple does not measure the temperature of its surroundings directly. Rather, it measures the effect the surrounding temperature has on the thermocouple itself. Knowledge of the dimensions and materials of the thermocouple can be used in an energy balance analysis to calculate this response time and evaluate whether it can be neglected or not.

In the case of a thermocouple heating up, an energy balance equation reduces to:

$$\dot{q}_{in} = \dot{q}_{stored} \tag{4.4}$$

or

$$hA(T_\infty - T) = \rho V c_p \frac{dT}{dt}, \tag{4.5}$$

with the initial condition being:

$$T(0) = T_0. \tag{4.6}$$

where A is the surface area of the bead tip of the thermocouple, and V is the volume of the bead tip. This equation can be integrated by separation of variables to give:

$$\frac{T_\infty - T}{T_\infty - T_0} = exp\left[-\frac{t}{\tau}\right] \tag{4.7}$$

$$\tau = \frac{\rho V c_p}{hA}.$$

τ is known as the time constant, and represents the time it takes the thermocouple to reach 63.2% of the environmental temperature in response to a step change in T_∞. This equation assumes lumped capacitance, which can be checked by calculating the Biot number of the thermocouple lead and comparing it to the threshold value of 0.1. With the thermal conductivity of the lead k=25 $\frac{W}{m \cdot K}$ and using the h value calculated below, the Biot number is 0.005, validating lumped capacitance.

To calculate the time constant, it can be assumed that the leading bead of the thermocouple is a sphere. The free stream thermocouple's dimensions are shown in Table 4.4.

Table 4.4 Thermocouple Dimensions: Dimensions of the free stream thermocouple for energy balance.

Diameter (mm)	0.25
Surface area A (mm^2)	0.197
Volume V (mm^3)	8.2E-3

Material properties can be taken as an average of the properties of chromel and alumel, shown in Table 4.5.

Table 4.5 TC Material Properties: Material properties of the free stream thermocouple.

	Chromel	Alumel	Average
Density ρ $\left(\frac{kg}{m^3}\right)$	8730	8600	8665
Specific Heat c_p $\left(\frac{J}{kg \cdot K}\right)$	448	523	486

The h value in τ can be calculated from a sphere in external flow according to Incropera et al. [21]:

$$Re_D = \frac{V \cdot D}{v} = \frac{4.7 \frac{m}{s} \cdot 0.00025m}{15.89E - 06 \frac{m^2}{s}} = 74.0$$

$$\overline{Nu_D} = \frac{\bar{h}D}{k} = 0.683 Re_D^{.466} Pr^{1/3} = 0.683 \cdot 74^{.466} \cdot 707^{1/3} = 4.52$$

$$h = \frac{k}{D} Nu_D = \frac{0.025 \frac{W}{m \cdot K}}{0.00025m} \cdot 4.52 = 464.7 \frac{W}{m^2 \cdot K}$$

With h known, τ can be calculated:

$$\tau = \frac{\rho V c_p}{hA} = \frac{8665 \frac{kg}{m^3} \cdot 8.2E - 3mm^3 \cdot 486 \frac{J}{kg \cdot K}}{464.7 \frac{W}{m^2 \cdot K} \cdot 0.197mm^2 \cdot 1000} = 0.376\ s$$

With τ known, the actual T_∞ time delay can be found from Equation 4.7, with the original free stream thermocouple data being T, and T_∞ being adjusted fluid temperature. A visualization of this adjustment can be seen Figure 4.18.
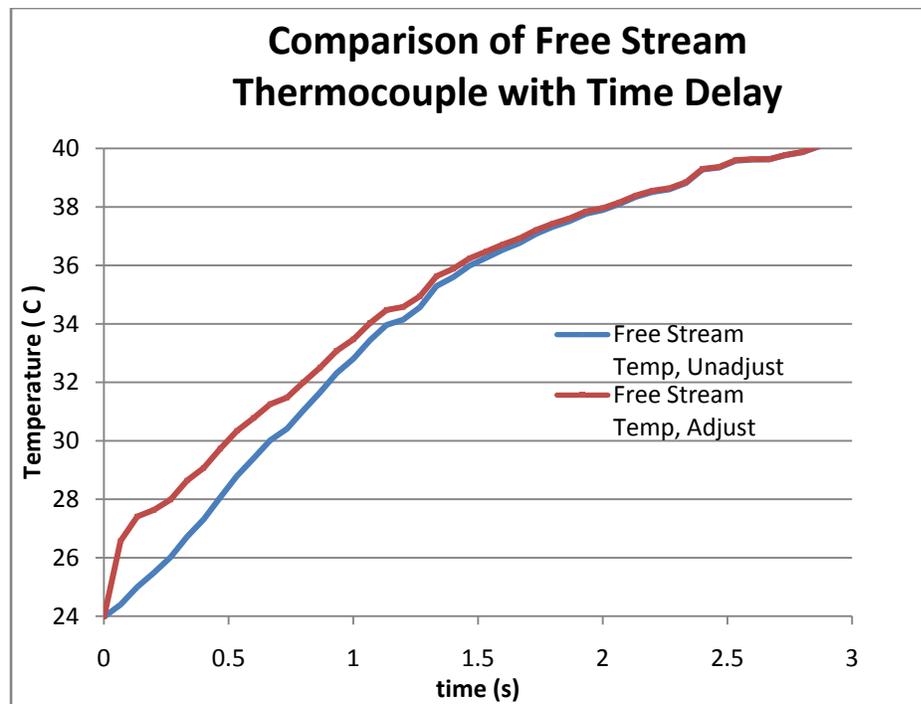
Figure 4.18 Time Delay: Time delay comparison of the 0.25 mm diameter free stream thermocouple. The blue line is raw thermocouple data, the red line is the adjusted data.

It can be seen that the difference between values is slight from 0 to 2 seconds, and not visible after 2 seconds. The difference was small enough to ignore except when comparing m_1, the first ramp of T_∞ calculated with inverse methods, to the free stream data.

Time delay values for the surface-mount thermocouples were provided in their data sheet. For the Style 2 surface-mount thermocouples from Omega used, a time delay *tau* of 2 to 5 milliseconds was given when mounted [50]. This was much less than the free stream wire thermocouple due to the comparative thinness of the material: 0.25 mm for the wire thermocouple versus 0.0125 mm for the surface-mount thermocouples. Because of this, it was decided no adjustments were needed for the surface-mount thermocouples due to heat transfer time delays.

The time shift caused by the data acquisition board is a function of its sampling rate, and the sampling rate is limited by the CJC time response. The TCIC series board used had 8 channels. Since the board cannot sample all 8 channels at once, it steps through them individually at a 400 Hz rate, equating to 50 Hz per channel. The 50 Hz individual channel sample rate can be neglected, since a much slower sampling rate of 15 Hz was used to correspond with the camera sampling rate. The time delay occurs when the DAQ board is switching between channels at the stated 400 Hz sample rate. This value corresponds to a 0.0025 second delay between each channel. Assuming a thermocouple is sampling on the last channel, a maximum delay of $0.0025 \cdot 7 = 0.0175$ seconds occurs, or 0.00875 seconds if four channels are used. Again, this value is small enough compared to the time scales used in the experimental trials to be neglected.

4.4.9 <u>LabVIEW</u>

LabVIEW (Laboratory Virtual Instrumentation Engineering Workbench) is a computer program published by National Instruments used to aid in the process of data acquisition. This program was the third main component of the temperature measurement system, and worked in conjunction with the thermocouples and TCIC DAQ board.

LabVIEW was used to interface the DAQ board with the computer and translate the signals it gathered into useable data. LabVIEW uses a visual-based block diagram programming language called a VI, or virtual instrument. It links the block diagram to a front panel, a virtual tool that allows the user to interact with the program and data acquisition.

The advantages of LabVIEW to this experiment was the visualization of data in real-time. Temperature curves could be seen instantly during experimental trials through the use of LabVIEW's graphing subroutines, and during calibration it aided in determining whether the test plate was at steady-state or not.

A disadvantage of LabVIEW is that it is designed to work with NI hardware, and therefore does not function as well when used with boards manufactured by other companies. Since the DAQ board used was manufactured by Omega, a significant amount of additional programming was needed for the components to work together.

4.4.10 <u>MATLAB</u>

MATLAB, short for MAtrix LABoratory, is a mathematical computer program designed for numerical computing. It is particularly useful for automating analysis of large amounts of data, and this was its primary function in this research. Since an experimental trial's data consisted of 1,000 images, each with thousands of pixels,

MATLAB was an essential tool. A summary of MATLAB scripts can be found in

Appendix B.

4.5 Data Analysis

Before data could be analyzed, the following attributes and assumptions of the

flow structure in the wind tunnel were determined or verified:

- Velocities

- Reynolds numbers

- Flow wall effects

Also, the following heat transfer attributes and assumptions were investigated:

- Nusselt correlations

- Semi-infinite solid assumption

- Constant h assumption

Next, the TLC hue to temperature calibration is described in the following steps:

1. RGB data was converted to hue data after noise-subtracting and averaging.

2. Steady-state and transient hue and thermocouple data was used to form a
   hue/temperature curve with a polynomial fit.

3. This polynomial calibration was applied to the experimental trial data to achieve a
   surface temperature vs. time curve at different test plate locations.

4.5.1 Correlations for the Test Plate

To evaluate the results of the experimental trials, the h values that were found

were compared to theoretical correlations.

4.5.1.1   Flat-Plate Correlations

The following flat-plate correlations were used to calculate boundary layer thicknesses and verify experimental heat transfer data. The correlations used were reported by Incropera et al. [21].

Boundary layer thickness δ is defined as the value for which $u = 0.99u_\infty$. Two different correlations exist, one each for laminar (found using a similarity method on the continuity and momentum equation), and turbulent boundary layers. To determine turbulence, a critical Reynolds number Re_c was assumed to fall between 10^5 and $3 \cdot 10^6$. Also, air properties were evaluated at a film temperature of 300 K for convenience, with a kinematic viscosity $v = 15.89 \cdot 10^{-6} \frac{m^2}{s}$. The equation to calculate the Reynolds number based on x is

$$Re_x = \frac{V \cdot x}{v}, \tag{4.8}$$

where x is the stream-wise distance from the start of the boundary layer.

The correlation for laminar boundary layer thickness on a flat surface is:

$$\delta = \frac{4.92x}{(Re_x)^{0.5}}. \tag{4.9}$$

The correlation for turbulent boundary layer thickness on a flat surface is:

$$\delta = \frac{0.37x}{(Re_x)^{0.2}}. \tag{4.10}$$

The boundary layer thickness equations were not needed for any calculations involving the flat plate, but were applied to calculations involving the wind tunnel test section in Section 4.5.2: Tunnel Side Wall Effects.

In a similar manner, there exists a Nusselt correlation for a flat plate, found using similarity methods on the energy equation. The Nusselt number for a flat plate is a

function of the Reynolds and Prandtl numbers. Table 4.6 contains a summary of Nusselt

number correlations for both laminar and turbulent flow, as well as two boundary

conditions: constant surface temperature and constant heat flux.

Table 4.6 Nusselt Correlations: Correlations corresponding to different conditions.

Conditions	Nusselt Correlation	
Laminar, $T_s = $ const	$Nu_x \equiv \dfrac{h_x x}{k} = 0.332 Re_x^{1/2} Pr^{1/3}$	(4.11)
Turbulent, $T_s = $ const	$Nu_x \equiv \dfrac{h_x x}{k} = 0.0296 Re_x^{4/5} Pr^{1/3}$	(4.12)
Laminar, $q_s^{''} = $ const	$Nu_x \equiv \dfrac{h_x x}{k} = 0.453 Re_x^{1/2} Pr^{1/3}$	(4.13)
Turbulent, $q_s^{''} = $ const	$Nu_x \equiv \dfrac{h_x x}{k} = 0.0308 Re_x^{4/5} Pr^{1/3}$	(4.14)

For these experiments, a Prandtl number of 0.707 was used based on a 300 K air

film temperature. In this manner, *h* can be found from generally accepted correlations and

compared to *h* values returned by the inverse method to check for accuracy.

4.5.1.2   Velocity Measurement

Since calculation of the Reynolds number was needed, an accurate measurement

of velocity was required. For this, a pitot tube and Dwyer Series 475 Mark III digital

manometer was used. The manometer had a reported accuracy of $\pm 0.005''$ W.C. and a

range of 0-1.000" W.C. The pitot tube was inserted into the top of the wind tunnel

slightly ahead of the leading edge of the flat plate, halfway between the plate and top of

the test section. The fan was activated with heater off, and pressure readings were taken

at 1% fan intervals from 10% to 20%. The pressure readings were then translated into

velocity readings using Equation 4.15, where ΔP is the pressure differential reading from

the manometer, and ρ is the density of air, evaluated at film temperature, $1.193 \frac{kg}{m^3}$.

$$V = \sqrt{\frac{2\Delta P}{\rho}} \qquad (4.15)$$

Table 4.7 shows the range of velocities corresponding to the pressure differences

measured with the Dwyer manometer.

Table 4.7 Velocity: A range of wind tunnel velocities measured.

% Blower	$\Delta P \pm 0.005$ (in. W. C.)	ΔP (Pa)	$V\left(\frac{m}{s}\right)$
10	0.016	3.287	2.35
11	0.023	4.724	2.81
12	0.028	5.751	3.11
13	0.035	7.189	3.47
14	0.042	8.627	3.80
15	0.049	10.065	4.11
16	0.060	12.324	4.55
17	0.067	13.762	4.80
18	0.078	16.022	5.18
19	0.090	18.487	5.57
20	0.102	20.951	5.93

4.5.1.3   Boundary Layer Start Position

An accurate value of x, the distance from the start of the boundary layer, was

needed to calculate both Reynolds and Nusselt numbers. In an ideal flat plate, this value

would be the distance from the leading edge. However, the reality is that a blunt edge

creates a stagnation point at the leading edge of the flat plate [51]. This causes the flow to

separate from the leading section of the surface of the plate, and reattach at some distance

downstream, as seen in Figure 4.19. The flat plate was positioned in the center of the

wind tunnel test section such that there was 10 cm between the top surface and the wall,

and 10 cm between the bottom surface and the wall.
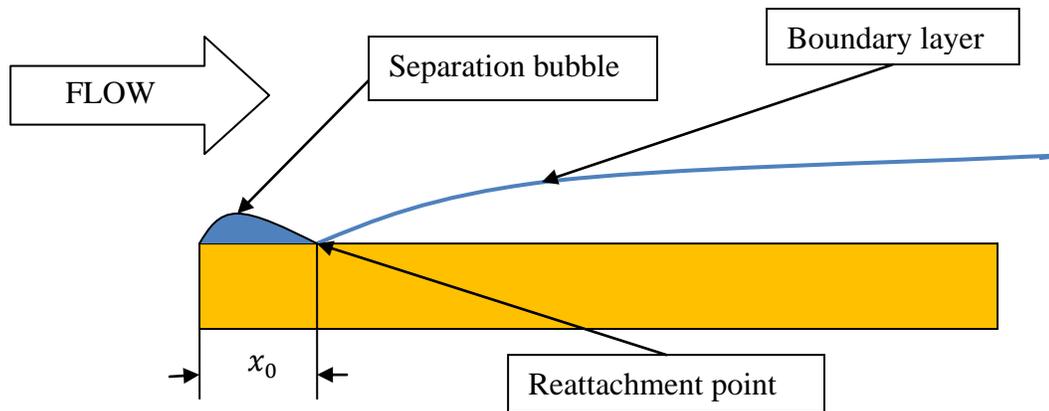
Figure 4.19 Separation Bubble: A blunted edge on a flat plate creates a separation bubble at the leading edge when it is immersed in a flow.

A rounded edge was added to the leading edge of the plate to minimize this effect. The distance from the leading edge of the plate to the reattachment point, x_0, needed to be found to determine experimental *x* values.

To find x_0, the image series from experimental trials was examined. From these, the start of the boundary layer could be identified as the position where the TLCs first started to change color. It should be recognized this position will change as velocity changes. Since red is the first color the TLCs experience, the red signal was examined.

Figure 4.20 shows an image closely corresponding to the start of the red signal to illustrate the location of the boundary layer start. The leading edge of the plate corresponds to the left edge of the image (the leading edge of the plate being the edge of the top of the half-circle leading edge). The green light at the lower edge of the image is an LED to denote activation of the tunnel air heater.
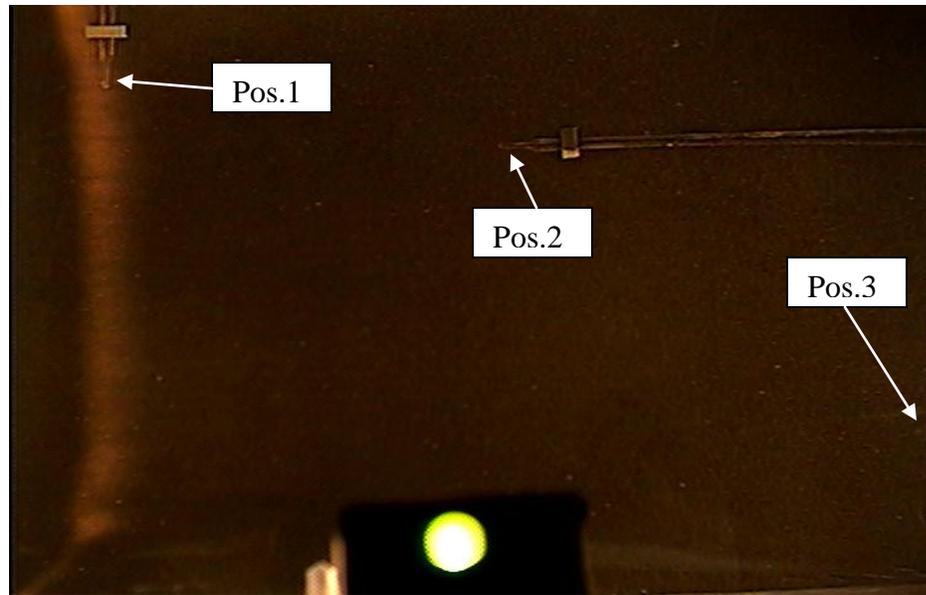
Figure 4.20 Boundary Layer Start: This image corresponds to the point where the TLCs just start to change color. The left edge of the image corresponds to the plate leading edge. It can be seen the boundary layer starts at a distance downstream of the leading edge.

The pixel location of the start of the red signal near the middle of the plate was correlated to position. For this trial (blower at 16.4%), x_0 was found to be 7/8" (2.22 cm) from the leading edge of the plate (considered to be the top of the rounded edge). The positions of the three surface-mount thermocouples (designated Position1, Position 2, and Position 3) were 1", 4", and 7" from the leading edge, respectively. Accounting for the position of x_0, the distance from the start of the boundary layer of the three thermocouples (*x*) becomes 1/8", 3 1/8", and 6 1/8".

4.5.1.4 Reynolds Numbers

With *V* and *x* found, the Reynolds numbers could now be calculated. A kinematic viscosity of air of 1.59E-05 $\frac{m^2}{s}$ was used. Table 4.8 contains a summary of these Reynolds numbers.

Table 4.8 Reynolds Numbers: A summary of the Reynolds numbers at different positions on the test plate.

%	$V\left(\frac{m}{s}\right)$	Reynolds Number		
		Pos. 1	Pos. 2	Pos. 3
10	2.35	469.01	11725.24	26737.24
11	2.81	562.32	14058.07	32056.82
12	3.11	620.44	15511.03	35370.04
13	3.47	693.67	17341.86	39544.91
14	3.80	759.88	18997.06	43319.28
15	4.11	820.77	20519.17	46790.17
16	4.55	908.23	22705.83	51776.44
17	4.80	959.75	23993.80	54713.43
18	5.18	1035.55	25888.63	59034.22
19	5.57	1112.35	27808.85	63412.93
20	5.93	1184.19	29604.78	67508.22

To determine whether the flow is laminar or turbulent, these values are to be compared to a *critical* Reynolds number, a number that denotes transition to turbulence. It has been found experimentally that the critical Reynolds number varies between 10^5 and $3 \cdot 10^6$ [21]. Referring to Table 4.8, this implies that all flows were laminar. However, roughness of the plate, flow inconsistencies, etc., could induce turbulence. Therefore, *h* values for both laminar and turbulent conditions were compared to experimental *h* values. These Reynolds numbers were used to calculate Nusselt number correlations to compare with the *h* values calculated with inverse methods.

4.5.2   Tunnel Side Wall Effects

The points to be analyzed on the plate were chosen to be outside the boundary layer generated by the side walls of the wind tunnel. The test section of the tunnel had a square cross section, with the plate positioned in the middle, flush with the side walls as shown in Figure 4.21.
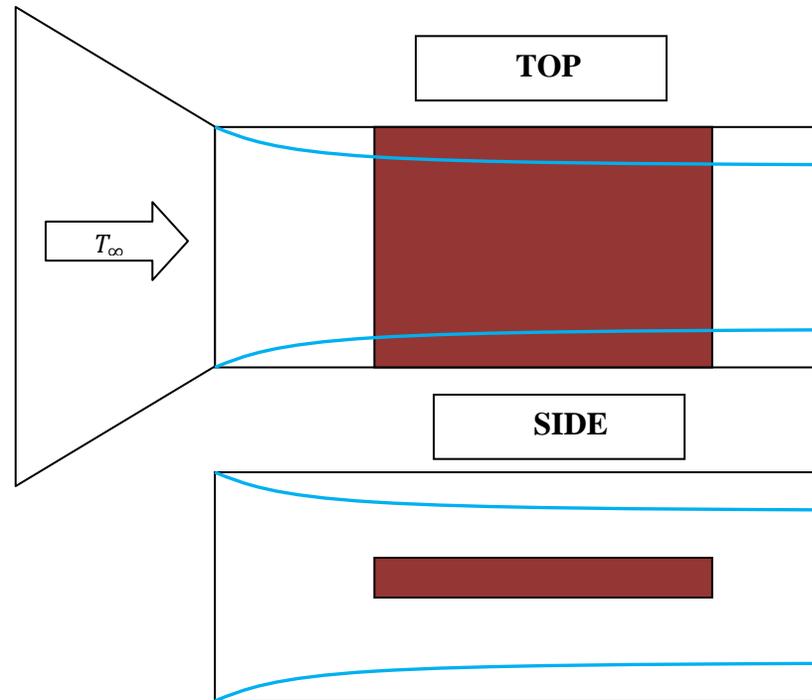
Figure 4.21 Test Section Views: A top-down and side view of the wind tunnel test section. The red object represents the flat plate, and the blue lines show where side-effect boundary layers were expected to reside.

To do this, boundary layer thicknesses, δ , were calculated for the range of

possible wind tunnel speeds, which ran at a minimum of 3 m/s to a maximum of 30 m/s,

with the test speed being around 6 m/s. The correlation equations for laminar and

turbulent flat plate boundary layers are given by Equations 4.16 and 4.17, respectively.

$$\delta = \frac{4.92x}{(Re_x)^{0.5}} \tag{4.16}$$

$$\delta = \frac{0.37x}{(Re_x)^{0.2}} \tag{4.17}$$

A plot of the results is shown in Figure 4.22.
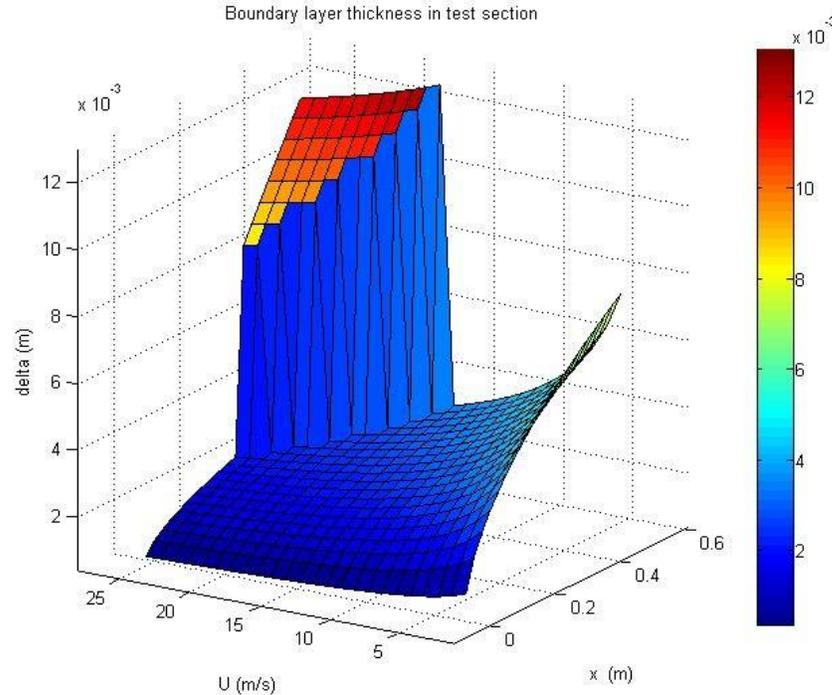
Figure 4.22 Boundary Layer Plot: Boundary layer thickness in test section. Plate starts at 23 cm and is 26 cm long. The large jump in δ that corresponds to large values of x and U represents the transition to turbulent flow.

The boundary layer growth started at the beginning of the test section. The forward edge of the plate started 23 cm into the test section, and the plate was 26 cm long. Generally, it can be seen that δ is below 0.8 cm. The discontinuity in the plot represents the boundary layer's transition to turbulence. In this case, any wind tunnel velocities over 20 m/s will result in a turbulent boundary layer, and a boundary layer thickness of 1.2 cm or higher. However, since no wind tunnel velocities over 20 m/s were used, it can be assumed any section of the plate significantly more than 0.8 cm from the side edges were unaffected by wall effects. The MATLAB algorithm for this calculation can be found in Appendix B.

4.5.3   Semi-Infinite Solid Assumption

For analysis, it was assumed that the flat plate could be treated as a semi-infinite solid: a solid that extends to infinity in all directions except for one, its surface [21]. For the assumption to be strictly true, the centerline of the plate would see no temperature rise over the duration of the experiment. To evaluate the semi-infinite solid assumption, a 1-D plane wall with convection model was compared to a semi-infinite solid model, as the experimental plate experienced heating on both sides.

The exact solution [21] for a plane wall with thickness 2L is

$$\frac{T(x,t)-T_\infty}{T_0-T_\infty} = \sum_{n=1}^{\infty} C_n \exp(-\zeta_n^2 Fo)\cos(\zeta_n x^*)$$

$$C_n = \frac{4\sin(\zeta_n)}{2\zeta_n + \sin(2\zeta_n)} \tag{4.18}$$

$$x^* = \frac{x}{L}$$

$$Fo = \frac{\alpha t}{L^2}$$

where x^* is the dimensionless distance from the centerline, Fo is the Fourier number, and ζ_n is the nth root of the transcendental equation

$$\zeta_n \tan(\zeta_n) = Bi \tag{4.19}$$

$$Bi = \frac{hL}{k}$$

with Bi being the Biot number. In this case, 2L would be the full-plate thickness of 1.7 cm, and L is the half-plate thickness.

For the half-plate thickness L of 0.25" (0.635 cm), the Fourier number was found to be 0.162 at a time of 60 seconds (the maximum experiment duration), indicating a

multi-term calculation was needed. Since the centerline temperature was the point of interest, $x^* = 0$. The equation becomes

$$T_{center} = T_\infty + (\text{T}_0 - \text{T}_\infty) \sum_{n=1}^{3} C_n \exp(-\zeta_n^2 Fo). \qquad (4.20)$$

Four terms were chosen because it was found the fourth term only changed the temperature value by 0.0024%, whereas the third term contributed 2.9% to the final answer.

Using this model and the conditions and material properties for polycarbonate listed in Table 4.9, the centerline temperature rise after 60 seconds was calculated to be 1.3 degrees Celsius.

Table 4.9: Variable and parameter values used in centerline temperature response calculation.

Variable	Value
T_0	24 °C
T_∞	45 °C
h	80 W/m^2-K
k	0.19 W/m-K
ρ	1190 kg/m^3
c_p	1470 J/kg-K

This was compared to a temperature rise in a semi-infinite solid, calculated at the same distance from the surface and the same parameters and temperatures. For this calculation, the step change to model was used from Section 3 with x=0.25" (.0635 cm):

$$\frac{T(x,t)-T_0}{T_\infty-T_0} = \left[\text{erfc}\left(\frac{x}{2\sqrt{\alpha t}}\right) - \exp\left(\frac{hx}{k} + \frac{h^2 \alpha t}{k^2}\right) \text{erfc}\left(\frac{x}{2\sqrt{\alpha t}} + \frac{h\sqrt{\alpha t}}{k}\right) \right].$$

At a time of 60 seconds, the temperature rise given by the semi-infinite step change model was 0.7 °C. Figure 4.23 shows a plot of the model comparison. It shows that the plane-wall model follows the semi-infinite model until about 25 seconds, when the two models begin to diverge. At 60 seconds, the maximum length of an experimental trial, the difference between the two models is about 0.6 °C. This discrepancy is mitigated somewhat by the fact that the h used was based on the highest h values produced by the experiment and the step change model produces a higher temperature rise than the series of ramps model used. A ΔT of 0.6 °C, 3% of the total temperature rise, was considered a tolerable error that did not need to be adjusted for.
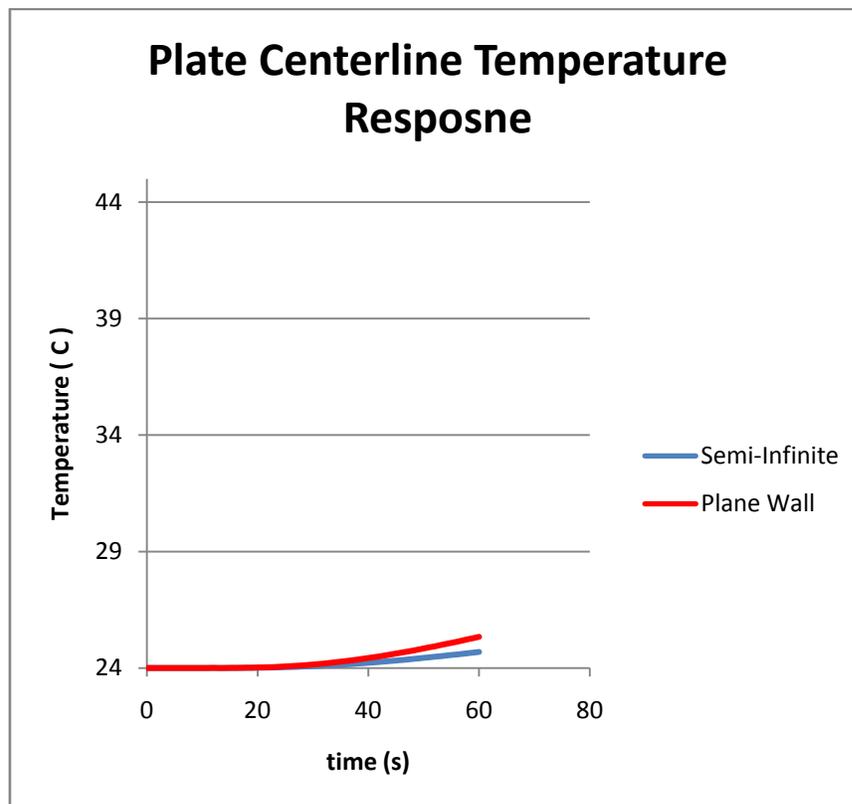
Figure 4.23 Centerline Model Comparisson: When comparing a "realistic" plane wall model with an "ideal" semi-infinite model, the difference is 0.6 °C at a time of 60 seconds.

4.5.4 Constant h Assumption

In the development of the series of ramps model assumed for the transient surface temperature response of the test plate, it was assumed that the heat transfer coefficient h parameter was constant. It is useful to reexamine the definition for h for this assumption:

$$h = \frac{-k\frac{\partial T_f}{\partial y}\Big|_{y=0}}{(T_\infty - T_s)}.$$

It is shown that h is a ratio of the temperature gradient at the surface to the temperature difference that drives convective heat transfer. It can be seen h is not strictly constant for a sudden temperature rise, as is the case for this study. In the transition from $T_\infty = T_s$ to $T_\infty > T_s$, h varies as the rate of increase/decrease of the temperature gradient and temperature difference equalizes.

Thiele [7] performed a numerical FLUENT analysis of heat transfer coefficient variation over a flat plate in time under similar experimental conditions to this study. It was found that Nusselt values, and thus h values, become constant in the first 0-0.3 seconds after the start of the flow temperature rise, and that for an acrylic plate (acrylic has very similar material properties to polycarbonate), the Nusselt numbers are very close to a constant surface temperature correlation, as seen in Figure 4.24.
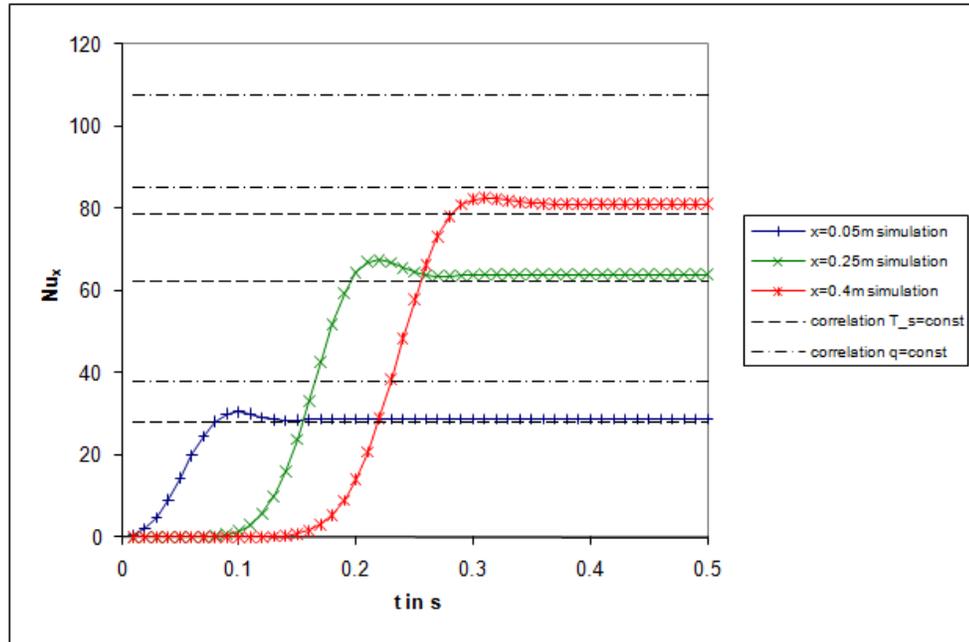
Figure 4.24 Constant *h* Assumption: *h* values become constant between 0.001 and 0.3 seconds [7]. The dotted lines indicate the bounds of the Nusselt correlations, with the curves representing different plate positions.

To understand why *h* becomes constant, it is useful to revisit the definition of *h*:

$$h = \frac{-k\frac{\partial T}{\partial y}\big|_{y=0}}{(T_\infty - T_s)}.$$

If *k* is constant, this implies that the gradient is proportional to the driving temperature difference regardless of time:

$$\frac{\partial T}{\partial y}\bigg|_{y=0} \propto (T_\infty - T_s).$$

As plate surface temperature increases, the difference between fluid temperature and surface temperature decreases. Also, the magnitude of the temperature of the gradient at the surface decreases. The above figure implies that these values decrease at the same rate during heating. Why this is true is not readily intuitive, however.

4.5.5 TLC Calibration

To achieve a relationship between hue and temperature, a calibration needed to be performed. This calibration was done *in situ* under experimental conditions. The calibration procedure was as follows.

Calibration was done after an experimental trial was performed, leaving enough time for the apparatus to reach equilibrium with room temperature again, about four or more hours. This assured that lighting and environmental conditions would be kept as similar as possible.

The wind tunnel was turned on to 16.4% (4.7 $\frac{m}{s}$), and 3-5 seconds of camera/thermocouple data was taken with the heater turned off to obtain a baseline value. Next, the heater was activated to 20%, and 5 minutes were allowed for the plate to come to equilibrium with the fluid temperature. Next, 3-5 seconds of camera/thermocouple data was taken. This procedure was repeated, increasing the heater setting, to give 9 calibration points between room temperature and the TLC's clearing temperature.

4.5.6 RGB Data Extraction and Averaging

At each calibration point, around 50 frames of data were taken. A pixel was chosen at each surface-mount thermocouple stream-wise position near the center of the test plate, designated Positions 1, 2, and 3, corresponding to distances from the leading edge of 1", 4", and 7", respectively. First, a "central pixel average" was taken at each frame, averaging the chosen pixel's RGB data with two pixels each to the left and right (at the same stream-wise position) – a five pixel average. After position-averaging, the RGB values were time-averaged over each frame to obtain a single RGB value to correspond with each thermocouple position at the calibration point. The thermocouple

data was time-averaged to obtain a single temperature value per thermocouple at each calibration point.

4.5.7   Background Gray Subtraction

Background "grayness" was then removed from the RGB values whose values could range from 0 to 255. The values of RGB for each position were examined with the TLCs in their transparent state (heater off), and the amount of background color present in the signal was determined by measuring the average RGB value's distance from zero. Table 4.10 shows the background gray offset values.

Table 4.10 RGB Noise: Values were subtracted from the RGB signal to make the unheated plate "black."

	Position 1	Position 2	Position 3
R	37.513	48.209	47.522
G	16.687	22.940	27.052
B	0.157	0.609	5.6

These values were then subtracted from the averaged calibration data at each calibration point.

4.5.8   RGB Data to Hue Data

The noise-subtracted RGB calibration data was then converted to a HSV signal using MATLAB's built-in functions. The hue value was paired with thermocouple data as shown in Table 4.11.

Table 4.11 Hue Calibration Data: Hue values were paired with thermocouple data to correlate the two.

Heater	T ($^{\circ}$C)	Hue		
		Pos. 1	Pos. 2	Pos. 3
0%	22.51	0.00	0.00	0.00
20%	24.58	0.00	0.00	0.00
35%	28.77	23.18	18.76	20.80
40%	31.21	80.88	85.19	90.43
45%	33.50	93.07	96.06	106.16
50%	36.06	111.24	112.35	124.34
55%	37.54	118.44	122.14	134.05
60%	41.60	144.08	143.08	148.49
65%	44.53	148.83	150.45	151.95

The trial RGB data was also converted to hue. This conversion can be seen in Figure 4.25.
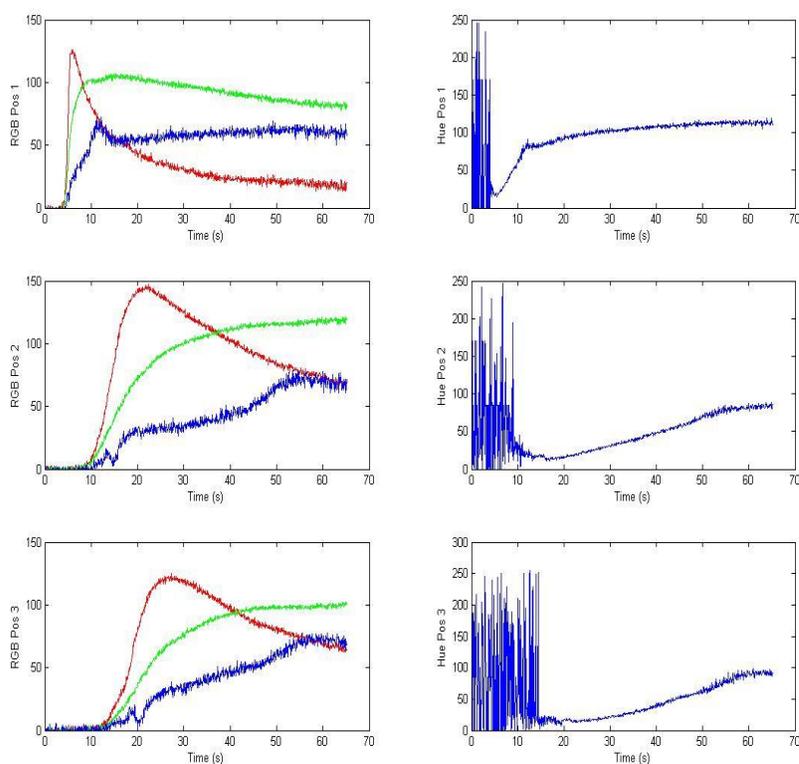
Figure 4.25 RGB to Hue Trial Data: RGB was converted to hue data.

It can be seen that in the initial few seconds of data where the plate is below the TLC color change start, the hue data alternates between its minimum value of 0 and maximum value of 255. This is due to the conical color model used for HSV, in which a position of 0 degrees and 360 degrees on the circle represents black. This manifests in the color model equations for hue:

$$M = \text{Max}(R, G, B)$$

$$m = \text{Min}(R, G, B)$$

$$C = M - m$$

$$H' = \begin{cases} \text{undefined} & \text{if } C = 0 \\ \frac{G-B}{C} \bmod 6 & \text{if } M = R \\ \frac{B-R}{C} + 2 & \text{if } M = G \\ \frac{R-G}{C} + 4 & \text{if } M = B \end{cases} \quad (4.21)$$

When the components of the RGB signal are all close to 0, M and m are very close to each other, causing alternations between which equation in the piecewise hue function is used.

This data was eventually negatively time-shifted 58 frames (4 seconds) to coincide with the start of the heater, negating some of this initial noisy hue data. Any additional noisy hue data was disregarded for the purposes of parameter estimation from the inverse method. An additional 2 seconds of data was ignored for Position 1, 12 seconds for Position 2, and 19 seconds for Position 3.

4.5.9 Hue/Temperature Calibration

Temperature versus hue was then plotted, and a 4th order polynomial fit was applied to the calibration data to achieve a calibration curve. The curve was very flat from 29 °C to 31 °C. In this portion of the curve, hue was changing very quickly over a

short temperature range. This sensitivity of hue to temperature made it difficult to obtain calibration points in this region.

As described in Section 4.3.2.1, the camera used in these experiments had a peculiarity where the blue signal experienced a "bump" at a certain threshold. At a blue value of 50, the signal rises unnaturally fast, then drops at a blue value of 75. This bump was translated to hue, and in turn to temperature. Unfortunately, this bump occurred in the region in which there was no steady-state calibration data.

To obtain calibration in this region of the hue signal, the experimental trial transient hue data was compared with the thermocouple data in the region of the bump. Figure 4.26 shows the hue data compared with thermocouple data. The hue data was given a 2nd order polynomial fit, while the thermocouple data was given a power fit.
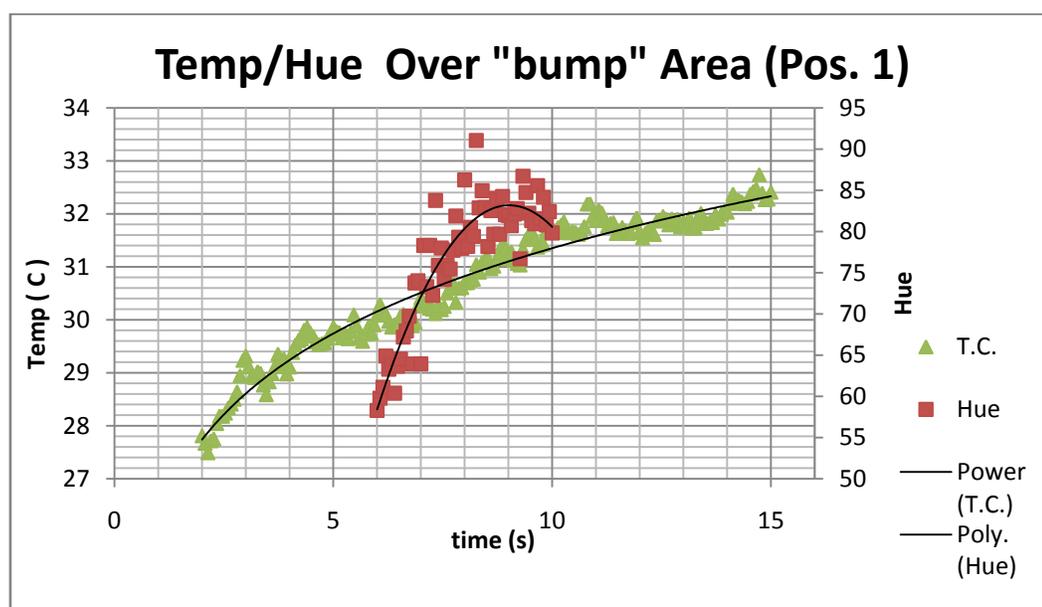
Figure 4.26 Parametric Fit Pos. 1: Hue data was compared to thermocouple data in the bump region.

The set of parametric temperature-time (power) and hue-time (polynomial) equations were evaluated at different times to form a set of hue-temperature points to add to the calibration. Table 4.12 shows that points were chosen at 0.5 second intervals.

Table 4.12 Parametric Fit Data: times were chosen to relate hue to thermocouple temperature in the chosen range.

Time (s)	Hue	T.C. (°C)
6	58.43	30.16
6.5	65.95	30.34
7	72.11	30.51
7.5	76.92	30.67
8	80.37	30.82
8.5	82.46	30.96
9	83.19	31.10
9.5	82.56	31.23

These points were then added to the calibration curve and given a 3rd order polynomial fit. Trial data was overlain to verify how well the calibration curve fit, as shown in Figure 4.27.
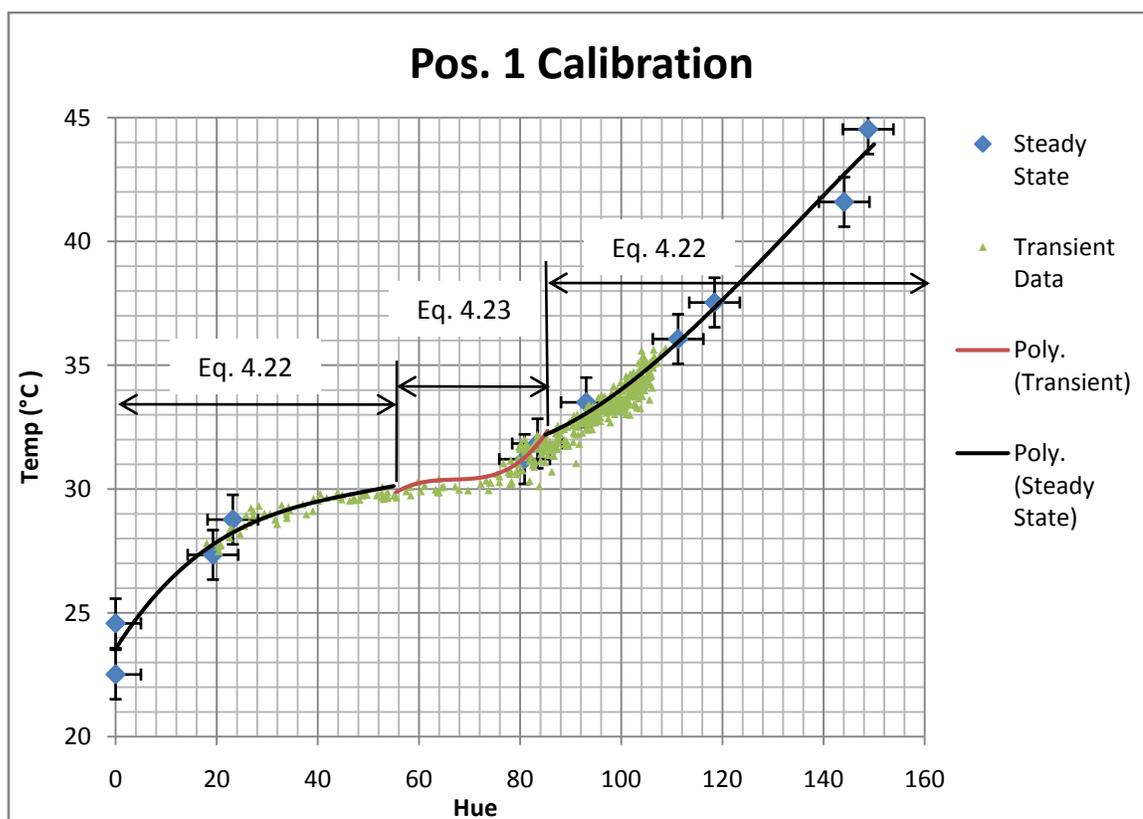
Figure 4.27 Pos. 1 Adjusted Calibration Curve: Points were added from transient data to provide more information in the flat region. Green points indicate how well the curve fits the trial data. The calibration was applied in a piecewise fashion.

It can be seen that, for this plate position, there was data to which the 4th order calibration curve applied, and data to which the 3rd order fit applied. So, a piecewise calibration was applied, with Equation 4.23 applied to data with hues between 60 and 85, and Equation 4.22 applied to all other data.

$$T = -1.56E - 07H^4 + 5.75E - 05H^3 - 6.35E - 03H^2 + 0.321H + 23.55 \quad (4.22)$$

$$T = 2.94E - 04H^3 - 5.84E - 02H^2 + 3.91H - 57.23 \quad (4.23)$$

The hue data was also very noisy in the bump region. To account for this, any data more than seven standard deviations from a power curve fit of the series was disregarded. This method discounted 15 data points. The final temperature trial data at Position 1 determined from hue is shown in Figure 4.28. In this figure, the hue temperature data compares well to thermocouple data verifying its accuracy. Also, it can be seen the hue temperature data has variable noise. In the 28-30 °C region, the data has very little noise, and the noise increases with temperature as more blue appears in the signal.
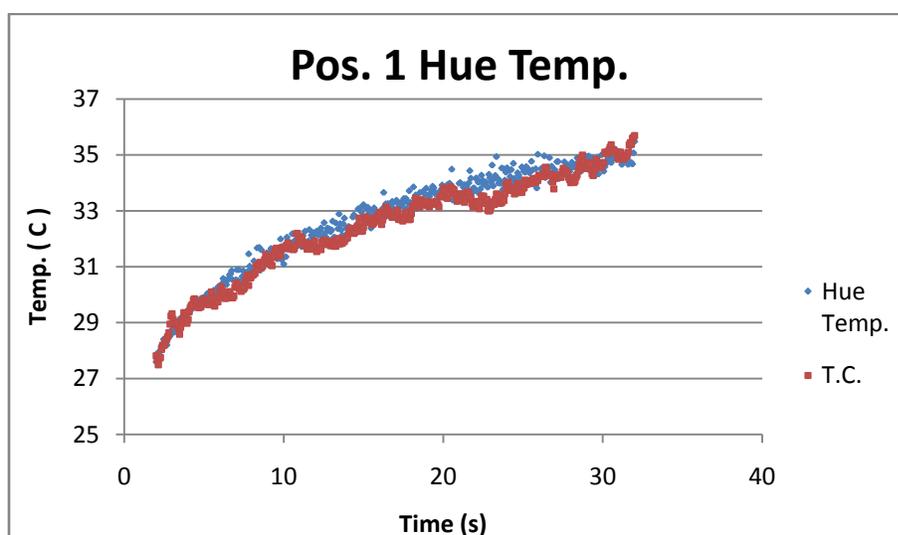
Figure 4.28 Pos. 1 Final Data: Here the hue data is compared to thermocouple data for verification.

The thermocouple signal in the above plot can be seen to be noisier than the hue temperature data. The patterns in the thermocouple noise indicate a frequency pattern to the noise. It was believed that the variable speed controller for the wind tunnel blower caused electrical interference with the data acquisition unit, as this noise only occurred when the blower was on. This noise was present in all thermocouples. However, as the thermocouples were calibrated with the blower turned on, absolute accuracy was not affected. This noise was present in all thermocouples.

A similar process was performed for Positions 2 and 3 on the plate. However, because heat transfer was slower in these regions, the "bump" region was more protracted, involving a much larger percentage of the data. In fact, almost all of the data lay in the "flat" portion of the calibration curve, determined by the transient calibration technique.

Figure 4.29 shows the Position 2 calibration. This steady-state calibration predicted a flatter curve in the 29-31 °C region than what best represented the data by the transient data.
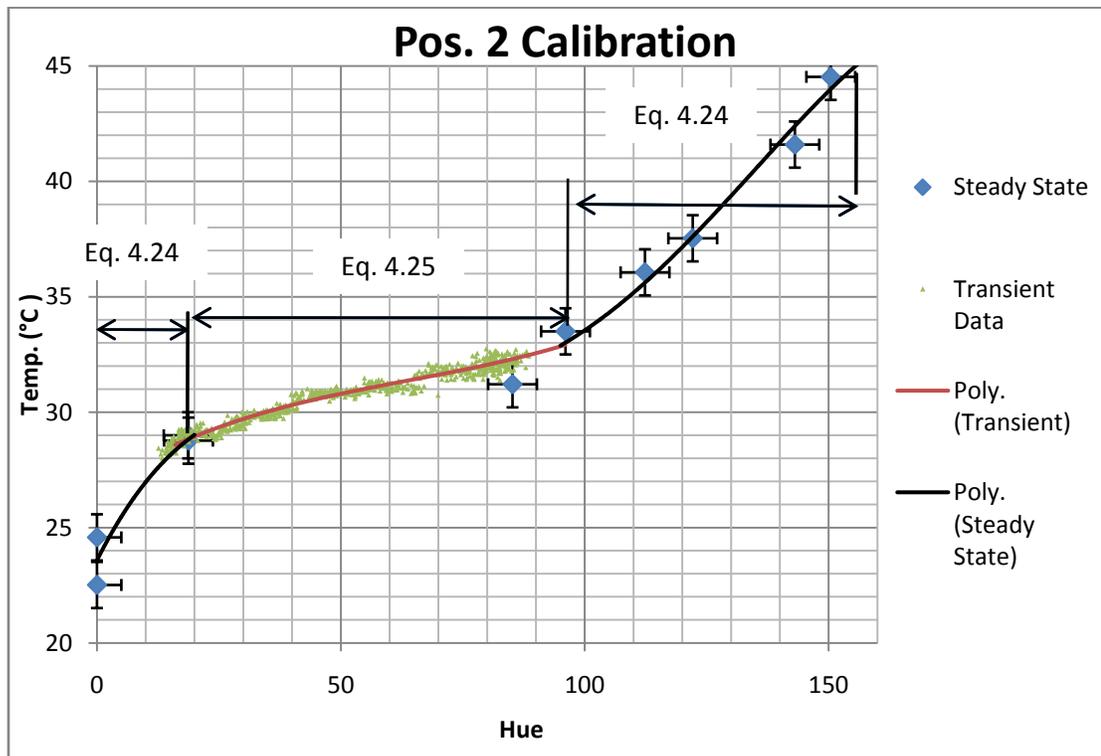
Figure 4.29 Pos. 2 Adjusted Calibration Curve: Points were added from transient data to provide more information in the flat region. Green points indicate how well the curve fits the trial data. The calibration was applied in a piecewise fashion.

Since all of the data lay in the flat region at Position 2, only the 3rd order polynomial, Equation 4.25, was applied to the trial data. Equation 4.24 shows the 4th order polynomial portion of the calibration.

$$T = -2.05E - 07H^4 + 7.76E - 05H^3 - 8.91E - 03H^2 + 0.42H + 23.6 \quad (4.24)$$

$$T = 8E - 06H^3 - 0.0015H^2 + 0.137H + 26.744 \quad (4.25)$$

Figure 4.30 shows the final calibrated data for Position 2 compared to the thermocouple data at that position. Here the best data lay between 15 and 46 seconds. The low noise region of the TLCs, which happened over a span of 3 seconds for Position 1 now accounts for the entire range of data here.
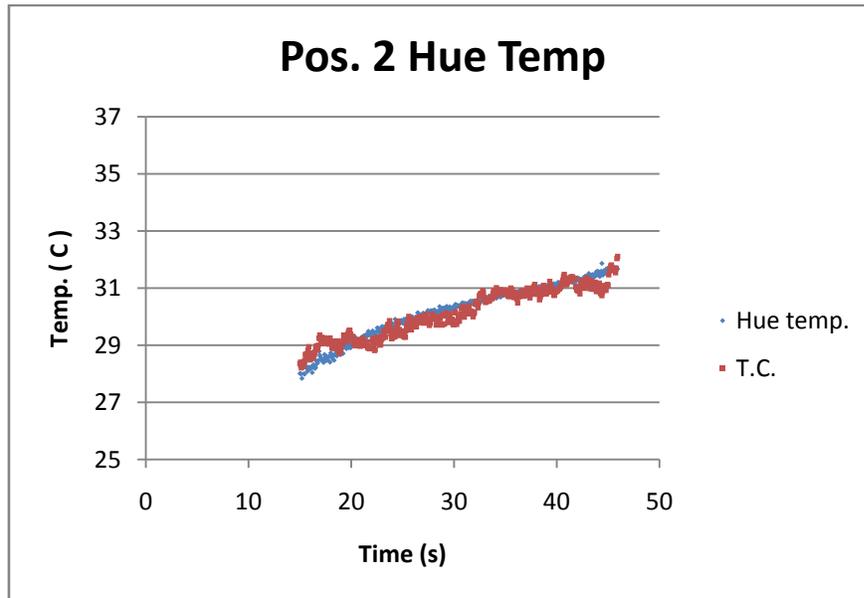
Figure 4.30 Pos. 2 Final Data: Here the hue data is compared to thermocouple data for verification.

Figure 4.31 shows the Position 3 calibration.
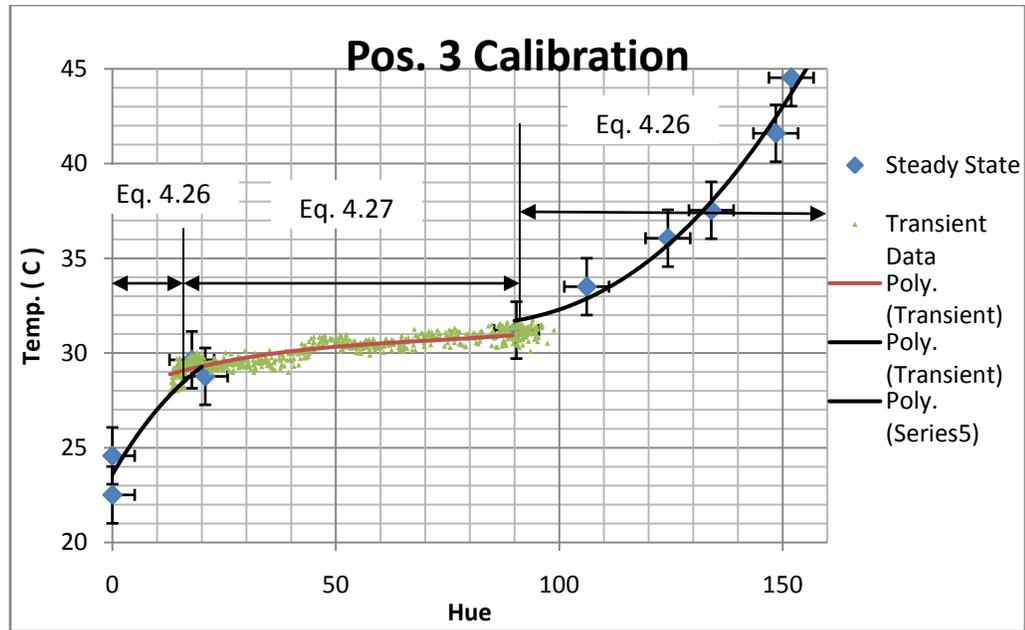
Figure 4.31 Pos. 3 Adjusted Calibration Curve: Points were added to provide more information in the flat region. Green points indicate how well the curve fits the trial data.

As with Position 2, only the 3rd order polynomial curve (Equation 4.27) was applied to the trial data. The 4th order calibration curve can be found in Equation 4.26.

$$T = -9.1E - 08H^4 + 5.3E - 05H^3 - 7.2E - 03H^2 + 0.41H + 23.6 \qquad (4.26)$$

$$T = 8E - 06H^3 - 0.0011H^2 + 0.09H + 27.896 \qquad (4.27)$$

Figure 4.32 shows the final calibrated data for Position 3 compared with the thermocouple data for that position. Here the best data lay between 20 and 48 seconds. While this position's thermocouple was particularly noisy, it can still be seen that the data fits well. However, since this thermocouple was used in the calibration curve above, this position also has the highest amount of uncertainty. As with Position 2, all of the data was within the low-noise portion of the TLCs.
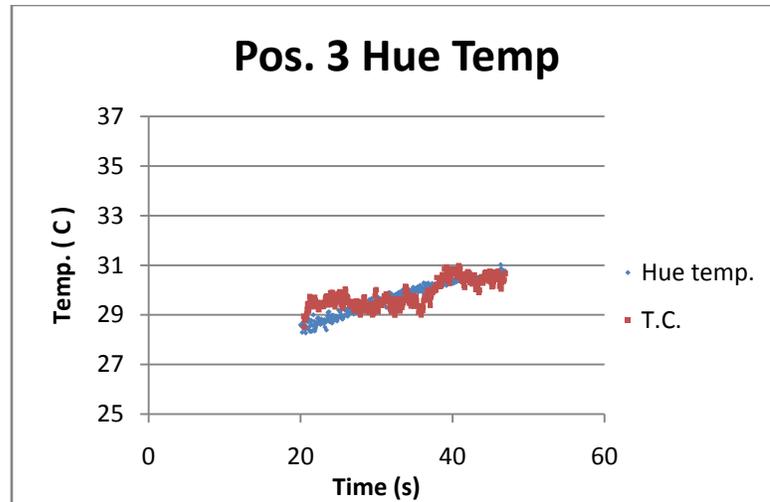
Figure 4.32 Pos. 3 Final Data: Here the hue data is compared to thermocouple data for verification.

4.5.10 Ramp Starting Parameter Guess

Five ramps were chosen to represent the data based on areas of linearity and high curvature observed in the measured fluid temperature history. Figure 4.33 shows the division of ramps and their slopes, along with the fluid temperature history.
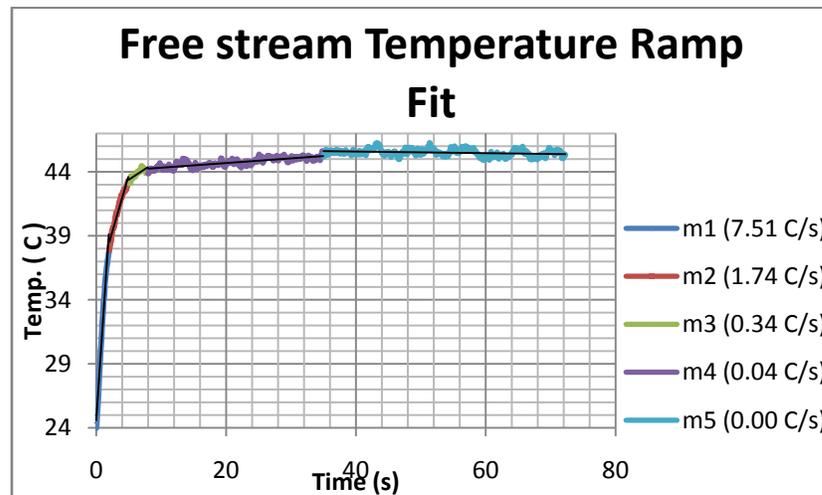
Figure 4.33 Ramp Segments and Slopes: free stream thermocouple data was divided into five ramps.

The free stream response was divided into four segments: the initial quick temperature rise, the "knee," and two sections following with a shallow rise to the asymptotic temperature. It was decided that a single ramp could be used to represent each

section except the "knee"; because this portion of temperature change was rapid, it was decided two ramps were needed in this region. Table 4.13 contains a summary of the ramp values and their associated transition times.

Table 4.13 Ramp Parameter Summary: A summary of ramp transition times and ramp slope used to start the inverse method.

i	m	t
1	7.51	2
2	1.74	5
3	0.34	8
4	0.04	34
5	0.00	72

To calculate the ramp slopes, a linear curve-fit was applied to each section of data. In Figure 4.33, the ramps are not continuous; the end point of one ramp was not required to be at the same point as the start point of the next ramp. Therefore, the ramp slopes calculated with this method were used only as guidelines. They were used as the starting guesses in the inverse method algorithm and as "ballpark" figures. Because of discontinuities, when placed "end to end" continuously, the ramps as stated would cause the fluid temperature to rise significantly past fluid data temperature.

This method calculated a slight negative slope for the final, asymptotic ramp. Since cooling of the wind tunnel air was not possible during an experimental trial, this slope was attributed to thermocouple noise. This slope value was ignored, and a ramp slope of 0 °C/s was used.

Initial equilibrium temperature of stream and plate was taken as the average of the four thermocouples over the time before the heater was turned on, resulting in $T_0 = 24.3$ °C.

5   RESULTS

Experimental data was analyzed with the inverse method in four steps:

1. Experimental data was used with a step change surface convection model to determine if two parameters, h and a constant T_∞, could be estimated with the conjugate gradient algorithm.

2. Generated data without noise was used with a series of ramps convection model to determine if five parameters, 4 slopes of T_∞ and h, could be estimated with the algorithm.

3. Generated data with Gaussian noise was used with the series of ramps to see how error effected parameter estimation compared with "perfect" data.

4. Experimental data was used with the series of ramps convection model (5 parameters) to determine how well the algorithm estimated parameters from real-world data.

To analyze the results of the inverse method, Nusselt numbers were first calculated from established correlations so heat transfer coefficient, h, values could be compared. The fluid temperature profile, $T_\infty(t)$, was compared against the free stream thermocouple data.

The initial inverse analysis using the TLC temperature data in the conjugate gradient inverse method algorithm employs a T_∞ step change surface convection model:

$$T_s = T_0 + (T_\infty - T_0)[1 - \exp(\beta^2)\mathrm{erfc}(\beta)] \tag{5.1}$$

$$\beta = \frac{h\sqrt{\alpha t}}{k}$$

This was a two parameter estimation problem (h and T_∞). This was done to verify the

data produced reasonable asymptotic values for T_∞ and h. Next, fictitious surface

temperature data was produced using the four-ramp (i=4) T_∞ surface response model:

$$T_s - T_0 = m_1 t \cdot g(\beta_0) + \sum_{i=1}^{n-1} [H(t - t_i)m_{i+1}(t - t_i)g(\beta_i) - m_i(t - t_i)g(\beta_i)]$$

$$H(\tau) = \begin{cases} 0 & \tau < 0 \\ 1 & \tau \geq 0 \end{cases} \tag{5.2}$$

$$\beta_i = \frac{h}{k}\sqrt{\alpha(t - t_i)}$$

$$g(\beta_i) = 1 - \frac{2}{\sqrt{\pi}\beta_i} + \frac{1 - \exp(\beta_i^2) \cdot \text{erfc}(\beta_i)}{\beta_i^2}$$

with five parameters (m_1, m_2, m_3, m_4, h). At first, no noise was added to this fictitious

data to confirm functionality of the inverse method. Then, Gaussian noise on the order of

the hue derived temperature standard deviation was added to see how this affected

results.

Next, hue temperature data at each of the three plate positions was used in the

five-parameter inverse method to determine if the algorithm would converge to a

solution, and how good that solution was. If a solution was not achieved or was

inaccurate, the same method was tried with fewer ramps.

5.1 Nusselt Correlations and Expected *h*

Table 4.6 in Section 4.5 contains a summary of Nusselt number correlations for a

flat plate. Using the Reynolds numbers of each position, distances from the boundary

layer start position, and $k = 0.0263 \frac{W}{m-K}$ (air), *h* values were calculated under laminar

and turbulent conditions. Both constant surface temperature and constant heat flux

conditions were calculated for comparison.

	Pos. 1	Pos. 2	Pos. 3
x in. (m)	1/8" (3.175E-03)	3 1/8" (7.9E-02)	6 1/8" (0.156)
Re	930.66	23156.63	45727.01
h W/m^2-K (T_s = const, lam)	72.49	14.53	10.34
h W/m^2-K (q'' = const, lam)	101.98	20.44	14.55
h W/m^2-K (T_s = const, turb)	51.8	27.2	23.7
h W/m^2-K (q'' = const, turb)	53.9	28.3	24.7

From Section 4.5.4, it was indicated that the true h values for this study would be

closer to the h values calculated from the T_s = const. boundary condition since the plate was

made of a low thermal conductance material.

5.2   Position 1 Analysis

5.2.1   Step Change Model

Hue temperature data was taken from Position 1 and used as the input to the step

change surface convection model conjugate gradient algorithm. This was done to verify

the data, give a good estimation of asymptotic T_∞, and a reasonable estimation of h.

Figure 5.1 shows the inverse method curve (blue) fit to the data (green) and the

corresponding estimated fluid temperature (red). It can be seen the fit is best at later

times, but diverges from the data at earlier times, from 2 to 10 seconds. Against the fluid

thermocouple reading of 45.17 °C, the fit estimated T_∞ to be 45.00 °C, a difference of

only 0.8% of ΔT_∞. Heat transfer coefficient h returned by the inverse method was

estimated to be 82.03 $\frac{W}{m^2 K}$, which is within the range of 72.49 to 102.98 $\frac{W}{m^2 K}$ predicted by

the Nusselt correlations, being closer to the constant surface temperature correlation.

Temperature Curve Fit
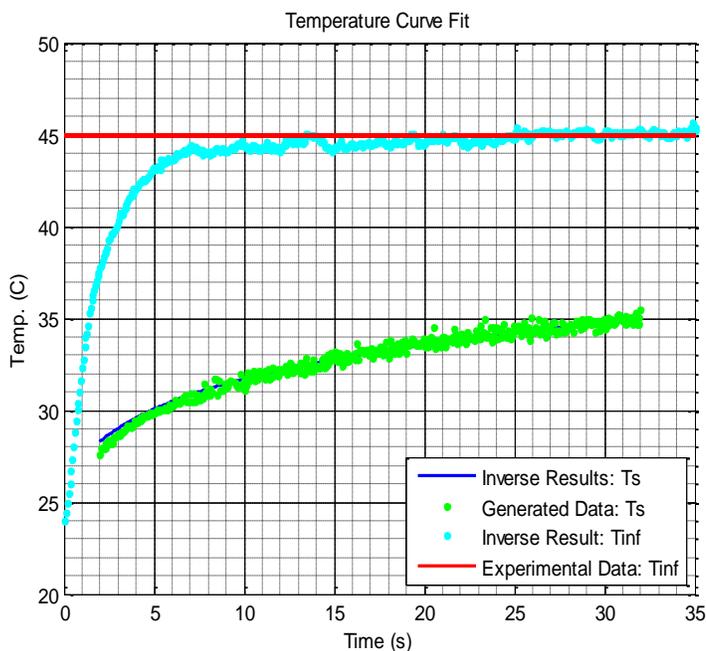
	"True" Parameter	Inverse Result
$T_\infty=$ (°C)	45.17	45.00
$h=$ (W/m-K)	72.5-102	82.03

Figure 5.1 Parameter Fit for Step Function: The step function model fits the data fairly well except near the initial rise in surface temperature, from 2 seconds to 10 seconds.

Figure 5.2 shows the convergence of the solution. The blue line represents average (RMS) error of the fit. The convergence is slow and even, reaching a minimum at 2944 iterations. The correlation curve indicates the convergence of the correlation between residuals and time. The ideal correlation value is 0, indicating no relationship between error and time. This would support the validity of the surface temperature response model used to fit the data. Here it can be seen correlation increasing with time to a value of 0.44. This indicates a flaw in the model, arising from the fact that the fluid temperature that produced the data did not take the form of a perfect step change. The gradient line shows the convergence of the objective function gradient with respect to the parameters T_∞ and h.
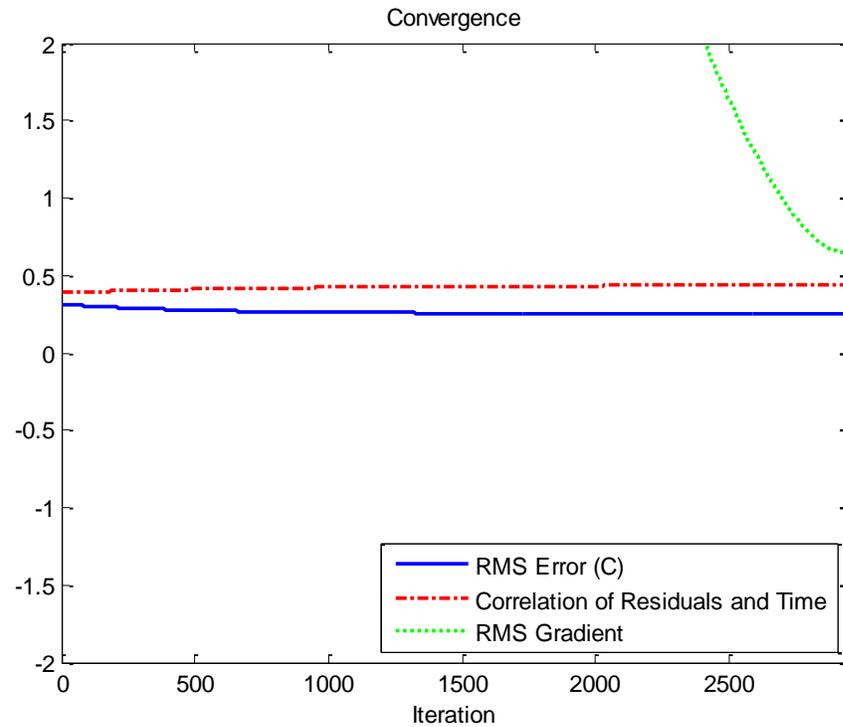
Figure 5.2 Convergence: Convergence criteria were plotted versus iteration.

Figure 5.3 shows the convergence of the chi-squared goodness of fit criterion. This value is the ratio of fit uncertainty to data uncertainty. For a perfect model and perfect fit, chi-squared would be one. The chi-squared converged value of 1.68 confirms that another model may fit the data better.

Chi-Squared Goodness of Fit
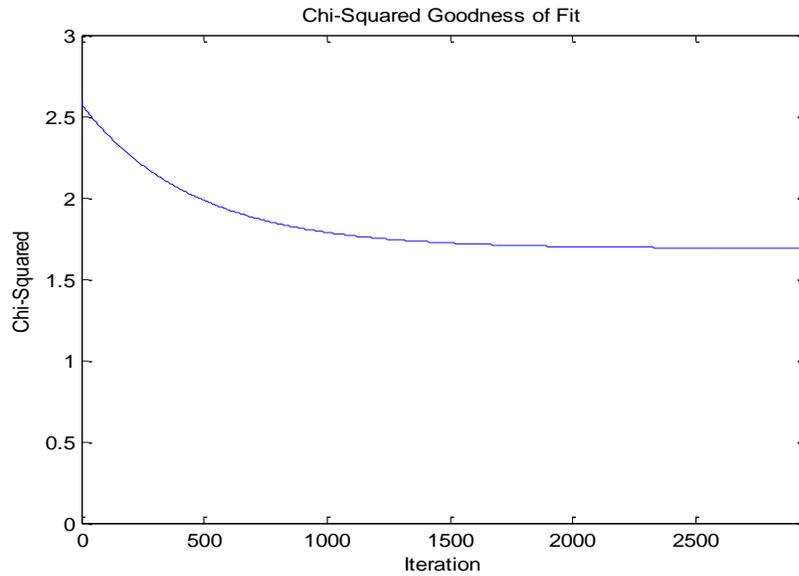
Figure 5.3 Convergence: Convergence of chi-squared goodness of fit was plotted versus iteration.

Figure 5.4 is a plot of the residuals of the final iteration versus time. This gives information about where the data fits best, and any "trending" in the residuals indicates a poor model. This figure shows little trending of the data past 7 seconds, indicating the model fits the data well. However, the figure confirms that another model may fit the data better in the region up to 7 seconds where clear trending is visible.
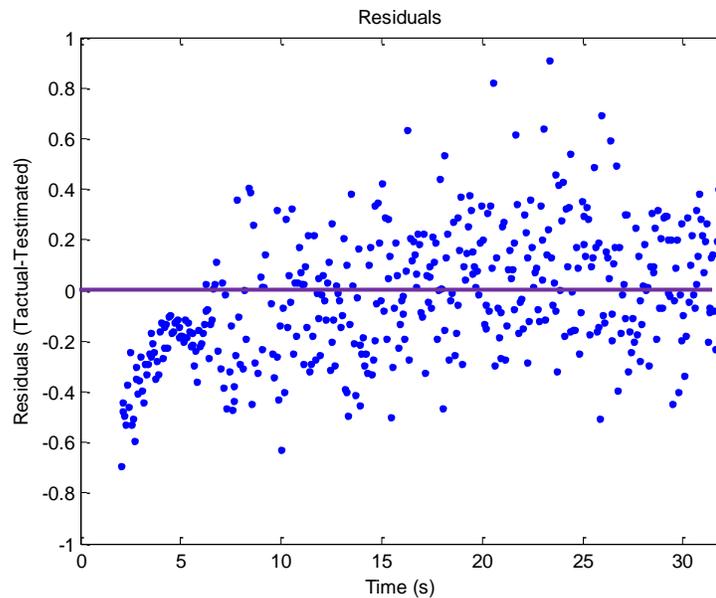
Residuals

Figure 5.4 Residuals: residuals were plotted against time for the converged solution.

Summaries of the estimated parameters and solution evaluation criteria are found in

Table 5.1. Also, the solution's 95% confidence interval is shown.

Table 5.1 Solution and Evaluation Summary

Solution and Evaluation			
Iterations			
2944			
Final RMS Error ($^\circ$C)			
0.2535			
Data Correlation			
0.4395			
Chi-Squared			
1.685			
	Result Value	Standard Deviation	
T_∞ (C)	45.00	σ_T ($^\circ$C)	0.4926
h (W/m2-K)	82.03	σ_h (W/m2-K)	3.1646
95% Confidence Interval			
44.03	$\leq$	T_∞ ($^\circ$C) $\qquad \leq$	45.97
75.83	$\leq$	h(W/m2-K) $\qquad \leq$	88.23

5.2.2 Generated Data

Data was generated from the series of steps surface convection model (4 ramps)

to test the conjugate gradient inverse algorithm for accuracy, stability, and convergence.

Since exact parameter values are known with user-generated data, accuracy can be

verified. In each case, starting guesses for parameters were held constant between trials,

being 10%-20% of their true values.

Convergence was assumed if error was minimized and gradient became small for

an extended number of iterations. The algorithm was allowed to run for a set number of

iterations, and then checked for convergence. If converged, the iteration parameter set

that produced the minimum error was used as the solution.

5.2.2.1   <u>No Noise, Full Data Set</u>

The first trial of generated data contained no noise, and represented a length of data

from 0 to 32 seconds. Actual Position 1 data represented 2 to 32 seconds. Table 5.2

contains the generated data's exact parameters m_i and h and their associated transition

times.

Table 5.2 Generated Data Parameters and Values: Generated data's parameters were known exactly.

Time	Value (s)	Ramp Slope	Value (°C/s)
t_1	2	m_1	7.51
t_2	5	m_2	1.70
t_3	8	m_3	0.30
t_4	32	m_4	0.04
		h	85

The generated data's parameters were chosen to closely mimic observed hue

temperature data parameters. When this noiseless generated data was used in the inverse

method, the algorithm was able to match the parameters exactly, as seen in Figure 5.5.

The algorithm converged within 1000 iterations, producing error and gradient both under

10^{-9}. For noiseless data, values for chi-squared, correlation, and residuals are

meaningless as the data had a standard deviation of 0.
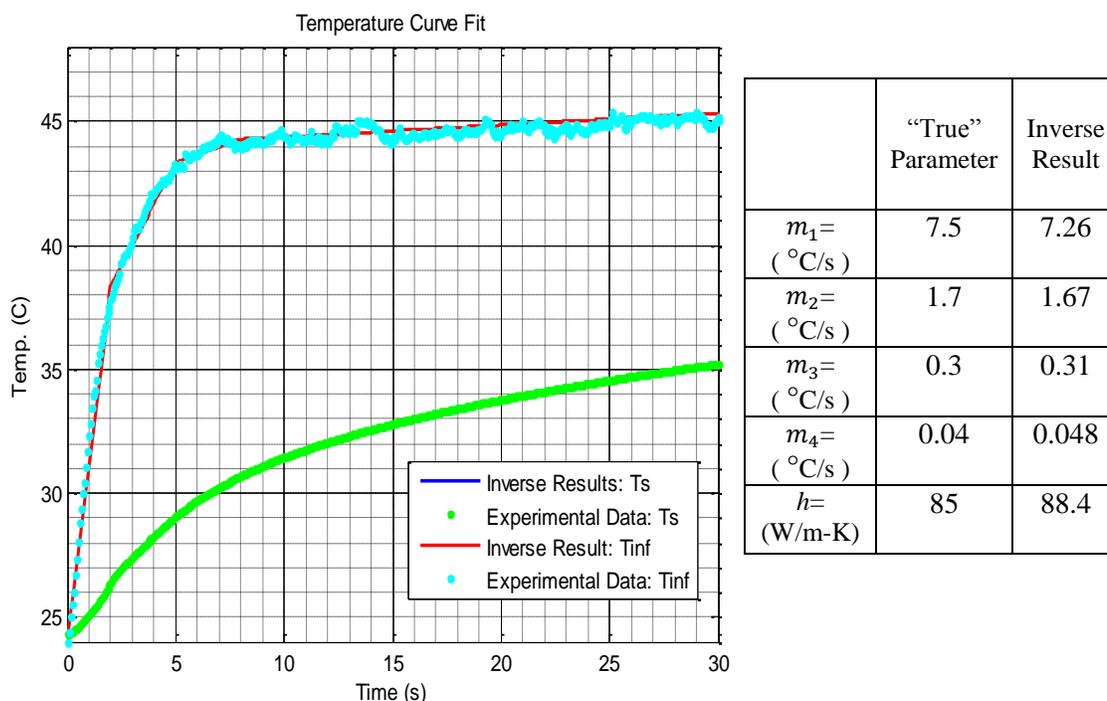
Figure 5.5 Generated Noiseless Data Result: The Estimated curve (inverse method results) matched the Data (generated) to three decimal places.

5.2.2.2 Noise, Full Data Set

After it was confirmed the algorithm worked with noiseless data, Gaussian noise was added with the same variance of the observed hue temperature data's, $0.038\ °C^2$ (0.2 °C standard deviation). Figure 5.6 shows the results of the parameter estimation. With noise added to "perfect" data, the algorithm estimated fluid temperature to be about 1 °C too low, and estimated h to be 99.7 $\frac{W}{m^2 K}$, above the true 85 $\frac{W}{m^2 K}$.

	"True" Parameter	Inverse Result
$m_1=$ ($^{\circ}$C/s)	7.5	7.0
$m_2=$ ($^{\circ}$C/s)	1.7	1.51
$m_3=$ ($^{\circ}$C/s)	0.3	0.36
$m_4=$ ($^{\circ}$C/s)	0.04	0.05
$h=$ (W/m-K)	85	94.47
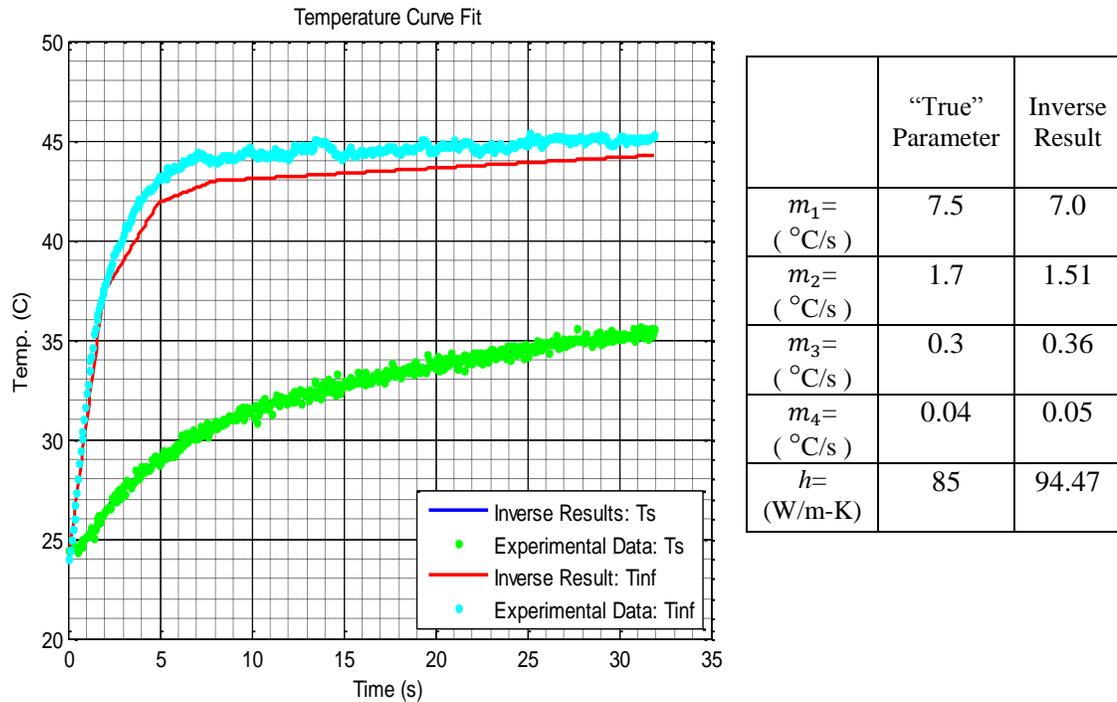
Figure 5.6 Generated Noisy Data Result: The algorithm estimated fluid temperature low and *h* high. Inverse results for T_s lies beneath the experimental data.

For convergence, chi-squared converged to 1 very quickly, within a few hundred iterations. Figure 5.7 shows other convergence criteria. RMS error converged to 0.2 °C within a few iterations. Correlation had a few oscillations then converged to a value of 0.02, indicating a slight correlation of temperature and time. This indicates a nominal amount of noise will cause a slight correlation. The gradient had periods of large variance, eventually dampening out as the algorithm converged to a solution at 2633 iterations. This could indicate a "wandering" of the algorithm when it arrives near the minimum of the objective function. A different conjugation coefficient might increase the efficiency of the algorithm. A plot of the residuals in Figure 5.8 indicates no obvious trend.
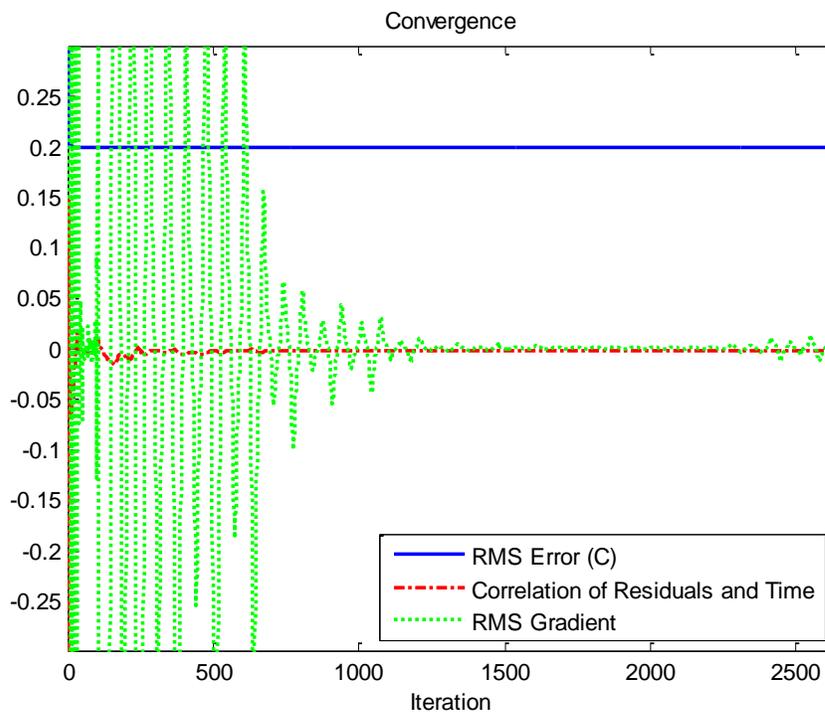
Figure 5.7 Convergence: All criteria are converged by 2633 iterations.
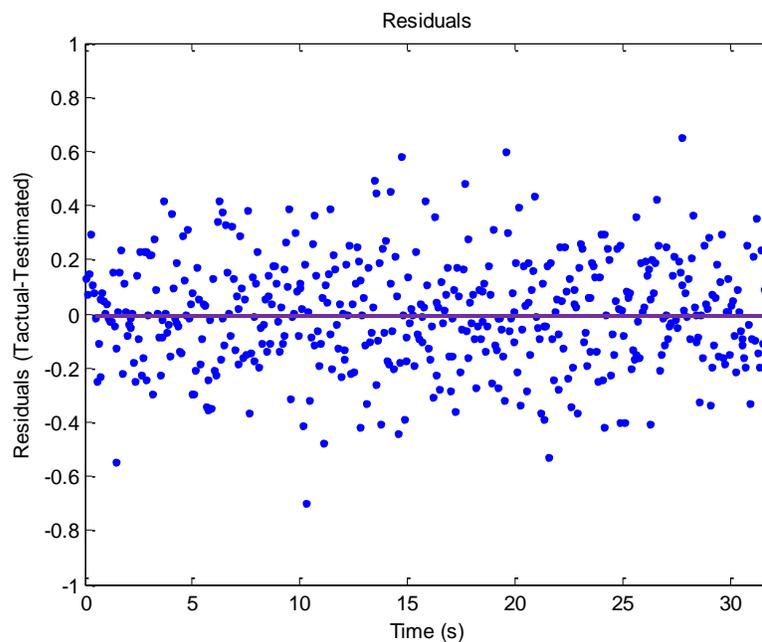
Figure 5.8 Residuals: The residuals for this trial show no obvious trend.

Table 5.4 summarizes the results of this trial. One troubling aspect is the very high standard deviations of the data. Recall the definition of covariance:

$$C_{m=}\sigma_d{}^2 C_d = \sigma_d{}^2 [J^T J]^{-1} \qquad (5.3)$$

Covariance is a function of the Jacobian scaled by the standard deviation of the data, and the values of variance are the diagonal values of C_m. The non-diagonal values indicate the degree of effect one parameter has on another. In the process of inverting the Jacobian product, large off-diagonal values of the matrix before inversion increase the diagonal values of the inverted matrix.

Table 5.4 shows the covariance matrix for this trial. Each non-diagonal value is the cross-covariance of a parameter pair. For example, position (5,1) indicates the covariance of *h* and *m1*. It can be seen that this is the largest covariance present, indicating any variation in *m1* has a great effect on *h*. Indeed, every covariance value of *h* and a slope has a large value. These heavy interdependencies translate into a high *h* uncertainty, as seen in position (5,5) of the covariance matrix. The interdependencies have a cumulative effect, indicating the more slopes used to represent the fluid temperature, the greater the uncertainty in *h* will be.

Table 5.3 Trial Covariance Matrix: large inter-parameter dependence creates high uncertainty in parameter estimation.

	m1	*m2*	*m3*	*m4*	*h*
m1	32.32286	4.321282	-2.77817	-0.74477	-595.089
m2	4.321282	1.071505	-0.62077	-0.10477	-86.2418
m3	-2.77817	-0.62077	0.408568	0.062044	53.86219
m4	-0.74477	-0.10477	0.062044	0.017994	13.82931
h	-595.089	-86.2418	53.86219	13.82931	11059.04

Table 5.4 Results of the Noisy Generated Data Parameter Estimation

Solution and Evaluation			
Iterations			
2633			
Final RMS Error ($^\circ$C)			
0.1990			
Data Correlation			
-0.0031			
Chi-Squared			
1.047			
	Result Value	Standard Deviation	
m_1 ($^\circ$C/s)	7.0	σ_{m1} ($^\circ$C/s)	1.22
m_2 ($^\circ$C/s)	1.51	σ_{m2} ($^\circ$C/s)	0.19
m_3 ($^\circ$C/s)	0.36	σ_{m3} ($^\circ$C/s)	0.12
m_4 ($^\circ$C/s)	0.05	σ_{m4} ($^\circ$C/s)	0.03
h (W/m2-K)	94.47	σ_h (W/m2-K)	19.37
95% Confidence Interval			
4.60	$\leq$	m_1 ($^\circ$C/s) $\leq$	9.38
1.15	$\leq$	m_2 ($^\circ$C/s) $\leq$	1.88
0.13	$\leq$	m_3 ($^\circ$C/s) $\leq$	0.60
0.00	$\leq$	m_4 ($^\circ$C/s) $\leq$	0.11
56.5	$\leq$	h(W/m2-K) $\leq$	132.44

5.2.2.3   Noise, Limited Data Set

Next, it was investigated how removing the first two seconds of data would affect

the results, since the hue temperature trial data was also missing the first two seconds.

It can be seen in Figure 5.9 that the algorithm again estimates a lower fluid

temperature and higher *h* than actual parameter values. When compared with the previous

trial containing the first two seconds of data, this trial does worse, estimating *h* to be

$101.11 \frac{W}{m^2 K}$ versus $99.7 \frac{W}{m^2 K}$, which is closer to the true value of 85. However, the

average error is only 0.0022 °C worse than the previous algorithm.

	"True" Parameter	Inverse Result
$m_1=$ ($^\circ$C/s)	7.5	6.88
$m_2=$ ($^\circ$C/s)	1.7	1.47
$m_3=$ ($^\circ$C/s)	0.3	0.38
$m_4=$ ($^\circ$C/s)	0.04	0.06
$h=$ (W/m-K)	85	96.7
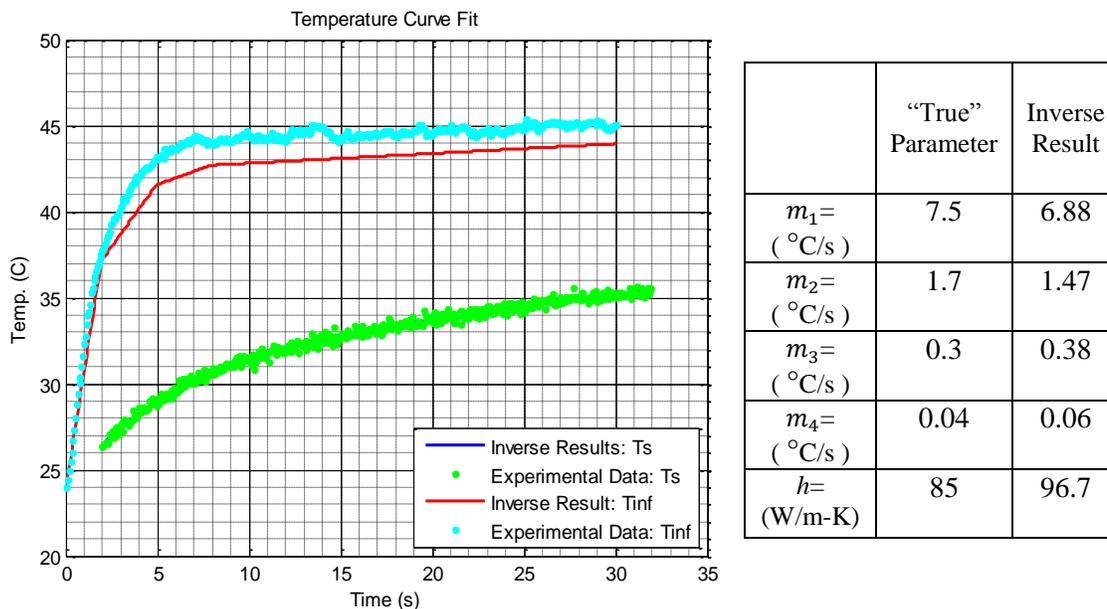
Figure 5.9 Generated Data Result: The algorithm estimates fluid temperature slightly high and *h* low.

Figure 5.10 shows that convergence is similar to the preceding trial, with correlation actually being better. Figure 5.11 shows chi-squared converged to just above 1, indicating an appropriate fit.
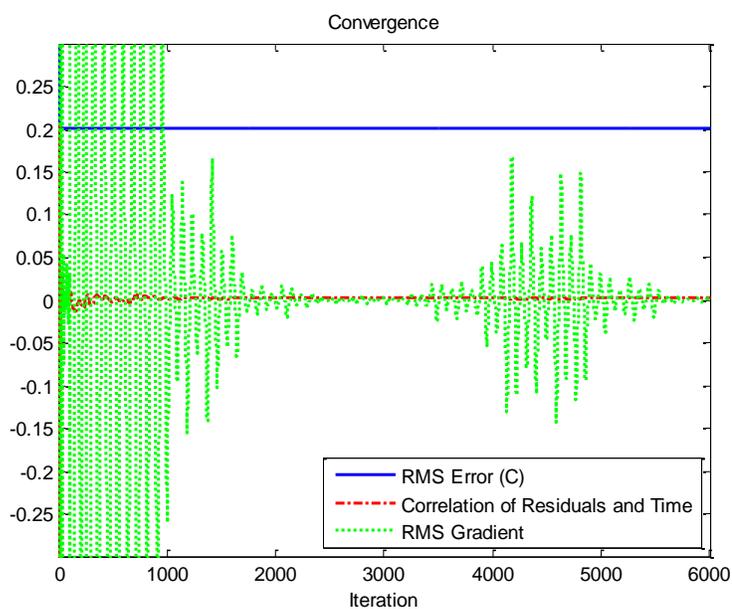
Figure 5.10 Convergence: The algorithm converged after 4980 iterations to similar values as other generated data trials.
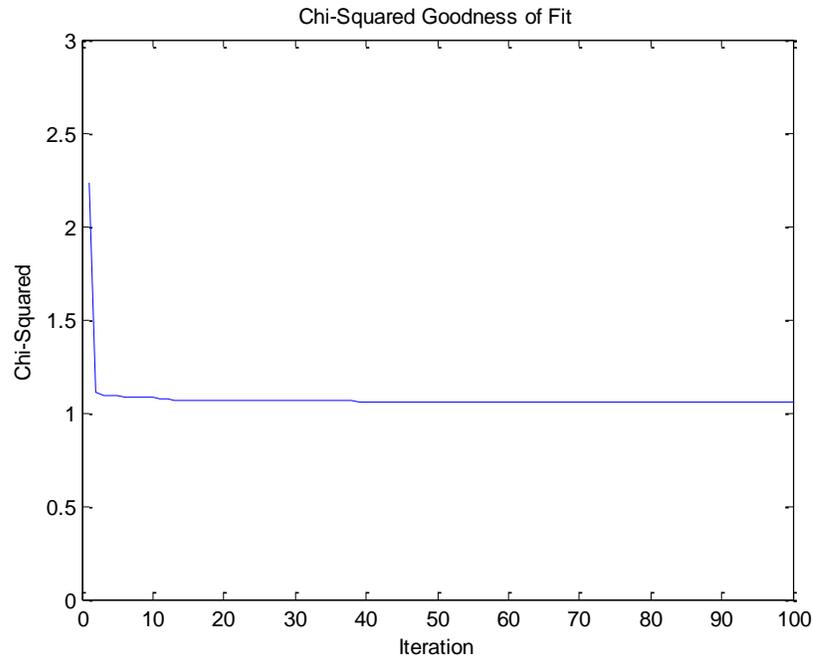
Figure 5.11 Chi-Squared: Chi-squared converged to just above 1, indicating a good fit.

In Figure 5.12, no trending can be seen visually in the residuals.
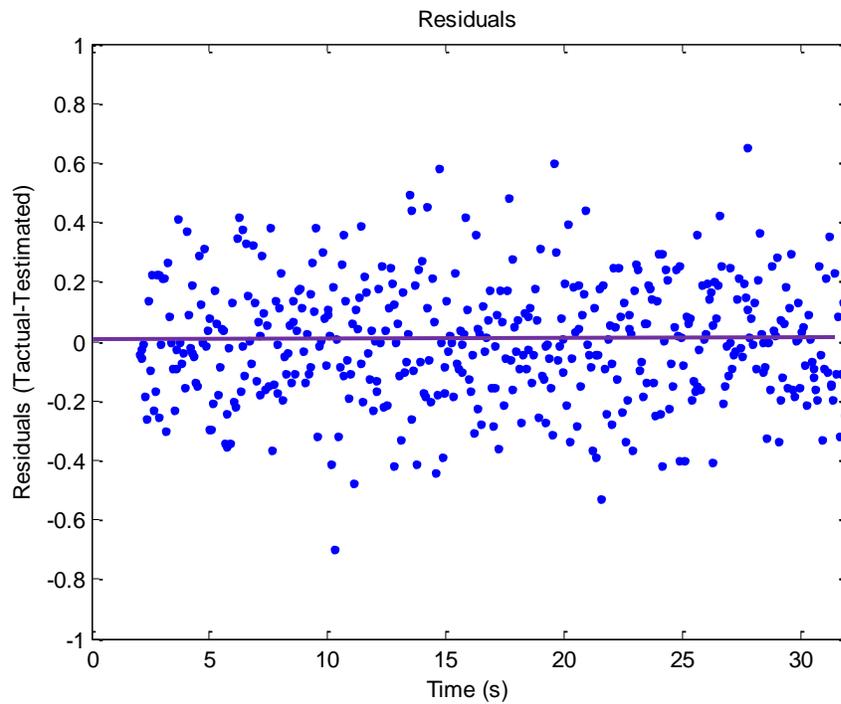
Figure 5.12 Residuals: no trending is evident.

The table below summarizes the results of this trial. Standard deviations and confidence intervals are similar to the previous data trial.

Table 5.5 Results of the Noisy Limited Data Set Generated Data Parameter Estimation

Solution and Evaluation			
Iterations			
4983			
Final RMS Error ($^\circ$C)			
0.20			
Data Correlation			
0.0018			
Chi-Squared			
1.069			
	Result Value	Standard Deviation	
m_1 ($^\circ$C/s)	6.88	σ_{m1} ($^\circ$C/s)	1.19
m_2 ($^\circ$C/s)	1.47	σ_{m2} ($^\circ$C/s)	0.19
m_3 ($^\circ$C/s)	0.38	σ_{m3} ($^\circ$C/s)	0.12
m_4 ($^\circ$C/s)	0.06	σ_{m4} ($^\circ$C/s)	0.03
h (W/m2-K)	96.69	σ_h (W/m2-K)	19.94
95% Confidence Interval			
4.53	$\leq$	m_1 ($^\circ$C/s) $\leq$	9.22
1.09	$\leq$	m_2 ($^\circ$C/s) $\leq$	1.85
0.15	$\leq$	m_3 ($^\circ$C/s) $\leq$	0.62
0.00	$\leq$	m_4 ($^\circ$C/s) $\leq$	0.11
57.62	$\leq$	h(W/m2-K) $\leq$	135.77

5.2.3 Hue Temperature Data

After it was investigated how well the conjugate gradient functioned with generated data with noise added, actual hue temperature trial data was used to estimate parameters. Like the generated data, the hue temperature data from Position 1 covered times from 2 to 32 seconds.

When using actual experimental data, the algorithm converges to an incorrect solution, as seen in Figure 5.13. The first ramp is very large, causing estimated free-

stream temperature to rise to 64 °C. The second ramp is negative, while there was no

cooling in the experimental trial. Heat transfer coefficient h was estimated to be $43 \frac{W}{m^2 K}$,

half of its likely true value. The results of the generated data with noise trials indicate that

the method is highly sensitive to variations in data.

	"True" Parameter	Inverse Result
$m_1=$ (°C/s)	7.5	21.22
$m_2=$ (°C/s)	1.7	-2.57
$m_3=$ (°C/s)	0.3	-0.34
$m_4=$ (°C/s)	0.04	-0.15
$h=$ (W/m-K)	72.5-102	42.93
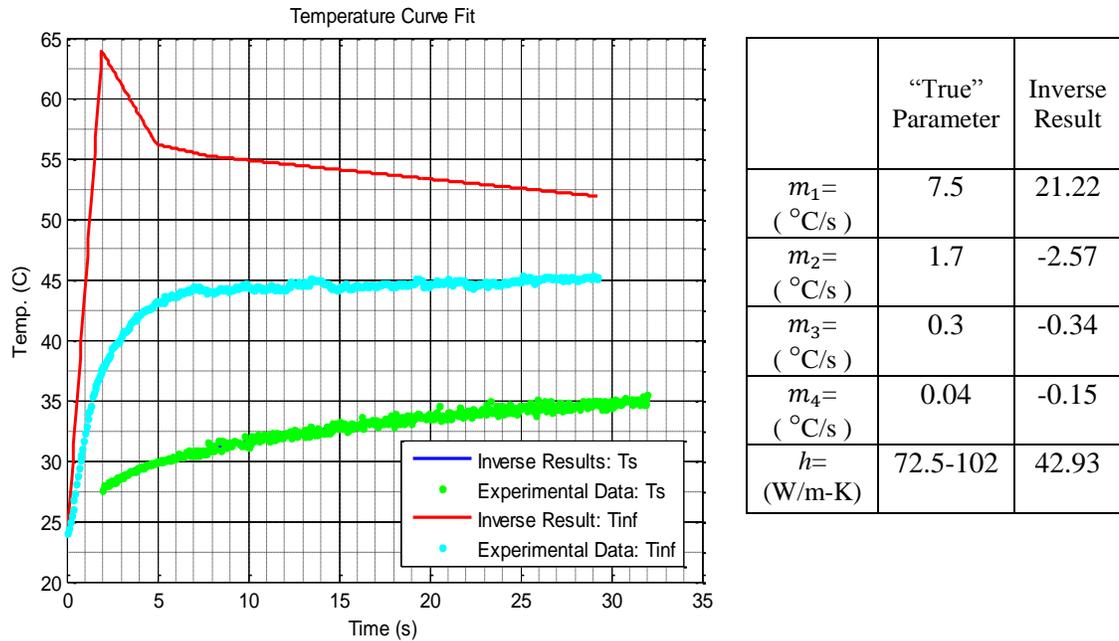
Figure 5.13 Experimental Trial Data Results: Parameters estimated were inaccurate.

While the parameters estimated by the algorithm were highly inaccurate, they

were the parameters that minimized the objective function, as shown by the convergence

criteria in Figure 5.14. Error converged to 0.22 °C, slightly above the standard deviation

of the hue temperature data. Correlation converged to -0.0045, also a good value for

experimental data. Gradient converged to $1.93 \cdot 10^{-6}$, indicating a minimized objective

function. Also, chi-squared indicated a good data fit in Figure 5.15, though slightly

higher than generated data.
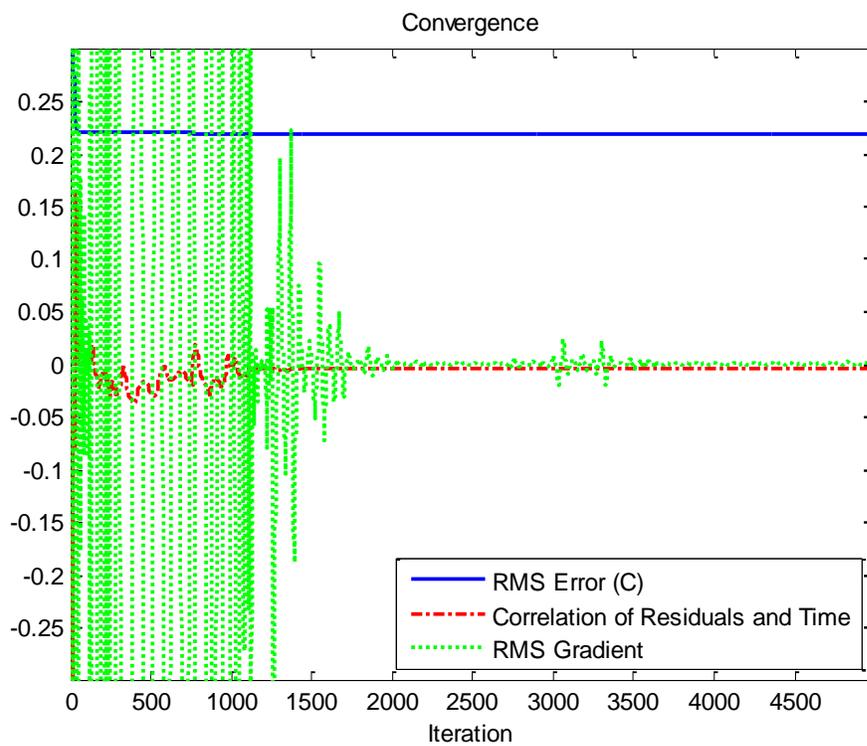
Figure 5.14 Convergence: All convergence criteria converged.
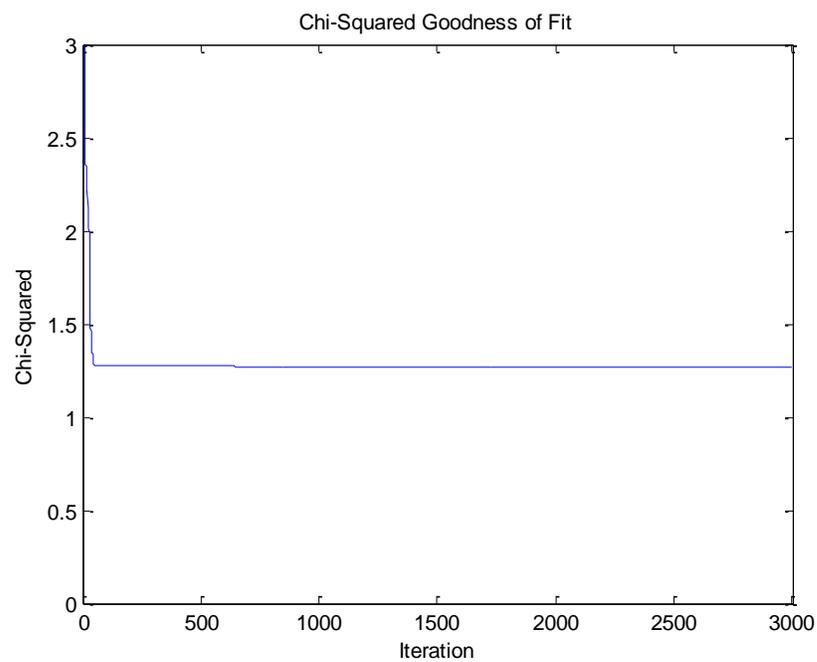
Figure 5.15 Chi-Squared: Chi-squared indicated fit error was slightly greater than data variance.

The plot of residuals versus time also indicated no trending of the data. The residuals around 5 seconds are visibly tighter as a result of less variance in hue data over that time.
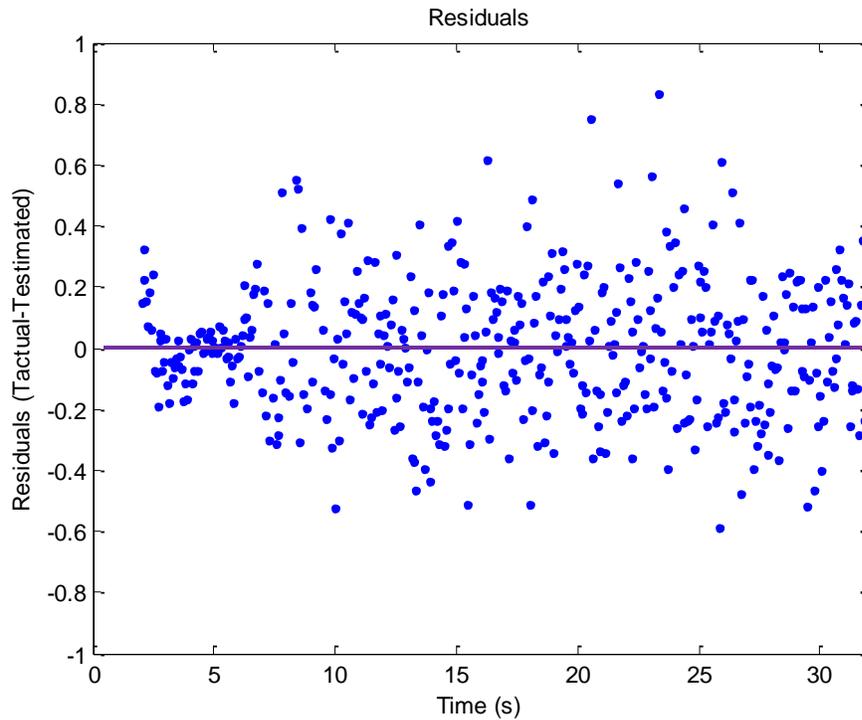
Figure 5.16 Residuals: While earlier data had less deviation, there was still no obvious trending of the data.

Table 5.6 summarizes the results of this trial. Again, standard deviations were high, indicating high parameter interdependence.

Table 5.6 Results of the Noisy Limited Data Set Generated Data Parameter Estimation

Solution and Evaluation			
Iterations			
4980			
Final RMS Error (°C)			
0.22			
Data Correlation			
-0.0045			
Chi-Squared			
1.269			
	Result Value	Standard Deviation	
m_1 (°C/s)	21.22	σ_{m1} (°C/s)	8.23
m_2 (°C/s)	-2.57	σ_{m2} (°C/s)	1.39
m_3 (°C/s)	-0.34	σ_{m3} (°C/s)	0.43
m_4 (°C/s)	-0.15	σ_{m4} (°C/s)	0.13
h (W/m2-K)	42.93	σ_h (W/m2-K)	18.08
95% Confidence Interval			
5.08	$\leq$ m_1 (°C/s) $\leq$		37.36
-5.30	$\leq$ m_2 (°C/s) $\leq$		0.16
-1.18	$\leq$ m_3 (°C/s) $\leq$		0.51
-0.48	$\leq$ m_4 (°C/s) $\leq$		0.11
7.5	$\leq$ h(W/m2-K) $\leq$		78.37

5.3   Position 2 and 3 Analysis

Position 2 was also analyzed initially with a step change model. The results are

shown in Figure 5.17. Position 2 covered a greater and later time span than Position 1:

15-45 seconds. The inverse method returned a fluid temperature of 46.96 °C and an *h* of

31.31 $\frac{W}{m^2 K}$. Nusselt correlations for laminar flow produce an *h* between 14 and 20 $\frac{W}{m^2 K}$.
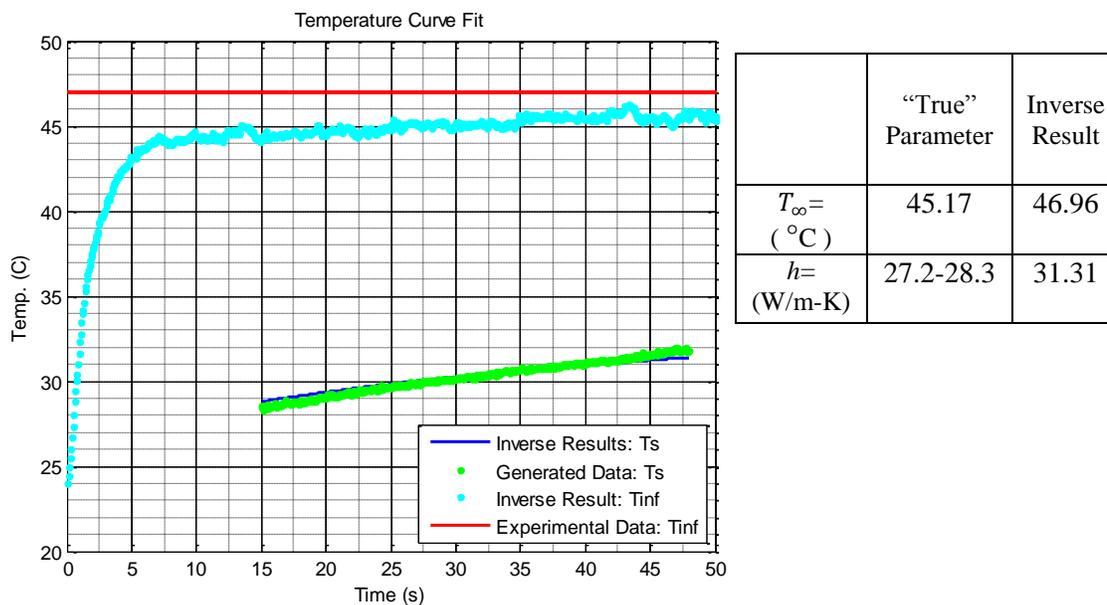
Figure 5.17 Position 2 Step Change Solution: The estimated fit had problems matching the slope of the data in this nearly linear region.

Figure 5.18 shows the convergence of the algorithm. Over a period of 5000 iterations, error and gradient were minimized at 188 iterations. The residuals proved to have a high degree of correlation, again suggesting the step change model is flawed for measured surface temperature response.
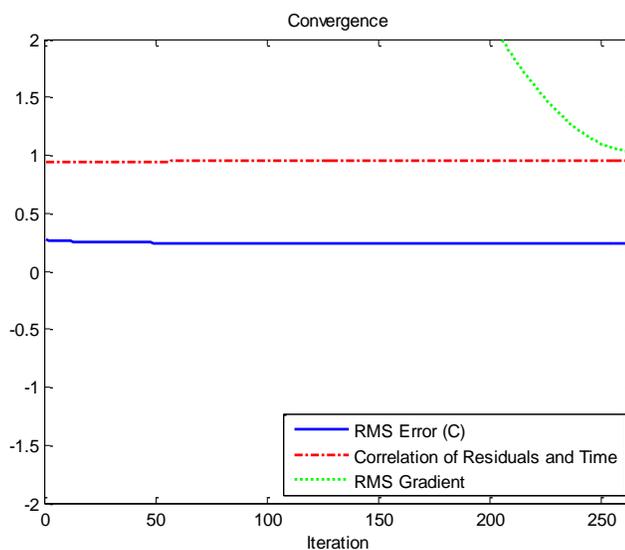
Figure 5.18 Position 2 Convergence for Step Change: The algorithm reached its best solution at 188 iterations.

Figure 5.19 corroborates the high correlation number, showing clear trending in the residuals.
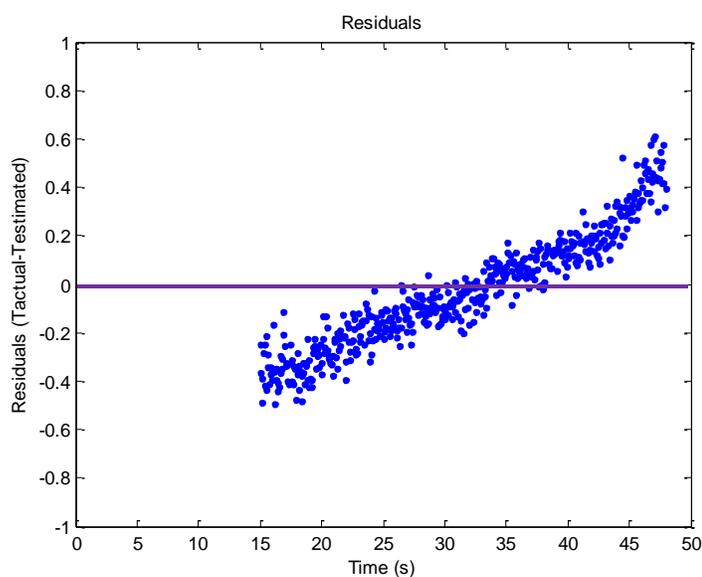
Figure 5.19 Position 2 Residuals for Step Change: Clear trending is present in the plot.

Despite the poor fit of the data, the confidence intervals in Table 5.7 are reasonable.

Table 5.7 Position 2 Step Change Results Summary

Solution and Evaluation			
Iterations			
266			
Final RMS Error ($^{\circ}$C)			
0.294			
Data Correlation			
0.894			
Chi-Squared			
1.465			
	Result Value	Standard Deviation	
T_{∞} ($^{\circ}$C)	46.96	σ_T ($^{\circ}$C)	2.1
h (W/m2-K)	31.31	σ_h (W/m2-K)	3.7
95% Confidence Interval			
42.9	$\leq$	T_{∞} ($^{\circ}$C) $\leq$	51.1
23.7	$\leq$	h(W/m2-K) $\leq$	38.3

Position 2 data was then used in the series of ramps algorithm. The results can be found below in Table 5.2. The algorithm fit the data well with an average error of 0.066, which is smaller than the standard deviation of the data as the chi-squared value of 0.11 shows.

	"True" Parameter	Inverse Result
$m_1=$ ($^\circ$C/s)	7.5	8.6
$m_2=$ ($^\circ$C/s)	1.7	-1.0
$m_3=$ ($^\circ$C/s)	0.3	10.2
$m_4=$ ($^\circ$C/s)	0.04	1.1
$h=$ (W/m-K)	27.2-28.3	34.2
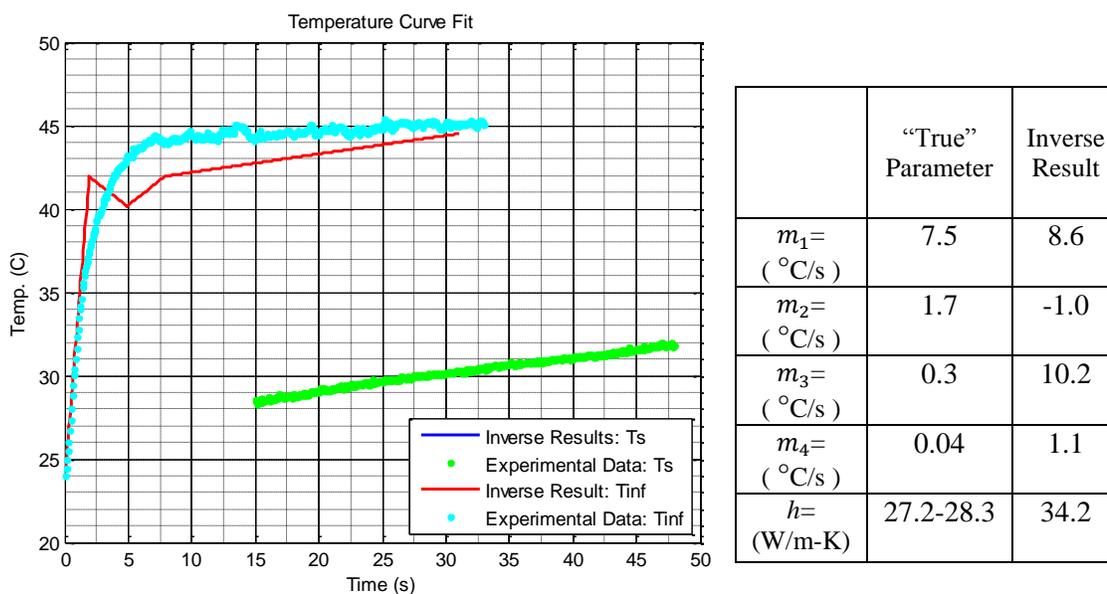
Figure 5.20 Position 2 Ramps Solution: Initial ramp estimation is poor.

Convergence occurs after 456 iterations as seen in Figure 5.21 Position 2 Ramps Convergence, though the algorithm ran for much longer. The convergence was good, with all three criteria reaching minimal values.
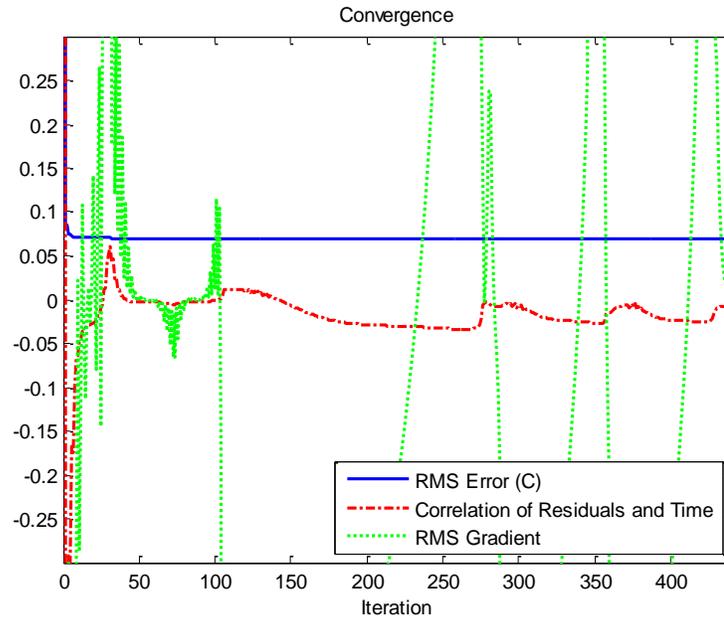
Figure 5.21 Position 2 Ramps Convergence: The algorithm converged after 456 iterations.

Figure 5.22 shows little trending, with some visible in the last two seconds of data as the residual values are primarily above 0 in this region.
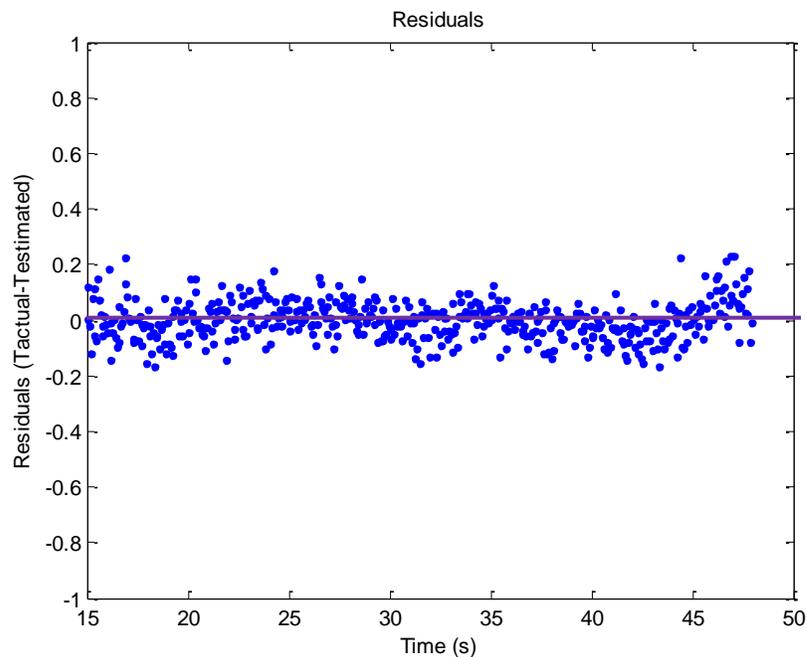
Figure 5.22 Position 2 Ramps Residuals: Little trending is visible except in the last few seconds.

Table 5.8 summarizes the results of this trial. Standard deviations are very large, as was the case with all series of ramps solution data.

Table 5.8 Position 2 Ramps Solution Summary

	Solution and Evaluation		
	Iterations		
	456		
	Final RMS Error (°C)		
	0.0657		
	Data Correlation		
	-0.0053		
	Chi-Squared		
	0.114		
	Result Value	Standard Deviation	
m_1 (°C/s)	8.60	σ_{m1} (°C/s)	1477
m_2 (°C/s)	-1.01	σ_{m2} (°C/s)	1343
m_3 (°C/s)	10.23	σ_{m3} (°C/s)	511
m_4 (°C/s)	1.05	σ_{m4} (°C/s)	0.6
h (W/m2-K)	34.2	σ_h (W/m2-K)	1026
	95% Confidence Interval		
-2886	$\leq$	m_1 (°C/s) $\leq$	2903
-2633	$\leq$	m_2 (°C/s) $\leq$	2632
-1000	$\leq$	m_3 (°C/s) $\leq$	1002
-1.1	$\leq$	m_4 (°C/s) $\leq$	1.2
-1976	$\leq$	h(W/m2-K) $\leq$	2044

Position 3 was analyzed in the same manner as Positions 1 and 2. The step change solution for Position 3 was very similar to Position 2 (see Figure 5.23), having the same trending and convergence profile. The algorithm produced a fluid temperature of 46.15 °C, slightly lower than Position 2. The *h* estimated was 27.6 $\frac{W}{m^2 K}$, again higher than the laminar Nusselt correlation range of 10-14, but close to the turbulent correlation range of 23-24.

	"True" Parameter	Inverse Result
$T_\infty=$ ($^\circ$C)	45.17	46.15
$h=$ (W/m-K)	23.2-24.7	27.6
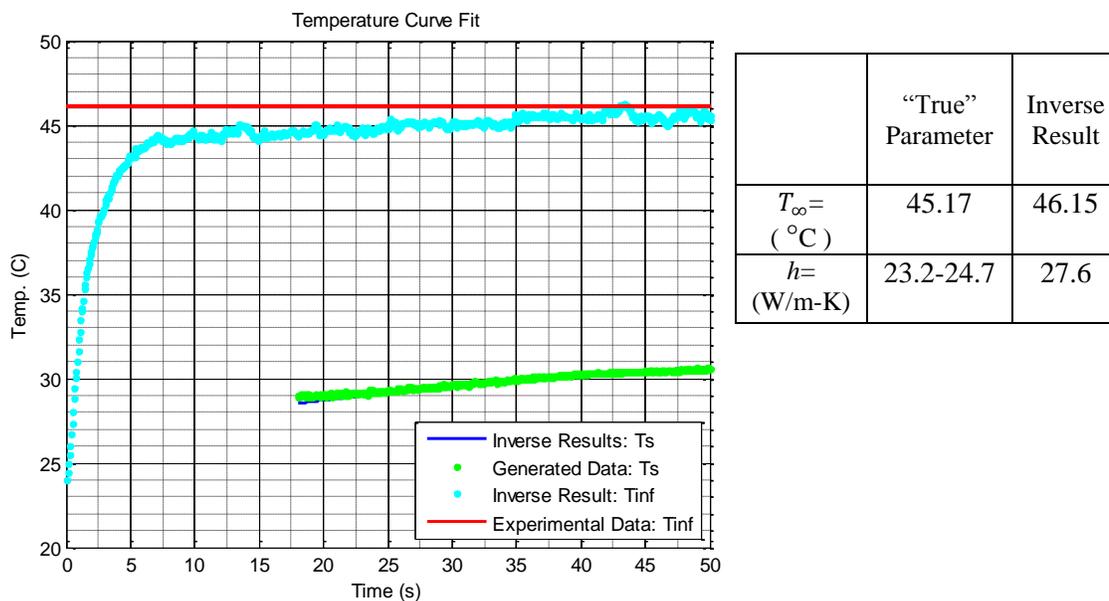
Figure 5.23 Position 3 Step Change Solution: The solution quality was similar to that of Position 2.

Convergence was also similar to Position 2, as seen in Figure 5.24. Also, Figure

5.25 shows clear trending.
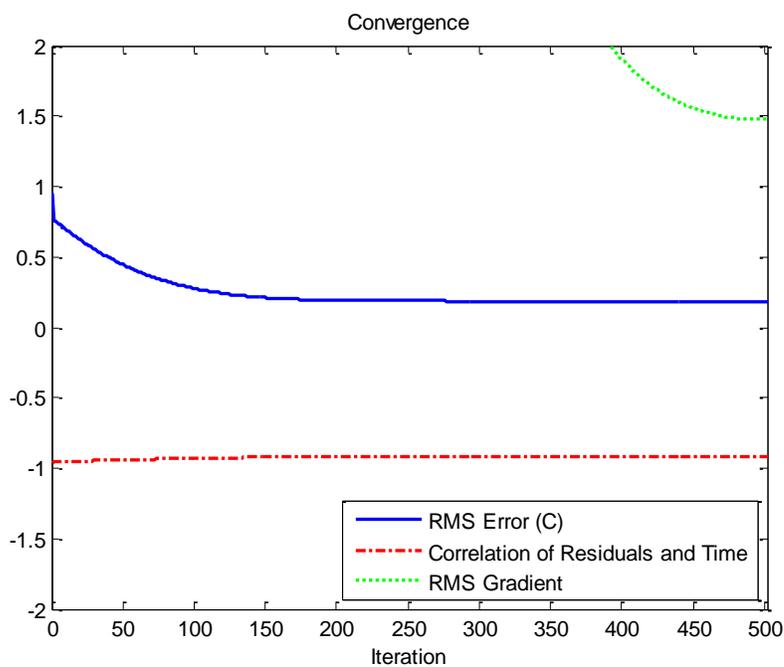
Figure 5.24 Position 3 Step Change Convergence: Convergence was similar to Position 2.
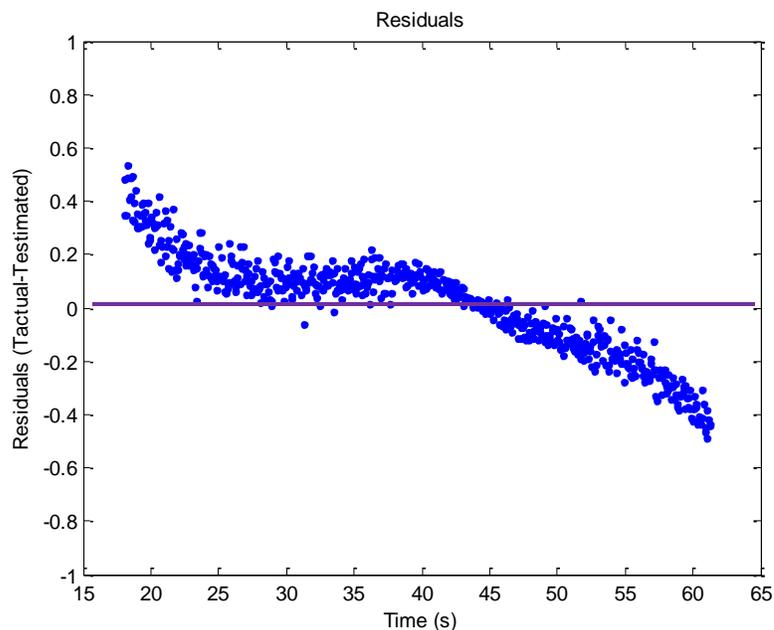
Figure 5.25 Position 3 Step Change Residuals: Clear trending can be seen.

Table 5.9 contains a summary of the solution.

Table 5.9 Position 3 Step Change Solution Summary

Solution and Evaluation			
Iterations			
501			
Final RMS Error (°C)			
0.267			
Data Correlation			
-0.947			
Chi-Squared			
0.8436			
	Result Value	Standard Deviation	
T_∞ (°C)	46.15	σ_T (°C)	1.77
h (W/m2-K)	27.6	σ_h (W/m2-K)	2.90
95% Confidence Interval			
41.59	$\leq$ T_∞ (°C) $\leq$		50.72
20.14	$\leq$ h(W/m2-K) $\leq$		33.30

Position 3 data was then used in the series of ramps algorithm. The results can be

found below in Figure 5.26. The algorithm fit the data well with an RMS error of 0.07,

which is smaller than the standard deviation of the data as the chi-squared value of 0.13

shows. The resulting ramp profile of the fluid temperature is still unreasonable, similar to

Position 1. An estimated h of $26.6 \frac{W}{m^2 K}$ was returned, similar to that found by the step

change solution.

	"True" Parameter	Inverse Result
$m_1=$ ($^\circ$C/s)	7.5	10.8
$m_2=$ ($^\circ$C/s)	1.7	3.80
$m_3=$ ($^\circ$C/s)	0.3	-2.9
$m_4=$ ($^\circ$C/s)	0.04	-0.12
$h=$ (W/m-K)	23.7-24.7	26.6
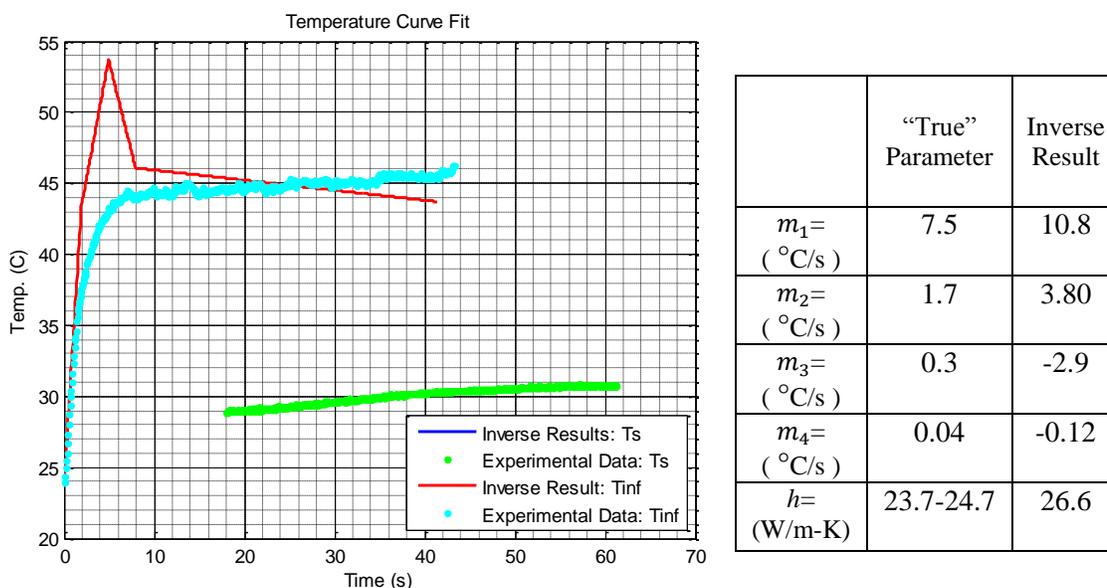
Figure 5.26 Position 3 Ramps Solution: Initial ramp estimation was poor, as with other positions.

Convergence occurs after 143 iterations, at which point the parameters were

taken, as seen in Figure 5.27, though the algorithm ran for much longer. The convergence

was good, with all three criteria reaching minimal values.
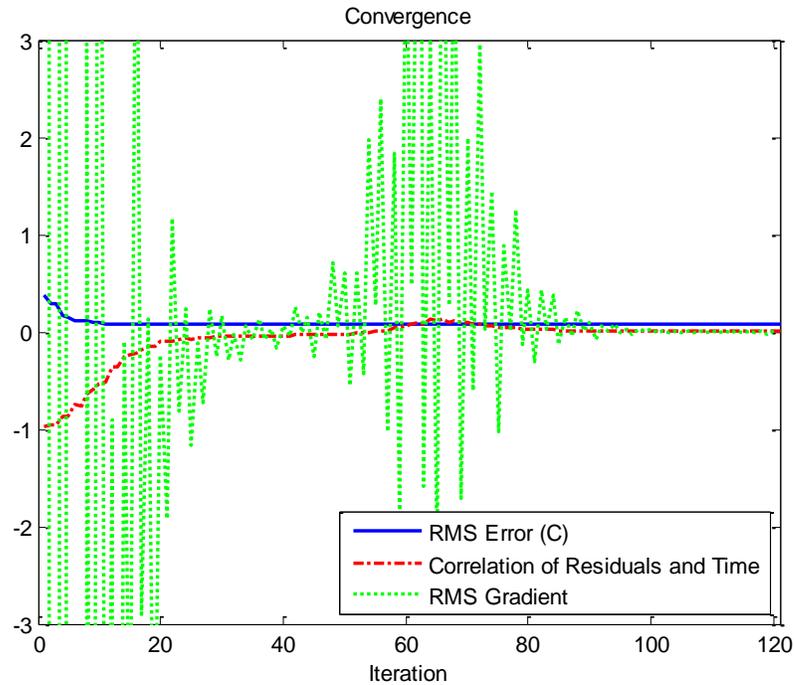
**Figure 5.27 Position 3 Ramps Convergence: Convergence was good, with all parameters reaching minimal values.**

Position 3 residuals show much clearer trending than either Positions 1 or 2 (see

Figure 5.28).
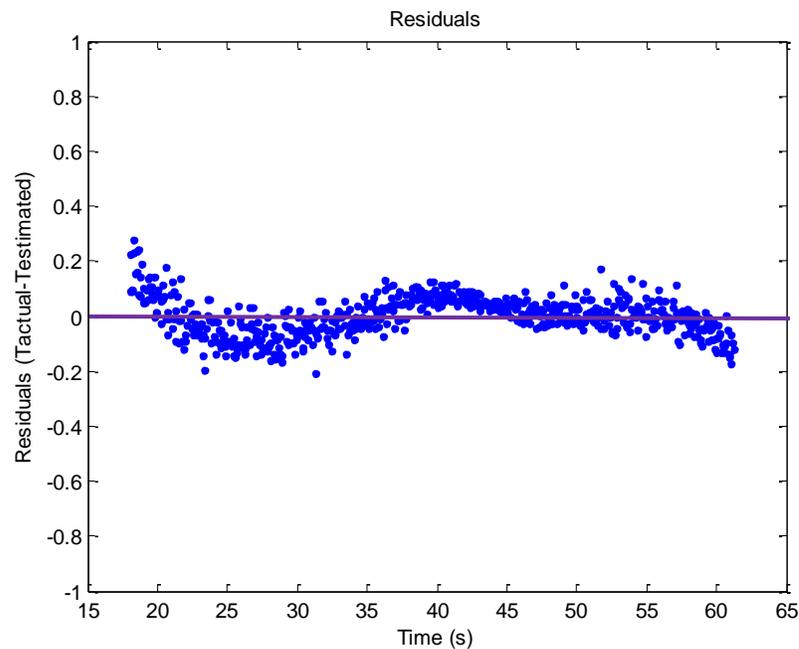
Figure 5.28 Postion 3 Ramps Residuals: Much clearer trending was present compared with Positions 1 or 2.

Table 5.10 contains a summary of the Position 3 series of ramps solution.

Table 5.10 Position 3 Series of Ramps Solution Summary

Solution and Evaluation			
Iterations			
143			
Final RMS Error ($^\circ$C)			
0.0705			
Data Correlation			
0.0046			
Chi-Squared			
0.1309			
	Result Value	Standard Deviation	
m_1 ($^\circ$C/s)	10.8	σ_{m1} ($^\circ$C/s)	823
m_2 ($^\circ$C/s)	3.80	σ_{m2} ($^\circ$C/s)	799
m_3 ($^\circ$C/s)	-2.90	σ_{m3} ($^\circ$C/s)	297
m_4 ($^\circ$C/s)	-0.12	σ_{m4} ($^\circ$C/s)	0.77
h (W/m2-K)	26.6	σ_h (W/m2-K)	205
95% Confidence Interval			
-1600	$\leq$	m_1 ($^\circ$C/s) $\leq$	1624
-1562	$\leq$	m_2 ($^\circ$C/s) $\leq$	1569
-585	$\leq$	m_3 ($^\circ$C/s) $\leq$	578
-1.6	$\leq$	m_4 ($^\circ$C/s) $\leq$	1.4
-376	$\leq$	h(W/m2-K) $\leq$	429

6   CONCLUSIONS

After analysis of the data, the results were considered and conclusions were formed. Major conclusions obtained are as follows.

A step change model was first considered for parameter estimation using surface temperature data.

- Two-parameter estimation using a step change model was successful at obtaining asymptotic free stream temperature within 10% for downstream Positions 2 and 3, and 1% for forward Position 1. This is provided the temperature rise of the free stream is "sharp" and not gradual.
- Heat transfer coefficients estimated with the step change model were within Nusselt correlation ranges for Position 1, and 10% higher than the upper Nusselt range for Positions 2 and 3.
- The quality of results for Position 1 compared to downstream positions indicated that the "knee" of the surface temperature response, the area of greatest change, is the most important area of the curve to capture.
- Residuals trending indicated another model may fit the data better.
- A flaw of the step change model was an inability to estimate the initial rise of the free stream temperature.

After a step change model was verified, surface temperature data was analyzed with a series of ramps analytical model.

- "Perfect" manufactured data determined the algorithm was stable and could converge given reasonable initial parameter guesses.

- Once noise was added to manufactured data, corresponding parameter estimation difference increased to 10% for ramp slopes and 17% for h. This indicated that the parameters have a high degree of sensitivity to noise, even when the data fits the model perfectly.

- Large inaccuracies were present when using trial data with the series of ramps model. These were characterized by a largely overestimated initial ramp followed by negative ramps attempting to correct for this initial overestimation.

- h values estimated at Position 1 were 50% of the values estimated with the step change model, and below the Nusselt correlation range. Values estimated for Positions 2 and 3 had a difference of 9%.

- The shape of convergence indicated large changes in parameters with comparatively small changes in RMS error. This indicates multiple parameter solutions nearly minimize the objective function, and the problem could be close to non-unique.

- Estimation of the first ramp appears to be most important, as its effects are contained within each subsequent ramp. Surface data in the initial few seconds would add greater constraints to the first ramp and improve results.

- Parameters are highly interdependent, as shown by the covariance matrix, particularly the cross-dependence of m_1 and h. This led to high confidence intervals.

- Additional constraints could reduce this interdependency and/or a weighting matrix scheme. A potential constraint could be forcing the estimated fluid temperature through a set temperature measured at the end of the experimental run.

The accuracy and quality of data could be improved in a number of ways for future experiments.

- Electrical interference of the thermocouples could be minimized through use of a DC blower or electrical isolation of the equipment.

- More steady state calibration data points could be taken in the area of greatest hue change.

- A new camera could be purchased with a Bayer filter that provides a smooth curve for the blue signal.

- A different HSV color model could be used to maximize the amount of initial useful TLC data.

- Alternatively, wider band TLCs could be used to avoid the singularity present at their clearing point.

7   SUMMARY

During the course of this study, a flat plate immersed in an air stream was studied. The plate was heated by a sudden increase in fluid temperature. Thermochromic liquid crystals, which covered the surface of the plate, were recorded during the temperature rise by a video camera, and this hue data was calibrated to reference thermocouples to provide a surface temperature history. It was desired to know if one could estimate the heat transfer coefficient h and the free stream temperature profile simultaneously with inverse methods, using the hue temperature history.

Three positions on the plate were analyzed at equally spaced intervals. Positions 1, 2, and 3 were analyzed with progressively sophisticated techniques. First, hue temperature data from the experimental trial was analyzed with a step change surface convection model. This revealed that a two-parameter inverse method was able to estimate asymptotic free stream temperature T_∞ and h with reasonable accuracy, using thermocouple data and Nusselt correlations to verify estimated parameters. However, it was found this method worked less well with data from Positions 2 and 3. Also, this method was poor at estimating surface temperature response in temporal regions experiencing rapid temperature increase. Through examining residual trending, it was indicated that another model may fit the data better.

After verifying that the method worked for two parameters, a five-parameter series of ramps convection model was examined. Generated data was first used with this method to verify accuracy and convergence. The algorithm was able to exactly estimate parameters with noiseless manufactured "data." Adding Gaussian noise to generated "perfect" data returned errors in the free stream temperature profile of up to 2 °C by the inverse method. While h values lay between Nusselt correlation boundaries, exact errors in h could not be determined. The Nusselt correlations give a minimum h at a constant surface temperature boundary condition, and a maximum h at a constant surface heat flux condition. Neither of these conditions were true for the experiment, meaning the true value of h lay somewhere between the two. So, unlike T_∞ which could be measured with a thermocouple, the exact value of h was unknown. When the algorithm estimated parameters using experimental data, large inaccuracies became present in free stream slope and h values. Also, large confidence intervals were produced.

From the results of the step change model trials, it is clear the experimental surface temperature response does not result from a perfect step change of the fluid temperature as suggested upon review of the residuals. Measurement of the actual free stream temperature confirms this. Variations in the surface temperature curve between actual and predicted values in the initial temperature rise region lead to inaccuracies that propagate with increasing time. After a short amount of time, surface temperature rise becomes nearly linear. Because initial surface temperature values do not match well between step change model and experimental data, the difference in slope of the linear portions of the surface temperature response curve become more pronounced with

increasing time. It is believed this is the reason Position 1 produced better results than the other two, because Position 1's data occurred earlier than the other positions.

Heat transfer coefficient results from the step change trials of Positions 2 and 3 suggest these points lay in a turbulent flow region. Turbulent flow Nusselt correlations for Position 2 predicted an h between 27 and 28 $\frac{W}{m^2 K}$. The h value estimated by the inverse method was 31.31 $\frac{W}{m^2 K}$, slightly above the upper bound of the correlations. The h estimated at Position 3 was 27.6 $\frac{W}{m^2 K}$, compared to a range of 23-24 $\frac{W}{m^2 K}$ predicted by the turbulent correlations. The T_∞ of 46.96 °C returned by the algorithm at Postion 2 was 1.8 °C above the measured value, a difference of 5% of the total temperature rise. Position 3 predicted slightly better with a value of 46.15 °C. These estimates were fair, though not as accurate as those at Position 1. This could indicate that the step change model is poorer for later data or that the small standard deviation of the Positions 2 and 3 hue temperature brings into contrast model errors more readily.

The results of the series of ramps trials show large inaccuracies. It was shown that, even given perfect generated data with a realistic amount of Gaussian noise, the inverse algorithm produced T_∞ results up to 2 °C below actual values. The hue temperature trial data contained inaccuracies the generated data did not, such as uncertainty from the hue/temperature calibration, noise in the thermocouple data, and missing portions of initial surface temperature response. These realities, combined with the sensitivity to noise of the method shown by the generated data trials, produced inaccurate and physically impossible parameter estimates. The algorithm did converge to a solution, however, and produced a curve that fit the data well, with little trending

visible in the residuals. The Position 1 *h* estimate of $43 \frac{W}{m^2 K}$ was poor, a value 50% of the value predicted by the step change model. The Position 2 ramp results fit fluid temperature the closest, and also returned an *h* that was close to the step change results: $34 \frac{W}{m^2 K}$ compared with $31 \frac{W}{m^2 K}$, a difference of 9%. Position 3 had a poor estimation of fluid temperature, estimating a maximum temperature of 56 °C. Estimated *h*, however, was closest to the step change solution, with an difference of 4%.

With nonlinear models, the possibility of multiple solutions that minimize the objective function – or solutions that nearly minimize it – become a possibility. With different combinations of parameters resulting in nearly the same RMS error, the four-ramp surface convection model showed these tendencies.

It is believed the largest culprit in the errors in the series of ramps trials was the absence of the initial few seconds of hue temperature data. It is believed that the first ramp of the fluid temperature response is the most crucial to estimate accurately. This is supported by the model in which m_1 appears in every term of the temperature response. This is also supported by the values of the covariance matrix, indicating a high dependency of other variables on the first slope. Heat transfer experienced during the initial ramp influences the surface temperature profile at every future time. Each successive ramp has diminishing influence. This was shown when the series of ramps model was used with a limited generated data set missing the first two seconds of data. In this trial, the error was increased in comparison to the full data set. Having that first two seconds of data in the experimental set would add greater constraint to the first ramp, limiting how much fluctuation it could experience.

The parameters in the series of ramps model are also highly interdependent, as shown by the large off-diagonal values in the covariance matrices. Uncertainties in one parameter have a multiplying effect into other parameters' uncertainty. The h parameter, in particular, has high codependency with the slope parameters. It is believed additional constraints to the inverse method may reduce these interdependencies.

The method of estimating fluid temperature response and the heat transfer coefficient seems to work with a limited number of parameters, as shown by the estimation of T_∞ and h in the step change model trials. It is believed that the large number of parameters in the series of ramps formulation appears to lead to large inaccuracies in the estimated parameters. Ramps must be estimated sequentially, that is, if only two ramps are to be estimated, it must be the initial two ramps. It is not possible to only estimate the last two ramps as each ramp contributes heat transfer to the ones following. Therefore, using fewer numbers of ramps contributes to even greater inaccuracies in the parameter estimates, as only data up to the end of the last ramp can be used. For example, in a two ramp problem, only the first 5 seconds of data could be used. Since the first two seconds were already missing, this increased the percentage of the data set that was absent, increasing parameter error.

It is believed that the series of ramps formulation could work better with additional constraints added to the objective function. An example constraint would be forcing the fluid temperature through a set temperature measured at the end of the experimental run, or holding the slope of the final ramp at a constant value of zero. This might reduce error in initial ramp estimations as well covariance of parameters.

In terms of data collection and preparation, there are a few suggestions for improving accuracy for future experiments. It was found that electrical interference, likely caused by the variable speed blower controller, caused large amounts of noise in thermocouple collection. Electrically isolating data acquisition equipment from the controller would greatly improve accuracy. Or, alternatively, use of a DC blower could also mitigate this problem.

The temperature/hue calibration is also a possible area of improvement. Steady-state calibration data was difficult to obtain in the narrowest portion of TLC color play. Special attention to this area is recommended, as many more calibration points should be recorded in this region than others. Transient calibration data was used to mitigate this. While the transient calibration method is an effective calibration tool, its downside lies in the fact that only locations on the plate with thermocouples could be calibrated. With the steady-state calibration, any plate location could be analyzed.

The camera used in this research had a peculiarity that caused a bump in the blue signal at certain hues. It is believed this was caused by a flaw in the Bayer filter in which green wavelength light bled into the blue sensors. It is recommended that a different camera be used in subsequent studies, and the color play of the camera checked carefully before use.

The HSV color model used is also a potential source of improvement. Because the model used was a piecewise function, based on the plane-projection of the RGB color cube, discontinuities were sometimes present when max (R, G, B) changed. This rendered initial hue values unusable since they fluctuated between 0 and 255 at low RGB values. To mitigate this, a circular HSV color model could be used. This model would

use an angular value for hue, a value whose angle could be shifted to avoid the discontinuity that occurs at low RGB values. Alternatively, TLCs could be used whose color play starts below room temperature. However, this could potentially lead to missing recording of the red max portion of the TLC color play, as this occurred early and quickly in experimental trials. The red max point is not a necessary portion of the TLC color play to record, but it does contain color data with the least amount of noise.

Errors in results could also be a result of problems with assumptions in the mathematical model. It was shown in Section 4.5.3 that the centerline temperature of the plate differed than the temperature calculated with a semi-infinite solid solution. This had potential to affect the surface temperature, causing a slower temperature response and lower h values. The constant h assumption could also be questioned. Very large h values that occur in the first 0.1-0.4 seconds of an experimental trial's effect on surface temperature could be investigated. Also, due to the separation bubble at the leading edge of the plate, there was potential lateral conduction in the plate at forward locations. Since the model assumed 1-D conduction, this was also a potential source of error.

In summary, the method developed for estimating heat transfer parameters in gas turbine engine cooling channels was partially successful, and shows signs of definite promise. The step change surface convection model trials showed that the method works with two parameters, and provides estimated fluid temperature within 10% error. The series of ramps model displayed problems with estimating the initial ramp value causing subsequent ramps to try to correct for this, ultimately resulting in errors in all ramps. Also, this method showed a tendency toward multiple solutions of nearly the same RMS error. However, it is believed that increasing the amount of viable data taken with TLCs

combined with additional objective function constraints could correct for these problems, leading to a heat transfer estimation technique that increases accuracy and becomes a valuable tool for cooling channel engineers.

8 WORKS CITED

[1] J C Han, S Dutta, and S Ekkad, *Gas Turbine Heat Transfer and Cooling Technology*.: Taylor & Francis, 2000.

[2] Jeff Dahl. (2007, December) Wikipedia: Gas Turbine Engine. [Online]. http://en.wikipedia.org/wiki/File:Jet_engine.svg

[3] D Hubble and T Diller, "A Hybrid Method for Measuring Heat Flux," *Journal of Heat Transfer*, vol. 132, no. 3, March 2010.

[4] Rico Poser, "Transient Heat Transfer Experiments in Complex Geometries Using Liquid Crystal Thermography," University of Stuttgart, Stuttgart, PhD Thesis 2010.

[5] Adrian Bejan, *Convection Heat Transfer*, 3rd ed. Hoboken, United States of America: John Wiley & Sons, 2004.

[6] Latif Jiji, *Heat Conduction*, 3rd ed. Berlin: Springer-Verlag , 2009.

[7] Holger Thiele, "Investigating the Relationship of Heat Transfer Coefficients found in Steady-State and Transient Experiments," University of Stuttgart, Stuttgart, Diplomarbeit 2010.

[8] P T Ireland and T V Jones, "Liquid Crystal Measurements of Heat Transfer and Surface Shear Stress," *Meas. Sci. Technol.*, vol. 11, pp. 969-986, 2000.

[9] P R N Childs, J R Greenwood, and Long C A, "Heat Flux Measurement

Techniques," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 213, no. 7, pp. 655-657, January 1999.

[10] R J Butler and Baughn J W, "The Effect of the Thermal Boundary Condition on Transient Method Heat Transfer Measurements on a Flat Plate With a Laminar Boundary Layer," *Journal of Heat Transfer*, vol. 118, pp. 831-837, November 1996.

[11] T.V. Jones, "Heat Transfer, Skin Friction, Total Temperature and Concentration Measurements," in *Measurements of Unsteady Fluid Dynamic Phenomena*. Washington D.C.: McGraw Hill, 1977, pp. 63-102.

[12] J. R. Ferguson, "Simultaneous Thermochromic Liquid Crystal Calibration and Calculation of Heat Transfer Coefficients," in *ASME Turbo Expo*, Montreal, 2007, pp. 841-847.

[13] Dathan Froerer, "Transient Testing to Determine Heat Transfer Coefficient and Well Thickness of Turbine Blades," in *ASME International Mechanical Engineering Congress and RD&D Expo*, Anaheim, CA, 2004.

[14] Sebastian Jochem, University of Stuttgart, Stuttgart, Studienarbeit 2009.

[15] Michael Maurer, "Experimental and Numerical Investigations of Convective Cooling Configurations for Gas Turbine Combustors," University of Stuttgart, Stuttgart, PhD Thesis 2007.

[16] M. R. Anderson and J. W. Baughn, "Hysteresis in Liquid Crystal Thermography," *Journal of Heat Transfer*, vol. 126, pp. 339-345, June 2004.

[17] W.D. Morris and S.W. Chang, "An Experimental Study of Heat Transfer in a Simulated Turbine Blade Cooling Passage," *International Journal of Heat and Mass Transfer*, vol. 40, no. 15, pp. 3703-3716, September 1996.

[18] A. Frackowiak, J. Wolfersdorf, and M. Cialkowski, "Solution of the Inverse Heat Conduction Problem Described by the Poisson Equation for a Cooled Gas Turbine Blade," *International Journal of Heat and Mass Transfer*, vol. 54, no. 5-6, pp. 1236-1243, February 2011.

[19] P. Heidrich, J. Wolfersdorf, S. Schmidt, and M. Schnieder, "Estimation of Internal Heat Transfer Coefficients and Detection of Gas Turbine Blades from Transient Surface Temperature Measurements," *Journal of Physics: Conference Series*, vol. 135, no. 1, pp. 1-9, 2008.

[20] P. Heidrich, J. Wolfersdorf, and M Schnieder, "Experimental Study of Heat Transfer in Gas Turbine Blades Using a Transient Inverse Technique," *Heat Transfer Engineering*, vol. 30, no. 13, pp. 1077-1086, 2009.

[21] Frank P Incropera, David P Dewitt, Theodore L Bergman, and Adrienne S Lavine, *Fundamentals of Heat and Mass Transfer*, 6th ed. Hoboken: John Wiley & Sons, 2007.

[22] G Nellis and S Klein, *Heat Transfer*.: Cambridge University Press, 2008.

[23] Necati Özisik, *Heat Conduction*, 2nd ed.: John Wiley and Sons, 1993.

[24] Richard C Aster, Brian Borchers, and Clifford H Thurber, *Parameter Estimation and Inverse Problems*. London, UK: Elsevier Academic Press, 2005.

[25] N.M. Özisik and H.R.B. Orlande, *Inverse Heat Transfer*. New York: Taylor &

Francis, 2000.

[26] Oleg Alexandrov. (2007, June) Wikipedia: Conjugate Gradient Method. [Online].

http://en.wikipedia.org/wiki/File:Conjugate_gradient_illustration.svg

[27] (2009, December) Correlation and Dependence. [Online].

http://en.wikipedia.org/wiki/Correlation_and_dependence

[28] Malcom Longair, "Light and Colour," in *Colour: Art & Science*, Trevor Lamb and

Janine Bourriau, Eds. Cambridge, UK: Cambridge University Press, 1995, ch. 3,

pp. 65-102.

[29] Transition Metal Complexes and Color. [Online].

http://www.wou.edu/las/physci/ch462/tmcolors.htm

[30] Denis Baylor, "Colour Mechanisms of the Eye," in *Colour: Art & Science*, Trevor

Lamb and Janine Bourriau, Eds. Cambridge, UK: Cambridge University Press,

1995, ch. 4, pp. 103-126.

[31] (2010, May) The Retina. [Online].

http://www.visionweb.com/content/consumers/dev_consumerarticles.jsp?RID=

36

[32] National Instruments. (2008, June) NI Visions Concept Manual. Reference Manual.

[33] Steven Shevell, *The Science of Color*. Oxford, UK: Elsevier, 2003.

[34] (2010, July) Color Spaces. [Online].

http://www.cs.berkeley.edu/~sequin/CS184/TOPICS/ColorSpaces/Color_0.html

[35] (2011, September) HSL and HSV. [Online].

http://en.wikipedia.org/wiki/HSL_and_HSV

[36] Toralf Scharf, *Polarized Light in Liquid Crystals and Polymers*. Hoboken, New

Jersey: John Wiley & Sons, 2007.

[37] Feodor Oestreicher. Der Begriff "Flüssigkristall" - ein scheinbarer Widerspruch.

[Online]. http://fluessigkristalle.com/grundlagen.htm

[38] (2010) Liquid Crystals. [Online].

http://www.doitpoms.ac.uk/tlplib/anisotropy/liquidcrystals.php

[39] Courtney Peterson, "How it Works: The Charge-Coupled Device, or CCD," *Journal

of Young Investigators*, vol. 3, March 2001,

http://www.jyi.org/volumes/volume3/issue1/features/peterson.html.

[40] (2011, August) Bayer Filter. [Online]. http://en.wikipedia.org/wiki/Bayer_filter

[41] (2011, March) XCap Software. [Online]. http://www.epixinc.com/products/xcap.htm

[42] Adobe Developers Association. (1992, June) Tiff Version 6.0. [Online].

http://partners.adobe.com/public/developer/en/tiff/TIFF6.pdf

[43] Gary Gordon and James L. Nuckolls, *Interior Lighting for Designers*, 3rd ed.: John

Wiley & Sons, 1995.

[44] (2011, September) Color Temperature. [Online].

http://en.wikipedia.org/wiki/Color_temperature

[45] Patrick F. Dunn, *Measurement and Data Analysis for Engineering and Science*, 1st

ed. New York, NY: McGraw-Hill, 2005.

[46] J. V. Nicholas and D. R. White, *Traceable Temperatures: An Introduction to

Temperature Measurement and Calibration*, 2nd ed. Chichester, England: John

Wiley and Sons, LTD., 2001.

[47] (July, 2010) Encyclopedia of Alternative Energy and Sustainable Living: Seebeck Effect. [Online].

http://www.daviddarling.info/encyclopedia/S/AE_Seebeck_effect.html

[48] (2010, July) Thermocouple Theory. [Online].

http://www.efunda.com/designstandards/sensors/thermocouples/thmcple_theory.cfm

[49] (2011, August) TCIC Series Data Sheet. [Online].

http://www.omega.com/DAS/pdf/TCIC.pdf

[50] (2011, August) "Cement-On" Thermocouples Data Sheet. [Online].

http://www.omega.com/Temperature/pdf/CO-K.pdf

[51] JC Lane and RI Loehrke, "Leading Edge Separation From a Blunt Plate at Low Reynolds Number," *Journal of Fluids Engineering*, vol. 102, no. 4, pp. 494-497, December 1980.

9   APPENDIX A

9.1   Series of Ramps Detailed Derivation

For convenience, let

$$\hat{T} = T - T_0.$$

Duhamel's Theorem gives the temperature response of a system with a time-varying boundary condition in which a *fundamental solution* to the problem is known [22]. The fundamental solution is defined as the response of the system with a zero initial condition to a single, constant non-homogeneous term with magnitude unity. This can be stated as

$$\hat{T}(x,t) = \int_{\tau=0}^{t} T_f(x, t-\tau) \frac{dB(t)}{dt}\Big|_{t=\tau} d\tau + B_{t=0} T_f(x,t) \tag{9.1}$$

where T_f is the fundamental solution and B is the time-varying boundary condition.

The first step to applying Duhamel's Theorem is finding the fundamental solution. For this problem, the fundamental solution solves

$$\frac{\partial^2 T_f}{\partial x^2} = \frac{1}{\alpha}\frac{\partial T_f}{\partial t} \tag{9.2}$$

With boundary and initial conditions

$$T(x = 0, t) = 0 \tag{9.3}$$

$$T(x \rightarrow \infty, t) = 0 \tag{9.4}$$

$$\frac{\partial T_f}{\partial x}\Big|_{x=0} - \frac{h}{k} T_f = \frac{h}{k} \tag{9.5}$$

The solution to the fundamental problem is well known and is given by Equation 9.6.

$$T_f(x,t) = \text{erfc}\left(\frac{x}{2\sqrt{\alpha t}}\right) - \exp\left(\frac{hx}{k} + \beta^2\right)\text{erfc}(\frac{x}{2\sqrt{\alpha t}} + \beta) \tag{9.6}$$

$$\beta = \frac{h\sqrt{\alpha t}}{k}$$

The time-varying term $B(t)$ is the ramp function that defines T_∞

$$B(t) = m \cdot t \tag{9.7}$$

and

$$\frac{dB(t)}{dt}\bigg|_{t=\tau} = m \tag{9.8}$$

Also, since $B(0) = 0$, the $B_{t=0}T_f(x,t)$ term of Equation 9.1 is canceled.

When these terms are substituted into Duhamel's Theorem, the resulting integral,

$$\hat{T}(x,t) = m \int_0^t \left[\text{erfc}\left(\frac{x}{2\sqrt{\alpha(t-\tau)}}\right) - \exp\left(\frac{hx}{k} + \beta_\tau^2\right) \text{erfc}(\frac{x}{2\sqrt{\alpha(t-\tau)}} + \beta_\tau) \right] d\tau \tag{9.9}$$

$$\beta_\tau = \frac{h\sqrt{\alpha(t-\tau)}}{k}$$

is easily solvable with analytical mathematical software such as Mathematica and yields

the single-ramp solution:

$$\hat{T}(x,t) = mt\frac{k^2}{\alpha h^2}\left(1 - \text{erfc}\left(\frac{1}{2}\sqrt{\frac{x^2}{\alpha t}}\right)\right)$$

$$-mt\frac{k}{h\sqrt{\pi}\sqrt{\alpha t}}\left(2 \cdot \exp\left(\frac{-x^2}{4\alpha t}\right) - \sqrt{\pi}\sqrt{\frac{x^2}{\alpha t}}\text{erfc}\left(\frac{1}{2}\sqrt{\frac{x^2}{\alpha t}}\right)\right) \tag{9.10}$$

$$-mt\frac{k^2}{h^2\alpha t}\exp\left(\frac{h^2}{k^2}\alpha t + \frac{hx}{k}\right)\text{erfc}\left(\frac{h\sqrt{\alpha t}}{k} + \frac{x}{2\sqrt{\alpha t}}\right) + mt.$$

Setting x=0 gives the surface response solution.

$$\hat{T}(0,t) = mt\left[1 - \frac{2}{\sqrt{\pi}\beta} - \frac{1-\exp(\beta^2)\text{erfc}(\beta)}{\beta^2}\right] \tag{9.11}$$

The multi-ramp solution can be achieved by breaking the integral in Duhamel's

Theorem into a series of integrals for each ramp [23].

$$\hat{T}(x,t) = \sum_{i=1}^{n} m_i \int_{\tau=t_{i-1}}^{t_i} T_f(x,t-\tau)d\tau \tag{9.12}$$

In Equation 9.12, the t_i terms indicate the beginning of each ramp, with $t_0 = 0$, $t_n = t$,

and the m_i terms representing each ramp function's slope, respectively.

To solve the series of ramps problem for any number of ramps, first solve for

three ramps. Let

$$g(\beta) = 1 - \frac{2}{\sqrt{\pi}\beta} - \frac{1-\exp(\beta^2)\mathrm{erfc}(\beta)}{\beta^2}. \tag{9.13}$$

Then, it is first useful to recognize

$$m \int T_f(0, t-\tau)d\tau\Big|_{x=0} = -m(t-\tau)g(\beta_\tau). \tag{9.14}$$

Now, Equation 9.12 can be expanded:

$$\hat{T}(t) = -\mathrm{m}_1(t-\tau)g(\beta_\tau)|_0^{t_1} - \mathrm{m}_2(t-\tau)g(\beta_\tau)|_{t_1}^{t_2} - \mathrm{m}_3(t-\tau)g(\beta_\tau)|_{t_2}^{t} \tag{9.15}$$

Evaluating the bounds of the integrals,

$$\hat{T}(t) = \mathrm{m}_1\big(t \cdot g(\beta) - (t-t_1)g(\beta_1)\big)$$

$$+\mathrm{m}_2\big((t-t_1)g(\beta_1) - (t-t_2)g(\beta_2)\big) + \mathrm{m}_3(t-t_2)g(\beta_2) \tag{9.16}$$

$$\beta_i = \frac{h}{k}\sqrt{\alpha(t-t_i)}$$

Each ith element of the summation is relevant only for times after t_i. For example, even though three ramps might ultimately be used, for times less than t_1, the surface temperature response still only includes the first element of the summation, being equal to the single ramp solution.

After three elements of the summation have been calculated, a pattern emerges.

$$\hat{T}(0,t) = m_1 t \cdot g(\beta_0) + \sum_{i=1}^{n-1}[H(t-t_i)m_{i+1}(t-t_i)g(\beta_i) - m_i(t-t_i)g(\beta_i)]$$

where
$$H(\tau) = \begin{cases} 0 & \tau < 0 \\ 1 & \tau \geq 0 \end{cases}$$

$$\beta_i = \frac{h}{k}\sqrt{\alpha(t-t_i)} \tag{9.17}$$

$$g(\beta_i) = 1 - \frac{2}{\sqrt{\pi}\beta_i} + \frac{1-\exp(\beta_i^2)\cdot\mathrm{erfc}(\beta_i)}{\beta_i^2}$$

10 APPENDIX B

10.1 MATLAB Code

The following code was used to calculate the boundary layer thickness along a

flat surface.

```
%Program for calculating boundary layer thickness delta in test section
clear

%Air parameters nu in m^2/s
nu=1.589*10^-5;

%Range of velocities in the wind tunnel U in m/s
Umin=2;
Umax=6;
U=[Umin:.25:Umax];

%Plate starts at 26 cm from edge of test section, plate is 23 cm long
X=[.001:.0025:.2];

delta=zeros(length(U),length(X));
RE=zeros(length(U),length(X));

for i=1:length(U)
    for j=1:length(X)
        re=U(i)*X(j)/nu;
        RE(i,j)=re;
        if re>5*10^5
            delta(i,j)=0.37*(U(i)*X(j)/nu)^(-0.2)*X(j);
        else
            delta(i,j)=4.92*(U(i)*X(j)/nu)^(-0.5)*X(j);
        end
    end
end

surf(X,U,delta);
xlabel('y (m)')
ylabel('U (m/s)')
zlabel('delta (m)')
title('Boundary layer thickness in test section')
figure
surf(X,U,RE);
xlabel('y (m)')
ylabel('U (m/s)')
```

The following code was used to calculate the centerline temperature response of

the test plate.

```
%% Set Up Constants
clear
clc
%Starting Surface Temperature T0 (K)
T0=24.0;

%Solid material=Plexiglas(top)/stanless steel(bottom)
%Thermal conductivity k of the solid (W/m-K)
k=.19;
%k=15.1;

%Specific heat capacity c of solid (J/kg-K)
cp=1470;
%cp=480;

%Density of solid p (kg/m^3)
p=1190;
%p=8055;

%Thermal diffusivity of solid alpha (m^2/s)
alpha=k/(p*cp);

%Gas Temperature Tg (K)
Tg=T0+21;

%Heat transfer coefficient
h=85;

L=0.00635;
x=0:.00635/100.00635;
x=x';
xs=x/L;

t=0:60;
t=t';
Fo=alpha*t/L^2;

Bi=h*L/k;
mybitranscendental;
G=length(xi);

for i=1:G
    C(i)=4*sin(xi(i))/(2*xi(i)+sin(2*xi(i)));
    S(:,i)=C(i)*exp(-xi(i)^2*Fo);
end

T1=sum(S,2);
T=(T0-Tg)*T1+Tg;
Tchange=T-T0;
```

```
Tchange=[t Tchange];
plot(t,S(:,1),'r')
hold on
plot(t,S(:,2),'y')
hold on
plot(t,S(:,3))
hold on
plot(t,S(:,4),'g')
figure

plot(t,T);

XI=0:.1:10;
CN=4*XI./(2*XI+sin(2*XI));
figure
plot(XI,CN)
xlabel('Xi')
ylabel('Cn')
```

The sub-function calculates the roots of the transcendental Biot number equation:

$$\zeta_n \tan(\zeta_n) = Bi$$

```
xi=0;
Bi=2.8408;
for j=0.1:0.1:10
    x=(j-0.1):.0001:j;
    Y1=x.*tan(x);
    Y2=Bi*ones(length(x));
    for i=1:length(x)-1
        Num1=Y1(i)-Y2(i);
        Num2=Y1(i+1)-Y2(i+1);
        if (Num2>=0) && (Num1<=0)
            xi1 = INTERP1([Y1(i);Y1(i+1)],[x(i);x(i+1)],Bi);
            xi=[xi;x(i)];
        end
    end
end
xi=xi(2:length(xi));
```

The following algorithm uses the conjugate gradient inverse algorithm to estimate two parameters using a step change surface convection model.

```
%This program computes parameters h and Tg for a step function using
the
%conjugate gradient method.
clear
clc
%% Set Up Constants
%Starting Surface Temperature T0 (K)
T0=24.3;

%Solid material=Plexiglas(top)/stanless steel(bottom)
%Thermal conductivity k of the solid (W/m-K)
k=.19;
%k=15.1;

%Specific heat capacity c of solid (J/kg-K)
cp=1470;
%cp=480;

%Density of solid p (kg/m^3)
p=1190;
%p=8055;

%Thermal diffusivity of solid alpha (m^2/s)
alpha=k/(p*cp);

%For reference: htrue=80, Tgtrue=333.15

%% Guess at Initial Parameters and Set Up Vector
%Input initial parameter estimation h (W/m^2-K) and Tg K
Tg=44;
h=35;
P=[Tg;h];
Pmatrix=P';

%Read in surface temp data
%tT=xlsread('hue1inversemethod.xlsx');
tT=xlsread('hue4inversemethod1.xlsx');
t=tT(:,1);
Ts=tT(:,2);
t=tT(1:256,1);
Ts=tT(1:256,2);
L=length(t);

%Define standard deviation of the data
var=0.0383;
stdev=sqrt(var);
%Max iterations
countmax=159;
%Chi squared=1 stopping criteria
```

```
chistop=0;

%% Check Initial Guess
%Calculate initial B
B=h*sqrt(t)/(p*cp*k)^.5;

%Find estimated surface temperature response
[Tes]=mystepfunc(T0,Tg,B);

%Find sum of squares error by comparing with actual response
error=sum((Ts-Tes).^2);
AvgE=sqrt(error/length(t));

 %Compare to stoping criteria
chi=error/(length(t)-2)/stdev^2;
% if chi-1<=chistop
%      return
% end
 CHI=[chi];
%Start storing error in vector
E=error;

%Set up correlation vector
e=Ts-Tes;
eavg=sum(e)/length(e);
tavg=sum(t)/length(t);
eavgvec=e-eavg*ones(length(e),1);
tavgvec=t-tavg*ones(length(t),1);
uppersum=sum(eavgvec.*tavgvec);
lowersum=(length(t)-1)*std(e)*std(t);
corr=uppersum/lowersum;
CORVEC=[corr];

%% First Iteration
%Start Count
count=1;

%Compute Jacobian Sensitivity Matrix
[DTDTg]=mydiffTTg(B);
[DTDh]=mydiffTTh(T0,Tg,B,h);
J=[DTDTg DTDh];

%Obtain Gradient Tensor S
S=-2*J'*(Ts-Tes);
Svector=[S'];

%For first step, use steepest descent; move in direction of gradient
%So conjugation coefficient G=0, so ignore
%For conjugation coefficent G=0, directon of descent d=S
d=S;

%Find search step size beta (minimizes objective function)
beta=((J*d)'*(Tes-Ts))/((J*d)'*(J*d));

%Compute new parameters Tg and h, new estimated after taking step
```

```matlab
Pnew=P-beta*d;
Pmatrix=[Pmatrix;Pnew'];
%Tg=Pnew(1,1);
h=Pnew(2,1);
B=h*sqrt(t)/(p*cp*k)^.5;
[Tes]=mystepfunc(T0,Tg,B);
error=sum((Ts-Tes).^2);
E=[E;error];

%Continue Correlation vector
e=Ts-Tes;
eavg=sum(e)/length(e);
tavg=sum(t)/length(t);
eavgvec=e-eavg*ones(length(e),1);
tavgvec=t-tavg*ones(length(t),1);
uppersum=sum(eavgvec.*tavgvec);
lowersum=(length(t)-1)*std(e)*std(t);
corr=uppersum/lowersum;
CORVEC=[CORVEC;corr];

%% Iterate in earnest
while   count<countmax
    count=count+1;
    %Calculate new estimated T, Tes
    B=h*sqrt(t)/(p*cp*k)^.5;
    [Tes]=mystepfunc(T0,Tg,B);
    %Find sum of squares error by comparing with actual response
    error=sum((Ts-Tes).^2);
    erroravg=sqrt(error/length(t));
    E=[E;error];
    AvgE=sqrt(error)/length(t);

    %Calculate Chi Squared
    chi=error/(length(t)-2)/stdev^2;
    CHI=[CHI;chi];

    %Continue Correlation vector
    e=Ts-Tes;
    eavg=sum(e)/length(e);
    tavg=sum(t)/length(t);
    eavgvec=e-eavg*ones(length(e),1);
    tavgvec=t-tavg*ones(length(t),1);
    uppersum=sum(eavgvec.*tavgvec);
    lowersum=(length(t)-1)*std(e)*std(t);
    corr=uppersum/lowersum;
    CORVEC=[CORVEC;corr];

    %Compute the new Jacobian Sensitivity Matrix
    [DTDTg]=mydiffTTg(B);
    [DTDh]=mydiffTTh(T0,Tg,B,h);
    J=[DTDTg DTDh];
    %Obtain new gradient tensor S
    Snew=-2*J'*(Ts-Tes);
    %Obtain conjugation coefficient G Using Fletcher-Reeves Formulation
    n=length(S);
    sum1=0;
```

```matlab
        sum2=0;
        for i=1:n
            sum1=sum1+Snew(i,1)^2;
            sum2=sum2+S(i,1)^2;
        end
        gamma=sum1/sum2;
        %Find new direction of descent dnew
        dnew=Snew+gamma*d;
        %Find new search step size betanew
        betanew=((J*dnew)'*(Tes-Ts))/((J*dnew)'*(J*dnew));
        %Compute new parameters
        Pnew1=[Tg;h]-.05*betanew*dnew;
        Pmatrix=[Pmatrix;Pnew1'];
        Tg=Pnew1(1,1);
        h=Pnew1(2,1);
        %Iterate
        d=dnew;
        S=Snew;
        Svector=[Svector; S'];
        disp(count);
end


%% Solution Evaluation

disp('Solution and Solution Evaluation');
%Plot the estimated temperature curve on top of the data curve
plot(t,Tes);
hold on
plot(t,Ts,'g');
hold on
plot(t,Tg*ones(length(t)),'r');
axis([0 t(length(t)) 24 48]);
title('Temperature Curve Fit')
xlabel('Time (s)');
ylabel('Temp. (C)');
legend('Estimated','Data','Fluid Temp');
grid on
grid minor
figure

Svectornorm=sqrt((Svector(:,1).^2)+(Svector(:,2).^2));

plot(sqrt(E/length(t)));
hold on
plot(CORVEC,'r');
hold on
plot(Svectornorm,'g');
axis([0 count+1 -2 2]);
legend('Average Error (C)','Correlation of Error and Time','Average
Gradient')
xlabel('Iteration')
title('Convergence')
figure
```

```matlab
%Plot chi-squared convergence criterion
plot(CHI);
title('Chi-Squared Goodness of Fit');
xlabel('Iteration');
ylabel('Chi-Squared');
axis([0 count+1 0 3]);
disp('Chi-squared converged value');
disp(chi);

%Calculate standard deviation of the parameters and 95% and 99%
confidence
COVm=stdev*(J'*J)^-1;
stdevm=[sqrt(COVm(1,1));sqrt(COVm(2,2))];
disp('Standard Deviation (T/h)');
disp(stdevm);
con99=[Pnew1-2.576*stdevm Pnew1 Pnew1+2.576*stdevm];
disp('99% Confidence Interval (T/h)');
disp(con99);
con95=[Pnew1-1.96*stdevm Pnew1 Pnew1+1.96*stdevm];
disp('95% Confidence Interval (T/h)');
disp(con95);
disp('Iteration count');
disp(count);
figure
plot(t,(Ts-Tes),'.')
xlabel('Time (s)')
ylabel('Residuals (Tactual-Testimated)')
title('Residuals')
```

The following are sub-functions for the CG algorithm.

```matlab
function TS=mystepfunc(T0,Tg,B)
TS=T0+(Tg-T0)*(1-exp(B.^2).*erfc(B));

function DTDTg=mydiffTTg(B)
DTDTg=1-exp(B.^2).*erfc(B);

function DTDh=mydiffTTh(T0,Tg,B,h)
DTDh=(Tg-T0)*(-2*B.^2/h.*exp(B.^2).*erfc(B)+2/sqrt(pi)*B/h);
```

The following algorithm uses the conjugate gradient inverse algorithm to estimate

five parameters (four fluid temperature ramps and *h* using a series of ramps surface

convection model.

```matlab
%This program computes parameters h and mi's for a series of steps
function using the
%conjugate gradient method.
clear
clc

%% Set Up Constants
%Starting Surface Temperature T0 (K)
T0=24.3;

%Solid material=Plexiglas
%Thermal conductivity k of the solid (W/m-K)
k=.19;

%Specific heat capacity c of solid (J/kg-K)
cp=1470;

%Density of solid p (kg/m^3)
p=1190;

%Thermal diffusivity of solid alpha (m^2/s)
alpha=k/(p*cp);

%% Guess at Initial Parameters and Set Up Vector

%Read in surface temp data
%tT=xlsread('E:\THESIS\MATLAB\CGSeriesramps\hue1inversemethodadjusted1.
xlsx');
%tT=xlsread('E:\THESIS\MATLAB\CGSeriesramps\noisytempgen.xlsx');
tT=xlsread('E:\THESIS\MATLAB\CGSeriesramps\hue7inversemethod.xlsx');
t=tT(:,1);
Ts=tT(:,2);
%t=tT(30:479,1);
%Ts=tT(30:479,2);
L=length(t);

%Thermocouple data
tTtherm=xlsread('E:\THESIS\MATLAB\CGSeriesramps\therm1inversemethod.xls
x');
ttherm=tTtherm(:,1);
Tstherm=tTtherm(:,2);
Ltherm=length(ttherm);
TCinf=xlsread('E:\THESIS\MATLAB\CGSeriesramps\TCinf.xlsx');

%Set ti's (s)
t0=0;
t1=2;
```

```
t2=5;
t3=8;
t4=32;
%t5=ttherm(length(ttherm));
tvec=[t1;t2;t3;t4];

%Input initial parameter estimation h (W/m^2-K) and mi's (K/s)
%"Real" Values
% m1=7.5117;
% m2=1.7401;
% m3=.3368;
% m4=.0362;
% m5=0;
%h=80;

m1=7.4;
m2=2;
m3=.3;
m4=0.06;
mvec=[m1;m2;m3;m4];
%Set Number of Ramps
R=length(mvec);

h=25;

P=[mvec;h];
Pmatrix=P';

%Variance and standard deviation of the temperature data
VAR=0.0383;
STDEV=sqrt(VAR);

% /////Set Max iterations/////
countmax=143;

%% Plot Initial Results
%Calculate initial Bi's
B=mybeta(h,k,alpha,t,tvec,R);

%Find estimated surface temperature response
[Tes]=myseriesrampfunc(T0,t,R,mvec,tvec,B,@myG);

disp(m1)
%Form initial Tinf
Tinf=zeros(length(t),1);
Tinf(1)=T0;
for i=2:length(t)
    if ttherm(i)<=t1
        Tinf(i)=Tinf(i-1)+m1*1/15;
    elseif (t1<ttherm(i)) && (ttherm(i)<=t2)
        Tinf(i)=Tinf(i-1)+1/15*m2;
    elseif (t2<ttherm(i)) && (ttherm(i)<=t3)
        Tinf(i)=Tinf(i-1)+1/15*m3;
    elseif (t3<ttherm(i)) && (ttherm(i)<=60)
        Tinf(i)=Tinf(i-1)+1/15*m4;
```

```matlab
        end
    end
%Plot the estimated temperature curve on top of the data curve
%figure
plot(t,Tes);
hold on
plot(t,Ts,'.g','linewidth',0.01);
hold on
plot(ttherm(1:211),Tinf(1:211),'r');
hold on
plot(ttherm(1:length(t)),TCinf(1:length(t)),'c');
title('Temperature Curve Fit')
xlabel('t (s)');
ylabel('T (K)');
legend('Estimated','Data','Gas Temp');
hold on
plot(ttherm,Tstherm,'r-.')
grid on
grid minor
figure

xlabel('Time (s)')
ylabel('Temp (C)')
title('Series of ramps solution compared with hue temp and
thermocouple')
legend('Analytical','Huetemp','Thermocouple','Tinf')

%% Check Initial Guess
%Calculate initial Bi's
%Find sum of squares error by comparing with actual response
error=sum((Ts-Tes).^2);
AvgE=sqrt(error/length(t));

%Compare to stoping criteria
chi=error/(length(t)-R-2)/VAR;
CHI=chi;
%Start storing error in vector
E=AvgE;

%Set up correlation vector
e=Ts-Tes;
eavg=sum(e)/length(e);
tavg=sum(t)/length(t);
eavgvec=e-eavg*ones(length(e),1);
tavgvec=t-tavg*ones(length(t),1);
uppersum=sum(eavgvec.*tavgvec);
lowersum=(length(t)-1)*std(e)*std(t);
corr=uppersum/lowersum;
CORVEC=corr;

%% First Iteration
%Start Count
count=1;

%Compute Jacobian Sensitivity Matrix
[dTdm]=diffTdm(t,R,mvec,tvec,B,@myG);
```

```matlab
[dTdh]=mydiffTdh(alpha,k,t,R,mvec,tvec,B,@diffGdh);
J=[dTdm dTdh];

%Obtain Gradient Tensor S
S=-2*J'*((Ts-Tes));
Svector=S';

%For first step, use steepest descent; move in direction of gradient
%So conjugation coefficient G=0, so ignore
%For conjugation coeffect G=0, directon of descent d=S
d=S;

%Find search step size beta (minimizes objective function)
beta=((J*d)'*(Tes-Ts))/((J*d)'*(J*d));

%Compute new parameters Tg and h, new estimated after taking step
Pnew=P-beta*d;
Pmatrix=[Pmatrix;Pnew'];
m1=Pnew(1,1);
m2=Pnew(2,1);
m3=Pnew(3,1);
m4=Pnew(4,1);
mvec=[m1;m2;m3;m4];
h=Pnew(5,1);


B=mybeta(h,k,alpha,t,tvec,R);


[Tes]=myseriesrampfunc(T0,t,R,mvec,tvec,B,@myG);
error=sum((Ts-Tes).^2);
AvgE=sqrt(error/length(t));
E=[E;AvgE];

%Continue Correlation vector
e=Ts-(Tes);
eavg=sum(e)/length(e);
tavg=sum(t)/length(t);
eavgvec=e-eavg*ones(length(e),1);
tavgvec=t-tavg*ones(length(t),1);
uppersum=sum(eavgvec.*tavgvec);
lowersum=(length(t)-1)*std(e)*std(t);
corr=uppersum/lowersum;
CORVEC=[CORVEC;corr];
Q=[0;0;0;0;0];
%% Iterate in earnest
while  count<countmax
    count=count+1;
    %Calculate new estimated T, Tes
    B=mybeta(h,k,alpha,t,tvec,R);
    [Tes]=myseriesrampfunc(T0,t,R,mvec,tvec,B,@myG);
    %Find sum of squares error by comparing with actual response
    error=sum((Ts-Tes).^2);
    AvgE=sqrt(error/length(t));
    E=[E;AvgE];

    %Calculate Chi Squared
```

```matlab
        chi=(error/(length(t)-R-2))/VAR;
        CHI=[CHI;chi];

        %Continue Correlation vector
        e=Ts-(Tes);
        eavg=sum(e)/length(e);
        tavg=sum(t)/length(t);
        eavgvec=e-eavg*ones(length(e),1);
        tavgvec=t-tavg*ones(length(t),1);
        uppersum=sum(eavgvec.*tavgvec);
        lowersum=(length(t)-1)*std(e)*std(t);
        corr=uppersum/lowersum;
        CORVEC=[CORVEC;corr];

        %Compute the new Jacobian Sensitivity Matrix
        [dTdm]=diffTdm(t,R,mvec,tvec,B,@myG);
        [dTdh]=mydiffTdh(alpha,k,t,R,mvec,tvec,B,@diffGdh);
        J=[dTdm dTdh];
        %Calc Cov M
        COVm=(J'*J)^-1;
        %Then scale by the residual values
        Qxx=VAR*COVm;

stdevm=[sqrt(Qxx(1,1));sqrt(Qxx(2,2));sqrt(Qxx(3,3));sqrt(Qxx(4,4));sqr
t(Qxx(5,5))];
    Q=[Q stdevm];
    %Obtain new gradient tensor S
    Snew=-2*J'*(Ts-Tes);
    %Obtain conjugation coefficient G Using Fletcher-Reeves Formulation
    n=length(S);
    sum1=0;
    sum2=0;
    for i=1:n
        sum1=sum1+Snew(i,1)^2;
        sum2=sum2+S(i,1)^2;
    end
    gamma=sum1/sum2;
    %Find new direction of descent dnew
    dnew=Snew+gamma*d;

    %Find new search step size betanew
    betanew=((J*dnew)'*(Tes-Ts))/((J*dnew)'*(J*dnew));
    %Compute new parameters
    Pnew1=Pnew-.5*betanew*dnew;
    Pmatrix=[Pmatrix;Pnew1'];
    m1=Pnew1(1,1);
    m2=Pnew1(2,1);
    m3=Pnew1(3,1);
    m4=Pnew1(4,1);
    mvec=[m1;m2;m3;m4];
    h=Pnew1(5,1);
    Pnew=Pnew1;
    %Iterate
    d=dnew;
    S=Snew;
    Svector=[Svector; S'];
```

```matlab
    disp(count);
    %Reform Tinfinity
Tinf=zeros(length(t),1);
Tinf(1)=T0;
for i=2:length(t)
    if ttherm(i)<=t1
        Tinf(i)=Tinf(i-1)+m1*1/15;
    elseif (t1<ttherm(i)) && (ttherm(i)<=t2)
        Tinf(i)=Tinf(i-1)+1/15*m2;
    elseif (t2<ttherm(i)) && (ttherm(i)<=t3)
        Tinf(i)=Tinf(i-1)+1/15*m3;
    elseif (t3<ttherm(i)) && (ttherm(i)<=t4)
        Tinf(i)=Tinf(i-1)+1/15*m4;
    end
end
end


%% Solution Evaluation

%Plot the estimated temperature curve on top of the data curve
%figure
plot(t,Tes);
hold on
plot(t,Ts,':g','linewidth',2);
hold on
plot(ttherm(1:length(t)),Tinf,'r');
hold on
plot(ttherm(1:length(t)),TCinf(1:length(t)),'c');
title('Temperature Curve Fit')
xlabel('Time (s)');
ylabel('Temp. (C)');
legend('Estimated','Data','Gas Estimated','Gas Thermocouple');
grid on
grid minor

%Display solution evaluation parameters
disp('Solution and Solution Evaluation');
disp('Final average error');
disp(AvgE);
disp('Data correlation');
disp(corr);

figure
grid on
grid minor
gradient=(Svector(:,1)+Svector(:,2)+Svector(:,3)+Svector(:,4)+Svector(:,5))/5;
subplot(4,1,1), plot(gradient);
title('Average Gradient');
xlabel('Iteration');
ylabel('Gradient');
axis([0 count+1 -5 5]);
subplot(4,1,2), plot(E);
title('Average Error');
```

```
xlabel('Iteration');
ylabel('Average error (C)');
axis([0 count+1 0 .2]);
subplot(4,1,3), plot(CORVEC);
title('Correlation of Error and Time');
xlabel('Iteration');
ylabel('Correlation');
axis([0 count+1 -0.5 0.5]);
plot(CHI);
title('Chi-Squared Goodness of Fit');
xlabel('Iteration');
ylabel('Chi-Squared');
axis([0 100 0 3]);
disp('Chi-Squared converged value');
disp(chi);
disp('Gradient')
disp(gradient(length(gradient),:))
figure

plot(E)
hold on
plot(CORVEC,'r')
plot(gradient,'g')
hold on
axis([0 count+1 -3 3]);
legend('Average Error (C)','Correlation of Error and Time','Average
Gradient')
xlabel('Iteration')
title('Convergence')

%Calculate standard deviation of the parameters and 95% and 99%
confidence
%First calculate covariance matrix
COVm=(J'*J)^-1;
%Then scale by the residual values
Qxx=VAR*COVm;
stdevm=[sqrt(Qxx(1,1));sqrt(Qxx(2,2));sqrt(Qxx(3,3));sqrt(Qxx(4,4));sqr
t(Qxx(5,5))];
disp('Standard Deviation (m1/m2/m3/m4/m5/h)');
disp(stdevm);
con99=[Pnew1-2.576*stdevm Pnew1 Pnew1+2.576*stdevm];
disp('99% Confidence Interval (T/h)');
disp(con99);
con95=[Pnew1-1.96*stdevm Pnew1 Pnew1+1.96*stdevm];
disp('95% Confidence Interval (T/h)');
disp(con95);
disp('Initial Parameters t/m');
disp([tvec mvec]);
disp('Iteration count');
disp(count);

figure
plot(t,(Ts-Tes),'.')
xlabel('Time (s)')
ylabel('Residuals (Tactual-Testimated)')
title('Residuals')
```

The following are sub functions for the series of ramps conjugate gradient algorithm.

This sub function takes the derivative of G with respect to *h*.

```
function [dGdh]=diffGdh(alpha,k,t,B)
dGdh=sqrt(alpha*t)/k.*(4/sqrt(pi)*1./B.^2+2./B.^3.*(exp(B.^2).*erfc(B)-
1)-2./B.*(exp(B.^2).*erfc(B)));
```

The following sub function takes the derivative of surface temperature with

respect to slope of the free stream ramps.

```
function [dTdm]=diffTdm(t,R,mvec,tvec,B,myG)

L=length(t);
dTdm=zeros(L,R);
for j=1:R
    if j==1
    dTdm(:,j)=t.*myG(B(:,1));
    for i=1:L
        if t(i)>tvec(j)
            dTdm(i,j)=dTdm(i,j)-(t(i)-tvec(j))*myG(B(i,j+1));
        end
    end
    elseif (1<j) & (j<R)
    for i=1:L
        if t(i)>tvec(j-1)
            dTdm(i,j)=dTdm(i,j)+(t(i)-tvec(j-1))*myG(B(i,j));
        end
        if t(i)>tvec(j)
            dTdm(i,j)=dTdm(i,j)-(t(i)-tvec(j))*myG(B(i,j+1));
        end
    end
    elseif j==R
    for i=1:L
        if t(i)>tvec(j-1)
            dTdm(i,j)=dTdm(i,j)+(t(i)-tvec(j-1))*myG(B(i,j));
        end
    end
    end
end
```

The following sub function takes the derivative of surface temperature with

respect to *h*.

```
function [dTdh]=mydiffTdh(alpha,k,t,R,mvec,tvec,B,diffGdh)
L=length(t);
dTdh1=mvec(1).*t.*diffGdh(alpha,k,t,B(:,1));
dTdh2=zeros(L,1);
for j=1:R-1
    for i=1:L
```

```
        if t(i)>tvec(j)
            dTdh2(i)=dTdh2(i)+mvec(j+1)*(t(i)-
tvec(j))*diffGdh(alpha,k,(t(i)-tvec(j)),B(i,j+1));
            dTdh2(i)=dTdh2(i)-mvec(j)*(t(i)-
tvec(j))*diffGdh(alpha,k,(t(i)-tvec(j)),B(i,j+1));
        end
    end
end
dTdh=dTdh1+dTdh2;
```

The following sub function calculates beta.

```
function B=mybeta(h,k,alpha,t,tvec,R)
B(:,1)=h/k*sqrt(alpha*t);
for i=1:R-1;
    B(:,i+1)=myheaviside(t-tvec(i))*h/k*sqrt(alpha*(t-tvec(i)));
end
```

The following sub function calculates the surface temperature response.

```
function T=myseriesrampfunc(T0,t,R,mvec,tvec,B,myG);
L=length(t);
T1=mvec(1).*t.*myG(B(:,1));
T2=zeros(L,1);
for j=1:R-1
    for i=1:L
        if t(i)>tvec(j)
            T2(i)=T2(i)+mvec(j+1)*(t(i)-tvec(j))*myG(B(i,j+1));
            T2(i)=T2(i)-mvec(j)*(t(i)-tvec(j))*myG(B(i,j+1));

        end
    end
end
T=T0+T1+T2;
```