# Temperature Control Lab F: Model Predictive Control


**Objective**: Implement a Model Predictive Control (MPC) application for the Temperature Control lab. Use a second order linear empirical model with heaters 1 and 2 ($Q_1$ and $Q_2$) along with temperatures 1 and 2 ($T_{c1}$, $T_{c2}$, $T_{h1}$, $T_{h2}$). The difference between the two heater temperatures ($\Delta T_h = T_{h2} - T_{h1}$) is used to approximate the heat transferred between the two heaters.

**Heater 1 / Temperature 1**

$$\tau_{12} \frac{dT_{h1}}{dt} + (T_{h1} - T_a) = K_1\, Q_1 + K_3 \Delta T_h$$

$$\tau_3 \frac{dT_{c1}}{dt} + (T_{c1} - T_a) = T_{h1} - T_a$$

**Heater 2 / Temperature 2**

$$\tau_{12} \frac{dT_{h2}}{dt} + (T_{h2} - T_a) = K_2\, Q_2 - K_3 \Delta T_h$$

$$\tau_3 \frac{dT_{c2}}{dt} + (T_{c2} - T_a) = T_{h2} - T_a$$

There are 5 unknown parameters with 2 time constants ($\tau_{12}$ and $\tau_3$) and 3 gains ($K_1$, $K_2$, $K_3$) that need to be estimated to minimize the difference between the measured and predicted values for both temperatures. A parameter estimation exercise (see Figure 1) has identified the parameters in Table 1.

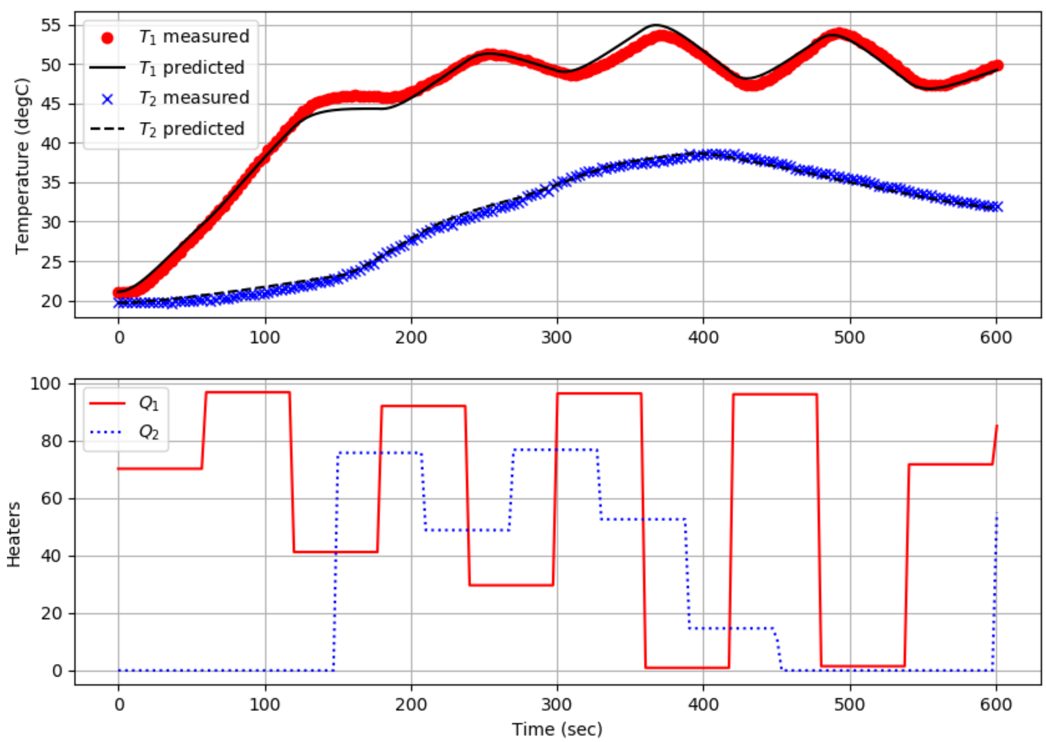
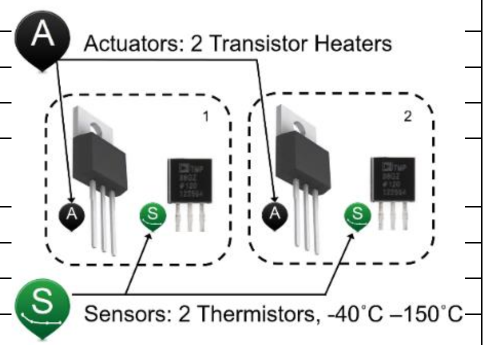



Figure 1. Parameter Estimation Results

The MHE or batch parameter estimation application can be rerun to improve parameter estimates, if desired. A more accurate ambient temperature can also be obtained by measuring the temperature with no heater output.

**Table 1**. Empirical Model Parameters and Variables

| **Parameter** | **Value** | **Description** |
|---|---|---|
| $K_1$ | 0.607 | Heater 1 Gain ($\sim\Delta T_{h1}/\Delta Q_1$) at Steady State |
| $K_2$ | 0.293 | Heater 2 Gain ($\sim\Delta T_{h2}/\Delta Q_1$) at Steady State |
| $K_3$ | 0.240 | Heat Transfer Between Heaters |
| $\tau_{12}$ | 192 sec | Time Constant for Thermal Capacitance of Heaters 1 and 2 (sec) |
| $\tau_3$ | 15 sec | Time Constant for Conduction to Temperature Sensor (sec) |
| $T_a$ | 23 °C | Ambient Temperature (°C) |
| $Q_1$ | 0-100% | Heater Output 1 (%) |
| $Q_2$ | 0-100% | Heater Output 2 (%) |
| | | |
| **Variables** | **Initial Condition** | **Description** |
| $T_{h1}$ | $T_a$ | Heater 1 Temperature (°C) |
| $T_{h2}$ | $T_a$ | Heater 2 Temperature (°C) |
| $T_{c1}$ | $T_a$ | Sensor 1 Temperature (°C) |
| $T_{c2}$ | $T_a$ | Sensor 2 Temperature (°C) |




With this model, implement an MPC application that follows setpoint changes for both temperatures. It is recommended to use a 10 minute test period where each setpoint changes 2-3 times during the testing. Include a plot of the setpoints, measured temperatures, and heater values. Discuss the controller performance and whether model mismatch or controller tuning can be improved.

**Questions for Consideration**

1. How did you decide on the appropriate control horizon time and cycle time? What are the tradeoffs between a shorter or longer time horizon?
2. How well does the MPC application track the setpoints? Are there regions where there is excessive overshoot (too aggressive) or a slow response (too sluggish)?
3. How much does the heater move each cycle of the controller when not saturated at an upper or lower limit? In some applications, excessive movement of the MV can cause problems such as with wearing out valves or motors. If it were important to have less MV movement, what tuning parameters could be adjusted?
4. What are the values for STATUS (optimizer status) and FSTATUS (measurement feedback status) for Q1 and Q2 (heater values as manipulated variables) and $T_{c1}$ and $T_{c2}$ (measured temperatures as controlled variables) for the MPC application? Create a table that shows how MVs and CVs are configured for Moving Horizon Estimation (MHE) versus Model Predictive Control (MPC).

See https://github.com/APMonitor/arduino/ at **2_Regression/2nd_order_MIMO** for starting files. The source code for the solution is located at **6_Model_Predictive_Control/2nd_order_linear**. Solution videos are available in Python (GEKKO) and MATLAB. You should strive to build your own applications versus relying on the solution templates.