Olin College of Engineering
ENGX 2360 Introduction to Thermal-Fluid Systems
Professor: Emily Tow

Final Project: Analytical Kite Design

Tuesday 12th December, 2023

Mira Chew, Camden Droz, & Nina Mindel

Contents

Introduction

Our group wanted to design a kite that was funny and had an attractive design to look at from the ground, so we decided to design our project around the fluid analysis of a fish-kite design to optimize the position of the kite's bridle point about the kite itself, and ultimately find the angle of attack that our kite should fly the most stable at, so that spectators could see our humorous kite design for as long as the kite was flying.

Design and Goals

Our group wanted to create a kite design that embodied the phrase "Fish out of water" and would make people laugh when seeing it fly in the air, so for our kite design, we went with a traditional fish shape, somewhat taking inspiration from the side profile of a carp. We chose a tail-to-head kite length of 27 inches so that our large kite design could be seen from the ground more easily than a smaller design, and constructed the kite from paper, wooden dowels as crossbeams, white yarn as the bridle cords, and yellow tape to smooth the outline of the kite shape and prevent the kite from ripping at the edges during flight. These design elements can be seen in Figure 1.

To approach our unorthodox intended design for the kite, we first created a piecewise function of the curve above the spine of the kite in Matlab that we would use for all analysis calculations of our kite. The piecewise defining the shape of the kite can be seen in the following equation:

$$f(x) = \begin{cases} 0.15(x-2.89)^2 - 2.137 & \text{if } -0.884476 \leq x < 2.89 \\ -1.07(x-4)^2 - 0.819 & \text{if } 2.89 \leq x < 4 \\ 0.026(x-16.384)^2 - 4.806 & \text{if } 4 \leq x < 16.384 \\ 0.085(x-16.384)^2 - 4.806 & \text{if } 16.384 \leq x < 22.919 \\ -0.669979(26-x)^{1/2} & \text{if } 22.919 \leq x \leq 26 \end{cases} \tag{1}$$

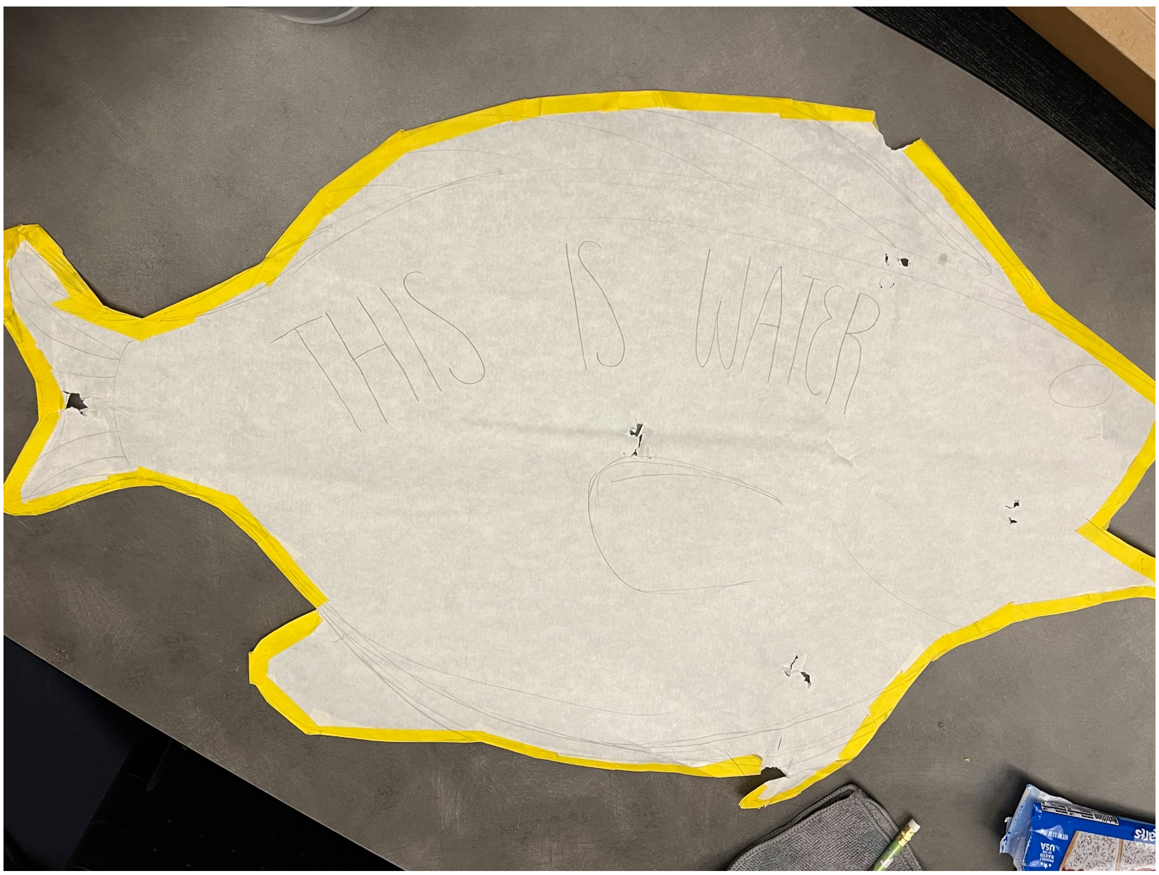

Figure 1: The final design of our kite.

While the design that is created using the piecewise function and its flipped curve is not fully accurate to the exact design of our kite, the extra area not taken into account in the piecewise function (such as the fish fins and lips) will have a negligible effect on the fluid interactions of our kite, so we can still use the piecewise to accurately analyze our kite design. The design created when the piecewise function is flipped and plotted against itself can be seen in Figure 2.

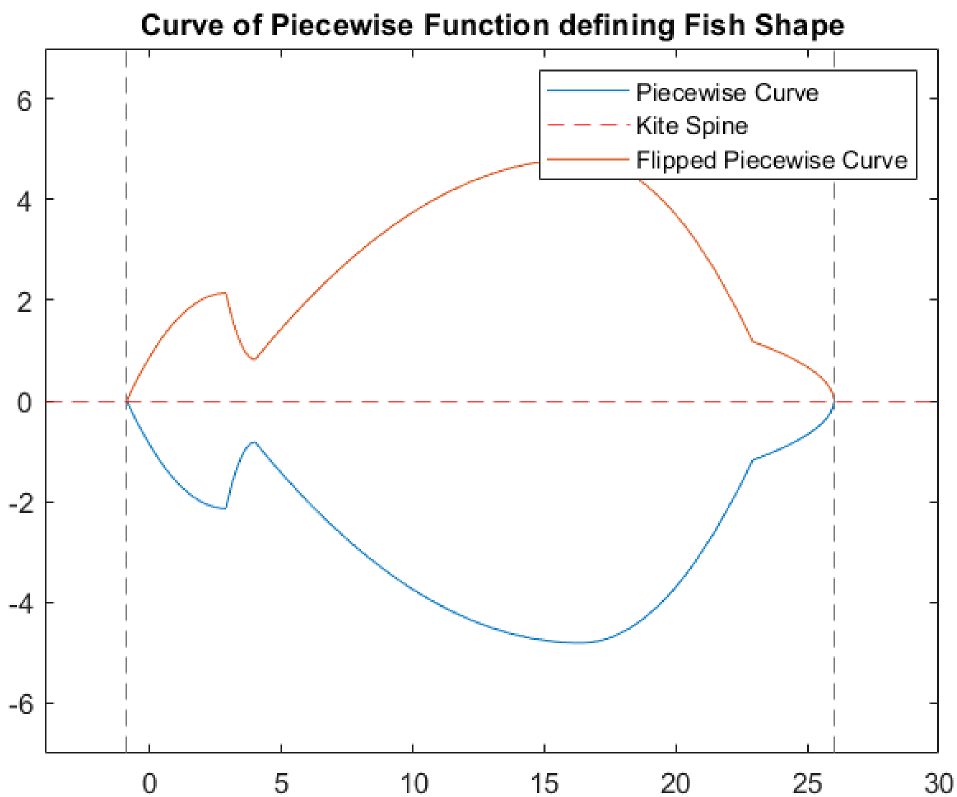

Figure 2: The fish design represented by the piecewise function of the curve below the x-axis and the same curve flipped.

We created the piecewise of only half of the kite so that during later analysis of the function, when finding the area of the entire kite we would only need to find the integral of this piecewise, and then multiply the result by 2 to find the total area of the kite. Based on our calculations, our kite would fly stable, and the net forces and torques acting on our kite's system would be at equilibrium, when the bridle point (BP, the point where all of the kite's bridle chords connected) was placed at approximately 5.28 inches away from the face of the kite, leading to an angle of attack (angle between the spine of the kite and the bridle point, starting at the tail of the kite) of approximately 13.8 degrees.

Theory of Operation

To find the bridle point position and angle of attack, we first needed to find the Center of Pressure (CP) and Center of Gravity (CG). The CP is used to analyze the pressure forces along large bodies, treating the pressure forces spread along the entire body as acting only at this point. The same holds for the CG, used to model the gravitational forces acting on a body at only one point. This is done to simplify any further analysis of a body, especially a kite, as when the forces and torques acting on these points are at equilibrium, the kite is in stable flight.

Furthermore, there are two important assumptions that we made to simplify calculations for the center of pressure position, as well as for the net torque acting on the bridle point. The first assumption is that the aerodynamic center of the kite is a close approximation for the center of pressure for our kite, and that the then assumed center of pressure sits 1/4 from the leading edge of the kite. This assumption is made because the true center of pressure of a kite varies with the angle of attack, so when calculating the center of pressure of our kite, we can use the aerodynamic center of different sections of the kite to approximate the center of pressure of the entire kite.

The second assumption is that there is enough tension in the bridle chords to be considered rigid struts, which is made so that the tension acting on the bridle point is always considered to be negative in both the x and y directions, and therefore all tension acting on the kite is through the bridle point, and tension can be removed from the net torque equation for the kite system, as the bridle point is the point of rotation for the system. A free-body diagram of our kite with these assumptions in mind can be seen in Figure 3.

It's important to note that for the rest of our calculations, the x and y axes used will be the ones shown in our free-body diagram, meaning the x-axis runs along the spine of the kite, and the y-axis runs perpendicular to the face plane of the kite.

To then calculate the CP x-position of the entire kite, we used the process outlined

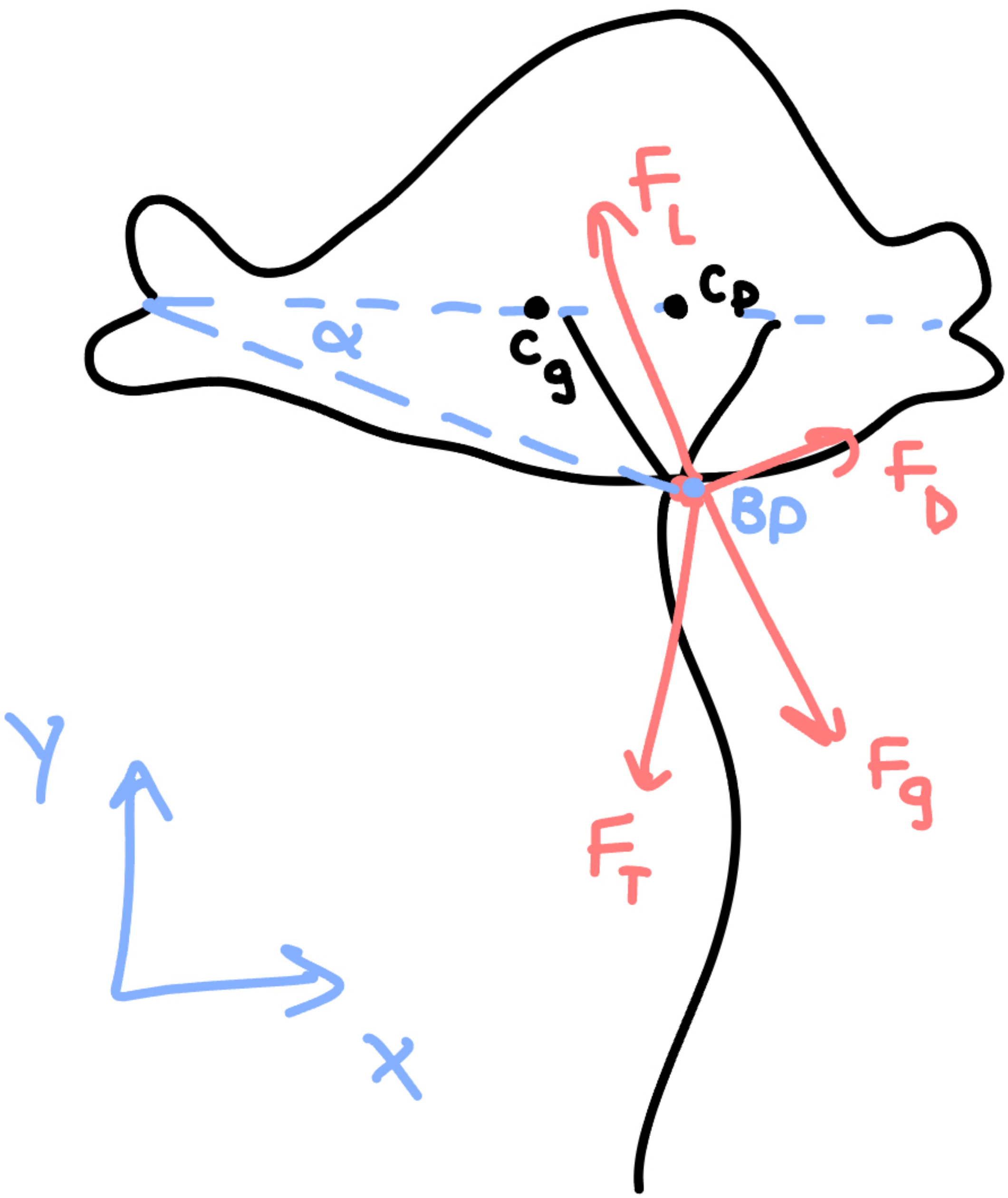

Figure 3: The Free-Body Diagram of our kite, depicting the x and y components of the pressure forces acting on our kite, as well as the gravitational and tension forces acting on the kite.

by the NASA Glenn Research Center: by taking the area-weighted Riemann Sum of the kite surface and finding the CP of each Riemann segment. The Riemann Sum equation for the x-position of the CP of the entire kite can be rewritten in integral form as:

$$x_{CP} = \frac{\sum_{n=1}^{N} A_n d_n}{\sum_{n=1}^{N} A_n} \tag{2}$$

$$x_{CP} = \frac{\sum_{n=1}^{N} (L_n \Delta x) d_n}{\sum_{n=1}^{N} L_n \Delta x} \tag{3}$$

$$x_{CP} = \frac{\int_a^b L(x) D(x)\, dx}{\int_a^b L(x)\, dx} \tag{4}$$

where L(x) is the function that determines the length of each Riemann rectangular segment, dx is the differential width of the Riemann segment, and D(x) is the function that gives the distance between the reference line (y-axis) and the differential CP (aeronautical center) of each Riemann segment, which is 1/4 the total kite length the from the head of the kite. Because the CP exists along the spine of our kite, and therefore its y-position is 0, the position of the CP is (x_{CP},0), where x_{CP} is the result of the previous equation.

To then find the x-position of the CG, we simply use the following equation:

$$x_{CG} = \frac{\int_0^L y(x) x\, dx}{\int_0^L y(x)\, dx} \tag{5}$$

where y(x) is the inverse integral of the piecewise function of our kite, and L is the total length of our kite. Our calculated positions for CP and CG are (21.3, 0) and (13.3, 0) (in inches) respectively.

Once we had the Center of Pressure and Center of Gravity, we then calculated the total Pressure Force (F_P) acting on the kite. The net Pressure Vector can be written as the sum of the drag and lift force vectors, assuming that the drag vector points horizontally and the lift vector points vertically. F_P can then be written as:

$$F_P = F_D\hat{x} + F_L\hat{y} \tag{6}$$

Using the equations for the magnitude of Lift Force (F_L) and Drag Force (F_D) provided by the NASA Glenn Research Center:

$$F_L = \frac{\pi\alpha p v^2 A}{1 + \frac{2A\alpha}{W^2}} \tag{7}$$

$$F_D = \frac{p v^2 A sin(\alpha)}{2}(1.28\alpha + \frac{A(\frac{2\pi A}{1+\frac{2\pi A\alpha}{\pi W^2}})^2}{0.7\pi W^2}) \tag{8}$$

where α is the angle of attack in radians, p is air fluid density, v is air fluid velocity, A is the full surface area of the kite, and W is the total width of the kite. The equation for F_L and F_D both include α (angle of attack) and therefore are both functions of angle of attack. From this point, we symbolically solved for net Torque, as eventually, we did a parameter sweep of the y-position for the bridle point (which is correlated to the angle of attack), to get the net Torque at each bridle point y-position.

To find net Torque, we used the following symbolic equation:

$$\Sigma\tau = (\vec{r}_{CP} \times \vec{F}_P) + (\vec{r}_{CG} \times \vec{F}_G) \tag{9}$$

where $\vec{r}_{CP}$ and $\vec{r}_{CG}$ are the spatial vectors that point to the CP and CG respectively. The equation does not include Tension Force (F_T) because the bridle chords are considered rigid struts, meaning all tension force is acting on the bridle point, and since the BP is the axis of rotation for the kite, there is no torque caused by tension force.

Once we had the equation for net Torque, we included this equation and the two for F_L and F_D in our parameter sweep of bridle point y-position to find the net Torque for each bridle point y-position tested. Once we had each net Torque, we wanted to only use the bridle point y-positions that would produce a kite that would fly without constantly moving around and would stay somewhat stable moving forward. To verify this, we used the following equations to calculate the x and y components of the F_T acting on the bridle point:

$$\Sigma F_{\hat{x}} = F_D + F_T\hat{x} \tag{10}$$

$$F_{T\hat{x}} = -F_D \tag{11}$$

$$\Sigma F_{\hat{y}} = F_L + F_G + F_T\hat{y} \tag{12}$$

$$F_{T\hat{y}} = -(F_L + F_G) \tag{13}$$

and once we had the y and x components of tension for each bridle point y-position, we removed y-positions where a component was positive, as then tension was not pulling away from the forces in that component direction, but rather pushing towards them. This quality is illustrated in Figure 4.

Finally, we had the bridle point y-positions with two negative tension components and their accompanying net Torque, and when graphed against one another, we could calculate the bridle point y-position that minimized net Torque. The plot of net Torque vs. BP y-position can be seen in Figure 5.

We then converted the y-position of the bridle point to angle of attack using the following equation to plot the angle of attack vs. the net Torque on the bridle point:

$$\alpha = tan^{-}1(\frac{BP_y}{BP_x}) \tag{14}$$

where BP_y and BP_x are the bridle point's y and x positions respectively. From that calculation, we plotted the angle of attack vs. the net Torque acting on the bridle point, seen in Figure 6.

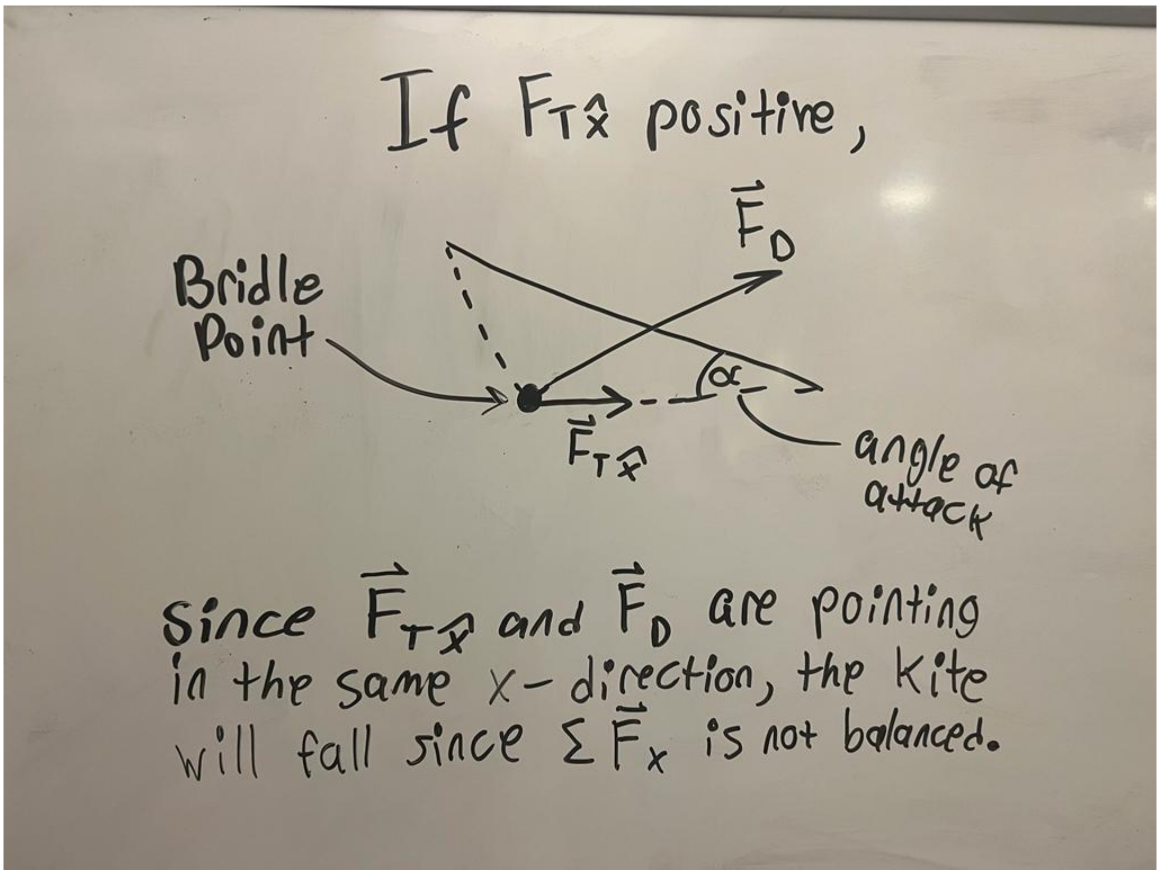

Figure 4: An example of the x-component of the Tension Force vector pointing in the same direction as drag. Because they are pointing in the same direction, the net Force in the x-direction is not balanced and therefore the kite is not stable.

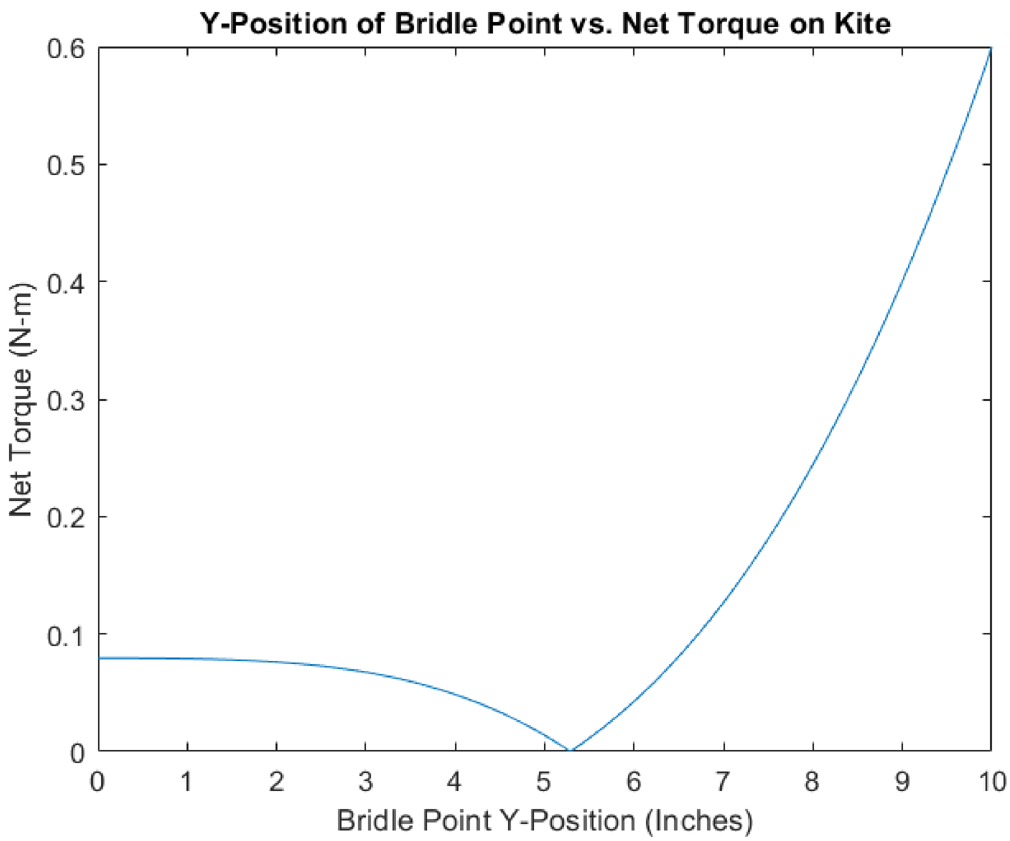

Figure 5: The graph of Y-position of the Bridle Point vs. the Net Torque acting on the kite at the bridle point.

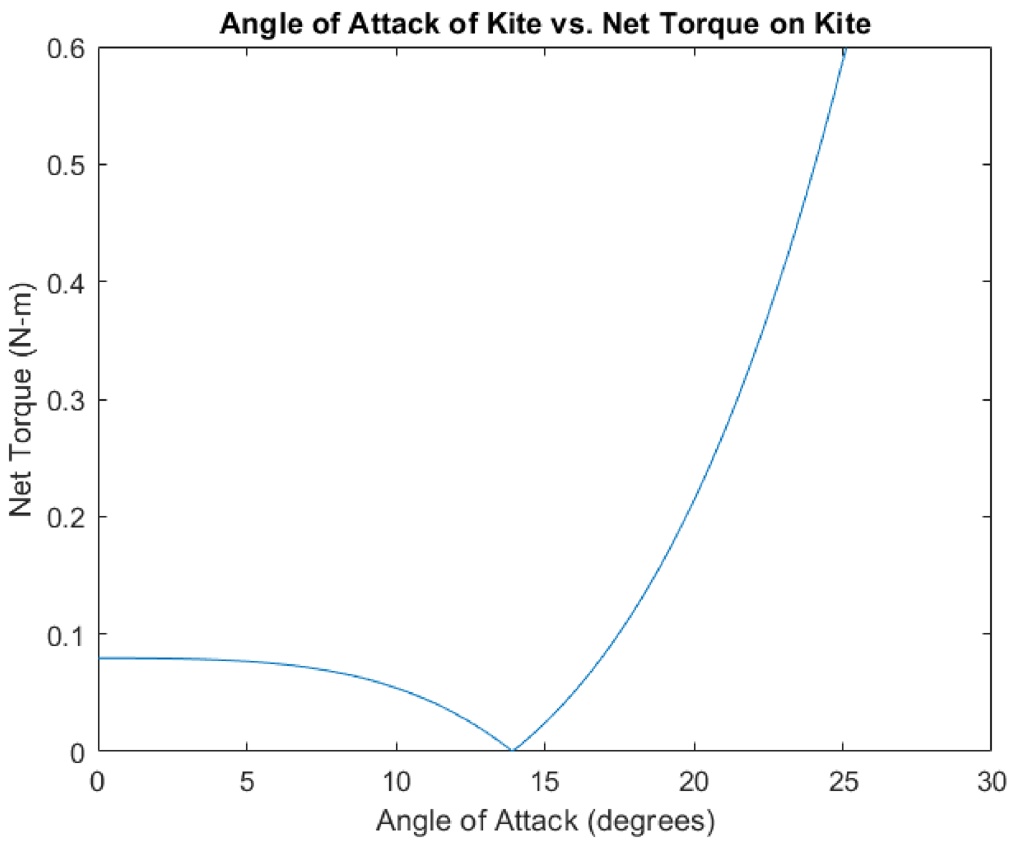

Figure 6: The graph of Angle of Attack of the Kite vs. the Net Torque acting on the kite at the bridle point.

From all of our calculations, we found that a bridle point y-position of approximately 5.27 inches, which equates to an angle of attack of approximately 13.8 degrees, created the closest net Torque on the bridle point to zero, being $5.115 * 10^{-}4$ Nm.

Results and Reflection

Initial Flying Prototype

Before calculating the ideal shape and design of our kite, we designed a fish-shaped kite to see how it would perform. Knowing the aeronautical center is about a quarter of the way down from the head of our kite, we estimated where to center the bridal point. We attached a string and tried to fly the kite by running with it dragging behind us to create a force against the kite and stabilize it in the air.

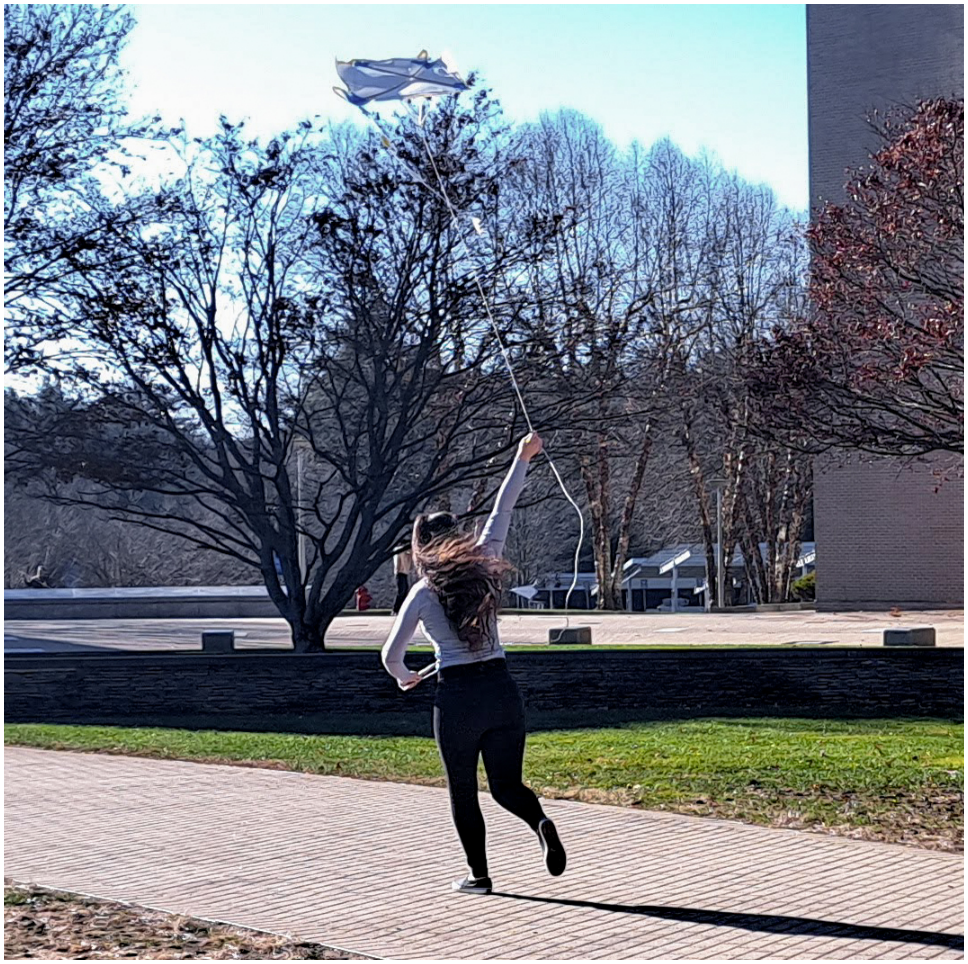

Figure 7: Photo of us running while flying our first prototype

Unfortunately, the kite flew in circles and proceeded to collide with the ground and every object in sight - including our professor's head... multiple times.

We learned that designing a non-traditionally shaped kite that would fly stable was challenging but we still wanted to keep the fish shape of the kite.

Updated Flying Prototype

For our final event, we did not end up flying a kite with our preferred specs as calculated previously in this report. Instead, we flew the same prototype after adjusting it. We adjusted the bridal point by moving it up the spine closer to the head of the kite, as that would account for the larger surface area closer to the head than the tail of the kite, as well as to account for the chance the center of pressure was being placed in front of the bridle point, which could have caused our kite to not have flight stability in the past. By changing the bridal point, the kite should spin less and be able to stabilize faster.

We also attached a significantly longer string so that we could take advantage of the greater wind speed that comes with higher altitudes.

When we took our kite out to fly... it flew! We had stronger wind this day which we estimated to be about 3 meters per second. We noticed less spinning and as we let out more string, our kite stabilized further. After letting out about 100 meters of string, our kite flew steady in the air. However, after a while the kite string would lose tension and the kite would begin to fall as it lost lift, meaning our tension was not negative in both the x and y directions, and the net force acting on the kite lost equilibrium.

Figure 8: Photo of us flying our kite high, more stable in the air

Improvements

With more time to design and fabricate our kite, we would incorporate the bridle point y-position we calculated of approximately 5.27 inches, as this would help stabilize our kite in flight as this distance minimized the net torque acting on the bridle point to near zero. Furthermore, the angle of attack that we calculated for the smallest net Torque on the bridle point, 13.8 degrees, is a small bit outside the optimal angle of attack range for traditional kites that we found through conversations with our professor and peers (angle of attack under 13 degrees), so given more time we would look into what could have caused this discrepancy, such as the unique geometry of our kite, and seek to change the kite design to lower our optimal angle of attack to fit this range.

Takeaways

Through the construction of our kites and physics calculations, we learned that flight is incredibly difficult to keep and maintain, and while many kite geometric settings such as angle of attack can have multiple values that achieve flight, only a small number of potential values will create stable flight, and by understanding the system of forces and torques acting upon the kite and its bridle point, we can better understand how to optimize a kite for stable flight.

Appendix

Code for finding bridal point

```
% Thermo Final Kite Project
clear all

syms L D CP x

% functions defining fish function

f1 = 0.15*(x-2.89)^2+7.863-10;
```

```
f2 = -1.07*(x-4)^2+9.181-10;
f3 = 0.026*(x-16.384)^2+5.194-10;
f4 = 0.085*(x-16.384)^2+5.194-10;
f5 = -0.669979*(-x+26)^(1/2);

% piecewise of initial fish function

y = piecewise((-0.884476<=x)&(x<2.89), f1, ...
              (2.89<=x)&(x<4), f2, ...
              (4<=x)&(x<16.384), f3, ...
              (16.384<=x)&(x<22.919), f4, ...
              (22.919<=x)&(x<=26), f5);

% plot of initial piecewise function

figure()
fplot(y)
xlim([-10 30])
ylim([-10 5])
yline(0, '--r')
hold on;
fplot(abs(y))
xlim([-4 30])
ylim([-7 7])
legend('Piecewise Curve', 'Kite Spine', 'Flipped Piecewise Curve')
title('Curve of Piecewise Function defining Fish Shape')

% integral of initial piecewise function

f = abs(eval((int(y,x,[-0.884476 26]))));

% functions defining inverse of fish piecewise

f_i1 = finverse(f1);
f_i2 = finverse(f2);
f_i3 = finverse(f3);
f_i4 = finverse(f4);
f_i5 = finverse(f5);

% piecewise of inverse of fish piecewise

```

```
y3 = piecewise((eval(subs(f1,x,-0.884476))>x)&(x>=eval(subs(f1,x,2.89))), f_i1,...
               (eval(subs(f2,x,2.89))>x)&(x>=eval(subs(f2,x,4))), f_i2,...
               (eval(subs(f3,x,4))>x)&(x>=eval(subs(f3,x,16.384))), f_i3,...
               (eval(subs(f4,x,16.384))>x)&(x>=eval(subs(f4,x,22.919))), f_i4,...
               (eval(subs(f5,x,22.919))>x)&(x>=eval(subs(f5,x,26))), f_i5);

% plot of inverse of fish piecewise

figure()
fplot(y3)
xlim([-10 10])
ylim([-5 30])

% integral of inverse of fish piecewise

eval(int(y3, x, [-2403/500, 0]));

% Calculation to find half-width of kite

i = -0.884476:0.1:26;
i_all = eval(subs(y,x,i));
half_w = abs(min(i_all));

% L(x) (function of length of each rectangular segment)
% can be written here as f_i5 - f_i4 - f_i3 - f_i2 - f_i1

L = f_i5 - f_i4 - f_i3 - f_i2 - f_i1;

% D(x) (function of that gives the distance from the reference line to the differential CP)
% can be written here as 1/4*L + f_i1

D = 3/4*L + f_i1+f_i2+f_i3;

% Calculation for Center of Pressure

CP = real(26 + eval(int(L*D, [0 half_w]))/eval(int(L, [0 half_w])));
```

```
% Calculation for Center of Gravity

CG = eval(abs(int(y*x, [-0.884476 26])) / abs(int(y, [-0.884476 26])));


CP = CP / 39.37;
CG = CG / 39.37;

A = 2*(f/39.37); % m^2
p = 1.204; % kg/m^3
W = (half_w/39.37) * 2; % m
mass = 0.06; % kg
v = 3; % m/s
FG = [0,9.81*-mass,0];

Torque = [];

% Parameter Sweep of bridle point y-positions to calculate drag force, lift
% force, net torque, and angle of attack.
for i=0:0.001:0.254
        BtoP = i;
        PtoG = CP - CG;

        angle = atan(i/CP+(0.884476/39.37));

        C_L = (2*pi*angle)/(1+(2*A*angle)/W^2);
        FL = 0.5*C_L*p*(v^2)*A;
        FD = ((p*(v^2)*A*sin(angle))/2) * (1.28*angle + ((A*(C_L^2))/(0.7*pi*W^2)));
        FP = [0,0,FL] + [FD,0,0];

        temp = cross([0,(0-BtoP)*sin(angle), 0], FP)+...
               cross([(CG-PtoG)*cos(angle),(0-BtoP)*sin(angle), 0], FG);

        Torque(end+1,1:6) = [abs(temp(3)),i,rad2deg(atan(i/CP)), FD, FL, 0];
end

```

```
[temp, indices] = sort(Torque(:,1),'ascend');

% Checking if both tension components are negative for each y-
    position
for i=1:length(Torque)
    Ty = -(Torque(i,5) + FG(3));
    Tx = -(Torque(i,4));
    if(Ty<0 && Tx<0)p
        Torque(i,6) = 1;
    end
end

rows = find(Torque(:,6));
Real_Torque = Torque(rows, :);

% plot of angle of attack vs net torque
figure()
plot(Real_Torque(:,3), Real_Torque(:,1))
xlabel('Angle of Attack (degrees)')
ylabel('Net Torque (N-m)')
title('Angle of Attack of Kite vs. Net Torque on Kite')

[~, temp] = min(Real_Torque(:,1));
min_angle = Real_Torque(temp,3)

% plot of bridle point y-position vs. net torque
figure()
plot(Real_Torque(:,2)*39.37, Real_Torque(:,1))
xlabel('Bridle Point Y-Position (Inches)')
ylabel('Net Torque (N-m)')
title('Y-Position of Bridle Point vs. Net Torque on Kite')

[~, temp] = min(Real_Torque(:,1));
min_BP_dist = Real_Torque(temp,2)*39.37

% Sweep for bridle point, check all combinations with ~0 net torque
    for negative
% tension vectors, and pick combination with highest angle value,
    because
% we want our 'unique design' to be as visible as possible
```