One-dimensional angle recovery potentiometer-servo feedback control system

Lyle Kenneth M. Geraldez*

National Institute of Physics, University of the Philippines - Diliman, Philippines
*Corresponding author: lgeraldez@nip.upd.edu.ph

Abstract

The paper successfully designed, developed, and validated a one-dimensional angle recovering system using a potentiometer as angle sensor and a servo as a feedback controller. The angle sensor was calibrated by sweeping through 15 data values of voltage and angle measured using *ImageJ*. From the calibration curve and fitted linear regression. Arduino transforms serial data input into the corresponding angle from a given reference point. The angle data was passed on to the servo motor rotating it towards the opposite direction equal to angle calculated. The device is only operational in the domain of angles with deviation less than 172.42° due to physical limitations of the used servo motor.

Keywords: angle sensor, feedback control

1 Resistors

Resistors are one of the common electrical components with the main purpose of providing opposition to electric current flow [1]. The resistance R to current given by an inverse proportionality constant to a potential difference V between two terminals can often be calculated conveniently via a proportionality relation $I \propto V$ where with $1/R$ as the proportionality constant. This relationship is called Ohm's law and circuit components following this relation is said to be Ohmic in nature. Hence, a resistor is 'Ohmic' if the current increases linearly with the voltage governed by R as the plot slope. For some resistive materials, however, the relationship is not easily governed by this linearity; current sometimes increases as a function of other variables. For instance, the resistive filament material inside a light bulb has a temperature-dependent resistance. As temperature depends as a function of operating time the light bulb has been passing current, so does the resistance. This gives the filaments its nonlinear resistive property which may not obey the direct linearity provided by Ohm's law. This nonlinear properties are being ubiquitously exploited in many circuit applications.

2 Voltage dividers

As a current-opposing material with direct quantifiable effect to the current flow, resistors can be applied in different ways. For instance, energy dissipated as heat from a restive material can be channeled for heating purposes such in a toaster, microwave, electric stove, and other heating appliance. For instance, in the previously discussed light bulb, the metal filament is heated up in such a way as to produce a white-hot glow from the induced high temperature. One interesting application is to accurately divided voltages as needed in sections of interests in a given electrical circuit system. Connecting resistors simply linearly adds up the resistances. As a consequence of Ohm's law, the current passing through a series of three resistors with total potential V is expressed as

$$I = \frac{V}{R_1 + R_2 + R_3} \tag{1}$$

It follows that the voltage fraction of across a resistor elements must be equal to the resistance fraction of that element. That is, rearranging to isolate V_n,

$$V_n = \sum V \frac{R_n}{\sum R} \tag{2}$$

This is the basis of a voltage divider. By using two resistors with resistances R_1 (connected to a potential V_1 on one-end and V on another) and R_2 (connected to a potential V_2 one one end and V on another) such that V is a circuit point between the two resistors, the voltage V can be directly calculated as

$$V = V_2 + (V_1 - V_2)\frac{R_2}{R_1 + R_2} \tag{3}$$

Given two potentials V_1 and V_2, tuning the values of R_1 and R_2 essentially tunes the value of V (i.e. $V_{V_1,V_2}(R_1, R_2)$). This way, if one designs a mechanism capable of controlling the resistances R_1 and R_2 and connecting this device to two terminals with potentials V_1 and V_2, one can design a mechanism to vary the voltage V at the middle of the two resistances. This is the principle of the *potentiometer*. A rotary potentiometer uses this mechanism by being able to vary the resistances R_1 and R_2 as function of the rotating angle θ. This is done so by using a resistive strip and a wiper. Given a resistive strip of length L, the position of the wiper (controlled by rotating the knob) along the strip strip divides it into two sections with lengths L_1 and L_2 such that $L = L_1 + L_2$. This material is fabricated such that $R_1 = R_1(L_1)$ and $R_2 = R_2(L_2)$. Hence, the voltage of interest V is a function of L_1 and L_2 and since $L = L_1 + L_2$, $L_1(\theta)$, and $L_2(\theta)$ where θ is the rotation angle, we finally have the function$V = V(\theta)$ parametrized by the two potential terminals V_1 and V_2. In other words, varying the knob angle of the potentiometer varies the voltage $V(\theta)$. This is the essence of most potentiometer applications as a voltage divider.

3 Angle sensing and feedback control

For this project, we aim to design a simple feedback system of angle recovery using a potentiometer as an angle sensor and a micro servo to demonstrate a simple feedback control. That is, we aim to restore the angle disturbance in the potentiometer θ by inducing an angle change of $-\theta$ on the servo to demonstrate the recovery feedback.

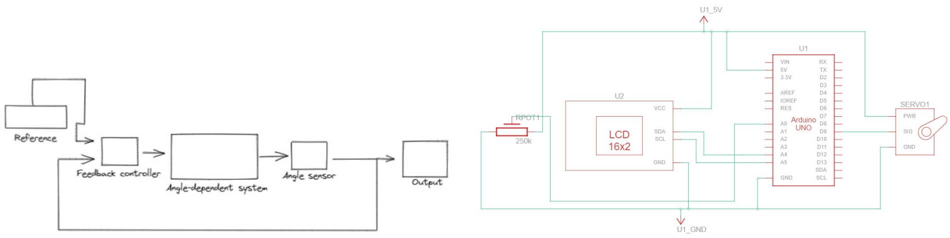

Figure 1: Figure shows control system diagram of angle-recovery feedback. Feedback controller takes input from reference data and angle sensor to readjust angle accordingly. Rotation angle from the potentiometer is proceesed in an Arduino UNO, transformed by its calibration curve, displayed on an LCD, and fed into the servo motor.

Since V is the measurable circuit quantity, we can find the function $\theta(V)$ by finding the calibration curve, modelling with a linear fit, and using the model to transform the detected voltage V on the circuit from the potentiometer to the corresponding rotating angle θ. The calculated θ is then translated into the recovery mechanism on the servo by rotating it by $-\theta$. However, due to resource constraints, there exists no physical connection between the servo and the potentiometer knob to reset or recover the angle and only exists to demonstrate a proof of concept. The constructed system is an attempt to design an auto-stabilization system by recovering the deviation angle from a set reference point. For instance, a drone is designed to be stabilized by preventing yaw, pitch, and roll.

An Arduino UNO was used as the main microcontroller to handle input, process, and output of data. Voltage output $V(\theta)$ from the middle pin of the potentiometer (connected to 5V and ground in each leg terminal) was computed from the serial data input at the analog pin A0. Serial data was converted to voltage by multiplying by a factor of 5/1023. To convert into the potentiometer rotation angle θ_p, the voltage is fed into the linear regression model from the calibration (discussed in the next section) by multipling by (-56.27) and adding 314.25 which is displayed into the LCD screen. Finally, this angle is fed into the servo motor rotating it (from an initial state of 180° to $\theta_s = 180° - \theta_p$. Given an initial angle state of the potentiometer to be θ_{p0} and an initial angle state of the servo to be θ_{s0}, an angle recovery would be where $\Delta\theta_p - \Delta\theta_s = 0$

3.1 Calibration

To find the function $\theta(V)$, we aim to plot the calibration curve. This is done so by turning the knob by some small angle, measuring the angle turned, and measuring the voltage on the middle pin of the potentiometer. To do so, a video is recorded showing both angle rotation and the LCD display of the corresponding voltage. The video is analyzed using *ImageJ* to measure the rotation angle at each discrete rotation. The data samples were recorded and analyzed accordingly to construct a linear fit model.

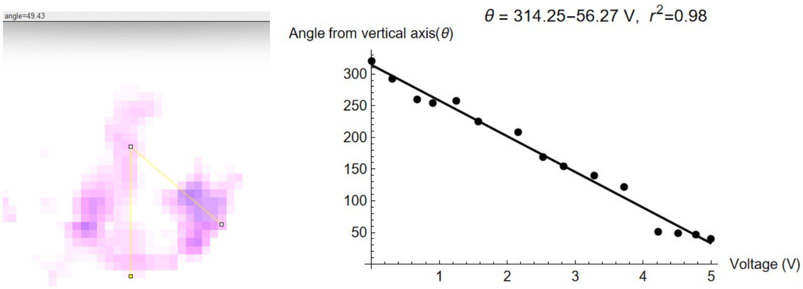

Figure 2: Left image shows angle extraction from each potentiometer state images by measuring angle between the horizontal axis and the written point mark. Linear regression was fit into the angle-voltage data with $r^2 = 0.98$

The video was separated into individual image frames, applied image corrections for convenience of angle measurement, and saved for replication purposes. A point mark was written on the potentiometer as a reference for rotation angle measurement. Linear regression fitting of the 15 data points gives a slope of -56.27 and an intercept of 314.25; these values are used to transform the data input (calculated from the serial data from the potentiometer) to find the respective angle value from a given voltage state $V(\theta)$.

4 Final product

The final product shows a prototype of the angle recovery system composed of a mechanically uncoupled potentiometer as angle sensor and servo motor as a feedback controller. The product aims to demonstrate that by rotating the potentiometer from a set reference initial state (in this case, from the right-most rotary setting), the servo motor responds by rotating towards the opposite direction by an angle equal to the displaced angle of the potentiometer state. The following mechanism can be seen in this demonstration: https://drive.google.com/drive/folders/1aBpl7dOehIvb93_pDI7_Yy7OQv1bbDGr

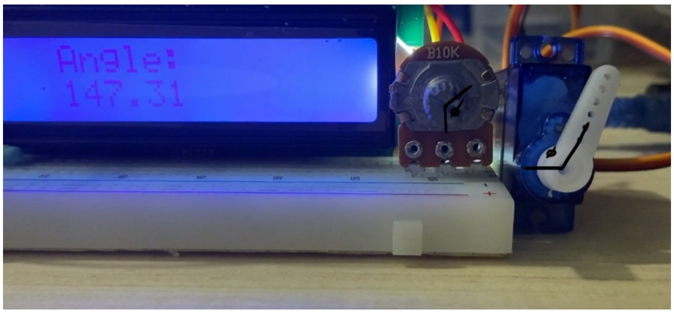

Figure 3: The final product is a servo motor reacting to the potentiometer rotation angle which is displayed on an LCD screen.

The Arduino code is uploaded at https://github.com/schwarzschlyle/electronics-and-instrumentation. The working mechanism of the code is as follows. The sensor pin and value, potentiometer voltage, corresponding angle, and linear regression quantities (slope and intercept) were initialized. Inside the void setup() function, the servo motor was attached and preset the blade at 180°. LCD display was also activated and cleared up. The void loop() function starts by resetting the servo to 180°, waiting one second, printing the potentiometer angle on the LCD, waiting another second, and inducing a feedback response to the servo by moving it by an amount equal to the change in potentiometer angle, waiting for 100 ms, and resetting the servo blade to 180° for another loop.

```
void setup()
{ servo.attach(9); // First, we tell the system what pin was the servo motor attached to
  servo.write(180);  // The servo was set to an initial angle state at 180 degrees
  lcd.begin(); // LCD was initialized as follows
  lcd.backlight();
```

```
  lcd.clear(); }
void loop()
{ servo.write(180); // Servo is reset to 180 degrees after each loop
  delay(1000);
  lcd.clear();
  lcd.setCursor(2,0);
  lcd.print("Angle: ");
  lcd.setCursor(2,1);
  angle = ((analogRead(sensorPin)* 5.0/1023)*slope) + intercept;
  lcd.print(angle); // Then, the angle of the potentiometer was displayed on the LCD
  delay(1000);
  servo.write(180-(angle-33.45)); // Feedback control instructions were sent to the servo
  delay(100); }
}
```

The project used contents from https://bit.ly/3iuzErK as a guide to setup LCD display using the LCD library from https://github.com/fdebrabander/Arduino-LiquidCrystal-I2C-library. Contents from https://docs.arduino.cc/learn/electronics/servo-motors was also used as a guide to setup the servo motor using the Servo library from https://github.com/arduino-libraries/Servo

5 Recovered angle validation

There are several limitations to the final product as can be observed in the following validation of input and output states. Both potentiometer has a maximum angle threshold being not able to fully realize a full 360° rotation. For the potentiometer, the angle domain is $[26.28°, 323.88°]$ while for the servo, the angle domain is $[7.18°, 172.42°]$

Calibration Angle	θ_p	θ_s	$\Delta\theta_p$	$\Delta\theta_s$	Recovery Error
33.45	26.28	172.42	0	0	0
59.58	53.85	145.83	27.57	-26.64	0.93
104.96	93.58	103.44	67.30	-69.03	-1.73
147.31	138.88	66.61	112.60	-105.86	6.74
186.91	180.00	29.65	153.72	-142.82	10.9
224.04	212.22	7.27	185.94	-165.2	20.74
271.9	257.56	7.32	231.28	-165.15	66.13
314.25	323.88	7.18	297.60	-165.29	132.31

The recovery error of the feedback system is the measured amount of angle offset deviation of the servo response from the measured input rotation on the shaft. Note that the servo(write) function is only capable of taking in values in the domain $[0, 180]$. Hence, at some point, the function servo.write(180-(angle-33.45)) takes a negative value. In such case, the servo could still only swipe to the 0° angle state. This can be observed from the ramping up of error as $\Delta\theta_p$ exceeds 180°. Also, observe that the calibration angle and the measured angle contains a significant mismatch. This may be due to mismatch in orientation in video recording the setup between calibration and validation.

6 Conclusion

Using a potentiometer as an angle-sensing device and a servo motor as a feedback controller, the paper successfully built and demonstrated a working prototype of a one-dimensional angle-recovering system that tries to reorient the system into a set reference angle. However, without a designed mechanical coupling of the potentiometer and the servo motor, the final product is only a proof-of-concept work and does not show the desired feedback control. Moreover, the system is only operational on angles less than 172.42° - the upper bound rotational degree of freedom provided by the servo motor.

References

[1] N. Storey, *Electronics: a systems approach* (Pearson Education, 2006).

7 Codes and Raw Data

All files regarding the paper can be found here: https://bit.ly/3VGgyNF

8 Technical correctness

The paper met the objective of contructing an angle sensor by converting voltage data output from the potentiometer and transforming it into the corresponding angle using the calibration curve. It shows an understanding of how rotation sensor can be implemented to different kinds of systems. The results was verified in the validation section of the paper; hence, it can be considered as both correct and complete. Codes were also filled with sufficient comments.

Self-score: 35/35

9 Presentation quality

The circuit diagrams and codes were presented in the final product section and a separate complete Github repository was also attached for validation and/or reflection. Figures contain descriptive captions to be able to stand alone without the referring texts. Plot of the calibration curve were properly labelled. Text is also clear and graphs are presented such that the linear relation is clearly seen.

Self-score: 35/35

10 Self reflection

The validity of the results was assessed on the validation section of the paper - discussing the limitations of the current developed prototype. For each aspect of the self-reflection with the corresponding self-score, compliance to the criteria for each category was briefly discussed justifying the set self-score.

Self-score: 30/30

11 Initiative

As an initiative, the paper extends the original objective of constructing an angle sensor by integrating it as a section of an angle-recovering feedback controller. Hence, this corresponds to a 10-point bonus.

Self-score: 10/30