Vehicle Thermal Control with a Variable Area Inlet*

Thomas R. Laynet‡ Milton E. Franke‡ and D. Brett Ridgely§

Air Force Institute of Technology
Department of Aeronautics and Astronautics
Wright-Patterson AFB, OH 45433-7655

Abstract

This study developed a variable area inlet and controller that regulated the temperature of an electrical component with ram air. The intent of the variable area inlet was to reduce vehicle drag by eliminating inefficiencies associated with component cooling and fixed area inlets. These inefficiencies arise from vehicles moving at varying speeds through varying air temperatures. The system was successful in regulating the component temperature. A nonlinear simulation model was built in MATLABSIMULINK™ and the thermal plant in the simulation was based on the electrical component’s convective heat transfer coefficient, empirically derived in terms of the Nusselt number. Proportional, Proportional-Derivative (PD), and Proportional-Integral-Derivative (PID) controllers were designed, built, and tested.

Nomenclature

\[ V_{\infty}: \text{ Freestream Airspeed (m/s)} \]
\[ \nu: \text{ Kinematic Viscosity (m}^2\text{/s)} \]
\[ \rho: \text{ Density (kg/m}^3\text{)} \]

1 Introduction

Automobiles and many other types of vehicles use ambient air as a heat sink for cooling components. Many rely on ram air for forced convection cooling with fans to augment the ram air at low speeds. The use of ram air for forced convection cooling has many advantages. The most significant advantage is that it is available as long as the vehicle is moving. In addition, ram air ducts are lightweight, reliable, and require low maintenance.

Most often, fixed area inlets are used to bring ram air into compartments or around components. A characteristic of the fixed inlet design is that the speed of the ducted air generally increases with vehicle speed. This characteristic along with varying ambient air temperatures and varying vehicle speeds pose trade-offs to the designer of fixed inlets. One design approach is to design the inlets for low speeds to reduce the need for augmented fan cooling. These designs often have large inlets and, unfortunately, they tend to provide excess air at high speeds.

Carr [1] indicated that a part of the aerodynamic drag of an automobile results from “cooling system drag”. Carr defined automobile cooling system drag as the drag difference when cooling air intakes were open and when cooling air intakes were closed. He obtained this data from 100 different automobiles that were tested in the MIRA Full-Scale Wind Tunnel between 1990 and 1994. The drag coefficient had a mean value of 0.019, which was approximately 6% of the typical total vehicle drag coefficient.

With the inefficiencies associated with the fixed area inlet, there appeared to be an opportunity to improve vehicle efficiency and performance. This opportunity was the motivation that led to the study of the variable area inlet.

During the conceptual stage of the development it was determined that by building an experimental model of a variable area inlet and by conducting simulations on a computer model, the temperature control concept could be proven and a modeling methodology could be established. It should be noted that the working model was not designed to be tested on an automobile. Instead, the

*This paper is declared a work of the U. S. Government and is not subject to copyright protection in the United States.
†Captain, USAF, Wright Laboratory, Eglin AFB, FL
‡Professor
§Assistant Professor
model was designed for testing in a laboratory wind tunnel. The prototype inlet was designed to automatically minimize the inlet area throughout the range of all vehicle speeds and all ambient temperatures while providing sufficient air for forced convection cooling. In other words, this inlet was designed to regulate the cooling airflow over a component to control the temperature of that component to a desired temperature.

To design a controller for this system, a thermal control model was needed. A background search of thermal control models revealed that models such as those in Franklin, et al. [2], Close & Fredrick [3], and Cannon [4] were available, but they did not use the cooling fluid velocity as the input variable. This study extended these models by using cooling airspeed as the input variable and by basing the component thermal control model on the Nusselt number. Since the cooling airspeed was variable, the convection coefficient was variable also.

2 Experimental & Analytical Set-Up

2.1 Hardware Model Description

The actual hardware model consisted of a 0.15 m inner diameter circular duct, a butterfly valve connected to a stepper motor, a PC and a small electronic board with an electrical component that generated heat. A cut away illustration, Figure 1 shows the air inlet and the relative position of each component. This assembly was positioned inside a wind tunnel. Temperature regulation occurs by the PC-based controller operating on the error between the reference temperature and the measured temperature \( \text{error} = T_{\text{ref}} - T_1 \). The measured temperature is obtained from a thermocouple mounted on the top center of the electrical component. The output of the controller is a pulse stream that is sent to the stepper motor. The stepper motor shaft rotates one step for every 5-volt pulse it receives. Each pulse corresponds to \( 360/4000 \) (0.09) degree of shaft rotation. The pulse rate corresponds to the angular velocity of the motor. Lastly, the sign of the error indicates the direction of motor travel.

Figure 1: Illustration of Experimental Model

2.2 Simulation Model

MATLAB\textsuperscript{TM} [5] SIMULINK\textsuperscript{TM} [6] was used to model the dynamic response of the system. The step size for each iteration was set to one second to match the hardware model's data acquisition and control step size. The actual hardware model was represented by a closed-loop negative unity feedback system modeled in block diagram form shown in Figure 2. Each block represents the modeled dynamics/characteristics of the hardware model. The actuator plant modeled the stepper motor dynamics, the inlet plant modeled the throttling characteristics of the butterfly valve, and the thermal plant modeled the component temperature with convective heat transfer losses.

3 Results

3.1 Proportional Controller Response

After the simulation model was correlated [7], the temperature regulating capabilities of a proportional controller were evaluated. There were two aspects to this evaluation. The first was to understand how the system regulated temperature and used inlet control when the freestream airspeed and the difference between the reference temperature and ambient temperature were varied. The second aspect was to develop an algorithm that would schedule the proportional gain to keep temperature responses within a desired performance range when the airspeed and temperature difference \( (T_{\text{diff}}) \) parameters changed.

The first aspect was accomplished by making 27 runs with variations in the gain, airspeed, and temperature difference. There were three variations in airspeed and three variations in temperature difference (at each airspeed) for a total of nine test conditions. At each of these nine conditions, gain was varied with values of 0.5, 2.0, and 8.0. Table 1 shows the specific values for the nine test conditions.

Figure 3 shows the temperature and inlet position responses for the simulation and hardware models, (Condition 1 in Table 1) with a gain of 0.5. For the hardware response, there is a large temperature overshoot of 40%, with little control usage. Figure 4 shows the responses (temperature and inlet position) at the same condition (Condition 1 in Table 1) with a gain of 8.0. For the hardware response, the temperature overshoot is much smaller (16%), but the control usage is much greater. Figures 3 and 4 both show that the simulation and hardware models are well correlated at Condition 1. Favorable corre-
Table 1: Test Condition Values

<table>
<thead>
<tr>
<th>Condition</th>
<th>Freestream Airspeed (m/s)</th>
<th>Temperature Difference (°C)</th>
<th>Ref Temp (°C)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>8.9</td>
<td>16</td>
<td>35</td>
</tr>
<tr>
<td>2</td>
<td>8.9</td>
<td>21</td>
<td>40</td>
</tr>
<tr>
<td>3</td>
<td>8.9</td>
<td>31</td>
<td>50</td>
</tr>
<tr>
<td>4</td>
<td>17.9</td>
<td>15</td>
<td>35</td>
</tr>
<tr>
<td>5</td>
<td>17.9</td>
<td>20</td>
<td>40</td>
</tr>
<tr>
<td>6</td>
<td>17.9</td>
<td>30</td>
<td>50</td>
</tr>
<tr>
<td>7</td>
<td>26.9</td>
<td>10</td>
<td>30</td>
</tr>
<tr>
<td>8</td>
<td>26.9</td>
<td>15</td>
<td>35</td>
</tr>
<tr>
<td>9</td>
<td>26.9</td>
<td>20</td>
<td>40</td>
</tr>
</tbody>
</table>

Table 2: Scheduled Proportional Controller Data

<table>
<thead>
<tr>
<th>Condition</th>
<th>Freestream Airspeed (m/s)</th>
<th>Temperature Difference (°C)</th>
<th>Gain</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>8.9</td>
<td>16</td>
<td>30</td>
</tr>
<tr>
<td>2</td>
<td>8.9</td>
<td>21</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>8.9</td>
<td>31</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>17.9</td>
<td>15</td>
<td>9</td>
</tr>
<tr>
<td>5</td>
<td>17.9</td>
<td>20</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>17.9</td>
<td>30</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>26.9</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>8</td>
<td>26.9</td>
<td>15</td>
<td>6</td>
</tr>
<tr>
<td>9</td>
<td>26.9</td>
<td>20</td>
<td>4</td>
</tr>
</tbody>
</table>

Figure 3: Comparison of Simulation and Hardware Model Responses: Condition 1, Gain = 0.5

Figure 4: Comparison of Simulation and Hardware Model Responses: Condition 1, Gain = 8

values varied in a second order nature with both airspeed and temperature difference. Using the least squares methods, the scheduled gain can be expressed as a second order function of both airspeed and temperature difference as

\[
Gain = 47.45 - (0.8767 \times T_{diff}) - (0.0032 \times T_{diff})^2 - (1.6 \times V_m) + (0.0223 \times V_m^2) \quad (1)
\]

An important point to recognize is the increased complexity of this temperature control system. With the fixed proportional controller, only one thermocouple on the electric component was required. For the scheduled proportional controller, the same component mounted thermocouple was required, but additional sensors were required to measure cooling air temperature and freestream airspeed.

The test condition that was shown previously was rerun on the simulation model to review the temperature response with the scheduled controller values. For Condition 1, when the scheduled gain of 20.14 was used in the simulation, the temperature overshoot was 11%. This was slightly greater than the 10% objective, but it was also significantly lower than the 40% shown in Figure 3.

An observation from this evaluation was that a scheduled proportional controller based on the least squares algorithm did keep the temperature response close to the performance objective. It also required increased hardware complexity due to the added sensors.
3.2 Linearized Plant & Root Locus Controller Design

The proportional controller designs in the previous section did regulate temperatures to the reference values, and by using the scheduled controller there was some performance control when either the freestream airspeed or the temperature difference changed. However, the proportional controller designs tended to have responses that oscillated before achieving their steady state values. Unfortunately, to design a controller that is more advanced (than the proportional controller) and capable of reducing the oscillations, the design tools and techniques usually require linearized plants. As a result, the system was linearized and a root locus diagram was used to design a controller. The proportional-derivative (PD) controller that was designed in this section shows significant improvements in control usage and oscillation reduction over the simple proportional controller.

The first task in this control design was to obtain a linearized model of the system. To start out, a first-order Taylor series expansion was performed on the thermal plant energy equation (details are given by Layne [7]). With the thermal plant linearized, the inlet and motor plants were combined to form a single linear system plant.

The energy equation consists of a convection term (with the convection coefficient written as a function of velocity), a thermal generation term, and a thermal storage term. The conduction and radiation losses were calculated to be small, so they were neglected. Therefore, the energy equation takes the form

\[-c\left(\frac{V L}{V}\right) AT + q_{gen} = \rho c_p V \frac{dT_1}{dt}\]  

which can be expanded and rearranged to get

\[-cV^m \left(\frac{L}{V}\right) A T_1 + cV^m \left(\frac{L}{V}\right) A T_\infty + q_{gen} - \rho c_p V \frac{dT_1}{dt} = 0\]  

Since

\[f(V) = f(V_0 + \Delta V) = f(V_0) + \frac{df}{dV} V_0 \Delta V\]  

we have

\[V^m = V_0^m + m V_0^{m-1} \Delta V\]  

and

\[T_1 = T_{10} + \Delta T_1\]  

After substituting, canceling out the nominal point and nonlinear terms, and converting to the s-domain, we get

\[-cV^m \left(\frac{L}{V}\right) A T_1 + cV^m \left(\frac{L}{V}\right) A T_\infty + \rho c_p V \frac{dT_1}{dt} = 0\]  

Rewriting the thermal plant in input/output format for the change in surface temperature with small changes in test section velocity produces

\[\frac{\Delta T_1(s)}{\Delta V(s)} = \frac{cV^m \left(\frac{L}{V}\right) A T_\infty}{\rho c_p V s + cV^m \left(\frac{L}{V}\right) A T_\infty}\]  

Next, the thermal plant needs to be combined with the inlet plant and the actuator plant. The linear region of the inlet valve can be approximated by

\[\frac{\Delta V(s)}{\Delta \theta(s)} = \frac{V_\infty}{62}\]  

so that

\[\frac{\Delta T_1(s)}{\Delta P(s)} = \frac{cV^m \left(\frac{L}{V}\right) A T_\infty}{\rho c_p V s + cV^m \left(\frac{L}{V}\right) A T_\infty}\]  

Finally,

\[\frac{\Delta P(s)}{\Delta \text{error}(s)} = \frac{K(s)}{s + \left(\frac{cV^m \left(\frac{L}{V}\right) A T_\infty}{\rho c_p V}\right)}\]  

By expanding and simplifying, the open loop system in its final form is

\[\frac{\Delta T_1(s)}{\Delta \text{error}(s)} = \frac{(0.09)V_\infty cV^m \left(\frac{L}{V}\right) A T_\infty}{s + \left(\frac{cV^m \left(\frac{L}{V}\right) A T_\infty}{\rho c_p V}\right)}\]  

The final form of the open loop transfer function reveals that the linearized system is a second order system with a pole at the origin and a pole in the left half plane. Two other observations can be made from this transfer function. First, the freestream airspeed $V_\infty$ and the difference between the nominal point temperature $T_{10}$ and freestream temperature $T_\infty$ act much like a proportional
gain. Second, the thermal capacitance will influence the speed of the response. A large thermal capacitance will slow down the system response. Condition 1 was evaluated with the following constants and nominal plant values:

\[ c = 0.2125 \quad m = 0.675 \quad V_o = 6.087m/s \]
\[ L = 0.0222m \quad \nu = 0.000016m^2/s \quad k_f = 0.026W/m \cdot K \]
\[ T_\infty = 292K \quad T_1 = 308K \quad A = 0.0016m^2 \]
\[ \rho c_p V = 5.86J/K \]

After the substitution, the open loop transfer function reduces to

\[ \frac{\Delta T_1(s)}{\Delta \text{error}(s)} = \frac{0.000696}{s(s + 0.0304)} \times K(s) \quad (15) \]

A PD controller was used, which has the form

\[ K(s) = K_p + K_d \delta = K_d \left( s + \frac{K_p}{K_d} \right) \quad (16) \]

By placing the zero at \( s = -0.1 \) such that the controller \( K(s) = K_d(s + 0.1) \), the system response speeds up with increasing values of \( K_d \). The root locus is shown in Figure 5.

![Root Locus Diagram with a Zero at \( s = -0.1 \)](image)

The performance of the PD controller was compared against the scheduled proportional controller from the previous section evaluated at Condition 1. The PD controller was designed to match the scheduled proportional gain controllers temperature overshoot for Condition 1. The values selected for this controller were \( K_p = 6.0 \) and \( K_d = 60 \). The PD controller design proved to be superior to the scheduled proportional controller design. Figure 6 shows a comparison of the simulation responses when the PD and scheduled proportional controllers were used at Condition 1 in Table 1. The PD controller significantly reduced the number of oscillations in the temperature and inlet responses for Condition 1 and the other conditions in Table 1.

![Comparison of Scheduled Proportional Gain and PD Controllers](image)

There are a few significant observations that can be drawn from this evaluation. First of all, the linearized transfer function requires a great deal of information about the nominal point being evaluated. Unless a prototype system is available, this data may not be available and will need to be estimated. However, a PD controller designed with one thermocouple outperforms a scheduled proportional controller design that utilized two thermocouples and an airspeed sensor.

### 3.3 MATLAB™ Nonlinear Toolbox Controller Design

The final method that was used to design a controller required the SIMULINK™ model. With this model and the Nonlinear Control Design (NCD) Toolbox [8], a proportional-integral-derivative (PID) controller was designed.

In this design, the controller was assumed to be of the form

\[ K(s) = K_p + \frac{K_i}{s} + K_d \delta = \frac{K_d \left( s^2 + \frac{K_p}{K_d} s + \frac{K_i}{K_d} \right)}{s} \quad (17) \]

so that the variables are \( K_p, K_i, \) and \( K_d \). Since the values of \( K_p, K_i, \) and \( K_d \) are adjusted with gradient methods, the simulation needs initial values for these variables. This need gives rise to a limitation. If poor initial values are used, the algorithm may not be able to obtain the desired performance response. For example, the NCD program failed when the constraint window shown in Figure 7 was used to design a PID controller with initial values of \( K_p = 8, K_i = 0, \) and \( K_d = 0 \). Consequently, the designer needs a good idea of the values that will be used for each of the controller variables. When the previous PD controller values \( (K_p = 6, K_i = 0, \) and \( K_d = 60) \) were used as initial conditions, the algorithm converged to the values of \( K_p = 4.72, K_i = -0.00072, \) and \( K_d = 100 \). This corresponds to a controller of the form

\[ K(s) = \frac{100(s^2 + 0.0472s - 0.000072)}{s} \]
\[ = \frac{100(s + 0.047)(s - 0.000152)}{s} \quad (18) \]
Figure 7: NCD Design Constraint Window with PID Controller Response

The root locus of this system is similar to the PD root locus in Figure 5 with the addition of a pole at the origin, a new zero at $s = 0.000152$, and the PD zero moved in to $s = -0.047$ from the $s = -0.1$ location. This PID controller was compared against the previous PD controller design. Figure 8 shows the comparison of the two controllers at Condition 1. The temperature overshoot was greatly reduced. There appeared to be no need for any gain scheduling when evaluating responses at the other test conditions in Table 1.

4 Summary and Conclusions

A working hardware model of a variable area inlet was built and tested. The experimental model validated the temperature control concept that a variable area inlet could regulate a component temperature with ram air. A simulation model of the physical system was constructed and the simulation responses compared favorably with the hardware model responses. As a result of the favorable comparison, the simulation model demonstrated that it did not require a linearized thermal plant. The simulation also modeled the convection coefficient as a variable with velocity as an input to the thermal plant. Future applications could use the same modeling methodology that was used in this work.

A scheduled proportional gain controller was designed for a varying freestream temperature and airspeed environment with least squares methods. This controller would require increased system complexity by requiring a freestream temperature sensor and a freestream airspeed sensor. The scheduled proportional gain controller performs better than a fixed gain proportional controller by regulating temperature better while using less control in a varying environment.

The linearized system revealed that the denominator of the open loop system used in this study has a root at the origin and another negative root. By using a root locus diagram and designing a PD controller with a zero that has a magnitude larger than the largest denominator root, an improvement to the dynamic proportional controller was realized. The major improvement was the reduction in control oscillations. Furthermore, unlike the proportional controller, the PD controller did not appear to require any gain scheduling for the range of freestream airspeeds and temperature differences that were used in this work. This eliminates the need for additional sensors.

Lastly, a PID controller designed with the nonlinear control design toolbox performed better than the PD controller by greatly reducing the temperature overshoot. Like the PD controller, the PID controller did not appear to require any gain scheduling for the range of freestream airspeeds and temperature differences that were used in this work. Excellent correlation was obtained between the simulation responses and the actual hardware responses.

References