1

# Detecting a Critical Transition in Flames with Varying Air-Fuel Ratios

Presented to the Faculty of Lycoming College in partial fulfillment
of the requirements for Departmental Honors in
Astronomy and Physics

By
David M. Surmick
Lycoming College
April 25, 2012

Approved by:

Dr. Christopher W. Kulp

Dr. David G. Fisher

Dr. Charles H. Mahler

Dr. Gene D. Sprechini

Dr. Santhusht S. DeSilva

## Table of Contents

## Table of Figures

## Abstract

The long term goal of this study is to develop simple mathematical models of combustion with direct applications in industrial burners. This will be accomplished by studying the Air-Fuel Ratio (AFR) of flames collected from video data. The computational goal of this study focused on recreating the phase space of the flames. Methods of doing this used the false nearest neighbors test. The results of this test were used to determine the complexity of the flames, as they appear to be highly nonlinear. In addition to the computations, this study developed several experimental techniques that will aid in the development of mathematical models of combustion. Flames in various AFR states were produced by a premixed gas burner and were video recorded. Time series data of the flames were produced from the video recordings. Those time series could be analyzed to obtain information about the flame's dynamics. In this project we focus on developing the necessary experimental and analytical techniques needed to accomplish this long term goal. An experimental procedure was established for controlling the AFR of the flames. The procedure allowed us to capture quality video recordings of flames in several AFR states. Techniques for generating time series from the video recordings were also established. These techniques allowed us to differentiate between the flame and the background in the video. Finally, we preformed several time series analysis algorithms to begin the process of understanding the dynamics of the flames in various AFR states.

# 1 Introduction

Flames have always been a point interest for humanity. Studies of combustion have been conducted in the past and mathematical models have been formulated from them. These studies use the Navier-Stokes equations to accurately describe the dynamics of flames [1, 2, 3, 4, 5]. Though they provide a detailed understanding of flame dynamics, the Navier-Stokes equations are extremely computationally intensive and are impractical for use in real world applications that require manipulations of flames in real time. The goal of this study is to develop simple models of flame dynamics that are easily used in real world applications such as improving the efficiency of industrial boilers. This study was performed in collaboration with the Oak Ridge National Laboratory.

The basis of combustion is the burning of fuel and air in a mixed ratio. This study attempts to capture essence of flame dynamics through a quantity known as the Air-Fuel Ratio (AFR). The AFR is a description of the mass ratio of the air-fuel mixture creating the flame. The AFR is the mass of the air divided by the mass of the fuel in the mixture. In this project, the fuel is natural gas. The stoichiometric mixture is the AFR where there is exactly enough air to burn all of the fuel in the mixture. For natural gas, the stoichiometric mixture has an AFR value of 17.2:1. The stoichiometric mixture is the most efficient state of the flame. The flame is at its hottest when in this state. Another useful tool in describing flame states is the equivalence ratio. The equivalence ratio, $\phi$, is the AFR of the flame divided by the AFR of the stoichiometric mixture. Hence, an equivalence ratio of $\phi=1$ represents the stoichiometric mixture. If the equivalence ratio is less than 1, then the AFR is less than the stoichiometric mixture and the mixture is said to be rich. This implies that there is too much fuel (or too little air) in the mixture. If the equivalence ratio is greater than one the flame is in a lean state. This implies that there is

too much air (or not enough fuel) in the mixture. This study is based on the assumption that the dynamics of the flame can be characterized simply by the equivalence ratio and that equivalence ratio can be estimated through images of the flame. It is the hope of this study that the "simple mathematical models" will be dependent on only one parameter, the equivalence ratio.

The focus of this study is based on previously completed but unpublished work related to the topic of increasing the efficiency of flames in industrial settings. Flames are most efficient when in the stoichiometric mixture because there is no loss of extra fuel and there is no loss of control and temperature due to excess air. Therefore, this study aims at developing mathematical models for the purpose of controlling a flame to the stoichiometric mixture. Unpublished previous work has shown evidence for a qualitative change in the dynamics as the stoichiometric mixture was approached. This evidence is apparent from changes in correlations in the data through autocorrelation calculations. This previous work suggests that there might be a critical transition occurring in the flame near $\phi=1$.

The study described in this paper took videos of flames in various AFR states. This is not a novel concept as previous studies have implemented this type of analysis before. Like the studies described in these papers, previous studies have also used video images of flames to learn more about them [4,5]. This study differs from previous works in one key aspect though. The goal of this study is to generate simple mathematical models of combustion where as the previous works used the old form of analysis with the Navier-Stokes equations.

### 1.1 Time Series

This study is based on creating time series from a video recording of a flame. A time series can be defined as a series of events that corresponds with time-ordered observations of a system.

Essentially, a time series is a data set. Time series are samplings of systems. While the systems sampled can either be continuous in nature or discrete, the series themselves are discrete. In this project, a video taken of a flame is turned into a time series through techniques developed in this study. Those techniques are discussed in this paper. Though video cameras record data and play it back in what appears to be a continuous fashion, they actually record data in discrete intervals. This occurs because a "continuous video" is actually a series of discrete frames. This means that every frame of the video is a discrete, time ordered measurement of whatever is being observed. Consider a video camera that records data at a rate of 30 frames per second (fps). For a 30 second video, this camera would produce 900 frames. Through methods discussed later in this paper, each frame of a video will be turned into a data point in a time series. Thus, if the videos produced above were used as a time series, the length of their time series would be 900 data points. The length of the data set has implications on the calculations that may be made on the time series. Time series analysis algorithms require long enough data sets in order to calculate a specific quantity or characteristic. If a series is too short, then "small sample effects" can arise. These effects can return false results. Knowing how much data is enough is a tricky endeavor. In this study the video lengths began at 30 seconds and ended at 5 minutes of flame video. As will be discussed later, the process of converting video into time series is computationally expensive and a balance needed to be found between having enough data and computational efficiency.

## 1.2 Probability

Probabilities are an integral part of analyzing a time series and understanding the dynamics of many systems. They are used extensively in many applications throughout the sciences, including this study. It is therefore beneficial to review the terminology and symbology of

probabilities as they are related to this study. The probability that an event, *x*, occurs is denoted as *p(x)*. Consider a six-sided die as an example. The probability that one of the sides comes up as a 3 when the die is rolled is $\frac{1}{6}$. In this case, the event is a measurement of *x*=3 with *p(x)*=$\frac{1}{6}$. This is calculated by dividing the frequency of an event of interest (a three on the die) by all the possible events (number of the numbers on the die) and is shown as follows:

$$p(3) = \frac{\#\ of\ 3's\ on\ the\ die}{\#'s\ present\ on\ the\ die} = \frac{1}{6}.$$

The multiplication of probabilities is inherent to the calculations being made in this study. This is because of what the multiplication of probabilities represents. The operation of multiplication on probabilities is equivalent to the probability of two independent events occurring. Consider having a lottery ticket which requires two numbers be picked correctly in order to win on the ticket. The probability of winning on this ticket is based on the multiplication of the probabilities of picking each number. If the probability of picking each number is one in a hundred, then the probability of winning on the ticket is

$$\frac{1}{100} * \frac{1}{100} = \frac{1}{10000}.$$

A probabilistic quantity that will be important in developing later calculations is the joint probability. The joint probability is the probability that two events, *x* and *y*, occur together. This quantity may be symbolized as $p(x, y)$. The joint probability is not limited to two events and can be used for as many events as needed. This probabilistic value is calculated exactly like the multiplication example above if two events, *x* and *y*, are independent events.

The sum of all probabilities for all events in a given system must be equal to one. In a situation where an event is never observed to occur, the probability has a value of zero, $p(x)=0$. Mathematically the "conservation of probability" is expressed as a summation over all the possible events $x$

$$\sum_{x} p(x) = 1.$$

### 1.3 Information Theory

Many of the calculations in this study are based on information theoretic values, thus it is useful to discuss information theory. Information Theory was developed by Claude Shannon in 1949 [6, 7, 8, 9]. He used it as a method for quantifying signals and eliminating noise from them. The basic tenet of information theory is the very definition of information. Information is defined to be a decrease of uncertainty. Decreasing the uncertainty that an event will occur, increases the probability of that event; *i.e*. if an event happens more frequently in a system and has a high probability of occurring, a measurement of this event is not highly uncertain as compared to an event with a smaller probability of occurring. This analysis may be applied to the measurement of events in a system and the information gained from them. If an event occurs frequently in a system, it provides little information about the dynamics of the system because a frequently occurring event is expected to occur. If an event that does not frequently occur is observed, more information is learned about the system and its dynamics. Knowing the probability that an event occurs, then becomes extremely valuable in gaining information about a system.

Information Theory requires that a series be symbolized in order for calculations to be made on it. If a binary symbolization was being used, each event or measurement in a system

could be symbolized as either a one or a zero by using a threshold. The threshold is a value such as the mean of the series. Elements in the series greater than the threshold are assigned the symbol 1 and elements less than the threshold are assigned the symbol 0. A series is not limited to two symbols. In this case, more thresholds would be required. Series need to be symbolized because Information Theory uses probabilities to analyze the series. Symbolization reduces the number of unique elements in the series. Information Theory algorithms work best when there are few unique elements instead of having many elements with lower probabilities. In what follows the series has already been symbolized.

Mathematically the information gained from an event is notated as [6]

$$I(x) = -\log_a[p(x)].$$

The first thing that pops out when looking at this definition is the minus sign which begs the question is information inherently negative? The answer is no. This is a mathematical artifact because the logarithm of a number that is less than one will always be negative regardless of the base. This is done to make information a positive quantity. Shannon presented a detailed and thorough argument for the mathematical expression for information, which is beyond the scope of this project [6]. The base of the logarithm determines what units are used for the information. Units in base ten are the familiar digit. If a base-2 logarithm is used, then the unit for the information is the bit. Hence, if the probability of a measurement is $\frac{1}{2}$, then the information it carries is 1 bit. In other words, that measurement can be represented by one binary digit. If the probability of an event occurring is $\frac{1}{4}$ then the information is 2 bits and the measurement requires two binary digits to be represented; 01 for example. Measurements based on the natural

logarithm (base *e*) have units of dits or dats. Now consider a system that has possible outcomes of 0's, 1's, and 2's and the flowing string of these symbols is observed by the system:

0 1 1 0 2 0 2 0 0 0 1 2

With this observation the probability of a two is 0.25 (25% chance of occurring), of a 1 is 0.25, and of a 0 is also 0.5. In this example, a measurement of 0 gives more information about the system than a 1 or a 2. This is reflected in the calculation of the information associated with each observation. The information associated with each event is as follows:

$$I(0) = -\log_2(.5) = 1\ bits$$

$$I(1) = -\log_2(.25) = 2\ bits$$

$$I(2) = -\log_2(.25) = 2\ bits$$

As has been alluded to, the value for the information is the number of digits of whatever base unit being used required to represent the measurement. In this example 1 bit of information would be required to represent a measurement of 0 and 1, and 2 bits of information would be required to represent a measurement of 2. This example also illustrates that events that occur less frequently (the event of 2 in this example) provides more information about the system. A measurement of zero could be represented with the binary number 0, one by 01, and two by 11.

Shannon addresses the issue of the average amount of information stored in a time series by introducing the concept of the entropy. The definition of entropy in terms of information theory is a measurement of the average amount of information associated with an event. Shannon defined a special quantity that averages the information of entire system. Called the Shannon entropy, *H* is defined as

$$H = \sum_x p(x)I(x) = -\sum_x p(x)\log_a[p(x)] \qquad (1)$$

where the sum is over the all the measurement values of the symbolized data set. The Shannon entropy is the minimum number of expected digits on average required to represent a measurement. This is also the average information of the system. From the previous example, the minimum required digits to represent that system would be

$$H = -\sum_{x} p(x) \log_a [p(x)] = 0.5 * 1 + 0.25 * 2 + 0.25 * 2 = 1.5$$

This gives the average number of binary digits needed to encode all the measurements into binary. Hence, a series with a large value for the Shannon Entropy means that each measurement requires more binary digits in its representation. This implies that the series is more complex and carries more information than a series with a smaller value for the Shannon Entropy.

### 1.4 Mutual Information

The simple mathematical models developed in this study will rely on the calculations made to determine correlations and directions of correlations within the flames; thus it is important to develop the mutual information. Mutual information is an information theoretic measurement that is based on the Shannon entropy [8, 9, 10]. This information theoretic quantity looks at two series, *X* and *Y*, and detects nonlinear correlations between them. It is the amount of extra information required to represent the systems as independent, as compared to the case when the series *X* and *Y* are related. This may be thought of as measuring a nonlinear correlation between the two series of events *X* and *Y*. Suppose that there are two systems which generated measurements simultaneously, such as the closing value of the Dow Jones and the DAX Market. We ask, whether or not knowing the current value $x \in X$ (the Dow Jones) gives any information about the current value $y \in Y$ (the DAX) or vice versa. The mutual information tells us if the

information from one series, *X*, is exchanged between series *Y* (or vice versa). However it does not tell us the direction of the information that is exchanged.

The mutual information is based on the comparison of two suppositions. The first is the two series, *X* and *Y*, occurring as independent to each other. The second supposition is that the two series are allowed to be dependent on each other. For each supposition there will be an accompanying Shannon entropy. The mutual information compares is essentially the Shannon entropy of the two events assuming that they are independent of each other ($H_1$), minus the Shannon entropy assuming that they are related ($H_2$).

The" independent" Shannon entropy is:

$$H_1 = -\sum_{x,y} p(x,y) \log[p(x,y)] = -\sum_{x,y} p(x,y) \log[p(x)p(y)]. \qquad (2)$$

$H_1$ gives the average number of digits required to symbolize each element of the series *X* and *Y* assuming independence. A second Shannon entropy ($H_2$) is used to describe the case where events *x* and *y* are not required to be independent of each other,

$$H_2 = -\sum_{x,y} p(x,y) \log[p(x,y)]. \qquad (3)$$

The Shannon entropy, $H_2$, gives the average number of digits needed to symbolize *X* and *Y* if they are not independent of each other.

The mutual information is then

$$M = H_2 - H_1 \qquad (4)$$

$$H_2 - H_1 = -\sum_{x,y} p(x,y) \log[p(x)p(y)] + \sum_{x,y} p(x,y) \log[p(x,y)]$$

$$= \sum_{x,y} p(x,y)[\log[p(x,y)] - \log[p(x)p(y)]]$$

$$M = \sum_{x,y} p(x,y)\ [\log[\frac{p(x,y)}{p(x)p(y)}]]$$

and gives the extra number of digits on average needed to symbolize *X* and *Y* assuming they are independent. Or put another way, this calculation will give the amount of information that is exchanged between the two series. If the mutual information, *M*, is zero then $p(x,y) = p(x)p(y)$, and there is no correlation between the two series and they are independent of each other. Values greater than zero determine the strength of the nonlinear correlation between the two series.

As an example, consider two strings of symbols which were generated from the same source. Each string of symbols consists of only two possible outputs, 1 or 0. The outputs of these systems are displayed below.

$$System\ X\text{:}\quad 1\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0$$

$$System\ Y\text{:}\quad 0\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 1$$

The mutual information may be applied to these two strings of symbols to see how strongly correlated they are. First consider the probabilities of each output in both systems.

$$p_X(0) = 0.4$$

$$p_X(1) = 0.6$$

$$p_Y(0) = 0.6$$

$$p_Y(1) = 0.4$$

The subscripts on each probability indicate which string of symbols they were calculated from. Now consider pairs of symbols between the two series. The subscript on these tabulations indicates which series the output originated from.

$$p(0_X, 0_Y) = 0.3$$

$$p(0_X, 1_Y) = 0.1$$

$$p(1_X, 0_Y) = 0.3$$

$$p(1_X, 1_Y) = 0.3$$

These probabilities are joint probabilities of both of the indicated outcomes occurring. This information may now be used to calculate the mutual information of the two strings of symbols. Base two logarithms have been used for ease of calculation and double checking the result.

$$M = 0.3 \log\left(\frac{0.3}{0.4 * 0.6}\right) + 0.1 \log\left(\frac{0.1}{0.4 * 0.4}\right) + 0.3 \log\left(\frac{0.3}{0.6 * 0.6}\right) + 0.3 \log\left(\frac{0.3}{0.6 * 0.4}\right)$$

$$= 0.03536$$

From this result it appears that the two systems are weakly correlated because there is a nonzero result for the mutual information. However, this example is only meant to illustrate the procedure of calculating *M.* They are weakly correlated though, because of the size of the result. The example series used are short. This opens the door to small sample effects which can create non-zero values of the mutual information when in fact no information is exchanged. There are means for detecting these effects [9, 11].

### 1.5 Phase Space

The analyses of this study were focused largely on recreating the phase space of the flame dynamics for various AFR states. The phase space plots the velocity of a system versus the position of the system [12]. Valuable information may be learned from the phase space of a system and this information is learned qualitatively by studying the plots of the phase space and how these plots change as different parameters of a given system are changed.

The phase space plot of a system has many, varied applications and is a good first analysis to perform on a system because much information may be determined from the phase space. One such behavior that is easily captured in the phase space is a critical transition. A critical transition (also known as a bifurcation) is a qualitative change in the dynamics of a system. This change in the dynamics occurs at a single instant and may or may not be a reversible process, depending on the system. A critical transition can easily be detected in a phase space plot as a qualitative change in the plot. This study is aimed at investigating flame states near a critical transition, which is why the phase space is a natural first analytical step.

An example system to illustrate the analytical abilities of the phase space is the undamped and damped harmonic oscillator. First consider the case with no damping. Physically this would represent a mass on a horizontal spring that never stops moving once it has been set in motion and always has the same amplitude of its oscillation (*i.e.* there is no friction in the system). The equation that governs the system state is of the form

$$\ddot{x} = -{\omega_0}^2 x,$$

where ${\omega_0}^2$ is an invariant parameter of the motion corresponding to the natural frequency of oscillation. After numerically solving the equation of motion on a computer using initial conditions of $x(0) = 1$ and $\dot{x}(0) = 0$, the plot of the phase space may be made.

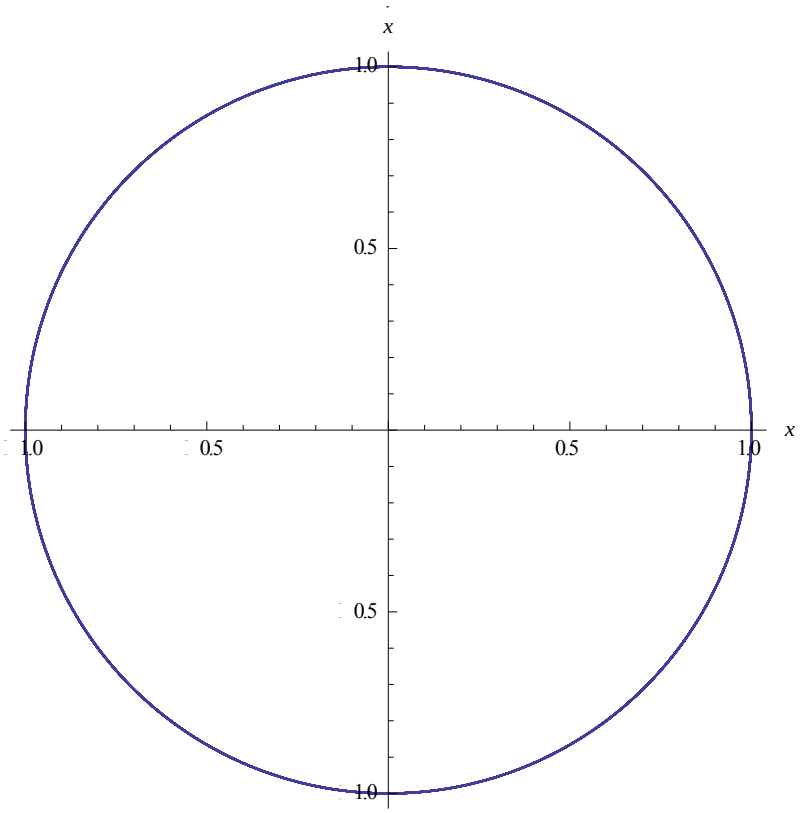


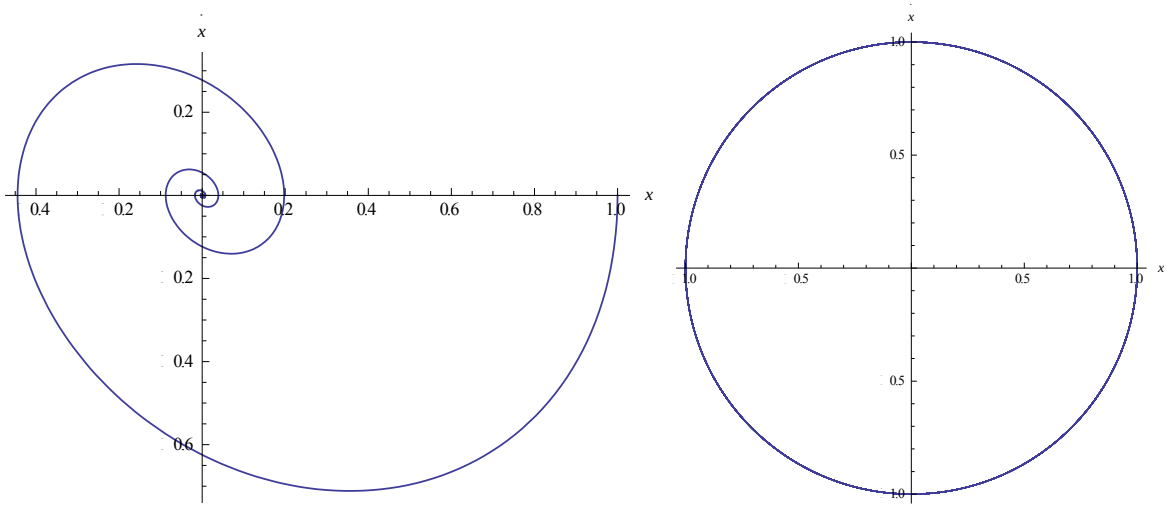

Figure 1. Plot of the phase space for an undamped harmonic oscillator. This plot is a circle of radius 1. The radius is dependent on initial conditions.

Figure 1 is a plot of the phase space for the undamped harmonic oscillator. The motion of the harmonic oscillator is periodic in nature. Intuitively the shape of the phase space could be worked out by thinking about the repetitive nature of the oscillator. As the harmonic oscillator moves back and forth it repeats the same motion over and over again (*i.e.* it continuously returns to its original position values of $\dot{x}$ and $x$). The object in phase space that best describes this motion is a circle. This is affirmed by studying the phase space plot. The plot shows a circle which describes the motion of the oscillator through time. Each rotation the circle represents one full oscillation.

Now consider the case where the harmonic oscillator is damped. The equation that governs the motion of this case is

$$\ddot{x} + 2\beta\dot{x} + {\omega_0}^2 x = 0.$$

The middle term of the equation is called the damping term as this term introduces the damping to the motion. This term can be considered as a variant parameter to the equation. Parameters such as these are how critical transitions occur in systems. They occur at a single point which correlates to a specific value for the given parameter. Critical transitions are qualitative and quantitative changes in the system. In this example the point at which the parameter changes the system is any nonzero number. In other words the critical vales of $\beta$ is $\beta_c = 0$. As soon as this parameter is introduced, the dynamics of the system change. In this case the amplitude of the oscillation of the pendulum decreases based on how large the damping coefficient is.




$\beta > 0$ $\beta = 0$

Figure 2. Phase space of the damped harmonic oscillator next to the undamped case. The initial conditions for these plots are $x(0) = 1$ and $\dot{x}(0) = 0$.

Figure 2 is a plot of the phase space of the damped harmonic oscillator next to the phase space plot for the undamped case. When the damping parameter is zero, the phase space of the damped case is exactly the same as the case with no damping. When there is damping the motion of the harmonic oscillator is accurately described. As the harmonic oscillator continues to be damped, the amplitude of its oscillation continually decreases until the motion of the harmonic oscillator stops. This sort of interpretation can be applied to all phase space plots. Figure 2 also shows the idea of a critical transition. Notice that the phase space qualitatively changes as $\beta$ changes. The critical transition (or bifurcation) is illustrated by a qualitative change in the phase space plot.

As it has been presented thus far, the phase space has been limited to two dimensions, however; if a system has more than one degree of freedom its phase space will have a higher dimension. When the leap is made to higher dimensions a new plot called the phase portrait is used. Instead of plotting against the first time derivatives, phase portraits plot simply in the coordinate plane for two of the system's variables. The phase portrait is a subspace of the phase space. When dealing with a large dimensional phase spaces it is often necessary to consider two or three dimensional "slices" of the phase space. These slices are known as Poincaré sections and can be thought of as a sampling of the phase space at certain time intervals. When considering “slices” of the phase space of a system, it is helpful to know how many dimensions there are in the system being considered. Based the results of this study, it is believed that Poincaré sections will be needed to accurately depict the phase space of the flames in this study.

In this project the times series generated from the flame videos will be used to reconstruct the phase space of the flames. The goal of this project is to bypass the complex equations used to describe combustion, hence data is needed from the flames to generate the phase space plots.

Well known techniques of phase space reconstruction which use time series measurements from the system of interest will be utilized to accomplish this [13].

# 2. The Experiment

The long term goal of this study is to use video recordings of flames in both constant and variable AFR states in order to develop simple mathematical models of combustion. To do this, flames in various AFR states were recorded using a video camera. Using the techniques outlined in this section, time series were generated from the flame videos. It hoped that the analysis of the resulting time series will lead to simple mathematical models of combustion. The methodology for the experiments performed in this study is a significant expansion of unpublished previous work done by Dr. Christopher Kulp. The results of these experiments and unpublished results were the impetus for this study.

## 2.1 Prior Work

The work and direction of this project is based off of previously completed unpublished work by Dr. Christopher Kulp. This section is a brief synopsis of that work. This study is aimed at increasing the efficiency of industrial burners by keeping their flames near the stoichiometric mixture. It was the purpose of his study was to investigate whether or not it is possible to produce useful time series from the videos of flames in various AFR states. This was done by video recording a flame from a Bunsen burner while its AFR was changed manually. By opening the slits on the sides of the burner more air was allowed to flow into the mixture. The AFR of the mixture for these flames was unknown. The videos were then used to produce time series by

looking at the pixels that make up the flame in each frame. The color of each pixel is made up of a specific intensity of red, green, and blue and with values ranging from 0 to 255. These values are known as RGB triplets. Each frame can be represented by a collection of points in "RGB space". A means of representing each frame as a single RGB triplet was necessary. This was accomplished with the development of the center of mass (COM) of each frame. The COM of the flame is just the average value of each color channel: red, green, and blue, for each frame of video data. In variable flame situations the COM plots were valuable in showing how the flame changed as its AFR was varied throughout a given video.

Results from previous COM plots showed that interesting dynamics occur near the stoichiometric mixture. Recall that the stoichiometric mixture is that AFR state in which there is enough air to exactly use all of the fuel being burnt by a flame. Plots of the COM data versus time show a dip in the autocorrelation (a measurement of self-similarity in a time series) near what was suspected to be the stoichiometric mixture. (Prior to this study the stoichiometric mixture was estimated by the emergence of a blue cone in the interior of the flame.)

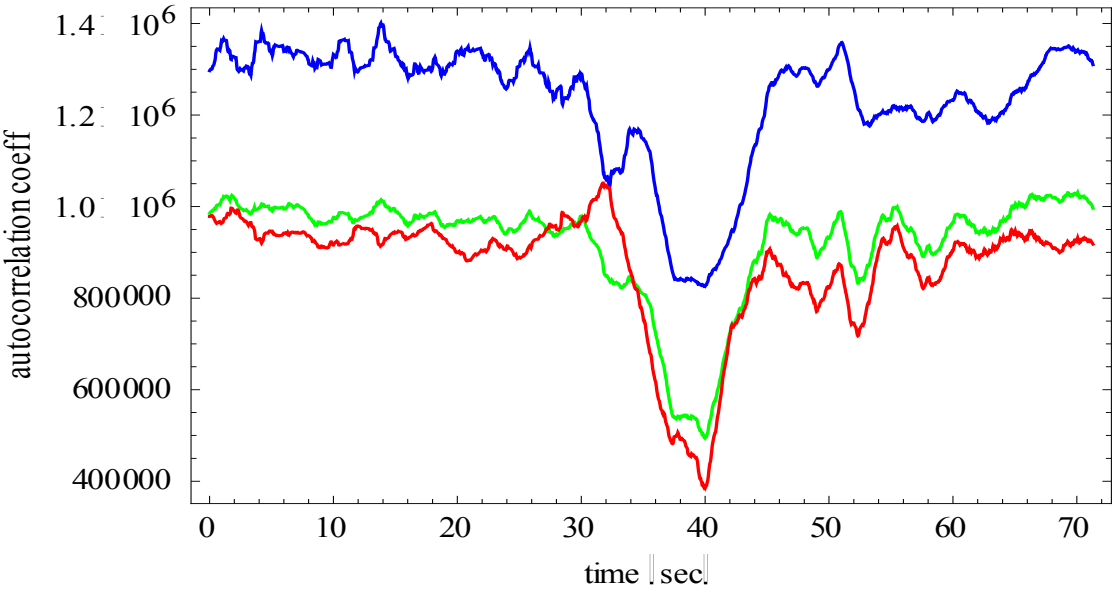



Figure 3. Autocorrelation plot of a variable AFR video that was transitioned from lean to rich.

There is a considerable drop in the autocorrelation COM data near what was approximated to be the stoichiometric mixture. This result suggested that a critical transition might be occurring at the stoichiometric mixture of a flame. This result is the impetus of the work of this study.

The experimental goal of the current study is to collect flame data for all three flame states; lean, rich, and variable. By targeting rich and lean states near the stoichiometric mixture (criticality), much can be learned about the state of the flame near the stoichiometric mixture of the flame. This study expands on the previous study by having a way to quantify the AFR for a flame state. This is done with new pieces of equipment that are on loan from the University of Tennessee and Oak Ridge National Laboratory that measure the flow rates and pressures of the gas and the air in the flame. The more precise apparatus for this study allows for more quantifiable data to be taken and more varied analysis may now be performed on the data. A major part of the experimental work done in this project was setting up and learning to use the equipment, as well as developing new experimental techniques for collection of data. This study also goes beyond the simple analysis of the previous work.

## 2.2 The Apparatus

The new equipment used in this project was critical to accomplishing the experimental goal. The major impact that this equipment had on the study was the ability to measure and control the flow rate of air and gas into the flame. There were several components to the apparatus for the experiments performed in this study. They consisted of: the air compressor, pressure and flow rate gauges, a rack for the pressure and flow rate gauges, a fume hood, a high temperature

burner, a video camera, and a chimney for the burner. All of these pieces needed to be assembled or integrated into the apparatus for controlling the mixture of the flame.

The air and gas was set up to flow through a rack of pressure gauges and flow meters. The air and gas were connected to the rack of pressure gauges and flow meters and then eventually to the burner with 5/8 inch plastic tubing. Each connection with the tube was secured with a pipe clamp of the appropriate size. After the clamps were applied to the connections, the connections were checked for leaks. This was done by listening for air passing through the leaks. When this occurred there was a hissing sound. Also, a mixture of soap and water was applied to the areas with connections and then compressed air was passed through the connection. If there was a leak the watery soap mixture would form bubbles and identify a leak. The leaks were fixed by tightening the clamps. Checking for leaks occurred on an as needed basis and was fixed with relative ease during data taking sessions. The air and the gas entered the rack at the bottom of the rack. From there the air was read by the pressure gauges and then passed through the flow meters as it left the rack and entered the burner. The plastic tubes connecting the pieces of the apparatus were made sure to be as short as possible to make sure that the pressure did not drop significantly before it entered the flame.

The experiments for this study were performed under a fume hood with a vent. The hood was used to keep gas from leaking out into the room as a safety precaution. The hood was equipped with several pieces of equipment needed to perform the experiment. This included the source for the natural gas and an electrical outlet. The source of gas for the flame was a common laboratory natural gas jet. Air was added to the flame using an air compressor. Pressurized air was used in these experiments to maintain a steady and controllable supply of air to the flame. The AFR of a flame is determined by the relative pressures of air and gas and the relative flow

rates of air and gas in the flame. The air was compressed using the standard air compressor and the air and gas flow rates were measured using two different flow rate meters. Figure 4 is an image of the air compressor used in this study along with the rack of pressure and flow rate gauges. Figure 5 is an up close image of the rack of pressure gauges and flow meters and Figure 6 is an image of the back of the rack.


Figure 4. Image of the air compressor and equipment rack used in the experiment.


Figure 5. Up close image of the front of the rack of pressure gauges and flow meters.


Figure 6. Image of the back of the rack of pressure gauges and flow meters. The two sets of meters on the right of the image were used in this study.

Since the air was compressed it came out a pressure greater than atmospheric pressure. The gas came out at atmospheric pressure. The pressure gauges used in this study were RobertShaw Fulton Controls Company Acragage pressure gauges. The pressure gauges read the gauge pressure, meaning they read the pressure of the system relative to atmospheric pressure. In addition to reading at gauge pressure, the pressure gauge that was used to read the pressure of the air entering the flame needed to be corrected by half of a psi (pound per square inch) due to an off-set. The psi is the unit that was used in both pressure gauges. Figure 7 is an image of the pressure gauges for the air and the gas entering the flame.

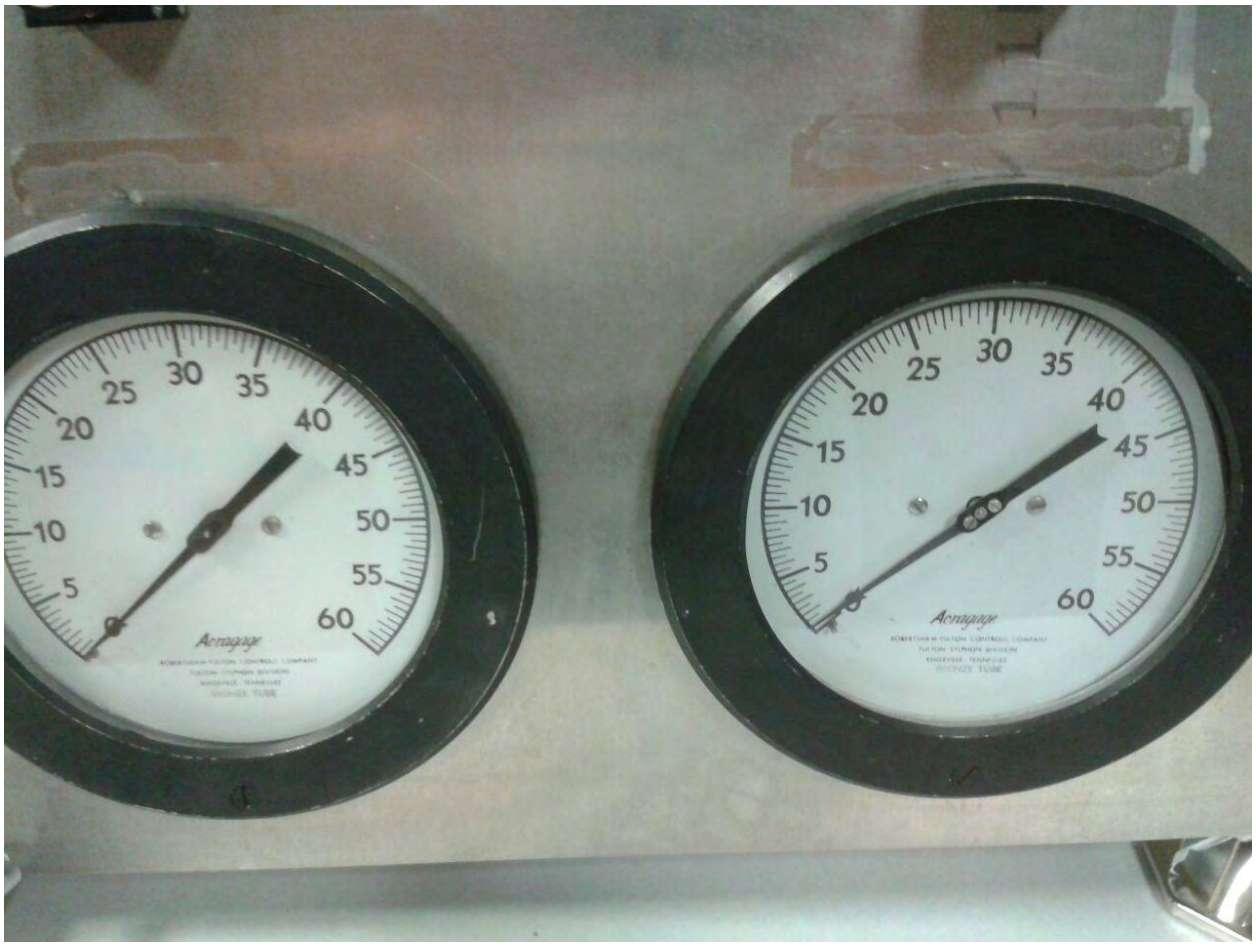



Figure 7. Image of the pressure gauges used in this study. The pressure gauges were RobertShaw Fulton Controls Company Acragage pressure gauges. The gauge on the left was used to measure the gas pressure and the gauge on the right was used to measure the air pressure in the flame. Note, the half psi off-set can be seen in the air pressure gauge on the right.

As was previously stated, two different types of flow meters were used. The flow meter used for the air was a ShoRate 1355EGL7CFGA flow meter and the flow meter used for the gas was an Omega Engineering FL-3404G-NV flow meter. Each flow meter used a different scale for measuring the flow rate. The important thing to note about the flow rate is that is the mass rate of flow for the air and the gas. A ShoRate flow meter was used to measure the flow rate of the air in the flame. This meter read the flow rate in units of standard liters per minute (SLPM). An Omega Engineering flow meter was used to measure the flow if the gas in the system. This flow meter read in units of millimeters. This unit need to be converted to the standard SLPM before a calculation of the system's AFR could be performed. The conversion chart was provided by the manufacturer. This conversion chart is provided in Appendix C. Figures 8 and 9 are images of the flow meters used for the air and gas respectively.

A correction needed to be made to the flow rate calculation because temperature and pressure affect fluid flow rates. The flow meters are calibrated for air at standard temperature and pressure. The equation that was used to determine the flow rate of the air and the gas is as follows

$$F_{actual} = f F_{measured} \qquad (5)$$

where $F_{measured}$ is the measured flow rate, $F_{actual}$ is the actual flow rate, and $f$ is a correction factor. This correction factor comes from the ideal gas law. The formula for $f$ is as follows

$$f = \sqrt{\frac{MW}{MW_{std}} \frac{P/T}{P_{std}/T_{std}}}$$

where $P$ is the measured pressure, $T$ is the measured temperature, $P_{std}$ is the standard pressure of one atmosphere, and $T_{std}$ is the standard temperature of 20℃. The flow rate is affected by changes in temperature and pressure of the flame. Equation 5 was programmed into a computer

to return the AFR of the flames with the appropriate substitution of *f* as well as converting mm to SLPM.

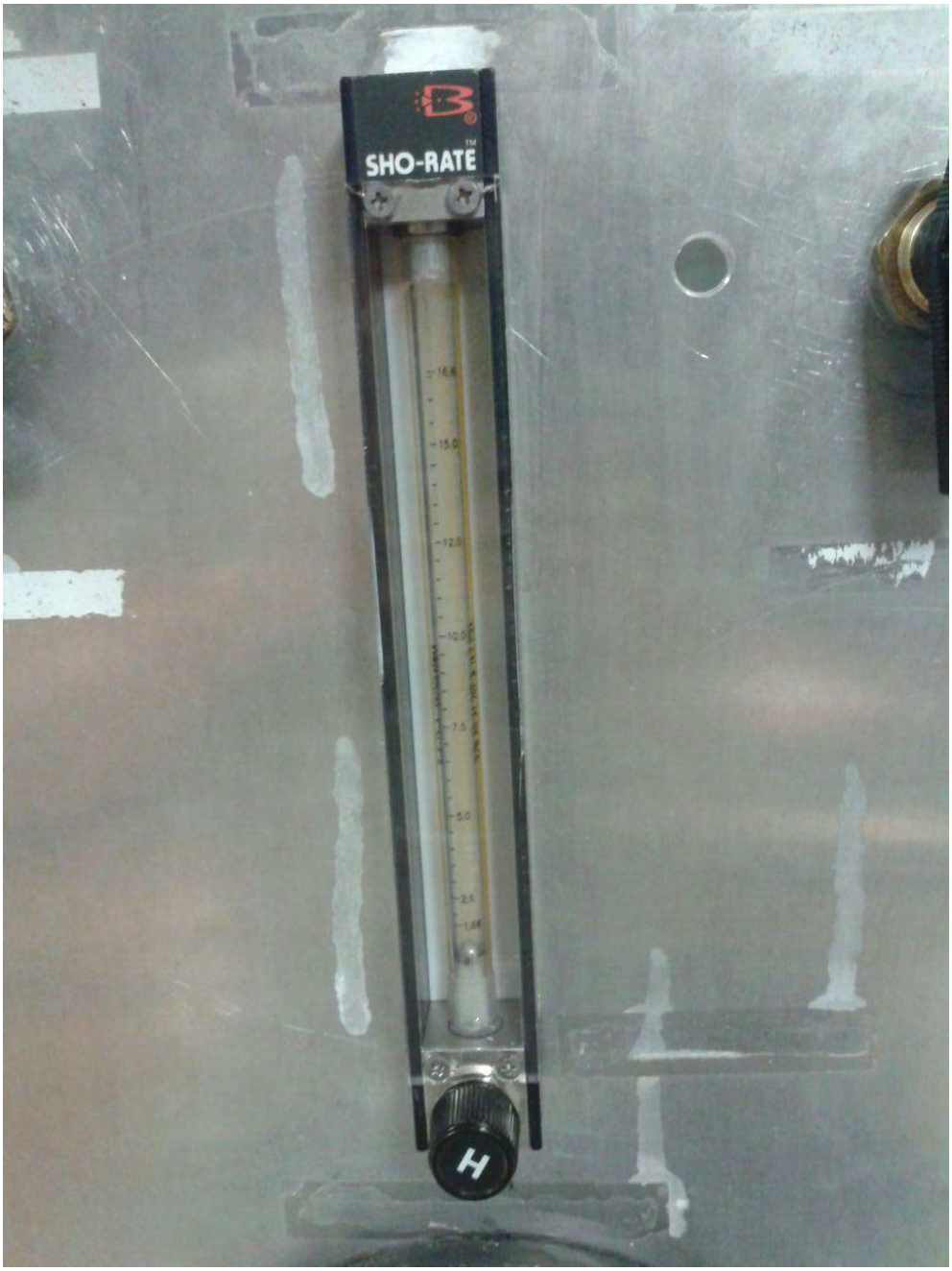



Figure 8. Image of the flow meter used for the air in the flame. The flow meter was a ShoRate 1355EGL7CFGA flow meter.

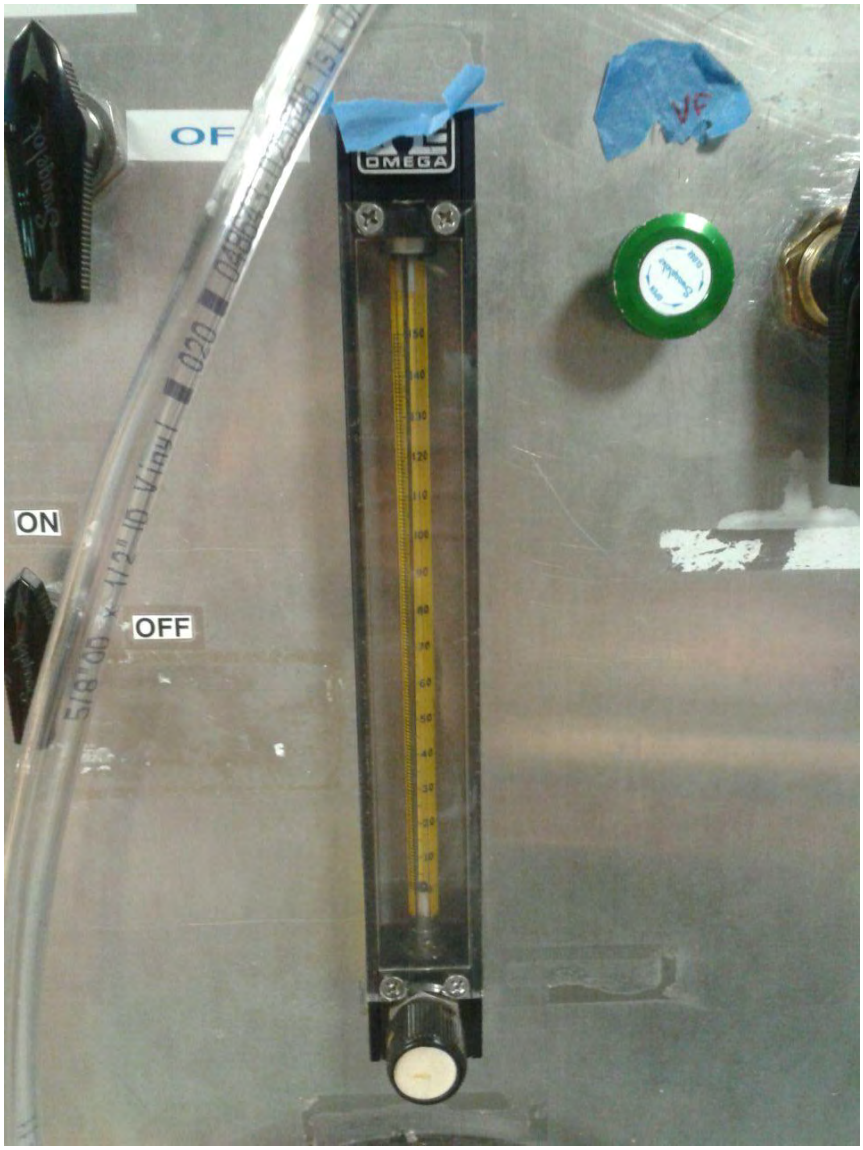



Figure 9. Image of the flow meter used for the gas in the flame. The flow meter was an Omega Engineering FL-3404G-NV flow meter.

The burner used in this study was a high temperature burner. The bottom of the burner was open via slits in the bottom of the burner. The system was closed by using plumbers tape to tape up the slits. The burner was lit and the flame emanating from it was video recorded. The video camera used in this study was an "off the shelf" Insignia NS-DV720P camcorder that was bought at BestBuy. The video camera took video at the rate of 30 frames per second. Because of

the air currents caused by the vent in the fume hood, the flame moved around quite a bit under the hood. This motion can add dynamics not naturally present in the flame. A moving flame was not ideal for videotaping a quality data set for data analysis. This was rectified by using a chimney. The chimney was placed over the burner after the flame was lit. The video was then taken through the clear face of the chimney. This face of the chimney was placed square to the camera so that no reflective affects distorted the videos of the flame. Figure 10 is an image of the set up for the chimney, burner, and video camera.

Figure 10. Image of the burner, chimney, and video camera.

The chimney went through several stages. Initial forms of the chimney were constructed out of sheets of Plexiglas that were super glued into the shape of a box. Holes were drilled in the

bottom of the box and metal legs were attached to the chimney. The dimensions of the final Plexiglas design were 15cm x 15cm x 50cm. This design was used until the sides of the Plexiglas began to warp as result of the high temperatures of the flame. A black background was placed behind the chimney hood when using this chimney. This was done to differentiate between flame and the background data. The black background was a sheet of plywood that was painted with flat black paint. The Plexiglas design was upgraded by constructing the next version of the chimney out of glass and thin aluminum sheets. The glass was purchased from Williamsport Mirror and Glass Company. The aluminum sheets constituted three of the sides of the chimney. These sides were constructed out of aluminum corners that had a sides of length four inches and height seven and three quarters inches. Six of these corners were constructed into three of the sides of the chimney. They were attached using thicker strips of aluminum which also turned into the legs of the chimney. Two of the legs were on the back of the chimney (opposite the glass). The other two legs were on the adjacent sides. The glass was tempered so that it would be heat resistant. The glass was the fourth side the chimney. The dimensions of the glass were four inches in length and nineteen inches in height. The aluminum sides of the chimney were a half inch longer than the glass. This is so that the glass could be attached to the three aluminum sides. Two aluminum strips (also a half inch longer than the glass) served as the clamps and applied the requisite pressure to hold the glass in place. The aluminum sides were painted using flat black paint, which no longer meant that the black background was required. When the glass was attached to the front of the chimney it was front heavy and unstable. This problem was fixed by making a wooden base with slots in it for the legs of the chimney. When the chimney was placed in the wooden stand it was stable and stood on its own. This is the current design of the chimney and is a more permanent form of the chimney because it uses tempered heat resistant glass

instead of Plexiglas as the side facing the video camera. Figure 11 is an image of the chimneys used in this study and Figure 12 is a close up image of the chimney with the burner in it.

Figure 11. Image of the chimneys used in this study. The middle chimney and the chimney to the left were used to take data. The glass is not in the chimney on the right. The chimney on the far left is the prototype for the middle Plexiglas chimney design and was not given legs.

Figure 12. Close up image of the current chimney design. The chimney pictured is the latest version of the chimney with the glass attached to the front of it and is in its position in the chemical hood. The burner is also in pictured in the position it was in for data runs. The white covering on the burner is the plumbers tape used to cover up the slits at the bottom of the burner. Also seen in the picture are inlets for the air and gas into the burner. They are in the lower left corner of the image.

Constructing the apparatus for this study is a key part of the results attained. Appendix A has more, larger, labeled images of the entire apparatus.

## 2.3 Experimental Methodology

Data sets were collected by recording video of the flame. Three different types of data sets were taken: constant lean flames, constant rich flames, and variable AFR flames. The analysis for much of this study has focused on flames in constant AFR states. The methodology for collecting constant AFR data and variable AFR data were slightly different.

Rich flame data were the easiest to take. The methodology behind taking the rich flame data was to "dial in" on the conditions prior to taking the data. This was accomplished by lighting the flame and placing the chimney over the flame. Then the flow rates of the air and gas were changed until a rich flame appeared to be rich. This was done visually. After the conditions were set, the readings from the flow meters and pressure gauges along with the temperature were entered into a computer program that calculated the AFR and equivalence ratio of the flame. The results of this program determined the exact state of the flame being observed (rich or lean). More often than not the visual description of the flame was correct. The visual description of the flame became questionable as the flame state approached criticality. After the AFR of the flame was confirmed to be rich, the video of the flame was taken. The length of the video taken ranged from thirty seconds to five minutes. The length was determined by the numerical analysis that was planned to be used on a particular data set. After the data sets were taken it was processed so that numerical analysis could be performed. Figure 13 is an image of a rich flame.

Figure 13. Image of a flame in a rich AFR state.

Initially the method for collecting the lean data was exactly the same as the method for collecting the rich video data. The method needed to change when a red spot began to occur in the lean flame. The red spot radiated out from the center of the flame and caused the color of the flame to change from its characteristic blue color to an orange red color which glowed very brightly and did not show up well on video. Because of the nature of the analysis that was to be performed on the video data, the flame not being in its characteristic color state caused several problems. It is believed that the red spot emerged due to the heat of the flame being produced and how compact the flame is when it in the lean state. The nature of the lean flame is that it is small and all of its heat is much closer to the burner. As a result of this the burner became too hot and the metal of the burner began to glow. This caused the flame to glow in a similar fashion. This effect was noticed in several flames and occurred after about a minute of video data, which in most instances was too short for a proper numerical analysis.

Attempts were made to reduce the effects of the red spot in the flame. None of these attempts were used to take a successful lean video of a length greater than a minute and a half. These attempts centered on cooling the burner before it was used. The best method for doing this was determined to be allowing air to run through the system the gas was introduced to the system. Methods for taking a good lean video of a length greater than one minute still have not been developed. It is likely that the red coloring may need to be handled during the process of generating time series from the videos.

The method for collecting the variable AFR video data was again similar to the method used to collect the rich AFR video data. Like the rich flame data, the initial flame state desired was a rich flame. This was accomplished in the same way as the rich flame videos. After the rich flame conditions were set and the flame was ignited, the rich flame was allowed to burn in a steady state for some time before the flame state was varied. The length of this time varied depending on the length of the entire video being collected at that particular time. The flame was then slowly changed from a rich state to a lean state. This was accomplished by reducing the flow rate of the gas in the flame using controls on the flow meters. After the flame state was varied it was allowed to stay in a lean state in a similar fashion to the rich state at the beginning of the video. The video was then processed for data analysis.

## 3. Numerical Analysis

After the video data were collected, it was then processed in many different ways in order to gain information about the desired flame states. AFR's of each flame were tabulated in the laboratory during data collection sessions so that the flame state video recorded was known and guided the data taking process as was needed. The first step in converting video to a time series was the

extraction of RGB values of the flame pixels in each frame. After the RGB values were collected they were then analyzed. The first sets of data had their phase spaces reconstructed. The autocorrelation method was abandoned for future studies because it produced confusing results. Instead the mutual information was used because it is more sensitive to nonlinear correlations. This calculation was done in concert with the false nearest neighbor, and modification of the false nearest neighbor analysis called the United Algorithm. The purpose of those was to give proper phase space reconstruction. A detailed account of the numerical analysis for set of code used on the RGB data follows. Appendix B contains the code for each analysis used in this study.

### 3.1 RGB Triplet Extraction and Data Filtering

The first thing that needed to be done was the COM coordinates needed to be extracted from each flame. The COM coordinates were chosen for the analysis because it is believed that they are a good representation of the state of the flame. Recall that the COM coordinate is a triplet value that represents of the average value for the red, green, blue color channels for the pixels that make up the flame. The triplet takes on the form of {red, green, blue} and each value ranges from 0 to 255. The COM for each frame of the video taken is the target of all of the analysis performed in this study.

The RGB coordinates were extracted using the computer algebra system Mathematica 8. Much of the code used to extract the RGB coordinates was code that was already built into Mathematica's software. The first thing that the code did was import the video flame videos. The videos were saved in .mov format. The data was imported using the Import command which imports data from selected files. The entire video needed to be imported. To ensure that the

entire video was imported, a Table command was used in the Import command. The Table command just organized the frames of the video into an ordered list beginning with the first frame of the video and ending with the last frame of the video.

After the video was in Mathematica, the background data needed to be subtracted from the video data. The first step in doing this was to cut the size of each frame down to a manageable size. This was done by subtracting pixels from each frame of the video which were known to have none of the flame in them. The pixels that were decided to be left out were picked from a representative frame of the video. However, not all the background pixels were removed. The exact pixels to be dropped from each frame were determined by moving the cursor over the image. The ImageTake command is what actually subtracted the pixels from the video frames. This command was mapped onto the entire video using the Function command. Mapping is a means of functional programming that increases the efficiency of the code. By mapping the ImageTake command to the video data it erased the need to manually perform the ImageTake command on each frame of the video. This type of functional programming was used whenever possible in this study to improve the efficiency of as much code as possible.

The next step in deleting the background was to threshold actual RGB values. The RGB values for each pixel on each frame of the remaining pixels of each frame were deleted from the frames using the ImageData command. The ImageData command produces several pieces of data about a particular image that range from the number of pixels in the frame to information on the color of the video. Each frame of the video was treated as a separate image. The particular option for ImageData that produces the RGB data for each pixel in a particular is the "Byte" call sign. This command was again mapped to the video data. The next set of background data subtraction occurred by deleting pixels that were considered to be black. Pixels that were black

had RGB values of {0,0,0}. These pixels were removed from the data set with the use of the DeleteCases command. Again, this code was mapped onto the entire video.

After all of the cropping was done, what was left of each frame was a list of RGB triplets of the flame in that frame. The COM data was relatively easy to produce from the RGB triplet data. All that needed to be done was to take an average of value for each color channel in each pixel. This was done using the Mean command which was then mapped to the data set. In the end the video was reduced to a list of the RGB COM of the flame in each frame.

Before the COM data could be exported, the data needed to be filtered. In general the data sets used in this study were relatively noisy. Noise refers to the number of perturbations from the characteristic behavior of the flame. Data sets were filtered using a mean filter. The mean filter filters the data by averaging the data over an overlapping window. Window sizes in this study were 15, 30, and 45 frames. These numbers were chosen because they represent a half second, a second, and a second and a half of data, respectively. The mean filter was coded using the built in MeanFilter package in Mathematica. After the data was filtered it was then exported using the Export command. This command has the ability for the user to choose what type of file is exported from the Mathematica program. The file type that was chosen for this study was CSV (comma separated value) files.

### 3.2 Variance and Phase Space Reconstruction

After the videos had their COM data generated, the first set of analysis that was performed on the data set was a variance test and phase space reconstruction. The variance is a statistical measure of how similar a system is to itself. The variance was applied to the data because previous studies have shown that decreases in the variance are a sign of critical slowing in a system [13,

15]. Critical slowing is the hallmark that a critical transition is about to occur in a system. The variance accomplishes this by comparing the system to itself. This is a windowed calculation, meaning that an overlap in the data is used as the calculation is made (similar to the Mean Filter) and calculation is done over the whole data set using this overlap. Mathematica 8 was used to perform this analysis as the windowed variance calculation is a built in package. The code for this calculation is called Variance and was mapped onto the data set to speed up the calculation.

The phase space of the flame video from 11/26/2011 was also created using Mathematica 8 but there was no built in package for the phase space reconstruction. The code for this analysis centered on calculating the autocorrelation of the data set to make ordered pairs of the data. These ordered pairs were then turned into a plot, which is the phase space of the system. The autocorrelation is used because it provides a measure of the correlation between one measurement and another made a time (delay) $\tau$ later. The formula for the autocorrelation is as follows

$$\rho(\tau) = \sum_{i=1}^{N} x_i x_{i+\tau} \qquad (7)$$

where $x_i$ is the $i^{th}$ element of the series and $N$ is the length of the data set. This formula was coupled with the Table command. The Table command was used because it loops the desired calculation back onto to the data set and created a list of points that could then be plotted to determine the delay with no autocorrelation.

The value of $\tau$ which gives zero autocorrelation was then used to create the ordered pairs of data for the phase space. When the time delay between measurements is zero then there is no linear correlation between $x_i$ and $x_{i+\tau}$ so they should serve well as two independent coordinates. This was done for two and three dimensions. The method for plotting the ordered pairs (or triplets if in three dimensions) is similar. The ordered pairs were organized into their proper lists

using the Table command. After this was done, plots could be made that were depictions of the phase space of the COM data. The ListPlot command was used to generate the two dimensional phase space and the 3DListPlot command was used to generate the three dimensional phase space.

### 3.3 Mutual Information and False Nearest Neighbors

A proper recreation of the phase space of the flame required a determination of the dimensionality of the flames used in this study. This needed to be done because results from the phase space reconstruction using the windowed autocorrelation returned results that were consistent with a system that had a dimensionality that was either fractional or greater than three dimensions. In order to test for these possibilities other methods needed to be considered. Unlike the autocorrelation, the mutual information has the ability to detect nonlinearly correlated systems. We suspect that our systems are highly nonlinear and therefore the results of the linear analysis needed to be checked with nonlinear analyses.

Using the mutual information we can perform an analysis similar to the autocorrelation. The "self mutual information" [10] is defined as

$$M(\tau) = \sum p(x_i, x_{i+\tau}) \log \frac{p(x_i, x_{i+\tau})}{p(x_i)p(x_{i+\tau})} \qquad (8)$$

and gives the amount of information transferred between $x_i$ and $x_{i+\tau}$ elements. A good choice for $\tau$ for identifying independent pairs for phase space reconstruction corresponds to the $\tau$ which gives the first minimum of $M(\tau)$.

False nearest neighbors is a test to see if two members of a time series are appropriate to pair together for phase space reconstruction. The false nearest neighbors tests for this by comparing points in many dimensions [16, 17]. A pair of points is said to be a false nearest

neighbor if the two points diverge as the dimension of the system goes up. The correct dimension will be signified by one of two events. The first is no false nearest neighbors. The second is a significant drop in the number of false nearest neighbors. A significant drop is constituted by at least a drop of fifty percent from one dimension to the next.

A false nearest neighbors analysis has the ability to determine an embedding dimension for a system. The embedding dimension is the integer number of coordinates required to recreate the phase space. The false nearest neighbors analyzes and set of ordered pairs and looks for divergences in this set based on an iterated calculation. The mutual information was used to create a list of ordered pairs. More exactly the point of the mutual information was to calculate a lag time in the time series that corresponded to pairs of numbers that are supposed to have little nonlinear correlation to each other. This analysis is similar to the autocorrelation technique described in the previous section, but works best for highly nonlinear systems. The false nearest neighbors calculation then used this lag time to perform its calculation to return an embedding dimension. If an embedding dimension is greater than three it would be necessary for Poincaré sections to be used to visualize the phase space. The embedding dimension is not the exact dimension of the system. It is just the integer number of dimensions required to represent the system in phase space. This means that if some system had an exact dimension of 4.23, the embedding dimension would be 5 or higher. Determination of exact dimensions of the system will be discussed later in this paper.

The code used to make the mutual information and false nearest calculations came from the TISEAN package [14]. This is a package of nonlinear time series analysis algorithms that was posted online by Rainer Hegger, Holger Kantz, and Thomas Schreiber. These algorithms

work through the DOS command line on the computer and are in the form of executable files. Version 3.0.1 of TISEAN was used in this study.

### 3.4 United Algorithm and False Nearest Neighbors

The results from the mutual information and false nearest neighbor analysis were not useful. A different method of determining and embedding dimension called the United Algorithm also uses a false nearest neighbors calculation [18]. The United Algorithm removes the need to use the mutual information to calculate a lag time. This method takes a different approach than the previous method and uses a series of iterations to determine the embedding dimension without an exact determination of the lag time from the mutual information. The United Algorithm uses the following function to make an inference about the embedding dimension of a system

$$S(m,\tau) = \frac{1}{N-(m-1)\tau}\sum_{n=1}^{N-(m-1)\tau}\sqrt{\sum_{l=1}^{m-1}(x(n+l\tau)-x(n))^2} \qquad (9)$$

where $N$ is the length of the data set, $m$ is the embedding dimension, and $\tau$ is the lag time. Equation 9 is called the average displacement.

The algorithm makes inferences about the lag time by comparing the two system states described under the square root sign in Equation 9. This method must be used in conjunction with a false nearest neighbors calculation [12]. The method for finding the embedding dimension begins with using a lag time of one and calculating the embedding dimension associated with this lag time. This embedding dimension is then substituted into Equation 9 and a plot of $S$ vs. lag time was made. The lag time for this calculation was taken to be the first extreme value from the plot of $S$ vs. lag time. Then another false nearest neighbor calculation was performed. This process was repeated over and over again until similar values for the dimension were returned. The TISEAN false nearest neighbor package was used to make the false nearest neighbor

calculation. The United Algorithm equation (Equation 9) was programmed using Mathematica. This was done by defining Equation 9 as a function and then making the necessary plots for the various values of the embedding dimension using the Plot command.

# 4. Results

The presentation of data in this study will be in the form of the filtered COM data. Data runs for this study were taken on the following dates: 11/21/2011, 12/05/2011, 03/26/2012, and 04/02/2012. The analysis that was performed on the data from these experiments followed logical pattern beginning with the data taken on 11/21/2011. The analyses performed were targeted at trying to recreate the phase space of the flames that were videotaped in the expectation that this would give information about a potential critical transition that might be occurring at the stoichiometric mixture of the flame's AFR state.

## 4.1 11/21/2011

One flame was video recorded on this date; a lean video each with a length of about thirty seconds. The equivalence ratio of the flame was 1.1. This video had a constant equivalence ratio and therefore no critical transitions were expected to be seen. The variance test was used to see if a constant data set produced a constant variance. Figure 14 is a plot of the COM for this flame. This data set is seen to be constant which is a good sign that no transitions are occurring in the system state.

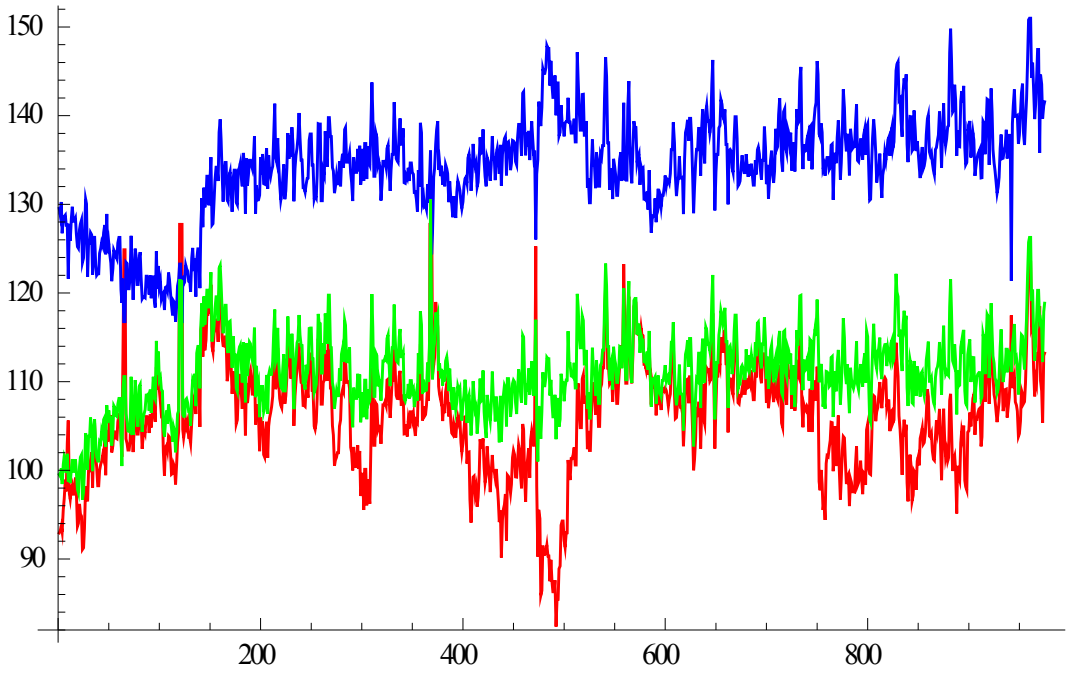



Figure 14. COM plot of the lean video taken on 11/26/2011. Blue is the blue COM, green is the green COM, and red is the red COM data.

The variance was used because it tests for critical slowing, a sign that a critical transition is about to occur. The critical slowing shows up in the windowed variance plot as a dip in the self-similarity in the system (dip in the variance). Figure 15 is a plot of the variance for this flame.

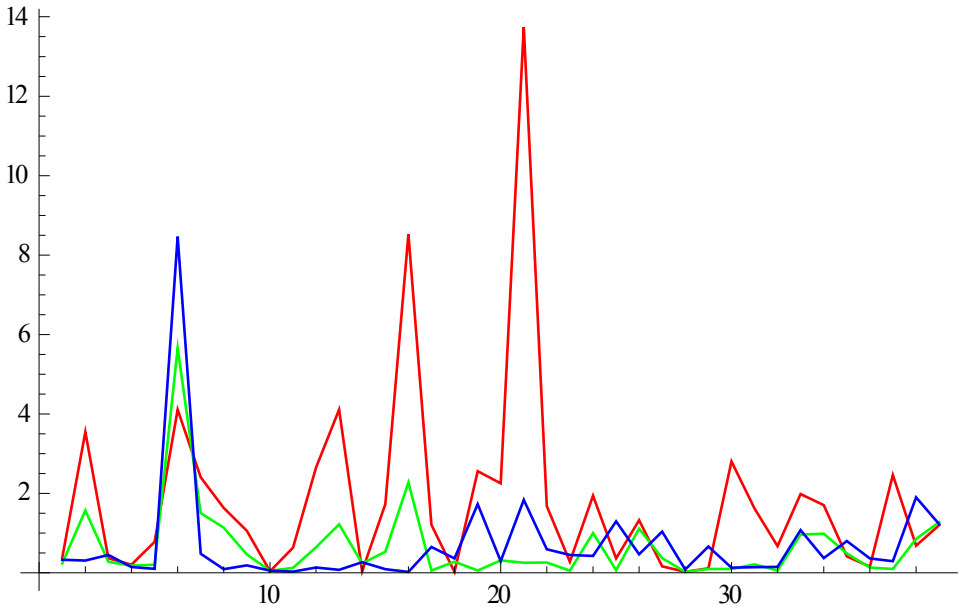



Figure 15. Plot of the variance for the rich flame video taken on 11/26/2011. Blue is the blue COM, green is the green COM, and red is the red COM data.

The data for the variance plot showed no obvious dip. Because it returned the expected results, the variance test was abandoned for further data sets. If the equivalence ratio was constant (as was with this data set) there should have been no dips in the variance plot. Based on the further results of the phase space reconstruction variance plots were abandoned as a means for detecting the possible critical transition.

The phase space reconstruction returned interesting results. The green color channel was used for the phase space reconstruction because it exhibited the best behavior to perform an analysis on. Figure 16 is a two dimensional plot of the phase space for this flame and Figure 17 is a three dimensional plot for the phase space of this flame. These plots are two dimensional and three dimensional subspaces (not Poincarè sections) of the full phase space. The time lags that were used to create these plots used the windowed autocorrelation to put together pairs of points

from the COM data for each color channel. This autocorrelation works best for linear systems, so a system that is nonlinear in nature may require further tests that are capable of dealing with nonlinear behaviors.

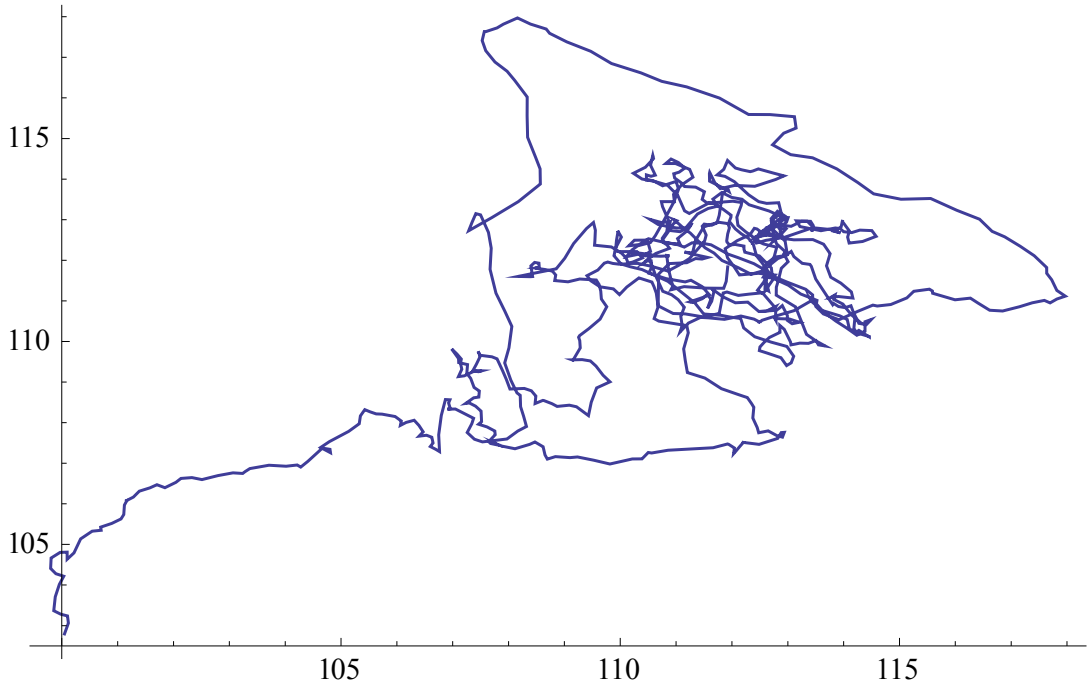



Figure 16. Two dimensional phase space plot for the green color channel of the flame video taken on 11/26/2011.

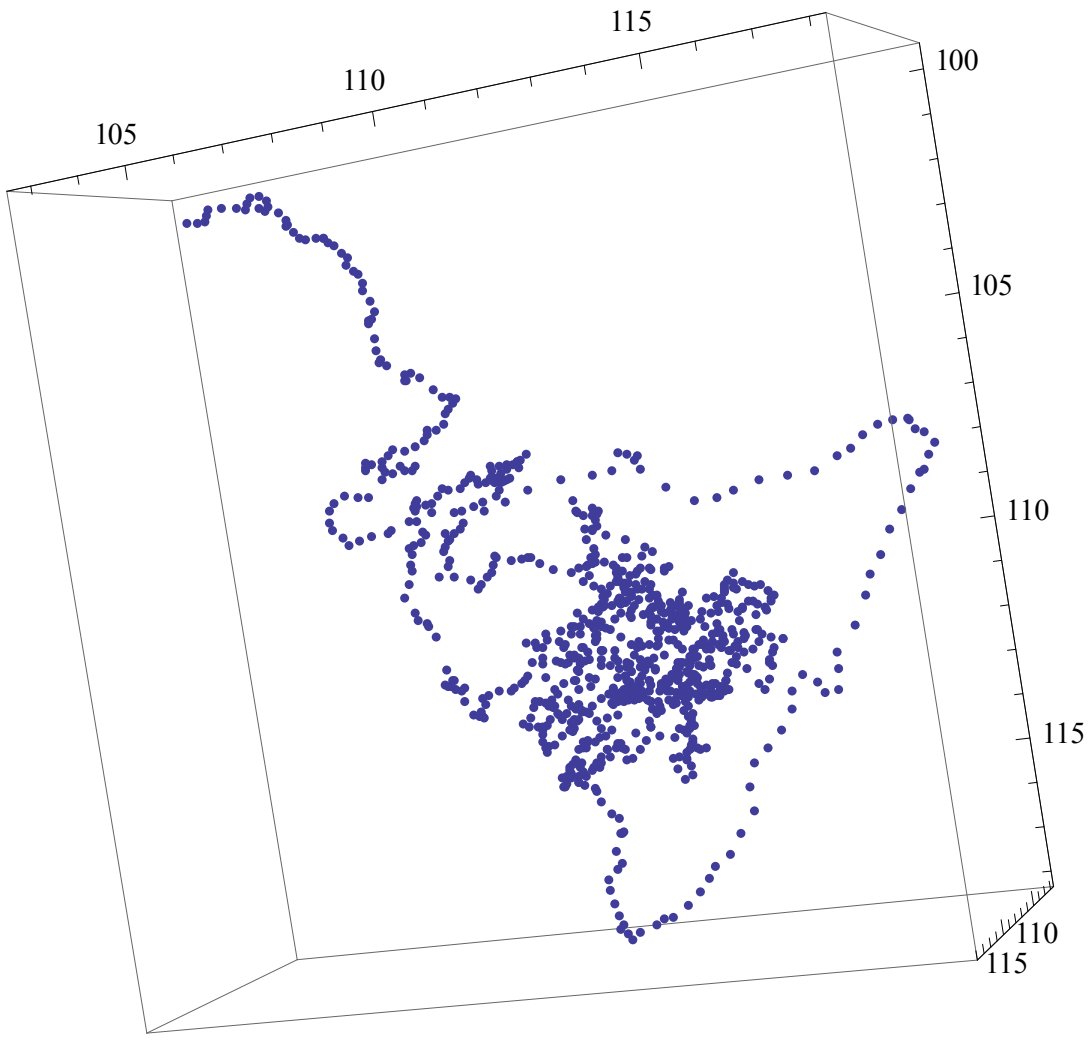



Figure 17. Three dimensional phase space for the green color channel of the flame video taken on 11/26/2011. The plot was oriented in way such the behavior of the phase space could best be seen.

The results of the phase space reconstruction were interesting because both the two dimensional and three dimensional phase space plots showed what are possibly fractal qualities. This behavior is the hallmark of a chaotic system, which would become an integral part of any mathematical model developed. Further, it appeared that the system was settling onto an attractor. In nonlinear systems there are stable and unstable sets in their phase space. The system

moves away from the unstable sets and moves towards the stable sets. For this reason the stable sets are called attractors. It appears from this data set that the flame is going towards an attractor of some sort. This would also be an integral part of any mathematical model developed. This result needed to be verified before any more work could be done with this result. More analysis is now required because only subspaces have been seen.

### 4.2 12/05/2011

Two flames were video recorded on this date as well; a lean and rich flame each of about a minute in length. The equivalence ratio of these flames were 1.2 and 0.96 respectively. Figures 18 and 19 are plots of the COM data for the lean and rich flames respectively. Each of these data sets showed that constant "flat" data sets are consistent with no changes in the dynamics (constant equivalence ratios.)

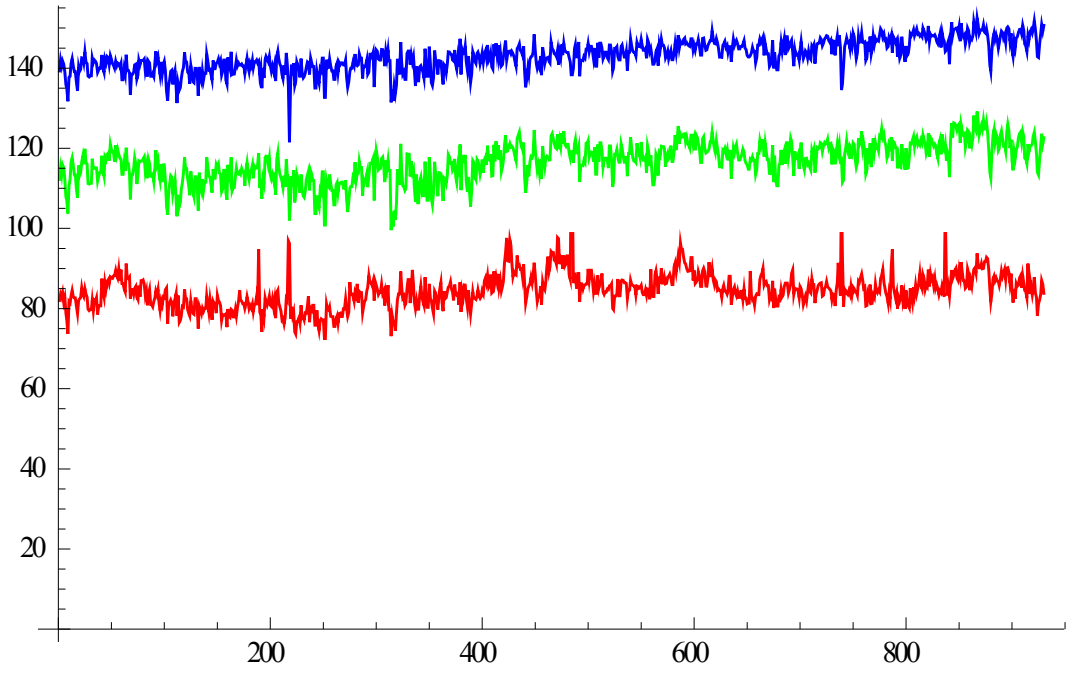



Figure 18. Plot for the lean COM data for the lean (φ=1.2) video taken on 12/05/2011. Blue is the blue COM, green is the green COM, and red is the red COM data.

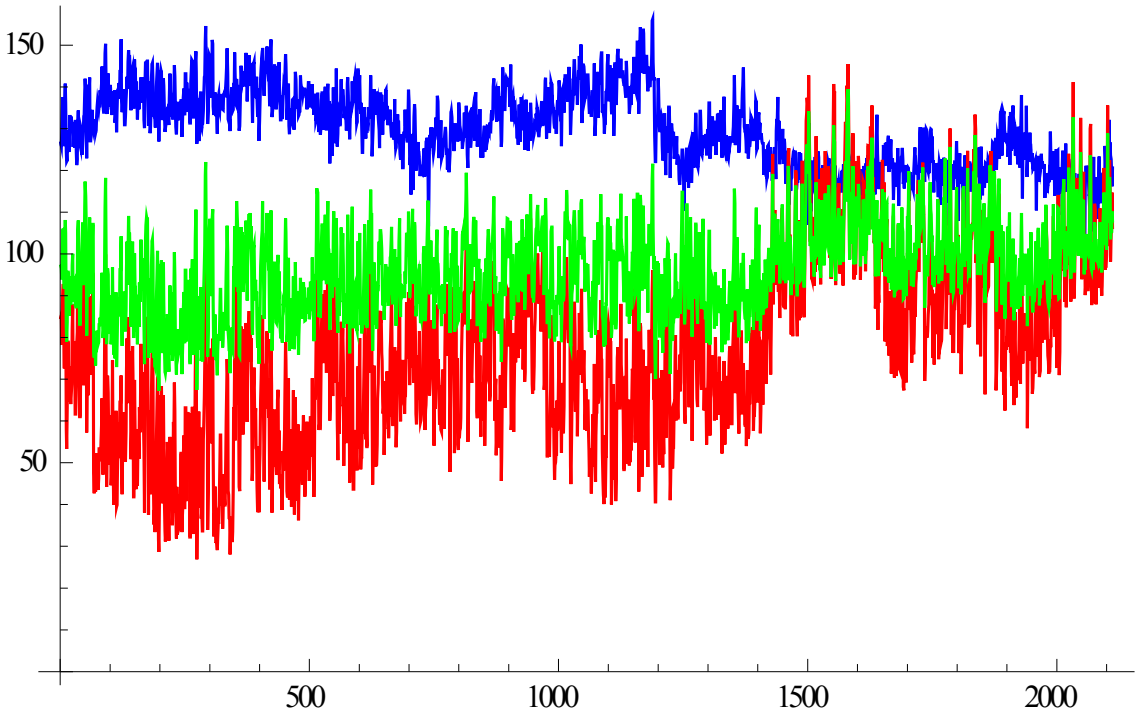



Figure 19. Plot for the lean COM data for the rich (φ=0.96) video taken on 12/05/2011. Blue is the blue COM, green is the green COM, and red is the red COM

The computational goal that was done on these flames was to affirm the result of the phase space reconstruction that was done on the previous data sets. The first thing that was done to these flames was their phase space was reconstructed. Figure 20 is the result of this reconstruction.

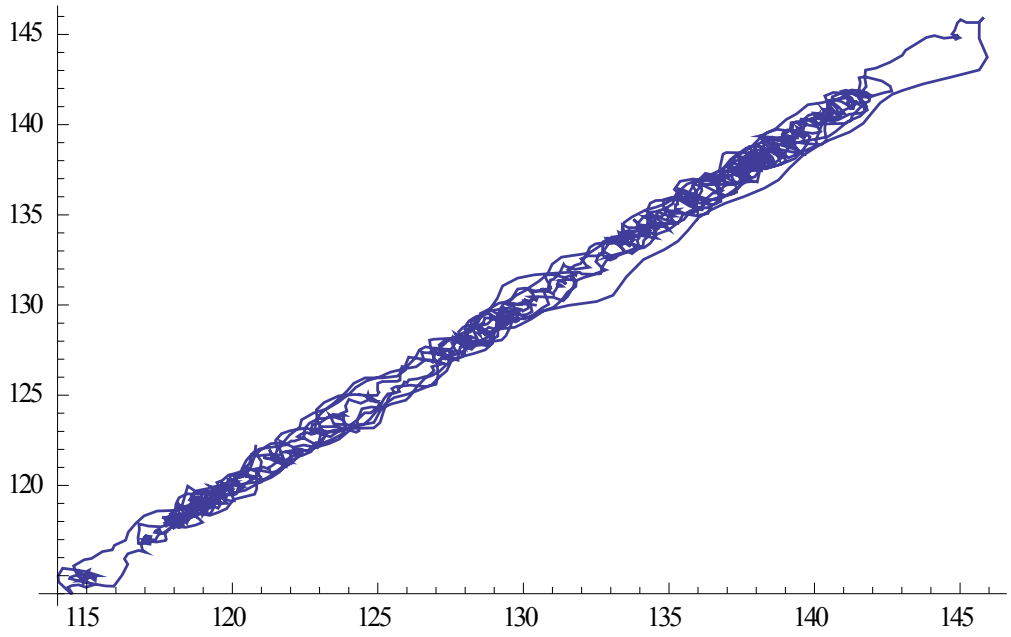



Figure 20. Phase space plot for the blue color channel for the rich flame video taken on 12/05/2011.

The result in Figure 20 gives no valuable information about how the flames may be changing as they approach the stoichiometric mixture because $\varphi$ is constant. However, this result was the impetus to determine the proper embedding dimension for the phase space of the flames. This was done to see if Poincaré sections would be required to make a proper depiction of the phase space of the flames.

The embedding dimension was found using the mutual information/false nearest neighbor analysis. The mutual information test was used because it has the ability to handle nonlinear systems without giving misleading results. Figure 21 is a sample mutual information plot.

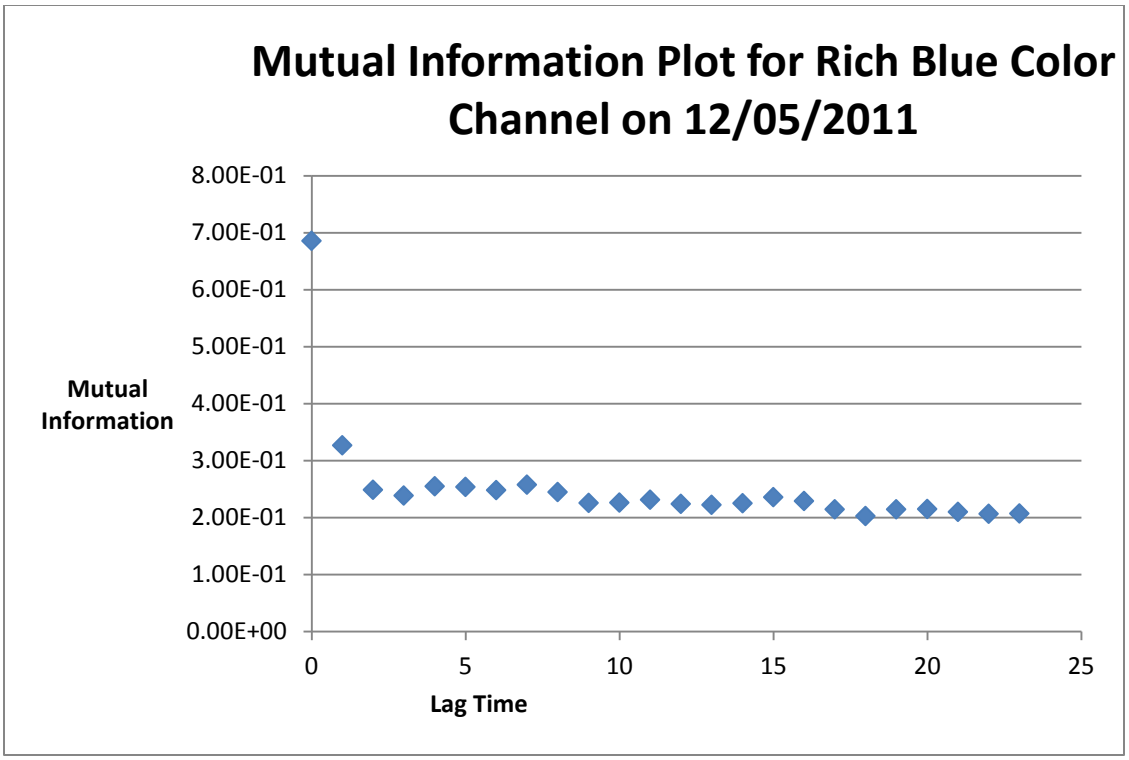



Figure 21. Mutual information plot for the blue color channel for the rich flame video taken on 12/05/2011. The number of delay vectors for this plot is 2 and the first minimum is 3.

Plots like Figure 21 were used to determine the lag time for the false nearest neighbor calculation. These lags were found by trying several different values for the number of symbols in the mutual information calculation. The number of symbols is essentially the window size of the mutual information calculation. The proper number of symbols was the first one that had a minimum in its mutual information calculation. Table 1 is a summary of the number of symbols and lag times used to perform the false nearest neighbor calculation for the rich and lean videos on this data.

| Flame Type | Color Channel | Number of Symbols in Series | Lag Time |
|---|---|---|---|
| Rich | Blue | 2 | 3 |
| | Green | 4 | 12 |
| | Red | 4 | 11 |
| | | | |
| Lean | Blue | 2 | 5 |
| | Green | 2 | 5 |
| | Red | 2 | 8 |

Table 1. Summary of the lag times and number of symbols used to calculate the false nearest neighbors for both the lean and rich videos on 12/05/2011.

Using these values the false nearest neighbors were calculated. Figure 23 is a sample plot of a false nearest neighbor calculation.

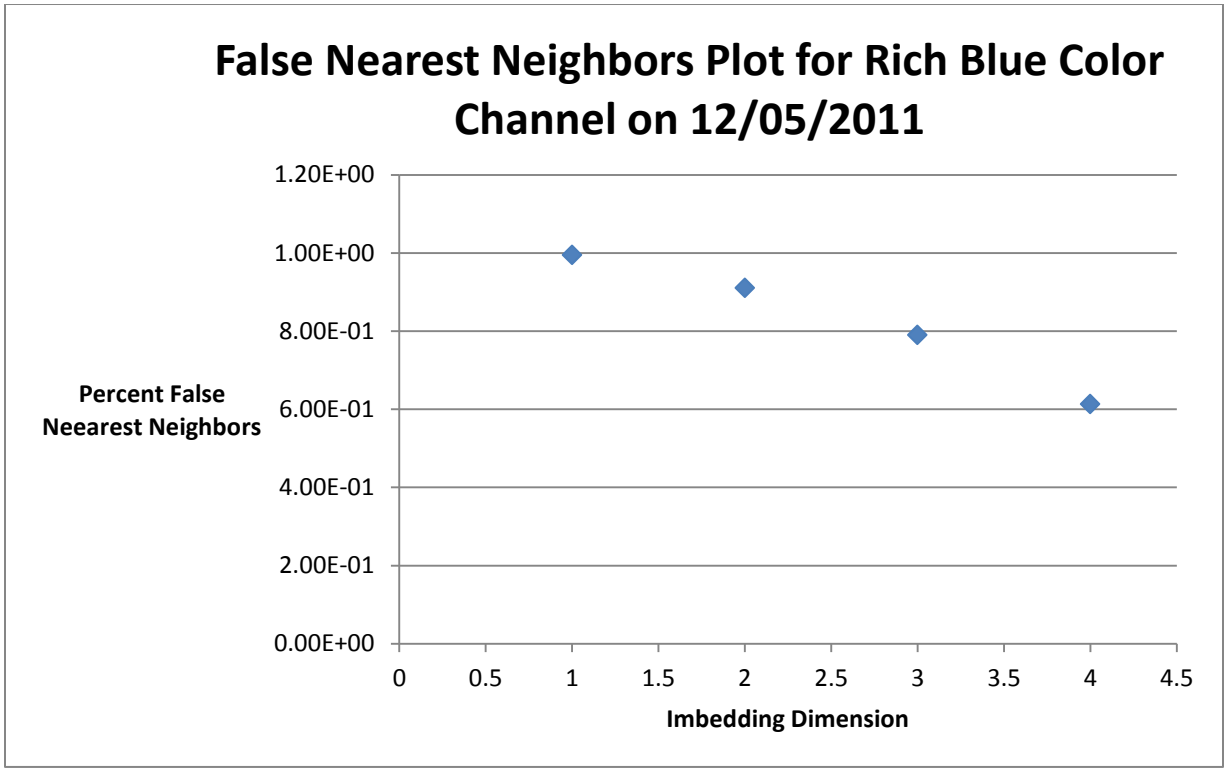



Figure 22. False nearest neighbor plot for the blue COM data of the rich flame video taken on 12/05/2011.

The result of the false nearest neighbor calculation was inconclusive because there was no embedding dimension that did not have any false nearest neighbors. A proper false nearest neighbors plot would have a significant change in the plot at an inflection point. This did not exist in this data set. It is suspected that this occurred because an insufficient number of frames of video were present to correctly complete the false nearest neighbor calculation. This was rectified by taking longer data sets in later experiments. The same tests were done for the lean data. The results of these computations was consistent with results from the rich data.

### 4.3 03/26/2012

Two videos were taken on this date; a rich video and a variable AFR video. The rich flame video had an equivalence ratio of 0.92 and the variable AFR video began with rich mixture ($\varphi$=0.52) and ended with a lean mixture ($\varphi$=1.1). Each of these videos was approximately five minutes in length. The primary computational goal of this data set was to be able to find the proper embedding dimension. Again this was first tried with the mutual information/false nearest neighbor analysis. This computation was performed only on the rich video. Computations were not performed on the variable videos because there was not enough time. Computations for this date's data focused on the rich data because it was consistent with the previous results from the study. Table 2 summarizes the lag times and number of symbols used to perform the false nearest neighbor analysis on the rich flame.

| Flame Type | Color Channel | Number of Symbols in Series | Lag Time |
|---|---|---|---|
| Rich | Blue | 16 | 97 |
| | Green | 10 | 113 |
| | Red | 10 | 67 |

Table 2. Summary of the delay vectors and lag time used to perform the false nearest neighbor analysis on the rich flame video taken on 03/26/2012.

This calculation was inconclusive. When the false nearest neighbors calculation was forced to be calculated past five dimension strange behavior was observed. Instead of a single inflection point that corresponds to a value for the embedding dimension, an asymptotic behavior was seen in the false nearest neighbor plot meaning this result was inconclusive and could not be considered for the recreation of the phase space of the flame. Figure 23 is a plot of this calculation. The embedding dimension for this calculation was forced to go out to fifteen dimensions.

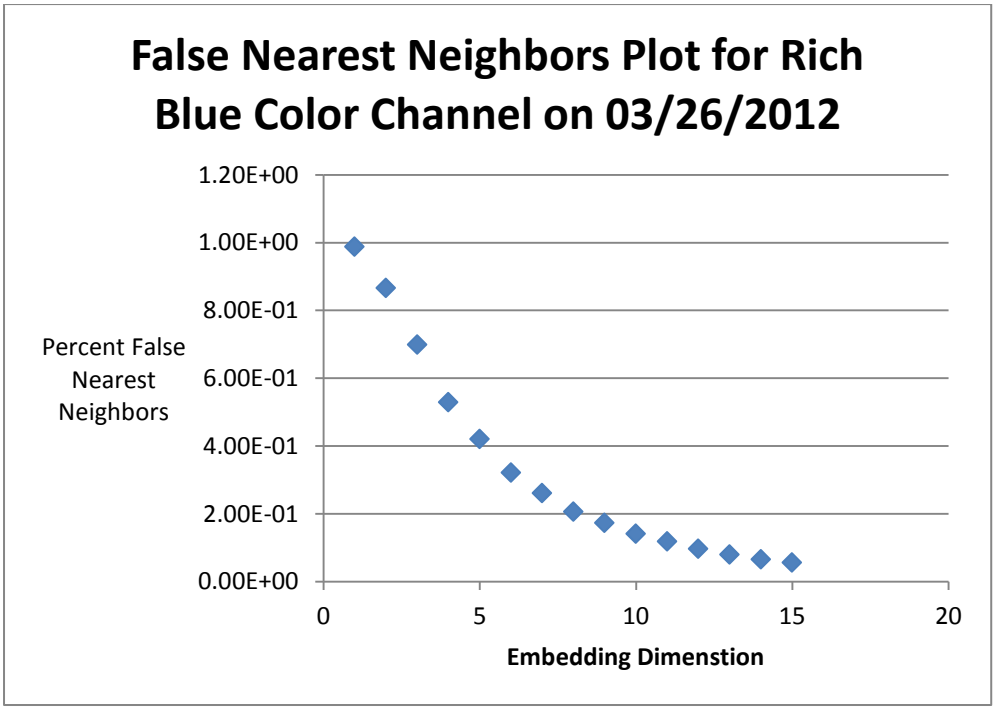



Figure 23. False nearest neighbor plot for the rich video taken on 03/26/2012. This goes to zero rather than giving an embedding dimension.

A second computation was performed on the flames taken on this date. This computation was also aimed at determining the embedding dimension of the phase space. The United Algorithm was used when the mutual information/false nearest neighbor analysis failed. This test was used on the rich flame taken on 03/26/2012. Figure 24 is a plot of the resulting false nearest neighbor plot after the United Algorithm was used just once.

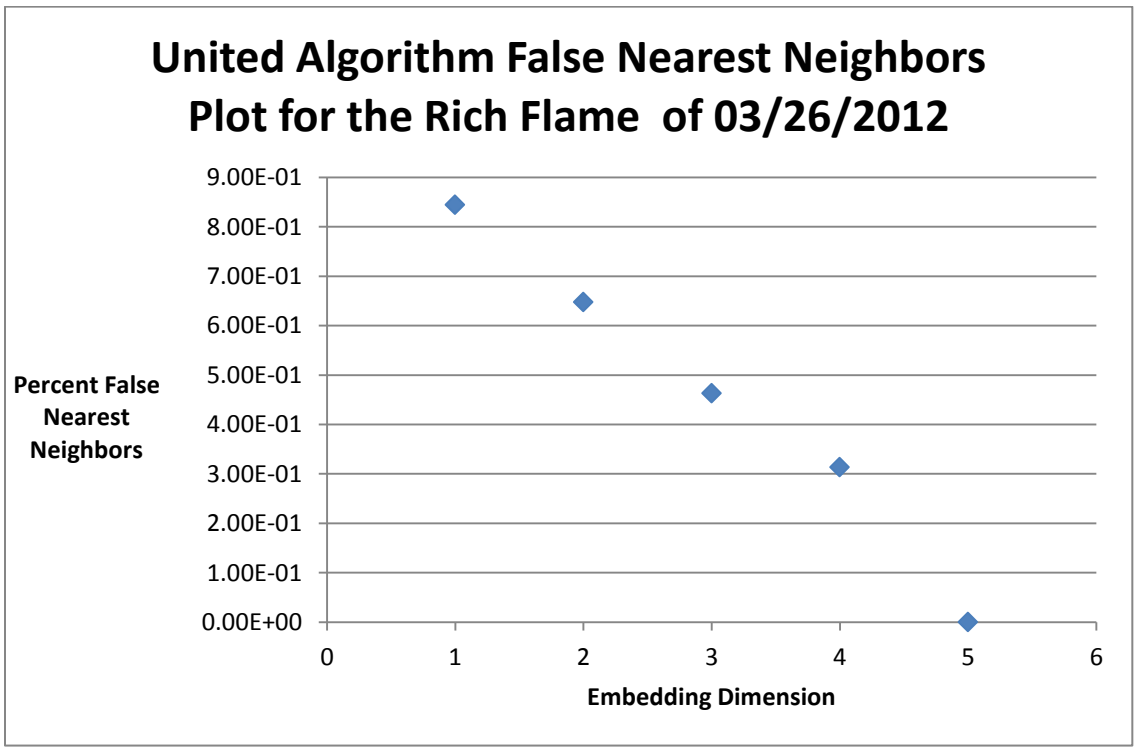



Figure 24. The false nearest neighbor plot associated with the United Algorithm. The equivalence ratio of this flame was

When using a lag time of 1, the embedding dimension was 5. After using the United Algorithm a lag time of 1411 frames was found to correspond with an embedding dimension of 5. This result is interesting but believable because the plot from Figure 24 is qualitatively different than the previous false nearest neighbor plots. Rather than the asymptotic behavior seen in the previous

results, this result had a more gentle change of thirty percent. The correlation between the first four dimensions was also significantly more linear than in the previous result. Another important aspect of this result is that when the computation was forced to have more than five dimensions be calculated, an error message was returned. This signifies that the calculation was done at five dimensions because of an insufficient length of data. This result needs to be confirmed by checking it with another rich flame. This result is promising and will be used in further studies to properly recreate the phase space of the flames. There are also several issues with this analysis that must be considered. The United Algorithm is a relatively new algorithm and no literature has been found that verifies its use with noisy systems such as the flames in this study. There is also an issue with the length of the data set because of using a delay of 1411 frames. This is possibly the reason the calculation could not be carried out past five dimensions.

The computations of this study provided good insights into the complexity of the flames. It is very clear that the flames are nonlinear in nature and have a high degree of complexity. The computations of this study are heading in the right direction and the results of these computations should be considered heavily. If the flame does have a dimensionality of five, the phase space will be able to be recreated in short fashion after the completion of this study.

## 5. Discussion

This study is one part of a larger study which is aimed at developing simple mathematical models of combustion. The goal of this project was to detect a critical transition in flames that were near the stoichiometric mixture. While a transition was detected steps were taken towards phase space reconstruction. These steps are large strides toward possibly finding a critical transition. Important aspects of the dynamics of the flame were discovered which need to be

incorporated into any mathematical model of the flame. The dimensionality of the flames in this study was clearly determined to be greater than three. This opens to the door to many nonlinear possibilities. The computations of this study focused on determining the embedding dimension of the phase space so that the proper phase space could be reconstructed.

The computational results of this study have had a clear impact on the future computational analyses that will be performed on flames. After a proper embedding dimension has been affirmed by similar computations to the ones used in this study, the exact dimension of the flames can and should be determined. A method for determining the exact dimension is the correlation dimension. This computation would be a direct extension of the work done in this study because the embedding dimension is required to make this calculation. In addition to the embedding dimension, a determination of the Lyapunov exponent is required. The Lyapunov exponent measures how much vectors diverge from each other in a chaotic system. The Lyapunov exponent is a computation that in of itself should be calculated from future data sets, in order to correctly classify the dynamics of the flame. These plots hinted at the possibility of the flames in this study being chaotic. Calculation of the Lyapunov exponent is the next logical step in testing for chaos in a system. These are all aspects that would need to be included in a mathematical model developed from the flames in this study.

In addition to the computational aspect of this project there is a large experimental component. The experiments of this project were a large improvement on previous studies. These improvements were made in order to make the "simple" mathematical models dependent on only one variable, the AFR of the flame. This study made the leap from the Bunsen burner studies previously mentioned to a state where experimental techniques will soon be developed for making the time series AFR dependent. This was a challenging endeavor and took much of

the time of this study. Challenges were met at almost every step of the way. They began with constructing the apparatus. Developing the experimental techniques for varying the AFR state of the flame also took several trials in the laboratory that did not produce results included in this study. These data runs culminated in the data used for the 03/26/2012 data sets.

Akin to the computational goals of this project, the over arching experimental goal of this study was to produce the AFR dependent time series. This study contributed greatly to this effort and put the project in a state to make this next step. Experimental improvements can be made by developing a technique to synchronize the varying AFR with the video of the flame. This would involve somehow digitizing the flow rates and it would more than likely require a computer program to do this. Computationally the goal of this would be to create "time series" of flame state vs. AFR. Currently the computations are performed on true time series; the state of flame as a function of time. This would done by automatically tracking AFR so series end up with as a function of the equivalence ratio. Though no experimental methodology was developed to do this, the experience gained by working with the new equipment will be invaluable as work towards this begins in earnest.

This project was a significant piece in a large puzzle. This study added valuable information to the overall puzzle of creating simple mathematical models of combustion. Experimental results and techniques have made the overall goal a not so daunting task. The computations of this study have also revealed quite a bit of information that would need to be included in any model of the flames. Integration of the results from this study to the overall goals provided a nice launching place for continued studies.

## References

1. D.B. Spalding, *Mathematical modeling of turbulent flames; a review*, Combustion Science and Technology, **1**, 3-25 (1976).

2. A. Sanz, J. Ballester, and M. A. Gonzalez, *Investigation of the characteristics and stability of air-staged flames*, Exp. Therm. Fluid Sci. **32**, 776-790 (2008).

3. B. Ugur Toreyin, Y. Dedeoglu, U. Gudukbay, A Enis Cetin, *Computer vision based method for real-time fire and flame detection,* Pattern Recognition Letters, **27**, 1, 49-58 (2006).

4. R. Hernandez and J. Ballester. *Flame imaging as a diagnostic tool for industrial combustion*, Combustion and Flame, **155**, 509-528, (2008).

5. A. Tuntrakoon and S. Kuntanapreeda, *Image-based flame control of a premixed gas burner using fuzzy logics*, in *Foundations of Intelligent Systems*, edited by N. Zhong *et al.*(Springer-Verlag, Berlin Heidelberg, 2003) pp. 673-677.

6. C.E. Shannon, *A Mathematical theory of communication*, The Bell System Technical Journal, **27**, 7, 378-423 (1948).

7. C. Daw, C. Finney, E. Tracey, *Symbolic analysis of experimental data*, Review of Scientific Intrsuments, **74**, 915 (2003).

8. T.D. Scheinder, *Information theory primer with an appendix on logarithms*, Available online: http://www.lmmb.ncifcrf.gov~toms/paper/primer/

9. A. Davis, *Small sample effects on information-theoretic estimates*, unpublished thesis (B.S. with honors), The College of William and Mary.

10. A. Fraser and H. Swinney, *Independent coordinates for strange attractors from mutual information*, Phys. Rev. A, **33**, 2, 1134-1140 (1986).

11. T. Schreiber and A. Schmitz, *Surrogate time series*, Physica D, **142**, 346-382 (2000).

12. S. H. Strogratz, *Nonlinear dynamics and chaos with applications to physics, biology, chemistry, and engineering*, Perseus Books Publishing, Cambridge, MA (1994).

13. M. Scheffer *et al.*, *Early-warning signals for critical transitions,* Nature, **461**, 3, 53-59 (2009).

14. R. Hegger, H. Kantz, and T. Schreiber, *Practical implementation of nonlinear time series methods: The TISEAN package*, Chaos, **9**, 413 (1999).

15. C. Khuen, *A Mathematical framework for critical transitions: bifurcations, fast-slows systems and stochastic dynamics*, Physica D, **240**, 1020-1035 (2011).

16. M. Kennel and H. Abarbaenel, *False neighbors and false strands: a reliable minimum embedding dimension algorithm*, Phys. Rev. E, **66**, 1-17 (2002).

17. M. Kennel and R. Brown, *Determining Embedding dimension for phase- space reconstruction using geometric construction*, Phys. Rev. A, **45**, 6, 3403-3411 (1992).

18. J. Yang *et al.*, *An united algorithm for state space reconstruction*, ICIST, 589-591 (2011).

## Acknowledgements

I would like to thank the following people or groups of people. Without them, this project would never have been reached its level of success.

Dr. Christopher Kulp (advisor), Lycoming College Department of Astronomy and Physics

Dr. David Fisher (honors committee), Lycoming College Department of Astronomy and Physics

Dr. Charles Mahler (honors committee), Lycoming College Department of Chemistry

Dr. Gene Sprechini (honors committee), Lycoming College Department of Mathematics

Dr. Santhusht DeSilva (honors committee), Lycoming College Department of Mathematics

Dr. Richard Erickson, Lycoming College Department of Astronomy and Physics

Lycoming College Chemistry Department

Lycoming College Buildings and Grounds

Lycoming College Night Cleaning Staff

Williamsport Glass and Mirror Company

Dr. Charles Finney, Oak Ridge National Laboratory

Dr. Stuart Daw, Oak Ridge National Laboratory

The University of Tennessee

Oak Ridge National Laboratory

# Appendix A Computer Program Code Developed for this Study

## RGB Triplet Extraction

```
Importing the Data
SetDirectory["C:\\Users\\surdavi\\Desktop\\David Surmick\\11 21 2011"];
Import["CLIP0003.mov",{"Frames",200}](* this command was used for decided what pixels could be cropped out of the data set *)
VidDat=Import["CLIP0003.mov",{"Frames",Table[i,{i,1,975}]}];
Cropping out The Black Background
Crop=Function[x,ImageTake[x,{1,470},{147,318}]]/@VidDat;
Length[Crop];
FrameData=Function[x,ImageData[x,"Byte"]]/@Crop;
Length[FrameData];
RGBTripData=Function[x,Flatten[x,1]]/@FrameData;
Length[RGBTripData];
Deleting Background Data
CroppedData=Function[y,DeleteCases[y,x_/;Max[x]<100]]/@RGBTripData;
Calculating Center of Mass Data
COMData=Function[x,N[Mean[x]]]/@CroppedData;
flatCOMData=Flatten[COMData,1];
length=Length[flatCOMData];
REDCOMData=Table[flatCOMData[[i]],{i,1,length,3}]
GREENCOMData=Table[flatCOMData[[i]],{i,2,length,3}]
BLUECOMData=Table[flatCOMData[[i]],{i,3,length,3}]
RED=ListPlot[REDCOMData,Joined→True,PlotStyle→{Red}];
BLUE=ListPlot[GREENCOMData,Joined→True,PlotStyle →Green];
GREEN=ListPlot[BLUECOMData,Joined→True,PlotStyle→Blue];
Show[RED,GREEN,BLUE,PlotRange→All]
```

## Data Filtering

```
filt=MeanFilter[gdata,15];
ListPlot[filt,Joined→True]
```

## Variance

```
g=ListPlot[greendata,Joined→True,PlotStyle→{Green,Thin}]
greenfilter=MeanFilter[greendata,15];
a=ListPlot[greenfilter,Joined→True,PlotStyle→{Thick}]
GREENPARTDATA=Partition[greendata,25];
Variance/@GREENPARTDATA;
greenvariance=ListPlot[Variance/@GREENPARTDATA,Joined→True,PlotRange→All,PlotStyle→Green]
r=ListPlot[reddata,Joined→True,PlotStyle→{Red,Thin}]
```

```
redfilter=MeanFilter[reddata,15];
c=ListPlot[redfilter,Joined→True,PlotStyle→{Thick}]
REDPARTDATA=Partition[reddata,25];
Variance/@REDPARTDATA;
redvariance=ListPlot[Variance/@REDPARTDATA,Joined→True,PlotRange→All,PlotStyle→R
ed]
bluefilter=MeanFilter[bluedata,15];
e=ListPlot[bluefilter,Joined→True,PlotStyle→{Thick}]
BLUEPARTDATA=Partition[bluedata,25];
Variance/@BLUEPARTDATA;
bluevariance=ListPlot[Variance/@BLUEPARTDATA,Joined→True,PlotRange→All,PlotStyle
→Blue]
Show[redvariance,greenvariance,bluevariance]
```

**Phase Space Reconstruction**

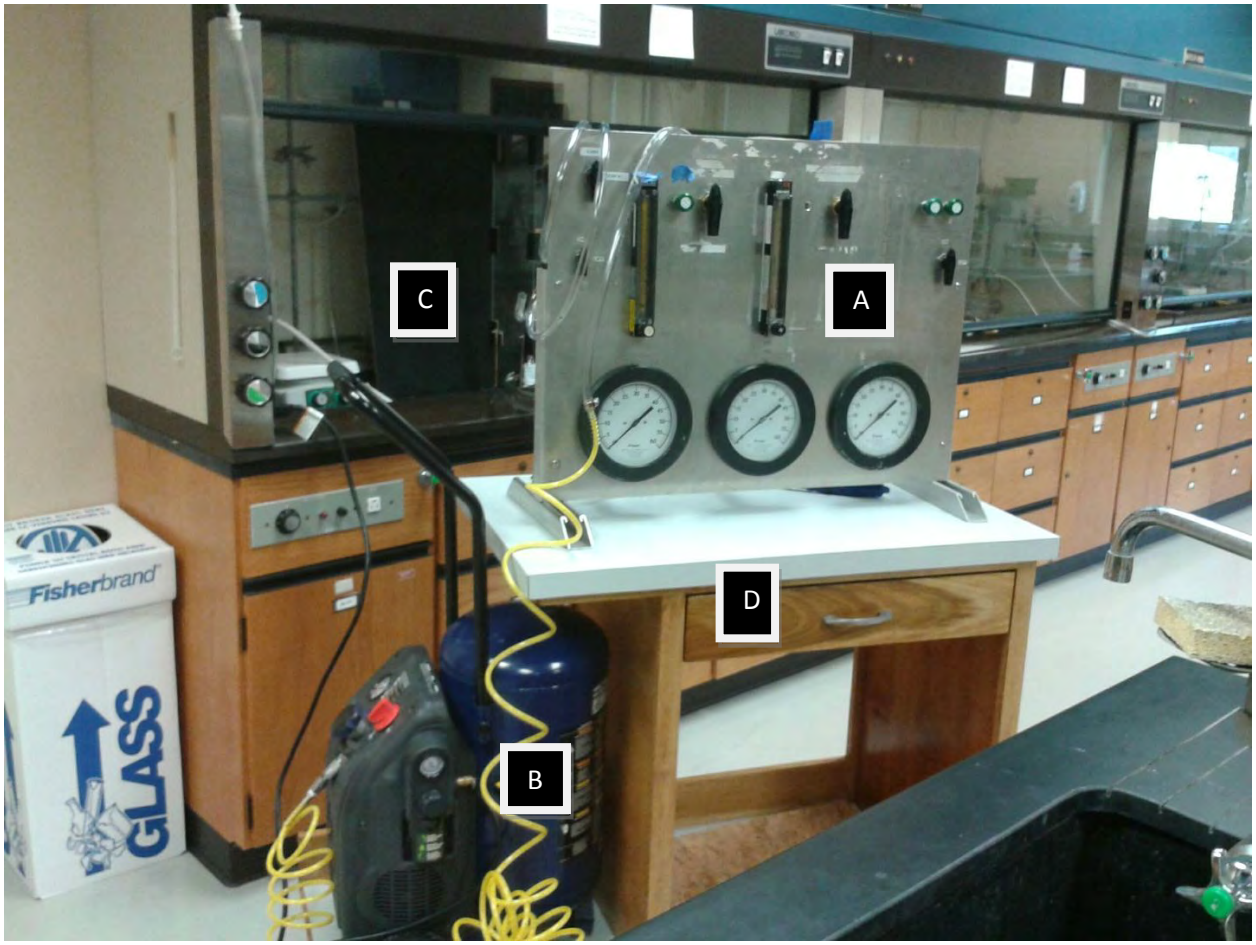
```
y=Table[filt[[i]],{i,Length[filt]}];
z=y-Mean[y];
Length[filt];
A=Table[1/(n-k1) Sum[z[[i]]z[[i+k1]],{i,n-k1}],{k1,0,150}];
ListPlot[A,PlotRange→{{104,108},{-.01,.01}}]
k1=105;
d=Table[{y[[i]],y[[i+k1]]},{i,870}];
ListPlot[d,Joined→True]
```

**United Algorithm**

```
SetDirectory["F:\\Research\\flame dynamics project (honors project and research
topics)\\Data\\03 26 2012\\Rich"];
blue=Flatten[Import["bluefilter.csv"],1];
green=Flatten[Import["greenfilter.csv"],1];
red=Flatten[Import["redfilter.csv"],1];
n=Length[blue]
m=5;
S[τ_]:=1/(n-(m-1)τ) (Sum[\[Sqrt]Sum[(blue[[j+l τ]]-blue[[j]])^2,{l,m-1}],{j,(n-(m-1)τ)}]);
sdat=Table[S[i],{i,1400,1450}];
ListPlot[sdat]
```

# Appendix B Labeled Images of the Apparatus

## Compressor, Rack, Cart, and Fume Hood



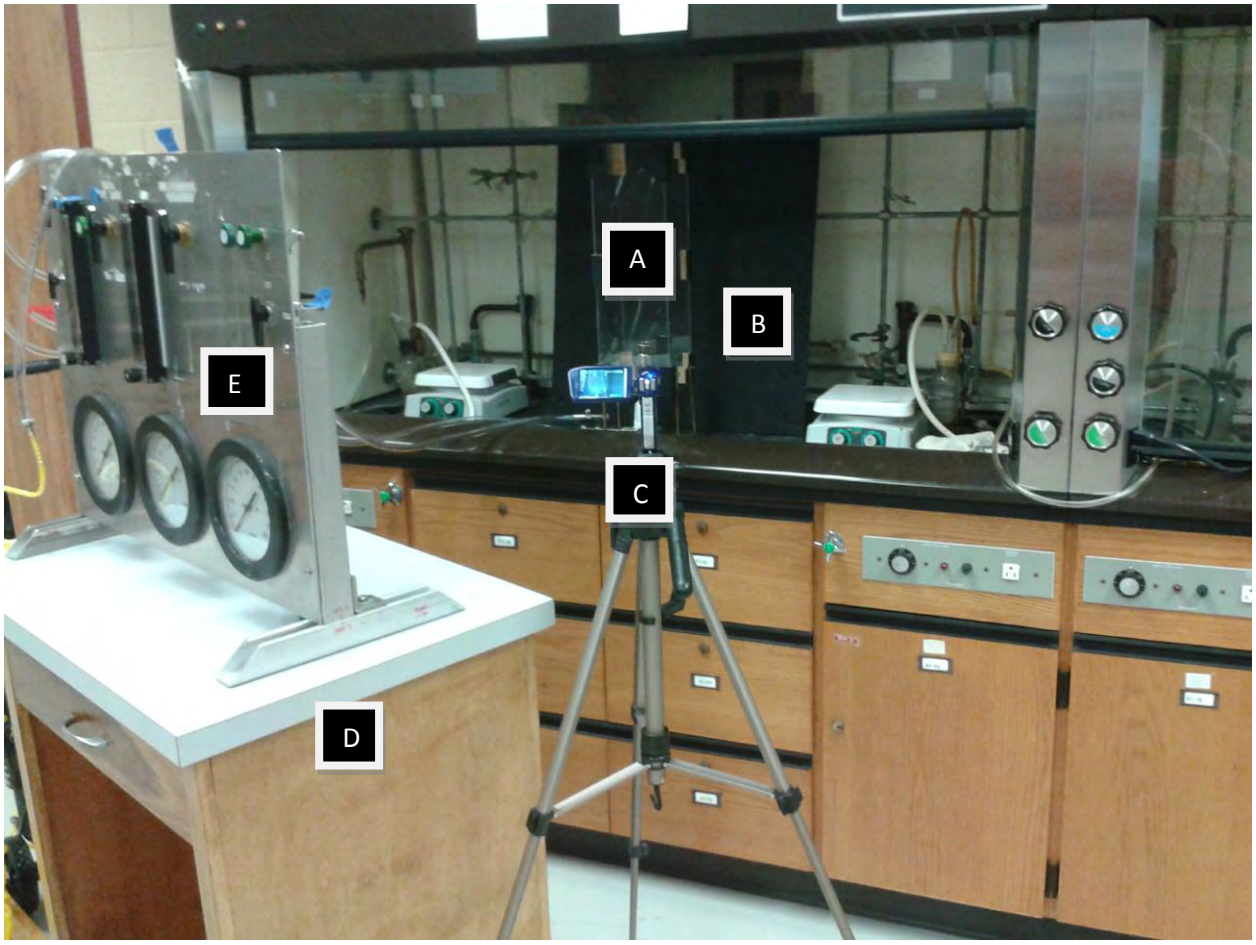

A- Rack of Pressure Gauges and Flow Meters

B- Air Compressor

C- Fume Hood

D- Cart for Transporting Apparatus

**Video Camera, Chimney, Black Background**




A- Chimney (old chimney)

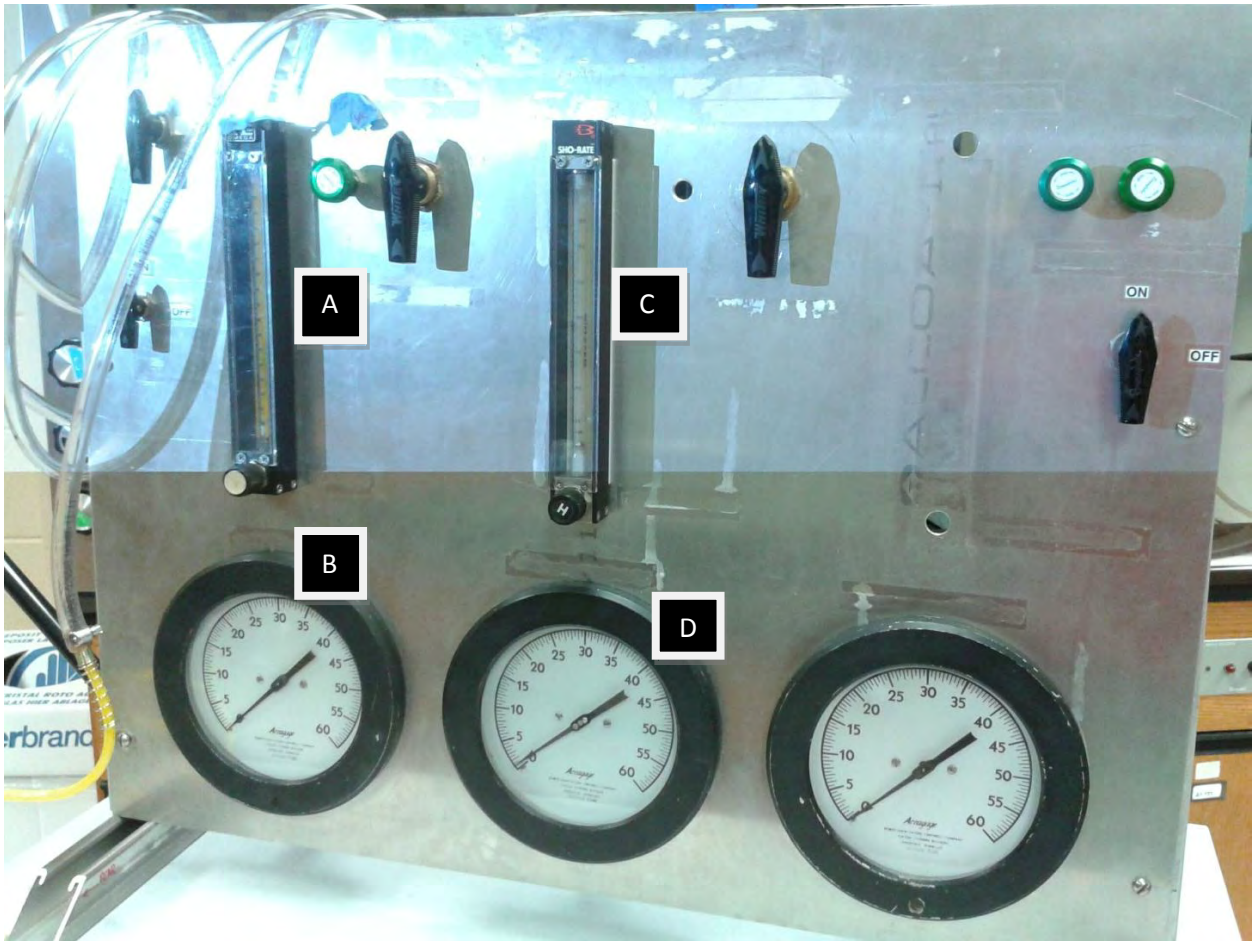
B- Black Background

C- Video Camera

D- Cart

E- Rack

**Pressure Gauges and Flow Meters**



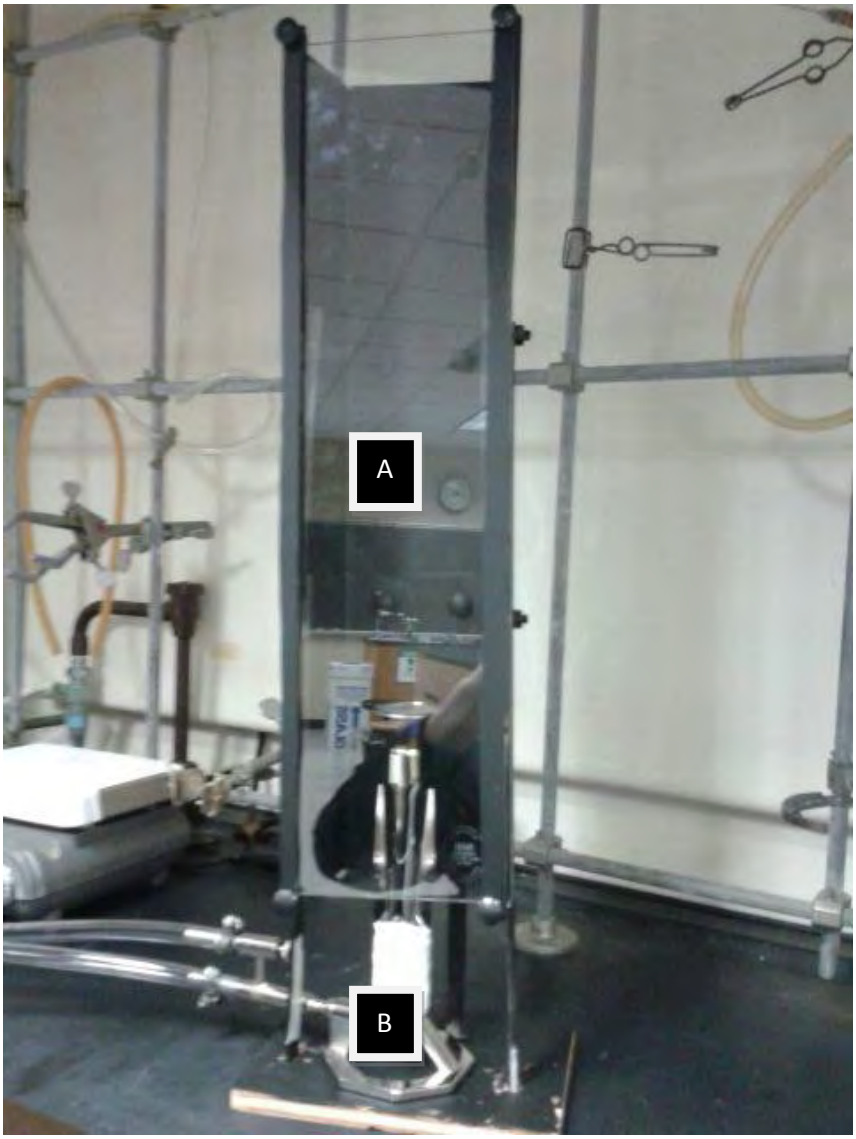

A- Gas Flow Meter

B- Gas Pressure Gauge

C- Air Flow Meter

D- Air Pressure Gauge

**Chimney and Burner**




A- Chimney (current design)

B- High Temperature Blast Burner

# Appendix C Manufacturer Flow Rate Conversion Chart

## 092-04-GL FLOWMETER CALIBRATION DATA 092/56

| CUSTOMER | CUST. P.O. No | REF. CURVE NUMBER |
|---|---|---|
| | | 1085-A1-AM-03D |

| Max. Flow | Min. Flow | Units | Metering Fluid | Date |
|---|---|---|---|---|
| 2313 | 185 | std. ml/min | air | 30-Nov-1999 |

| | | | |
|---|---|---|---|
| **Model Number** | FL-3404G | **Metering Temperature** | 70.0 °F |
| **Tube Number** | 092-04-GL | **Metering Pressure** | 14.70 psia |
| **Serial Number** | | **Metering density** | 0.001200 g/ml |
| **Float Material** | glass | **Metering Viscosity** | 0.01812 cp |
| **Float Density** | 2.53 g/ml | **Density at STD.Cond** | 0.001200 g/ml |
| **STD. Conditions** | STP: 1 atm @ 70 ° | **Accuracy** | ñ2% F.S. |
| **Room Temperature** | 70.0 °F | **Barometric Pressure** | 14.70 psia |

**SCALE READINGS AT CENTER OF FLOAT**

| Scale Reading (mm) | Flow |
|---|---|
| 150 | 2313 |
| 140 | 2170 |
| 130 | 2050 |
| 120 | 1932 |
| 110 | 1807 |
| 100 | 1674 |
| 90 | 1535 |
| 80 | 1392 |
| 70 | 1240 |
| 60 | 1085 |
| 50 | 928 |
| 40 | 763 |
| 30 | 590 |
| 20 | 395 |
| 10 | 185 |