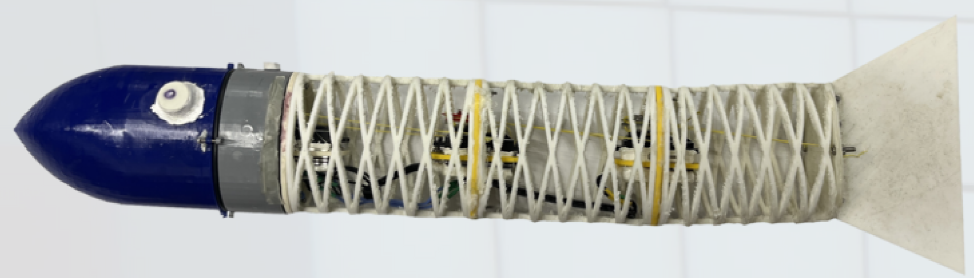
Soft Robotic Eel

Christopher Hunt (RBE), Dexter Stark (RBE), Pranav Jain (RBE), Natalie Essig (RBE)
Advisors: Cagdas Onal (RBE), Robin Hall (RBE)

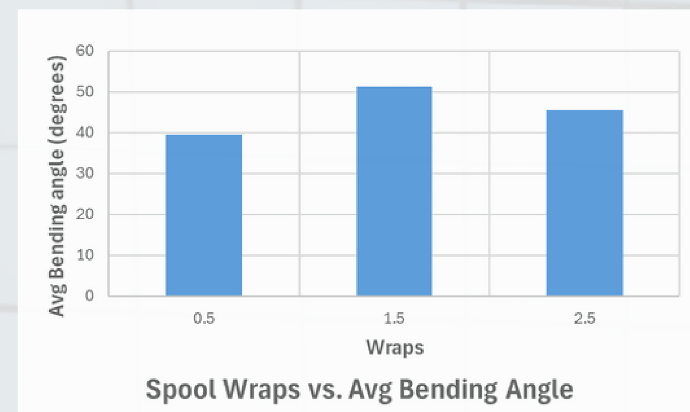
Overview and Objectives

- Expanded on previous project
- Increase movement speed
- Propulsion through anguilliform movement
- Allow for future scalability
- Modular design



Testing

- Bending angle
- Speed
- Material
- Water tightness



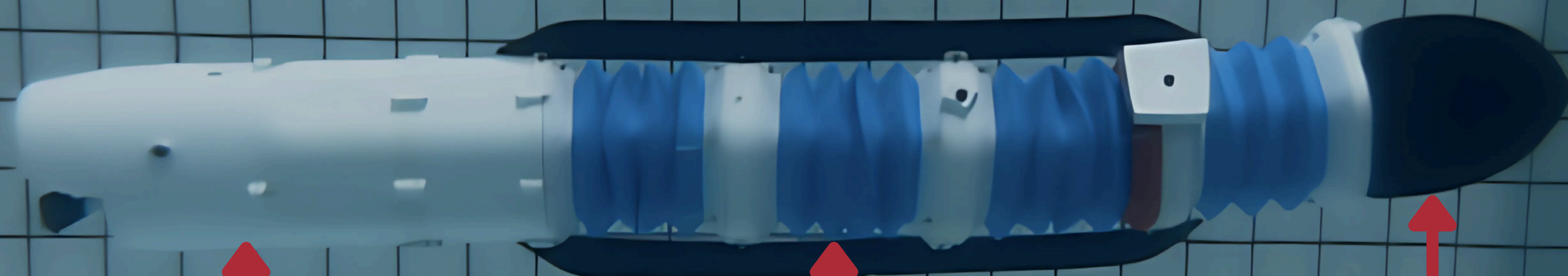
Software

- Bend angle is based on sine wave function
- Amplitude: 120°
- Frequency: 1 Hz
- Phase Shift: $2\pi/3$

```
// Most important function
// Frequency is the oscillation frequency (how fast the servos move back and forth)
// Phase shift controls coordination between servos
// Amplitude is how much each servo moves from center (one for each servo)
 void swim(float frequency, float phaseShift, float amplitude1, float amplitude2, float amplitude3){

  // 2 * PI converts frequency from Hz to radians per second
  float omega = frequency*2*PI;   //rad/second

  // Calculates the angle for each servo
  // angle = center + amplitude * sin(omega*time + phase shift)
  angle1 = center1 + amplitude1*sin(omega*elaspedTime);
  angle2 = center2 + amplitude2*sin(omega*elaspedTime - phaseShift);
  angle3 = center3 + amplitude3*sin(omega*elaspedTime - 2*phaseShift);

  angle1 = shortestPath(servo1.getAngle(), angle1);
  angle2 = shortestPath(servo2.getAngle(), angle2);
  angle3 = shortestPath(servo3.getAngle(), angle3);

  // Write angles to servos
  servo1.rotateTo(angle1);
  servo2.rotateTo(angle2);
  servo3.rotateTo(angle3);
 }

float shortestPath(float currentAngle, float targetAngle) {
    float diff = targetAngle - currentAngle;
    if (diff > 180) return targetAngle - 360;  // Adjust to take the shorter path
    if (diff < -180) return targetAngle + 360;
    return targetAngle;
}
```

Challenges

- Accordion Segments
  - Difficult to print and repair
  - Prone to tearing
- Motors
  - Limited torque output
  - Constrained by internal space
- Electronics
  - Small watertight compartment
  - Tight wiring layout
  - Limited space for battery



Head

- Houses battery and control board
- All components are 3D printed from PLA to protect delicate parts during motion
- The battery compartment is also designed to house future sensors
- Acrylic conformal coating was added to all compartment surfaces

Waterproofing

- Protecting the ESP32, PCB and servos from water was a top priority
- Marine-grade epoxy was applied to servos and wire connectors
- The ESP32 is enclosed in a watertight compartment within the head module

Modules

- Houses 360° servo motor and spool of fishing line
- Bends by rotating spool to pull on the passive end
- Each module can bend up to ~51° in either direction from center

Tail

- Final module of the eel
- Passive design
- Uses existing momentum for forward propulsion
- Tail end is constructed from NinjaFlex, a more rigid yet flexible TPE