THE TEMPERATURE CONTROLLER
AN INTRODUCTION TO SERVO LOOPS

1 INTRODUCTION

In the geiger experiment the computer was a passive spectator, just gathering data. For the laser experiment it caused a change (using a stepper-motor) and recorded the effect. We now move to the most complex type of task: Real-time control of a system. To do this you need to be able to measure the state of a system and then act in a way that will drive that system to a desired state in a controlled manner, and then keep it there.

In this experiment the computer is used to control the temperature of an aluminium block. The computer reads the amplified output of a thermocouple using an Arduino shield to determine the block temperature, compares this measured temperature with the set, or desired, temperature and then adjusts the power supplied to a heater using a signal sent to a PWM output port. Such a system, including the algorithm for determining what change in heater power to use, forms a *servo control loop*, examples of which may be found in any situation where some quantity is to be maintained at a particular value by adjusting a control parameter. Examples range from the mechanical governor which controls the rotational speed of a steam engine by adjusting the steam pressure, to the highly sophisticated computerised systems which control the power output of a nuclear reactor. Your own body is full of systems that regulate such things as pH, pressure and glucose levels in your blood, and a failure in one can cause immediate and rather permanent damage.

Servo controllers used to be analogue instruments because the discrete nature of computer sensing and control tended to reduce both accuracy and control stability. However they have long since been superseded by digital systems as faster computers and higher resolution ADC/DAC became widely available. Computer-based control systems also provide greater flexibility, and the development of far more sophisticated control algorithms fed by a complex sensor array can allow highly complex response behaviours to be programmed into the servo loop.

2 THEORY OF CONTROL

2.1 GENERAL

The aim of any temperature controller is to reach the desired temperature (set point) as quickly as possible with the minimum of overshoot, then hold at the set point as accurately as possible. When a steady state is established, the heating provided by the controller will exactly balance the heat lost by the system to its surroundings. A further function of the controller is to follow any changes either in the set point or in the surroundings as rapidly as possible. Thus the criteria for good control are:-

CONTROL ACCURACY	The mean temperature of the system should be as close as possible to the desired temperature.
CONTROL STABILITY	The fluctuations above and below the mean temperature should be small.
CONTROL RESPONSE	The system should follow changes in the set point as rapidly as possible.

In the following sections a number of possible control systems of increasing complexity are described, culminating in Three–Term or P.I.D. control.

2.2 OPEN LOOP OPERATION

In an open loop system a fixed heater power is applied and the system is allowed to come to equilibrium. There is no control as such, since the heater power can only be changed by the intervention of a human operator. The system takes a long time to reach equilibrium and any changes in the heat loss from the system produce corresponding changes in the system temperature. The boiling rings on a domestic cooker and cheap soldering irons use this form of "control".

2.3 ON–OFF CONTROL

In an on–off control system the heater power is either full on, if the temperature is below the set point, or off, if it is above. This kind of system is simple and easy to implement. The control accuracy and response can be made very good with this form of control and the system can be made largely immune to changes in heat loss. However, the control stability can never be very good since the system temperature must always cycle above and below set point. The magnitude of the temperature fluctuations depends on the the thermal properties of the system. For some systems, where temperature fluctuations are not important, this is perfectly satisfactory and provides a simple, robust system of control (e.g. the domestic electric oven). However, on–off control is inherently noisy since the heater current switches frequently between maximum and zero, and it cannot be used where sensitive electronics are to be operated.

One common compromise, which reduces noise at the expense of greater temperature variation, is to add hysteresis to the controller (remember the Schmitt trigger?) so that once the set point has been exceeded and the power switched off, it does not come back on again until the temperature is a few degrees below the set point. This is commonly achieved mechanically in domestic heating systems for example.

2.4 PROPORTIONAL CONTROL

A proportional control system overcomes the problem of temperature cycling and switching noise by allowing the heater power to be continuously varied. The heater power at any instant is proportional to the error between the measured and desired temperatures. Thus a large negative error (measured temperature below desired temperature) will produce a large heater power in order to correct that error.

If the output power were proportional to the error over the whole range of the instrument (e.g. $0 - 250^o$C in the case of a domestic oven), a negative error equal to half the total span of the instrument would be necessary to generate the full output voltage and a similar positive error would be necessary to reduce the output voltage to zero. Thus, although the controller would not suffer from temperature fluctuations, the accuracy would be very poor.

NOTE: this is an example of negative feedback, the behaviour is very similar to op–amps except the gain is much smaller.

2.4.1 PROPORTIONAL BAND

The PROPORTIONAL BAND of a controller is defined as the band of input signals over which the output is proportional to the error. It is may expressed either as a percentage of the total input span of the instrument or in degrees and is centered about the set point. For the purposes of the experiment here, it is most useful to work in degrees.

In the situation described in 2.4, where the output is proportional to the error over the whole span, the proportional band is 100% or 250^oC. By reducing the proportional band, the accuracy of the controller may be improved since a smaller error will then be necessary to produce a given change in output.

(The proportional band thus provides a convenient method of defining the gain of the controller in a way which is independent of the type of sensor or heater voltage in use. A small value of proportional band represents a high value of controller gain).

One might then expect that, by sufficiently reducing the proportional band, any required control accuracy could be obtained. Unfortunately, as the proportional band is progressively reduced, there will come a point at which temperature oscillations reappear. (In the limit, a controller with a proportional band of 0% is an on–off controller, as described above).

The reduction in proportional band, which can be achieved before the onset of oscillations, will depend largely on the design of the system being controlled. In some systems, it may be possible to achieve the required control accuracy without oscillations but, in most cases, this will not be so.

2.4.2 INTEGRAL ACTION

To overcome this problem, INTEGRAL ACTION is introduced. Consider a system controlled by proportional action as described above, with the proportional band sufficiently large to prevent oscillation. The result will be stable control, but with a residual error between the measured and desired temperatures. Suppose this error signal is fed into an integrator, the output of which is added to the existing controller output. The effect of this will be to vary the overall output until control is achieved with no residual error. At this point, the input to the integrator will be zero (as there is no proportional error signal) and this will therefore maintain a constant output. Integral Action has thus served to reduce the residual error associated with a Proportional control system. Provided the contribution from the integrator is only allowed to vary slowly, the Proportional Action will prevent the occurrence of oscillations. The response of the integrator is characterised by the INTEGRAL ACTION TIME (IAT). This is defined as the time taken for the output to vary from zero to full output in the presence of a steady error of 1 Proportional Band.

To ensure that the integrator itself does not give rise to oscillations, it is usual to employ an Integral Action Time of at least twice the response time of the system. The actual response time of a system has many components including sensor coupling, signal and control delays, and the rate at which changes can occur. Optimisation (or tuning) can involve detailed system modelling, direct measurement, inspired guesswork, or some combination of all three.

2.4.3 DERIVATIVE ACTION

The combination of Proportional and Integral Action will suffice to ensure that accurate and stable control can be achieved at a fixed temperature. However, it is possible by the use of DERIVATIVE ACTION to improve the response of the system to changes in the set point. Without derivative action, many systems will tend to overshoot the desired temperature. Derivative Action monitors the rate at which the measured temperature is changing and modifies the control output such as to reduce this rate of change. In this way, overshoot can be reduced and in many systems completely eliminated. (Derivative Action is exactly analogous to the use of velocity feedback in mechanical servo systems and serves the same function).

Like Integral Action, Derivative Action is characterised by an Action Time. If the measured temperature is changing at a rate of 1 Proportional Band per DERIVATIVE ACTION TIME (DAT), Derivative Action will contribute a signal sufficient to reduce a full output to zero or vice versa.

Do I need to suggest that you check out the *wikipedia* page on PID control?

3 THE EXPERIMENT

3.1 THE HARDWARE

You are provided with a small aluminium block with a $\sim 14\,\Omega$ cartridge heater and a chromel–alumel thermocouple. Both are connected your Arduino. The heater power comes from a 14 V_{rms} transformer (the black brick). You also have a fan that can be controlled by your Arduino. This will save you some time by allowing you to cool the block more quickly between test runs. It will also allow you to check the impacts of perturbations.

The short program that is provided simply reads the temperature and can be used to send a fixed output power to the heater. There is a simple python plotting package provided for real-time visualisation of system response.

The heater power is controlled by adjusting the *duty cycle* (the fraction of time that it is ON) rather than using an adjustable power: time proportioning. This is simpler to implement with the Arduino as its analog outputs rely on pulse-width modulation rather that being genuine analog signals. We have adjusted the cycle time to one second (resist the urge to fiddle with this) and as the solid state relay (the blue brick) switches at the zero crossings of its input, that leaves you with control adjustments of 1/120, or ~0.8%.

3.2 SOME SIMPLE TESTS

As a first step to make sure everything is connected and working, look at the system's response to a fixed (less than 100%) power input. Note the slow approach to equilibrium. Can you estimate the heating time constant? Try some different values to get an idea of the power needed to reach various temperature. **Do not exceed 150^{o}C.**

With the block at some elevated temperature, cut the power and observe the cooling. Can you estimate the cooling time constant? Is it different from the heating time constant? What impact might this difference have on your control system?

Try adding the fan.

Look at the initial response to a power input – How long after the power is first applied does the thermometer start to respond? What impact might this delay have on your control system?

3.3 NOW THE REAL WORK

Now implement the on/off controller. Notice how the output of the power supply oscillates rapidly near the set point due to noise on the thermometer signal. Investigate the effect of adding hysteresis to the control response – put a Schmitt trigger into the software – try different amounts of hysteresis and observe how it affects stability and accuracy. Note how the heating and cooling response times are different. What effect will this have on control stability?

3.3.1 Proportional control

Now write a *real* temperature control program using proportional response and investigate the effects of increasing the gain (reducing the proportional band) on the response time (i.e. the time required to reach stable control), system stability and the residual control error (the difference between the steady state temeperature and the set point).

3.3.2 P–I control

Add an integral term to the controller. Observe how this eliminates the residual control error, and allows accurate control at lower gain.

WARNING: Once you decide to add the integral and/or derivative terms to the control response you have to pay attention to *time*. A free-runnning loop with no clock cannot be used as the action times would be undefined.

The time response of this term (the integral action time or "IAT") must be adjusted to match that of the system or it will cause the controller to oscillate – a common problem with servo–loops. What is the origin of this problem?

3.3.3 P–I–D control

Try adding a derivative term – why does this make things worse? How would you correct for this? As with the integral action time, the derivative action time (DAT) also needs to be tuned to the specific system.

One really simple way to implement (fake?) the derivative term if a rapid response is less important than avoiding an overshoot, is to just limit the rate of change of the *setpoint*. Instead of passing a new setpoint directly to the temperature controller and having it dump lots of power into the heater as it races off to reach the new temperature, have the control code ramp the control setpoint up to the target so that the system is under contol all of the way to the desired setpoint. This approach has many limitations and will not suit all applications, but it is simple to implement.

As a test of the effectiveness of your various controllers, investigate their response to a sudden change in environment. With the block at a steady temperature, switch on the fan (to increase the losses from the system) and observe the new control point and how long the system takes to settle. Comment.

3.3.4 An exercise

Estimate the heat capacity of the system by determining the power required to hold the block at a number of temperatures and then measuring its cooling rate in the same temperature range. Assuming that the block is solid aluminium, compare your result with the recognised value. Comments should be made both on the tabulated value (the physics behind it) and on any differences between what you measure and the expected value. Do not weigh the block. This is an exercise in estimation and errors, not precision.

4 SOME GUIDELINES

(i) In computer control systems the proportional band is most usefully expressed as a temperature range and the error normalised to it:

$$E = \frac{T_{set} - T_{measured}}{Band}$$

(ii) The controller should supply half power at the set point (Why?) i.e. with E = 0.

$$Power = (0 \cdot 5 + E) \times P_{max}$$

(iii) If noise is a problem on the temperature signal, then it should be sampled more often and a number of readings averaged to obtain T_{meas}.

(iv) The integral term should only be used when the temperature is inside the proportional band, and should be set to zero otherwise. This improves the control response. Try not doing this to check its effect. (Comment)

(v) The full form of the power output is:

$$P = (0 \cdot 5 + E + I + D) \times P_{max}$$

$$Where\ E = \frac{T_s - T_m}{B} \qquad Proportional\ Response$$

$$I = \int Edt \qquad Integral\ Response$$

$$D = -\frac{dE}{dt} \qquad Derivative\ Response$$

(vi) Pick a reasonably high temperature for your control tests. If you work too close to room temperature the system takes a long time to cool after an overshoot, if you try to work too hot, the supply may not be powerful enough to balance the losses. Somewhere between 100^{o}C and 150^{o}C works well. The upper limit is around 200^{o}C (please do not test this limit!).

Notes:

- Make sure that you use the same conditions when comparing performance, *i.e.* starting point and set temperature.
- Plot data on the same scales so that comparisons make sense.
- Specify conditions : band IAT DAT on plots.

5 THE CHALLENGE

Brandon wants to present a trophy for the fastest controller in the class.

This will be awarded to the control program that can get from a steady state at 70^{o}C to 100^{o}C in the shortest time with minimal overshoot. If you are wondering what is possible, check this out: *Rev. Sci. Instrum.* **61** (1990), 2214–9.

You may not make any changes to the hardware, but you may use any algorithm you want, as long as you can explain (to me!) how it works.

If you want to be considered for this completely optional component of the course, add a page to your report showing the 70^{o}C to 100^{o}C response with enough time shown before the step to establish that the system started in equilibrium.

The rewards:

- A one-of-a-kind trophy created by Brandon
- Unrestricted bragging rights
- Transient glory