To: President Frank O. Simpson

From: Kirthi Shankar Sivamani, Oliver Balicanta, Godfred Mantey, Spencer Dorsch.

RE: Data analysis for new client

Date: April 16, 2018

Part 1, Introduction

The client requires us to construct an algorithm to analyze the time history data for designated thermocouple designs. The algorithm should help improve their testing protocols, equipment, and data analysis of their designs. The deliverables the team will be providing to the client are as follows; a detailed description of our analysis on the data provided by FOS with graphical visuals that provide a clear summary of the data, an analysis of the error for our approach to calculate the time constant, and an honest statement about what FOS can say to their customers concerning the performance of their new design. The criteria for measurement of performance of our algorithm will be the SSE values and the coefficients of correlation of our regression model. Our solution is restricted by the data provided by the client, only being able to use MATLAB as our program to develop our algorithm and being limited by the time of the deadlines. Thus, these are some of the constraints. Our algorithm is based on the using mean values of temperatures to calculate the four parameters. (ts = start time, tau = time constant, yl = low temperature, yh = high temperature). The structure of the algorithm involves nested looping structures and vector manipulation techniques to attain required results. The key features of the algorithm are that the entire code is easily readable, sections for heating and cooling are separated, and each line of code is properly commented for easy understanding.

The first critical decision we made while developing the algorithm was to improve the accuracy of identifying the start temperature parameter. After brainstorming multiple ideas, our solution was to use the average of the first 4% of the temperature values as our value for initial temperature. This is a rational because based on all provided test data, the temperature will not change within the first four percent of data recorded. Averaging this amount of the data gives us a good average while also ensuring that the average does not include data from the actual temperature shift. The implementation of this decision has resulted in the best generated values for start temp so far. The improved SSE values for each FOS confirm these results (Table 1).

Secondly, our team found a new way of calculating tau. Based on feedback and SSE values from the last milestone, it was apparent that our method for calculation was lacking. The new way we of calculating tau included firstly calculating the temperature of tau as we did in the previous versions of the algorithm by using the equation Y(tau) = yl + 0.632 (yh - yl) (for cooling we switch yh an yl). Afterwards, we use this value in conjunction with the data set to count the number of data points greater than the temperature at time equals tau. Subtracting this value from the total number of data points gives us an index value. The value of time at this index value minus start time will give us tau. In testing, this method has proven to be more accurate.

Thirdly, our team found a better way of finding yh and yl. After careful observation, our team noticed that steady state was only reached at the very end of the data for FOS 4 and 5. This was causing an issue based on the way we calculated the final values for temperature. For FOS 4 and 5, out method didn’t work so decided that to improve overall accuracy and use the average of the last 2 percent of the temperature values. Based on visual comparison of plots from before and after implementing this change we were able to conclude that Tau had improved accuracy. The new SSE values affirmed the validity of the change (Table 1).

Part 2, Procedure (parameter identification)

The algorithm receives data for time and temperature from the user. The type of data (heating or cooling) is decided by comparing the initial and final temperature. If final temperature is greater, the type is heating; cooling otherwise. We then calculate yh for heating and yl for cooling by taking mean of the first 4% of temperatures. We then calculate yl for cooling and yh for heating by taking the mean of the last 2% of temperatures. We then calculate temperature at tau using the equation y(tau) = yl + 0.632(yh-yl) for heating (yh and yl are flipped for cooling). To find ts, we find all such time data points such that temperature at that data point is lowest among all data points from this time value till the end of data. Among all these time values, we select the largest time value (as ts) for which temperature is lower than a certain threshold. This threshold is yl for heating and yh for cooling. We then find the number of data points for which temperature is greater than the temperature at tau. We subtract these number of points from the total number of data points to get the time where 63.2% of increase in data has taken place. We then subtract ts from this time to obtain tau.

Part 3, Results

The results of applying this algorithm to the specified data set is described in the tables and figure presented below. As shown in Figure 1, there as a clear correlation between thermocouple performance and price. This relationship is that shorter response times will correspond to increasingly expensive prices. Our algorithm found this to be consistently true for FOS thermocouples (see figure 1). As shown in Table 1 the values described for each FOS thermocouple have become reasonably accurate in their predictions with SSE values as low as 0.34 and maximum of 0.39. The trends in this data compared with those in Table 2 show that the more expensive a thermocouple is, the smaller the Tau value will be. The relationship between price and Tau is a power function, as can be seen from the regression equation in Figure 1.

Part 4, Interpretation

The experiment by FOS is not done long enough (10 seconds currently) to delineate accurate results from the data provide based on our team's algorithms parameter identification process. After careful analysis of the plots of the data received, our team noticed that in FOS 4, and 5, the steady states of those thermocouples were not reached until the last 4-5% data. This affected the results for calculating for the parameters Y(h) and Y(l) because of the structure of our algorithm. (See Parameter identification section for steps to calculate parameters). The possible errors in our algorithm for parameter identification are the use of undefined thresholds such as 4% and 2% in calculating mean values for the data. The results may vary depending on these thresholds as they are not calculated measures, but values based on a trial and error.

FOS can advertise their products in a manner that warns the buyers from using certain thermocouples for specific operations. Due to the detrimental effects a poor thermocouple can have in real world applications, it is essential that thermocouples such as FOS 3/4/5 be used for simple application such as personal experiments due to their very poor response times (see table 1 in references). They could market FOS 1 for use in critical situations such as safety systems, ICUs in hospitals etc. due to the excellent response time of 0.14 seconds. The consistency of manufacturing of the FOS is very good. The SSE of the tau values are within 0.035 meaning the FOS is very efficient at measuring tau values at a consistent range (Table 1). The price of each thermocouple is appropriate to its performance and need not to be changed. The first thermocouple FOS 1 registers a Tau value of 0.14 and is priced at $17.02. FOS 5 registers a tau value of 1.63 and is priced at $0.70. (see table 2 in references). We concluded that the price is appropriate for its performance level based on Table 2.

Figure 1: **Regression plot for Milestone 4**

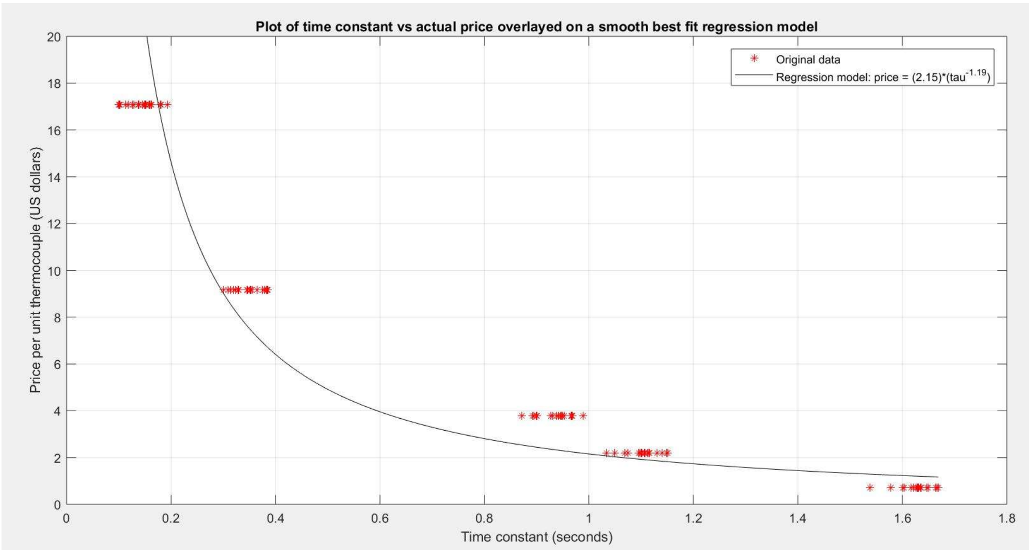

Table 1: Algorithm performance comparison for FOS designs

Model Number	M3 Algorithm			M4 Algorithm		
	τ Characteristics		Mean SSE_{mod} ($degF^2$)	τ Characteristics		Mean SSE_{mod} ($degF^2$)
	Mean (sec)	Standard Deviation (sec)		Mean (sec)	Standard Deviation (sec)	
FOS-1	0.14	0.030	2.39	0.14	0.028	0.34
FOS-2	0.34	0.038	2.46	0.34	0.028	0.34
FOS-3	0.89	0.073	2.96	0.93	0.032	0.35
FOS-4	1.05	0.087	2.83	1.10	0.031	0.35
FOS-5	1.56	0.143	2.70	1.63	0.030	0.39

Table 2: FOS model pricing

Model Number	Unit Price ($)
FOS-1	17.02
FOS-2	9.16
FOS-3	3.77
FOS-4	2.19
FOS-5	0.70

Source: ENGR132_Sp18_ProjectM3_Memo.pdf