Biomimicry: The Goliath Project

Robert Ensley

Department of Physics & Astronomy, University of Northern Colorado

Introduction

In studying the locomotion of a spider, it was observed that they perform pulling motions with the front four legs, and pushing motions with the back four. The front two legs perform similarly to those of a human with one leg maintaining contact while the other moves into position before contact. In general, a spider moves four of its legs at once with each of those four being a part of a pair, with each pair responsible for a different aspect of the spider's motion. To better study such behavior, I built Goliath, a robot which is designed to mimic the walking pattern of an arachnid using micro servo motors, several microcontrollers, and a 3D printed frame.

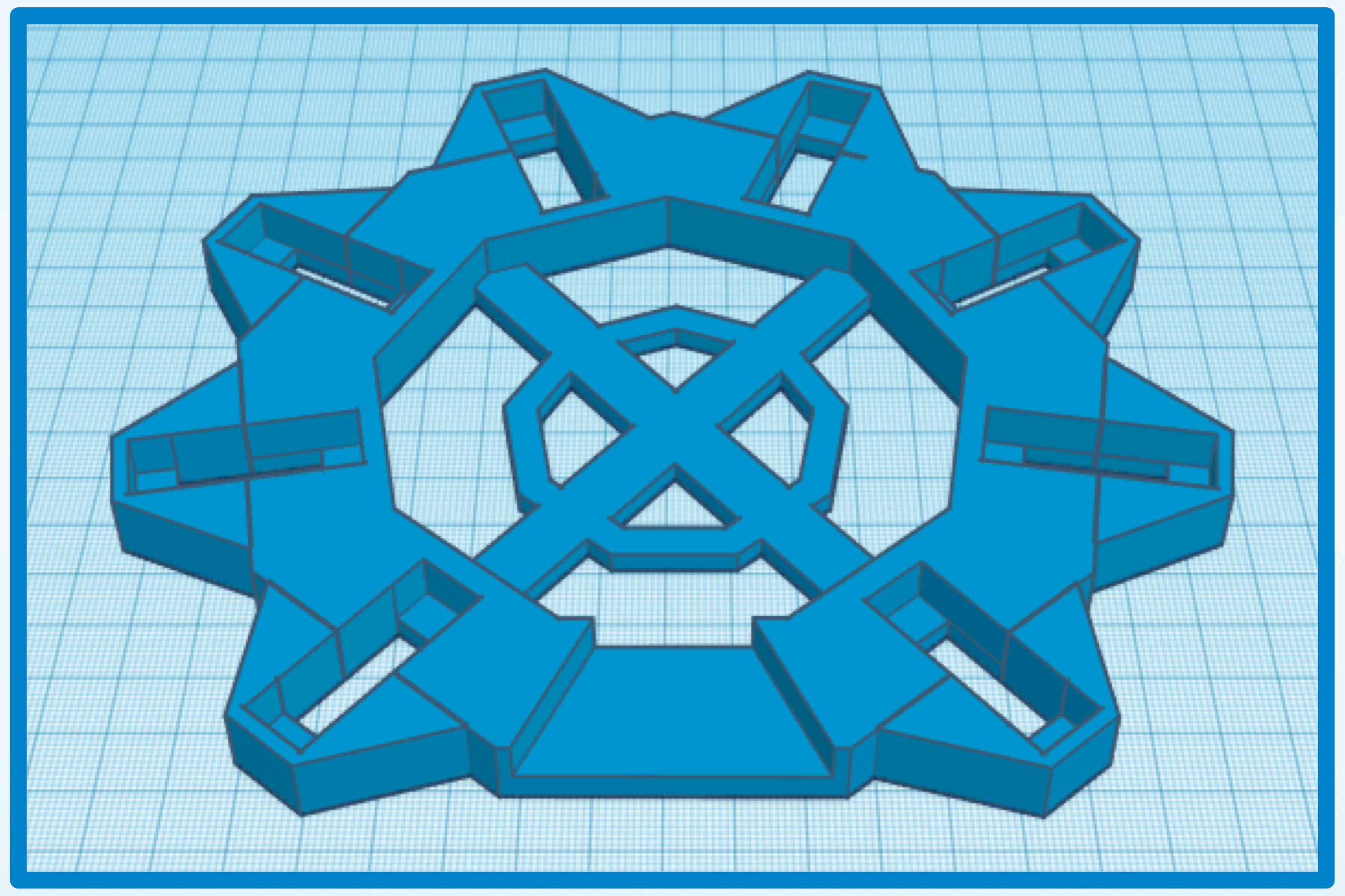


Figure 1 – This is the design for Goliath's body, or central mechanical hub, to which the legs are attached. It was 3D printed with ABS plastic.

Design

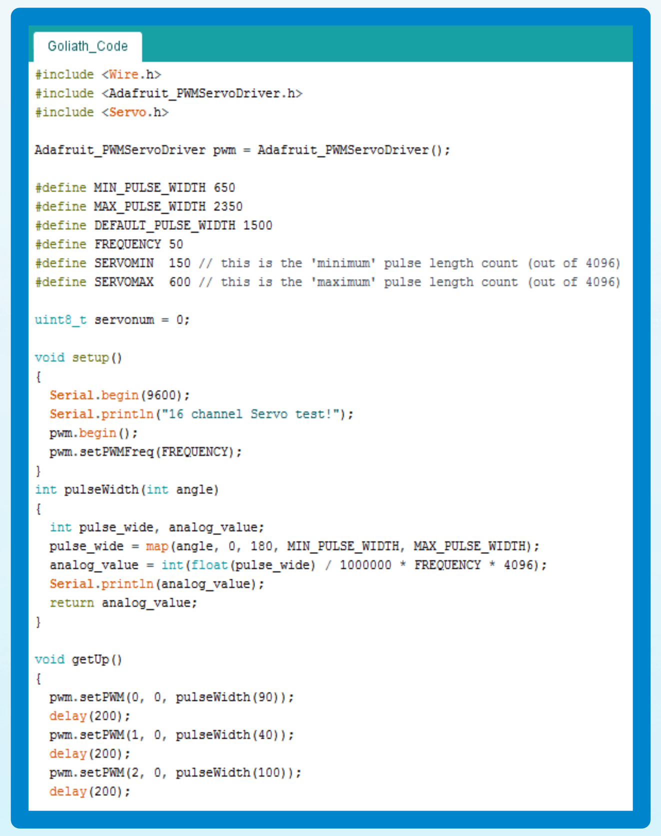
Studying the anatomy of a common wolf spider lead me to the current design for Goliath's center frame, or hub, to which the legs are attached. The front legs of the spider have a wider breadth than the rear legs. To achieve the desired front gap, the shape of the center frame was changed from an Octagon to a Nonagon with leg joints on all sides except for one (Figure 1). The center frame and all appendages were designed using the TinkerCAD online software and printed with a 3D printer using ABS plastic. Each leg consists of three parts and is controlled by three servo motors which provide a limited range of motion horizontally and vertically. Each motor fits statically in a leg joint and its rotating arm is hot glued to the subsequent leg part. The center frame was adapted into a ring shape to reduce total weight and allow for the microcontrollers to sit inside the frame rather than rest on top.

Code

All of algorithms for running Goliath were written using Arduino software and was uploaded into an Arduino Uno microcontroller. The Uno served as the primary electronic hub for two Adafruit PCA9685 servo controllers, which were able to communicate individual commands to each servo motor controlling a leg. Using the Adafruit code library, algorithms were written for each major portion of Goliath's movement then called upon by a primary line of code that controlled the total movement (Figure 2). The leg movements were coded in such a way to best mimic the motion of a typical spider as alluded to in the introduction.

In order to best mimic the walking pattern, the legs would move in groups of four. Legs eight, two, six, and four would move in one group with legs eight and two committed to pulling motions and legs six and four committed to pushing motions (Figure 3). The same happens for legs one, seven, three, and five with one and seven committed to pulling motions. This, of course, is strictly for walking forward, as walking backwards would cause legs three through six to perform the pulling motions instead of pushing.

Furthermore, legs one, eight, four and five perform mainly crawling motions while legs two, three, seven, and six perform sweeping motions side to side.



```
Goliath_Code
#include <Wire.h>
#include <Adafruit_PWMServoDriver.h>
#include <Servo.h>

Adafruit_PWMServoDriver pwm = Adafruit_PWMServoDriver();

#define MIN_PULSE_WIDTH 650
#define MAX_PULSE_WIDTH 2350
#define DEFAULT_PULSE_WIDTH 1500
#define FREQUENCY 50
#define SERVOMIN  150 // this is the 'minimum' pulse length count (out of 4096)
#define SERVOMAX  600 // this is the 'maximum' pulse length count (out of 4096)

uint8_t servonum = 0;

void setup()
{
  Serial.begin(9600);
  Serial.println("16 channel Servo test!");
  pwm.begin();
  pwm.setPWMFreq(FREQUENCY);
}
int pulseWidth(int angle)
{
  int pulse_wide, analog_value;
  pulse_wide = map(angle, 0, 180, MIN_PULSE_WIDTH, MAX_PULSE_WIDTH);
  analog_value = int(float(pulse_wide) / 1000000 * FREQUENCY * 4096);
  Serial.println(analog_value);
  return analog_value;
}

void getUp()
{
  pwm.setPWM(0, 0, pulseWidth(90));
  delay(200);
  pwm.setPWM(1, 0, pulseWidth(40));
  delay(200);
  pwm.setPWM(2, 0, pulseWidth(100));
  delay(200);
```

Figure 2 – An example of the code used for Goliath's motion

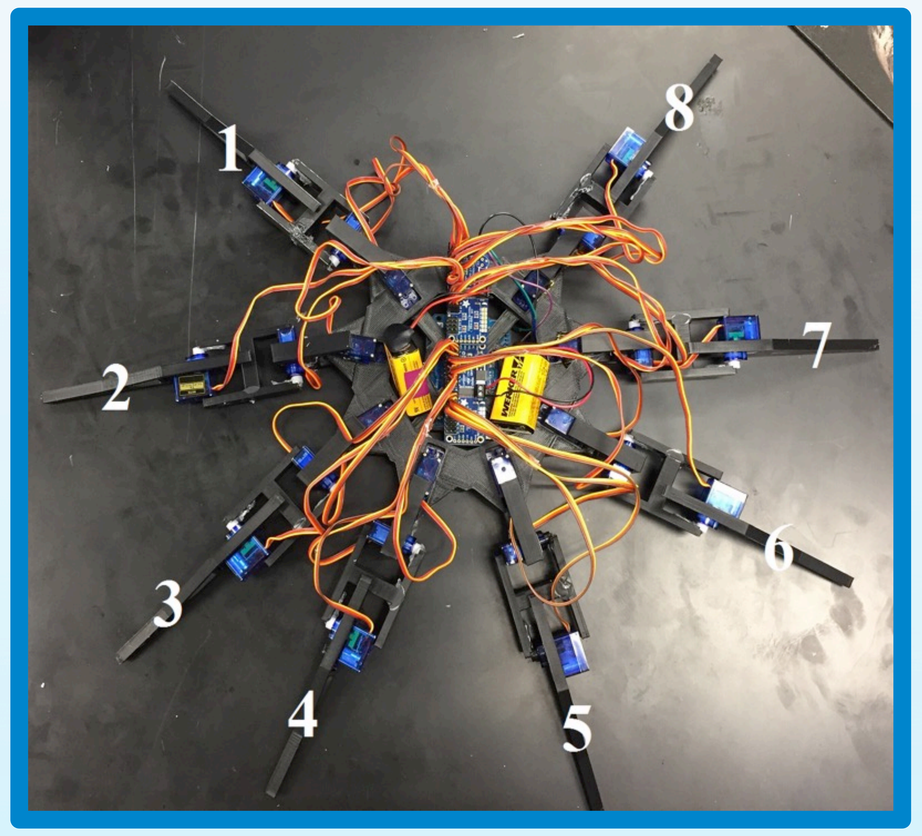


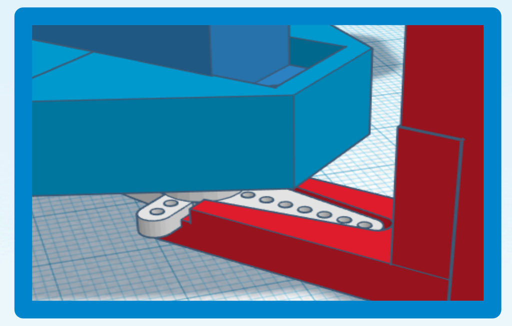
Figure 3 – The numbering scheme for Goliath's legs

Conclusion

Fortunately, the potential for Goliath goes far beyond what has been researched and reported. Now that the foundations have been laid for this project, improvements can be made and more nuanced ideas can be applied. For example, adding more adaptive code and touch and ultrasonic sensors would allow Goliath to reach a near-autonomous state, and adjustments to its mechanical design would only increase its stability.

Figure 4 – This drawing shows each component of one leg.

a.



b.

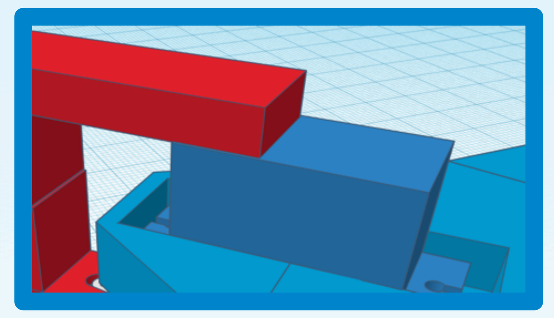


Figure 5 – Close-up of first (red) leg joint shown in Figure 4: a. One can see the white arm of the servo motor attached to the red leg joint. The blue mass is the central hub of Goliath's body b. The top of the leg joint is seen to float over the central hub.

Results and Discussion

One of the primary flaws with Goliath concerns the design for the first leg joint, seen in red in Figure 4. The servo motor that is connected to the outer rim of the central hub is attached to the first leg joint and is the sole point of support for the entire leg. So, the weight of the leg is only supported by one point of contact (Figure 5a), while the top of the first leg joint merely floats over the top of the central hub (Figure 5b). This situation causes strain and unnecessary shaking has been observed as the servo motor is driven.

Additionally, due to the nature of the software, commands must be run in order of appearance and cannot be executed simultaneously. This is a problem since the legs have to move in groups of four to properly mimic the motion of a spider. This was remedied in part by moving one joint at a time for each leg in these groups. For example (see Figure 3), if legs one, three, five, and seven were moving, the code would call for the first joint for leg one to move, then the first joint for leg three, and so on. As the first leg joint for leg seven finished moving, the second joint for leg one is called on to move and the cycle continues.

Acknowledgments

I'd like to acknowledge Dr. Galovich and Dr. Semak of UNC Physics and Astronomy. Also, I am grateful to Chris Koehler and Bernadette Garcia of the Colorado Space Grant Consortium/ NASA for their continued support.

Goliath was designed with the use of the online software, TinkerCAD, and programmed using Arduino software. Also, hardware from Adafruit Industries was used.