Embedded Control Laboratory 6

Gondola PD

SIMULATED LABORATORY

→ Readings

- Lab Manual: *Chapter 7 - Control Algorithms*

→ Laboratory Goals

General

- Provide preliminary experience in developing a PD control system.
- Perform sensor analysis to produce speed from position measurements.

→ Motivation

Simple proportional control is rarely the solution to systems requiring complex control. This laboratory provides introductory hands-on experience with an embedded implementation of the Proportional-plus-Derivative Control (PD) technique and attempts to provide a physical understanding of the use behind each component.

→ General Lab Description

In previous incarnations of ENGR-2350, there existed a blimp that was flown around in the Armory. These have subsequently been decommissioned due to the cost of upkeep (and helium!); however, the gondola portion was kept and placed on a turntable, as shown in Figure 1.

The gondola consists of three fans: two thrust and one tail. Typically, the thrust fans would be used to control altitude and speed, although in this implementation they may be used to control the heading direction as well. The fans are all controlled by the same speed controller of the drive motor on the car, therefore, their pulsewidth limits are the same. In the lab, you would need to also initialize the fans to neutral for 1 second - this is not necessary in the simulator.

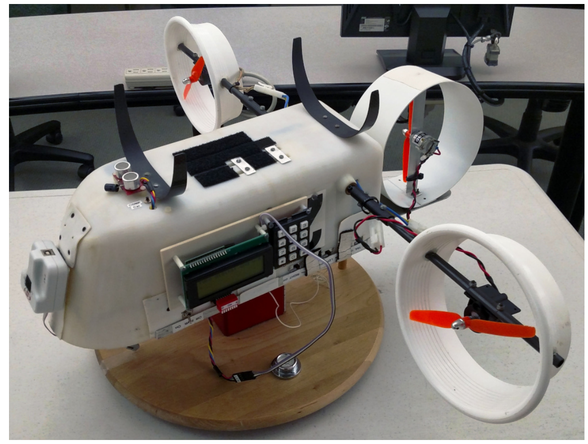

Figure 1: LITEC Gondola

A compass and ranger are both mounted to the gondola (within and on, respectively). The ranger was meant to detect the blimp's height off the ground when attached to the blimp; it will not be used in this incarnation of the laboratory. The compass is used as discussed below.

This laboratory requires the development of control routines to cause the blimp to orient itself in a particular heading using the compass measurements. To do so, the fans must be spun in the appropriate direction to rotate the blimp. Once the gondola gains a rotational velocity, there is little to no resistance attempting to cause the blimp to stop. This is unlike the car, where placing the drive motor in neutral causes the car to stop quickly due to the friction and drivetrain inertia. These physics effectively resist movement, which is referred to generally as system "damping;" that is, the car has high drive damping and the gondola has very low rotational damping.

The goal of this laboratory is to program the gondola with a PD control algorithm to quickly achieve a desired heading with little oscillation. The control must also account for steady state offsets.

The simulator will specify which fans must be used to complete these tasks. It may be assumed that the `XBR0` configuration used since Lab 3 is still employed such that the tail fan is connected to P0.4 (CEX0), the left fan is attached to P0.5 (CEX1), and the right fan is attached to P0.6 (CEX2).

The desired heading, or target heading, is also provided through the simulation via a second I^2C compass-like device with an address of 0x42. This device acts exactly like the laboratory compass except that instead of detecting magnetic north (or a specific object as done within Laboratory 5) it will return the heading that the gondola should try to achieve. The target heading is indicated within the simulation window with a blue line. This target heading will toggle between being stationary (randomly selected) and moving every 10 seconds. The simulation will always start with this value being targeted towards South.

→ Introduction to PD

Considering the proportional control equation used in labs 3-5, where $u(t)$ is the control output, u_0 is the "neutral" output, k_p is the gain constant, and $e(t)$ is the control error:

$$u(t) = u_0 + k_p e(t)\,, \tag{1}$$

it is clear that when the error is large, the output control signal will also be large. Likewise, the control signal will be small when the error is small. This is fine in situations where the system has high damping or the control signal enforces "neutral." This is not true in the case of the gondola: there no forcing on the system when the error is small, other than minimal rotational resistant from the turntable. This will cause the gondola to oscillate about $e(t) = 0$, or cause the gondola to possibly spin uncontrollably if the system is not symmetric. Even if the fans are symmetric, given an initial velocity, it is possible for the gondola to continue to spin and not oscillate.

To fix this, damping, or resistance to movement, must be provided by the control scheme as it is lacking in the physical system. This may be done through the addition of the "Derivative" term, so named as the contribution of this term is proportionally dependent to the time derivative of the error, $e'(t)$, through the derivative gain constant, k_d:

$$u(t) = u_0 + k_p e(t) + k_d e'(t)\,. \tag{2}$$

In this case, when error $e(t) = 0$ or $k_p = 0$ but $e'(t) \neq 0$, there will be a control force applied resisting the velocity[1], assuming the sign of k_d is correct. Selecting proper values for k_p and k_d will allow for these two control inputs to work constructively to (a) push the gondola towards the desired heading and (b) slow the gondola down as it approaches the desired heading. This type of control is known as "PD" control.

→ Embedded Implementation of PD

The full PD equation provided in Equation 2 assumes that the controller is a continuous time device, that is, the control is enforced at every possible value of time, t. This laboratory's implementation is *discrete*, where the control equation is updated at a fixed time period:

$$t_k = t_{k-1} + t_p = kt_p. \tag{3}$$

Here, the current update time is t_k, the previous update time is t_{k-1}, and the period between updates is t_p, where k is an integer noting the current update number. In this format, it is common to use k as shorthand for the time kt_p, such that the discrete error at time kt_p is simply $e(k)$, etc. Using this format, the discrete PD control equation becomes:

$$u(k) = u_0 + k_p e(k) + k_d \frac{e(k) - e(k-1)}{t_p} \tag{4}$$

→ Laboratory Success Conditions

- Successfully implement a PD control routine.
- Select gains for the PD control routine to allow the gondola to settle with little oscillation to a desired heading with error less than ±5°. An error of less than ±10° is acceptable but may be cause for a point deduction.
- The gondola successfully tracks both stationary and moving desired headings.
- The Gondola must be able to stop itself from spinning quickly and then achieve the desired heading. This can be ensured by pressing the "Force Spin" buttons on the top left of the simulation window.

Demonstration of the functionity for this laboratory is provided here: https://youtu.be/Q2Juc5yv2PE

[1]One will notice a discontinuity in the response when the gondola opposite the desired heading. This is due to the sign of the error flipping. We will ignore this effect in this lab.

→ General Lab Guidance

- Unlike Laboratories 4 and 5, this lab does not require multiple functional states.
- Using either your complete lab 4 code or lab 3-3 code is a good start.
- The easiest way to experimentally determine proper gains for this system is to attack them one-by-one. Do not try to find appropriate values for each all at the same time! It is suggested to:
  1. Ignore the simulation required desired heading and pick a constant one.
  2. Set $k_d = 0$, start k_p with a low value.
  3. Iteratively increase the value of k_p until the system goes unstable (spins uncontrollably). In some situations, this may not be possible as the system starts out unstable. For these cases, see the next bullet point.
  4. Reduce k_p back to "stable" value.
  5. Slowly increase k_d until the desired response is achieved (fast settling time).
  6. Once a good response is achieved, it is possible to alternate increases in k_p and k_d to achieve a quicker and more accurate response. This is possible as k_d provides additional stability to the system.
- The fans are not symmetric. In the forward direction, the fans provide about twice as much thrust as the reverse direction. In performing the gain selection steps above, this can be problematic. An easy solution to find a good set of gains is to change your starting point to be on the other side of the desired heading. For example, if starting at $0°$ with a desired heading of $90°$ is always unstable for any k_p when $k_d = 0$, try targeting $270°$ instead. Alternatively, starting with a low value of k_d may work as well.
- The fans do not all point in the same rotational direction. The direction of each fan is marked in the simulation. If using opposite facing fans, the pulsewidth needs to be inverted about neutral while the magnitude of the pulsewidth with respect to neutral should be the same; that is: $\mathrm{fan}_1 - \mathrm{PW}_{\mathrm{NEUT}} = \mathrm{PW}_{\mathrm{NEUT}} - \mathrm{fan}_2$ for opposite facing fans.
- It must be ensured that the compass has a new measurement each time the control equation is evaluated and the spacing between the measurements (40 ms) is consistent, otherwise the derivative gain will provide inconsistent control.
- It is your responsibility to ensure that all goals of the lab are implemented completely.
- Ensure that your RIN is added at the top of your code.

```
#define RIN  xxxxxxxxx  // Must be above #include"C8051_SIM.h"
```