Arduino Control System for a Temperature Bath

Winston Chu[1], Theo Siegrist, Jeff Whalen
[1]Florida State University

ABSTRACT

The goal of this work is to develop a process control system for a temperature bath that will ultimately control the reaction rate of a solution crystallizer. More specifically we wish to determine the efficacy of an Arduino-based control system for controlling temperatures to within 0.1 ºC of a target temperature. The target temperatures for the crystallization reaction are in the range of 20 to 90 ºC. In addition to the implementation of hardware, this requires the development of a quantitative model to describe the thermodynamics of the system and the implementation of a PID control algorithm into the Arduino. By acquiring thermal equilibrium data using the Arduino in data acquisition mode, we were able to develop a quantitative model of the thermodynamics of the system. We tested a number of process control algorithms that approached our target precision.

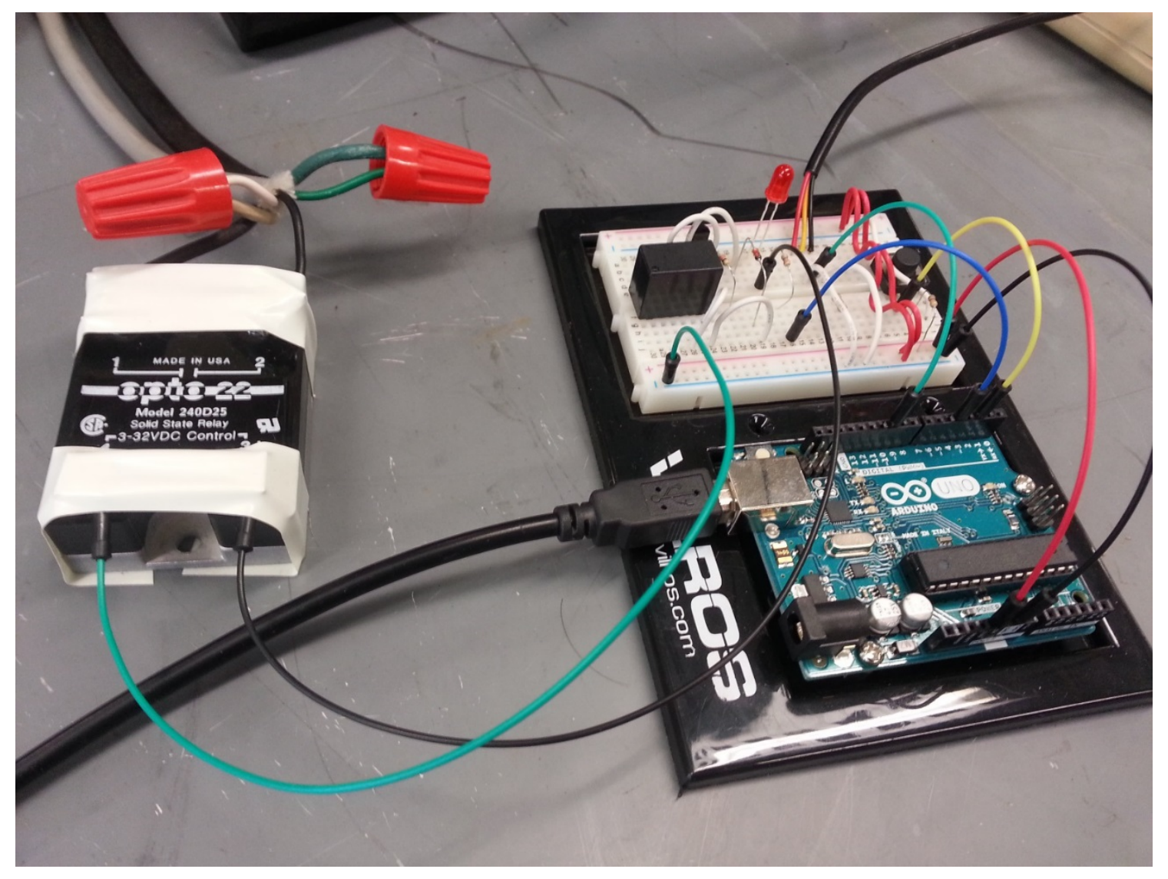
INTRODUCTION

The Arduino UNO is a microcontroller board that facilitates the intelligent and precise control of a nearly limitless range of applications via a large number of sensors and control elements. We want to test its efficacy in controlling temperature because it offers a very economical and simple platform to implement specialized and precise control of the system.

The temperature bath being controlled is intended to be used in a solution crystallizer. The specific crystallization reaction is the formation of KDP crystals (KDP = KH_2PO_4). The growth process for these crystals is controlled by the supersaturation of an aqueous solution, with an extreme sensitivity to temperature, requiring an accuracy of at least 0.1 ºC.

SYSTEM DESCRIPTION

The system consists of a 3.5 liter glass vessel filled with DI water. The system is constantly stirred by a magnetic stir-bar to maintain a homogeneous temperature profile. The system is heated by two 75 watt immersion heaters that draw power from a variac. This allows us to alter the maximum power output of the heaters. The system is controlled by an Arduino UNO that switches a solid state relay between the heaters and the variac. The power output of the heaters can also be controlled at the Arduino level by control of the heater ON state for a variable amount of time.



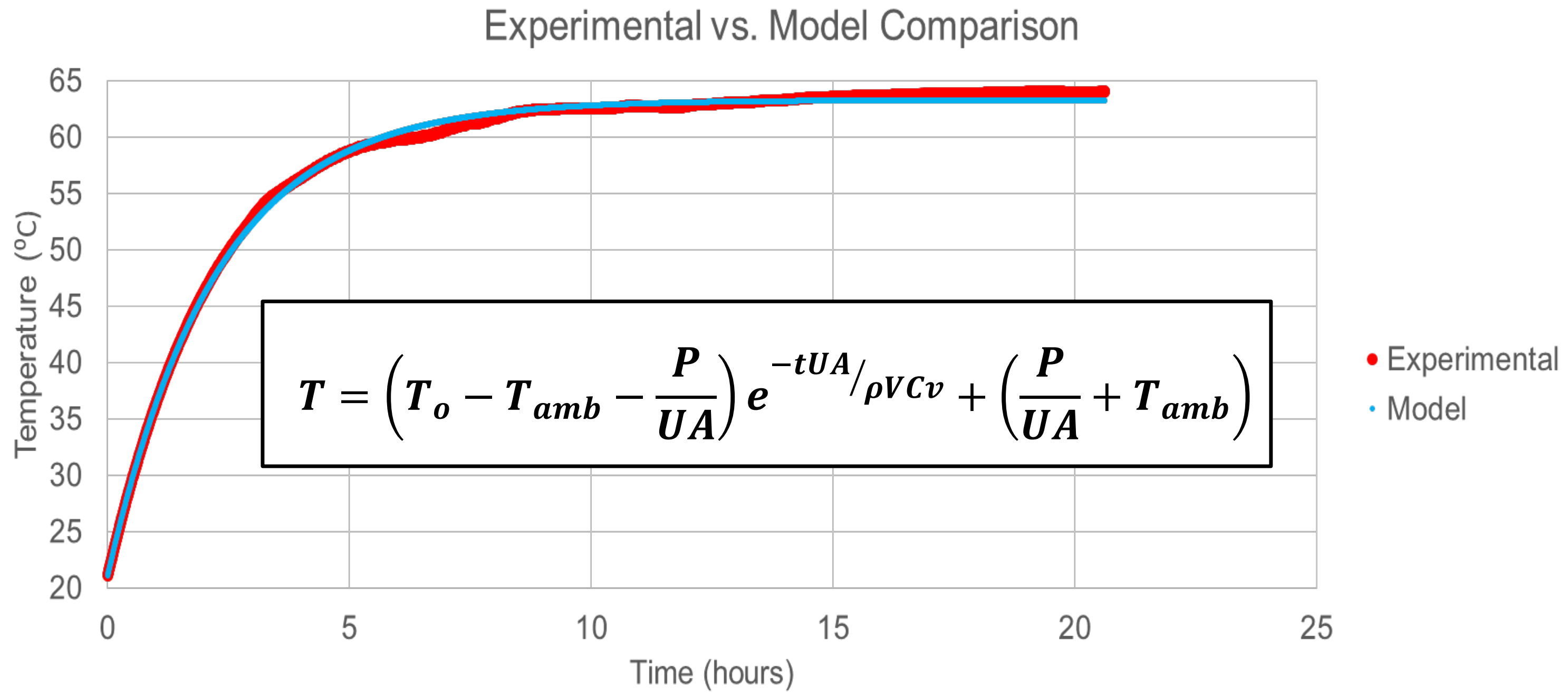
QUANTITATIVE MODELS

The quantitative model created predicts the temperature response of the system given a specific power input. The model was created starting with an energy balance that combines the heat losses into a single overall heat transfer coefficient that represents a heat loss proportional to the temperature difference between the ambient temperature and the system temperature. Integration of the differential equation gives the final expression for the temperature as a function of time.

A number of control algorithms were tested. The most basic control algorithm is the Overshoot. The Overshoot algorithm simply turns ON when the sensor reads below the target temperature and turns OFF when the sensor reads above the target. The expected drawback to using the Overshoot algorithm is an oscillatory response rather than a stable lock on the target temperature. With the Overshoot algorithm as a baseline, a more sophisticated PID algorithm was also tested. The PID algorithm contains three components that sum to the Manipulated Variable (MV). The three components are proportional, integral, and derivative components. Each of these responses can be adjusted by corresponding tuning parameters (K_p, K_i , K_d).

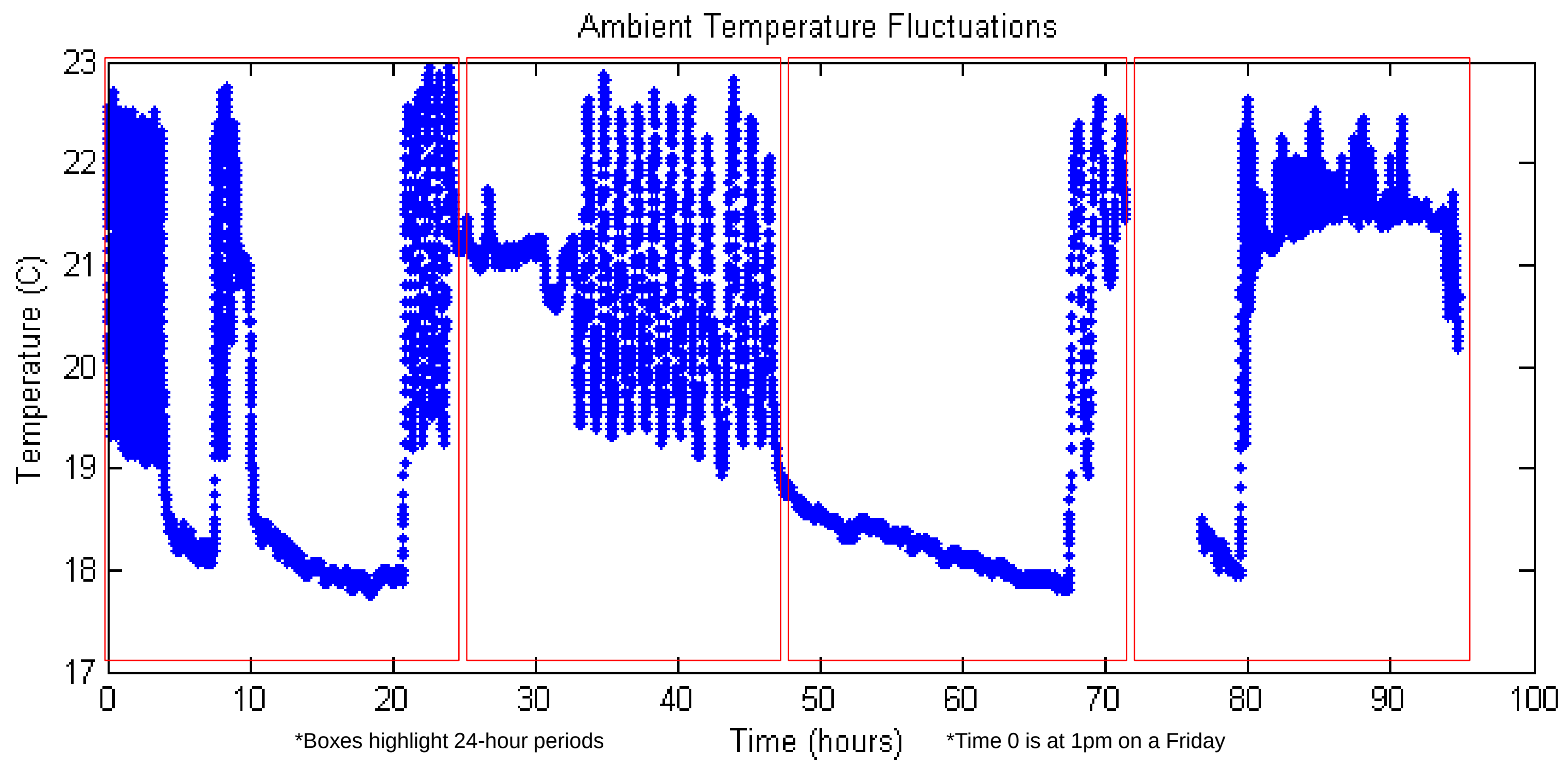
RESULTS

The quantitative model shows a strong match to the experimental data. All the variables with the exception of the overall heat loss transfer coefficient (U) and the power input (P) were defined. These values were determined from the measured data to be 13.1 J/m² K and 109.3 W respectively.


Experimental vs. Model Comparison

$$T = \left(T_o - T_{amb} - \frac{P}{UA}\right) e^{-tUA/\rho V C v} + \left(\frac{P}{UA} + T_{amb}\right)$$

One particularly interesting observation that we made in collecting thermal equilibrium data was that the curves consistently had small deviations from the expected curve. Further investigation into this issue revealed drastic temperature fluctuations in the ambient temperature. Ambient temperatures on the third floor of the Maglab were found to fluctuated by as much as 4 ºC.

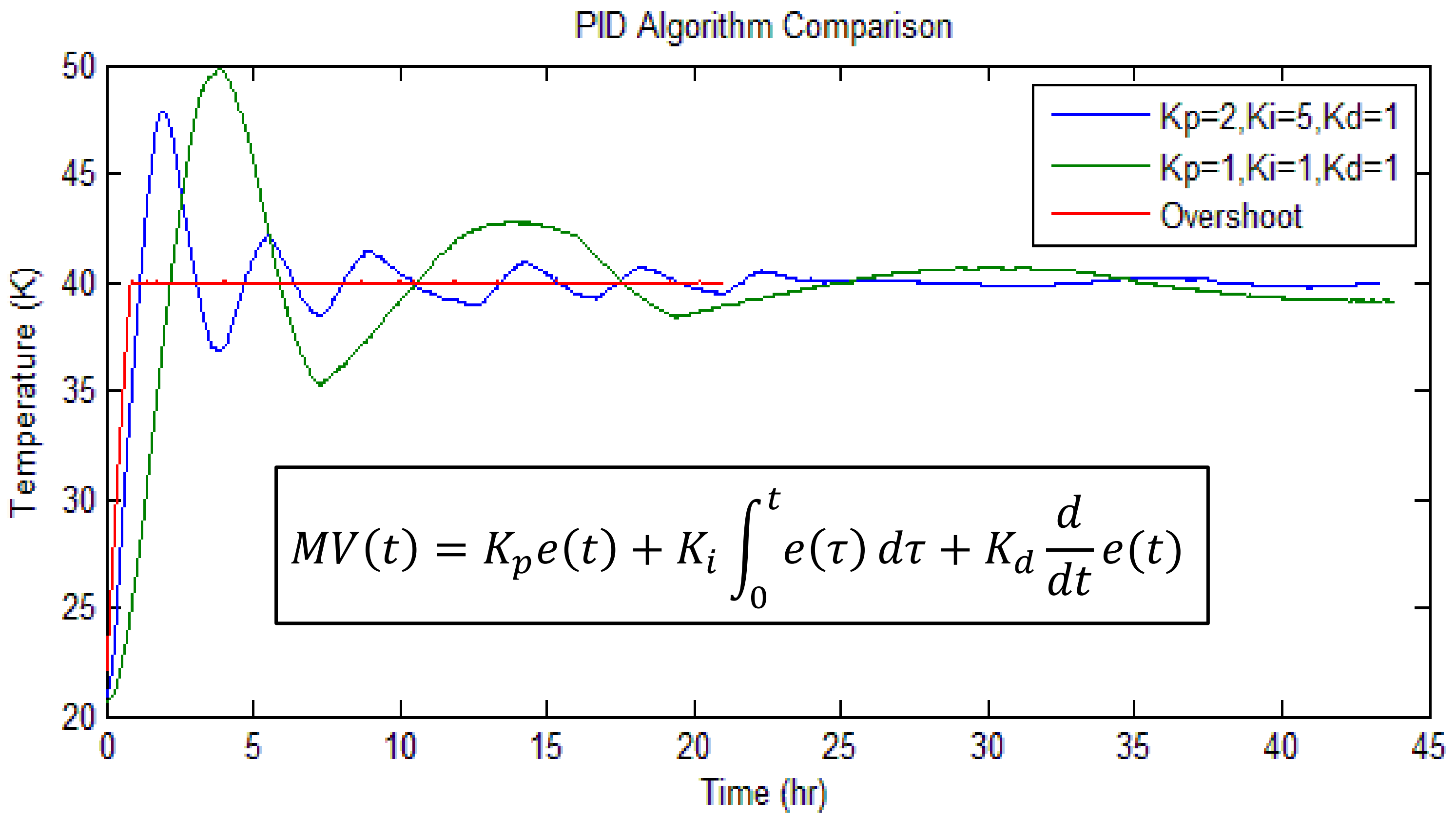

Ambient Temperature Fluctuations

*Boxes highlight 24-hour periods

*Time 0 is at 1pm on a Friday

These results indicate the need for a large mass of water to buffer the temperature fluctuations of the ambient air.

The initial control algorithm trials indicate that the tested PID tuning parameters do not out-perform the Overshoot algorithm. The Overshoot algorithm was able to control the temperature to within 0.1 ºC of the target however the response was oscillatory rather than stable.


PID Algorithm Comparison

$$MV(t) = K_p e(t) + K_i \int_0^t e(\tau)\, d\tau + K_d \frac{d}{dt} e(t)$$

FUTURE & ACKNOWLEDGEMENTS

Future work on this will include more refinements of the PID algorithm. Once the PID algorithm has been optimized for the system, a crystal growth chamber will be added and the temperature of the growth chamber controlled.

I would like to thank Dr.Siegrist for guiding me through this learning process and allowing me to utilize the workspace. This research was funded through DMR1157490