1 Overview

The QNET heating and ventilation trainer (HVACT) is shown in Figure 1. The system consists of a plexiglass duct, with a heater in one end and a blower in the other end. The heater is a halogen lamp and the blower is a variable-speed fan. There is a thermistor sensor placed inside the duct to measure the temperature of the chamber and another thermistor sensor outside the chamber to measure the room temperature.
The temperature measured at the thermistor inside the chamber is to be controlled using the heater voltage while the fan is ran at a constant speed. Heat is transferred to the thermistor by radiation from the heater and by convection from the air stream. A linear approximate model will be identified from data for the nonlinear heat transfer dynamics.

2 Overall Goals

The overall goals of this experiment can be summarized as

1. Identify a model for the heating dynamics of the HVAC.
2. Implement an ON-OFF controller to regulate the chamber temperature.
3. Implement a proportional-integral (PI) controller to regulate the chamber temperature.

3 Labview and Matlab Files

The necessary files to carry out the laboratory are available in the “controls drive” (T drive). The drive should be visible from the windows “Computer” pane

T:\ME389_PRO01_HVAC\

NOTE: You can read files in this folder but cannot edit them. Copy the folder to the local C drive.

To carry out the experiment you will make use of LabView. LabView is the manufacturer of a software suite called virtual instruments. LabView essentially allows us to use one single data-acquisition (DAQ) card to perform an assortment of functions such as measuring voltages and currents and generating signals. Prior to LabView such functions were done with dedicated singular instruments that could be cumbersome and expensive. Along with a highly functional DAQ card we can use LabView to build all sorts of tools through software. LabView programs are labeled with extension .vi, standing for “virtual instrument”. Typically, the programs are called “VIs” for short. These “VIs” collect data and control the hardware, enabling us to carry out the experiment. In most cases, the LabView files will send data to a .tdms file stored in the directory \HVAC_WorkingData. The LabView data can be converted to Matlab data for easy analysis and plotting.

3.1 ELVIS Configuration

To startup the Educational Laboratory Virtual Instrumentation Suite (ELVIS), follow the procedure below.

1. Note that there is a reference printed on the wall for starting the ELVIS.
2. First confirm the prototyping board power is switched OFF, down position. Also switch the Communications to BYPASS.
3. Turn on the main power to the ELVIS with the switch located on the back of the ELVIS. The system power LED should light up.
4. Run QNET_init_elvis_bypass.vi. The bypass is successful when the LED in the VI is lit.

5. Turn on the prototyping board power. The four LEDs +B, +15, -15, +5 should light up. If any are not fully lit, consult the TA.

6. **Warning**: If either the heater or the blower start working when switching ON the prototyping board power, immediately switch OFF the Prototyping board power.

### 3.2 Data Storage

1. In each of the VIs that are used to collect experiment data, the data is saved in a format that can be imported to MATLAB for easier analysis.

2. For example in HVAC system identification VI, there is a block for saving the data

   ![Save Data](image)

3. You can specify whatever name you like. The data is stored in the directory `\HVAC_WorkingData` and can be opened with MATLAB.

### 4 Modeling

#### 4.1 Temperature Ranges

Before designing and testing any controller it is important to get familiar with the different temperature ranges that can be achieved by the heater as a function of the air flow created by the blower. Fill out Table 1 below by letting the system evolve to a steady-state temperature after selecting a combination of voltages for the heater and the blower. It is also important to register the ambient temperature at the moment of studying the temperature ranges.

<table>
<thead>
<tr>
<th>Blower Voltage</th>
<th>0V</th>
<th>2.5V</th>
<th>5V</th>
</tr>
</thead>
<tbody>
<tr>
<td>0V</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5V</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15V</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20V</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Temperature ranges
4.2 Model Identification: Open-loop Step Response Analysis

Assume the plant $G(s)$ is of the form

$$G(s) = \frac{k}{\tau s + 1},$$

i.e. a first order linear system with a stable (LHP) pole. The objective of the system identification procedure is to determine the steady state gain $k$ and the time constant $\tau$ from experimental data. If the system, $G(s)$, is excited by a step input of magnitude $A$, the response will be

$$U(s) = \frac{A}{s}, \quad Y(s) = G(s)U(s) = \frac{Ak}{s(\tau s + 1)}. \quad (2)$$

Then, the steady-state value $y_{ss}$ of $y(t)$ can be computed by the final value theorem, i.e.,

$$y_{ss} = \lim_{t \to \infty} y(t) = \lim_{s \to 0} sY(s) = Ak, \quad (3)$$

and the steady-state gain of the system can be obtained as

$$k = \frac{y_{ss}}{A}. \quad (4)$$

Note that $y_{ss}$ can be obtained directly from the experimentally step response of the system as shown in Figure 2 (left). In this example, the figures show the response of the output $y(t)$ to a step input of magnitude $A$. The value of the flattop of the curve is by definition $y_{ss}$, the stationary value reached by $y(t)$.

By computing the inverse Laplace transform of $Y(s)$ in (2), we can write

$$y(t) = Ak(1 - e^{-t/\tau}) = y_{ss}(1 - e^{-t/\tau}). \quad (5)$$

Therefore,

$$\frac{y(t) - y_{ss}}{y_{ss}} = -e^{-t/\tau} \iff \ln \left( \frac{y_{ss} - y(t)}{y_{ss}} \right) = -\frac{t}{\tau}, \quad (6)$$

which implies that if we plot $\ln((y_{ss} - y(t))/y_{ss})$ versus $t$ as shown in Figure 2 (right), we can obtain $-1/\tau$, and therefore $\tau$, just by computing the slope of the plotted line. Note that Figure 2 (right) is just a different representation of the same data shown in Figure 2 (left). In this case, instead of plotting the output $y(t)$ versus $t$, we plot the natural logarithm of the relative error between the flattop value $y_{ss}$ and the output $y(t)$ versus $t$. 

Figure 2: System identification by step response.
4.3 Identification of Linear First-order Transfer Function

Based on the method described in Section 4.2, you will identify in this section a linear first-order response model for the temperature in the chamber. This model will be identified around an equilibrium point characterized by a blower voltage of $V_b = 20\,V$ and a heater voltage of $V_h = 2.5\,V$.

**Lab Work 1: Identification of Heater Model**

1. Select $V_b = 20\,V$ and $V_h = 2.5\,V$ and let the chamber temperature evolve until it reaches a stationary temperature. Denote this equilibrium temperature as $T_e$.

2. Apply a step signal of amplitude $A$ to the HVAC heating-lamp voltage $u(t)$ at a time $t = t_{ini}$ and observe the temperature response $y(t)$. Note that the step input must be relative to the equilibrium value of the heater voltage ($V_h = 2.5\,V$). For instance, to impose a step input of amplitude $A = 1$, you must increase the heater voltage from $V_h = 2.5\,V$ to $V_h = 3.5\,V$. Note that the input of our system is the heater voltage, i.e. $u(t) \triangleq V_h(t)$, and the output of our system is the chamber temperature, i.e. $y(t) \triangleq T(t)$.

3. Shift the temperature evolution by a value of $T_e$, i.e. $T \rightarrow T - T_e$, (temperature evolution is relative to the equilibrium temperature $T_e$) and shift the time evolution by a value of $t_{ini}$, i.e. $t \rightarrow t - t_{ini}$, (time evolution is relative to the initial time $t_{ini}$). Using these new coordinates you should be able to plot a figure very similar to Figure 2 (left) and apply the method described in Section 4.2.

4. Determine the model constants $k$ and $\tau$ in (1) by applying the method described in Section 4.2.

5. Compare in the same plot the experimental step response with the simulated step response obtained based on the identified model (1).

6. LabView and Matlab files are available to carry out the system identification

   • Use LabView QNET_HVAC_Lab05_SysID_ME389_ConstBlower.vi to collect the system identification data.
   
   • Once you stop the VI, the relevant data is saved to \\HVAC\WorkingData. You can load this into Matlab with plotHVACSysIDData.m
   
   • The key steps of the method described in Section 4.2 are implemented in the Matlab script firstorder_id_step.m. This script assumes the data is given in an Excel file (e.g. HVAC_Lab_01_Data.xlsx), but it can be modified to load the data in a different format. For instance, you can modify the script to load the data in the format created by the plotHVACSysIDData.m script.

5 ON/OFF Temperature Control

ON/OFF control or relay feedback is one of the simplest HVAC control strategies. The heater is switched ON when the temperature is lower than the desired value and switched OFF when the temperature is higher than the desired value. This is a form of bang-bang control which can lead to rapid activation of the switch as the measured temperature makes small deviations above and below the desired temperature.
This rapid switching behavior called “chattering” is often undesirable because it can cause significant wear and fatigue on the actuators. To avoid rapid switching, it is common to add a hysteresis to the relay switch, as in Figure 3. Instead of the output switching as the input goes from positive to negative, the switch goes from $M$ to $-M$ and when the input falls below $-T$ and does not switch back to $M$ until the input increases above $T$.

![Figure 3: Relay switch with hysteresis.](image)

For a given temperature set point, $T_r$, the switching logic is determined by the relay switch (Figure 4). The relay is parameterized by $T_{h\_on}$ and $T_{h\_off}$. The state of the heater switch can be described by

$$V_{h\_state} = \begin{cases} \text{ON} & T_c < T_{h\_on} \\ \text{OFF} & T_{h\_on} < T_c < T_{h\_off} \\ \text{previous} & T_{h\_off} < T_c \end{cases}$$  \hspace{1cm} (7)$$

![Figure 4: Relay switch with hysteresis for the heater.](image)

Take $V_{h\_min} = 0 \text{ V}$, $V_{h\_max} = 5 \text{ V}$ and $V_h = 20 \text{ V}$. Design a controller for the following specifications. Use a non-model based control approach (i.e. tune the parameters by hand).
Lab Work 2: ON/OFF Control Experiment

1. Take first the desired set point to be $T_r = T_e$. You should see that for this set point the heater voltage should be (in average) at its minimum value the same amount of time it is at its maximum value. Move later the desired set point value within the ranges obtained in Section 4.1 and repeat the design. Try at least one case with $T_r > T_e$ and one case with $T_r < T_e$.

2. The chamber temperature peak-to-peak oscillation, $\Delta T_{c_{p2p}}$, around the desired set point level should be no more than $0.5^\circ$C. Is this possible?

3. The control parameters are
   - $\Delta T_{h_{on}}$
   - $\Delta T_{h_{off}}$

4. LabView and Matlab files are available to implement the ON/OFF controller
   - Use LabView QNET_HVAC_Lab07_ON_OFF_Control_ME389_ConstBlower.vi to implement the controller and collect the system response data.
   - Once you stop the VI, the relevant data is saved to \HVAC\WorkingData. You can load this into Matlab with plotHvacOnOffControlData.m

6 Proportional-Integral (PI) Temperature Control

In this section you will design a PI controller (proportional + integral) to regulate the temperature around a set point. The PI control structure implemented in the VI is shown in Figure 5.

![Figure 5: PI control structure for the heater mechanism.](image)
Figure 6: PI control for the heater mechanism.

Lab Work 3: Heater PI-Control Design

1. For the closed-loop configuration shown in Figure 6, where $G(s)$ is the model identified in Section 4.2, design a PI-Controller ($K_p$ and $K_i$) to meet the following step response parameters:
   - Rise time $t_r \leq 6\text{s}$
   - Overshoot $M_p \leq 10\%$
   You must show all the details of your design.

2. Is integral control necessary to remove the steady state error in tracking a step reference?

3. Plot the step response to an increase of $3^\circ\text{C}$ in the set-point temperature $T_r$. Use the MATLAB command `stepinfo.m` to verify the performance requirements are met.

4. Implement the PI controller in the experiment. Change the set-point temperature from $T_e$ to $T_e + 3^\circ\text{C}$. Are the performance requirements met? In the negative case, tune the PI gains empirically until the performance requirements are met.

5. LabView and MATLAB files are available to implement the PI controller:
   - Use LabView QNET_HVAC_Lab06_PI_Control_ME389_ConstBlower.vi to implement the controller and collect the system response data.
   - Once you stop the VI, the relevant data is saved to \HVAC\WorkingData. You can load this into MATLAB with `plotHvacPIControlData.m`