Modeling Stone Skipping

Elias Ayoub

April 26, 2019

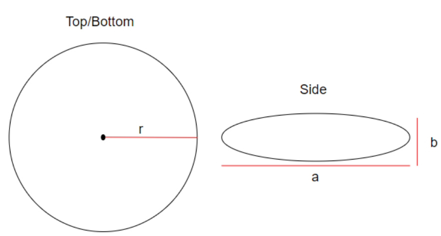
Contents

1 Introduction

Stone skipping or as others say, stone skimming, is the action of throwing a stone onto a surface of water in a manner causing it to skip repeatedly until it sinks. Many of us grow up skipping stones for fun just to see how far or how many skips we can get the stone to do while others have mastered it. According to Guinness World Records, the most consecutive skips of a stone on water is held by Kurt Steiner with 88 total skips. This might seem impossible but there are certain factors which affect a stone in motion that if mastered, one could maybe see beating the record as a possibility. These factors include the forces exerted onto the stone when released, the angle of attack, how fast the stone is spinning, how flat the stone is, etc. One of the assumptions that we make is that the stone has enough angular momentum(spin) and does not experience torque in order to be stable throughout its motion, meaning that the orientation of the stone is constant. These assumptions greatly help with the application of the model.

1.1 Using a discus

The Mathematics of Projectiles in Sport by Nevile De Mestre is a book that shows how numerous different sport projectiles are affected by forces in flight. For a stone to skip well, It will need to be flatter and since a flatter stone is closest to the shape of a discus than nearly any other projectiles, we will use a discus to define the shape of the stone mathematically. The discus' dimensions will be given like so: The radius of the circular top and bottom of the discus is



given by r, the height of the side ellipse by b and the length of the side ellipse by a which could also be thought of as $2r$.

1.2 Defining Force

From the moment of release out of the thrower's hands, the stone is automatically enduring many different forces. Netwon's Second law states that the force(F) on an object is given by its mass(m) multiplied by its acceleration(a).

$$F = ma$$

Our model is going to be split into two equations: the forces in the horizontal direction(x), and the vertical direction(y). This is necessary due to how forces such as gravity only affect objects in certain directions. We know that the mass is going to be a constant. However, the acceleration is a value that will always be changing so we need a way to rewrite a. Velocity is given by the change in position over time so the x component of the velocity is given by $v_x = \frac{dx}{dt}$ and the y component by $v_y = \frac{dy}{dt}$. Acceleration is given by the change in velocity over time so $a_x = \frac{dv_x}{dt}$ and $a_y = \frac{dv_y}{dt}$ which can both be rewritten as $a_x = \frac{d^2x}{dt^2}$ and $a_y = \frac{d^2y}{dt^2}$. By substitution into the original force equation, we have:

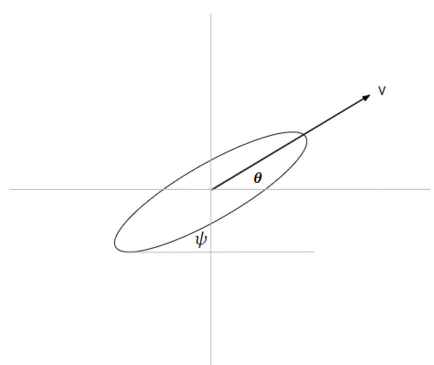
$$F_x = m\frac{d^2x}{dt^2} \quad (1)$$

$$F_y = m\frac{d^2y}{dt^2} \quad (2)$$

All of the forces will be split into an x and y component in order to be compatible with these equations.

2 Creating the Differential Equations

First, we will assume that the stone is thrown in a space in which there are no other forces acting upon it. This implies that $m\frac{d^2x}{dt^2} = 0$ and $m\frac{d^2y}{dt^2} = 0$ because



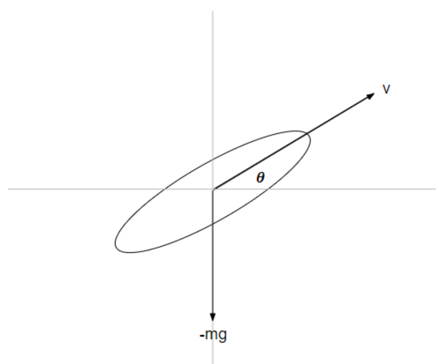
our accelerations are non-existent if there are no other forces. We introduce two new orientation values: θ is the angle of the translational velocity(velocity direction) and ψ is angle of the orientation of the discus with respect to the horizontal axis or in other words, the angle of attack. From the figure above, it is important to notice that the initial angle of attack is equal to the initial angle of the velocity. The assumption from earlier states that there is enough angular momentum to keep the orientation of the stone the same throughout the motion, therefore ψ is a constant. Using this as a starting point, we can begin to add different forces on the object one by one.

2.1 Gravity

The simplest force that can be added here is the force of gravity. This force is given by the mass of the object(m) multiplied by the acceleration due to gravity(g). g depends on the distance of the object from the center of the earth. It decreases as an object is further away. However, we will keep this as a constant since there will be very negligible change in altitude relative to the radius of the earth. The force due to gravity acts in the y-direction and causes the velocity to increase in the downward direction so we can subtract it, giving us:

$$m\frac{d^2x}{dt^2} = 0 \tag{3}$$

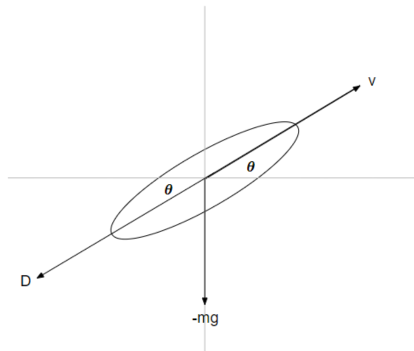
$$m\frac{d^2y}{dt^2} = -mg \tag{4}$$



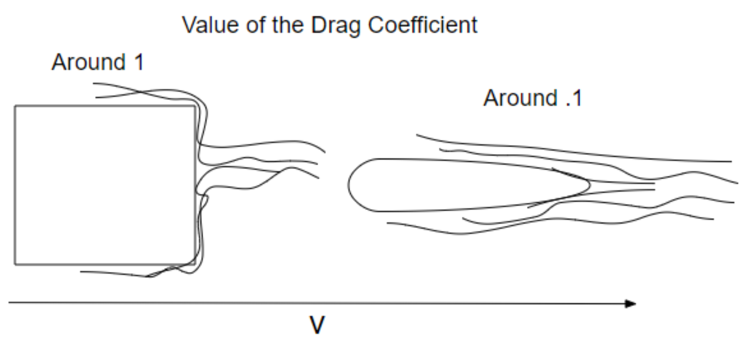
Since gravity only acts in the y-direction, F_x is still 0.

2.2 Drag and Lift

Any object in motion is affected by air/fluid resistance in different ways. The drag force is the force due to this resistance which exactly opposes the direction of the velocity.



Drag is calculated by $D = -\frac{1}{2}\text{p}Av^2C_D$,(Mestre 39) where p is the density of the fluid the object is in, A is the cross sectional Area of the object, v^2 is the square of the velocity and C_D is the drag coefficient. The Cross Sectional Area(A) depends on θ since it is decided by the direction of the translational velocity. It is calculated as the surface area of the discus which is experiencing the air resistancde. This will be discussed with better detail in another section. The drag coefficient C_D is a number that ranges from 0 to 1.1 (Mestre 157). This number depends on the shape and orientation of the object. For Example, an object with a flat surface in the direction of the velocity such as the square,

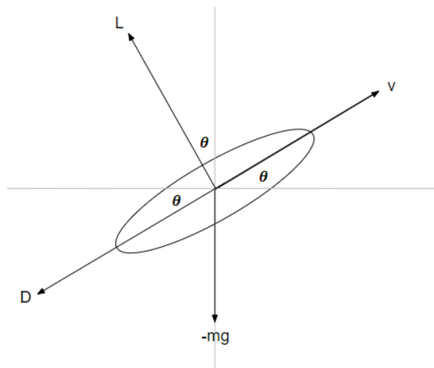


C_D is nearly as high as it could be while it is very low for the object on the right due to the air easily being able to move around the object without putting as much force against it. The higher the drag coefficient of an object, the more drag it experiences compared to another object with a lower coefficient even if their cross sectional areas are equal. The fact that the formula is negative works with the drag force being against the velocity direction. We now want to add the x and y components of D to our differential equations. By looking at the figure at the top of the page, is is obvious that the x-component is given as $D_x = -\frac{1}{2}pAv^2C_D\cos\theta$ and the y-component given as $D_y = -\frac{1}{2}pAv^2C_D\sin\theta$.

Our equations are now,

$$m\frac{d^2x}{dt^2} = -\frac{1}{2}pAv^2C_D\cos\theta \quad (5)$$
$$m\frac{d^2y}{dt^2} = -mg - \frac{1}{2}pAv^2C_D\sin\theta \quad (6)$$

Next is the lift force(L) which, unlike the drag force, is perpendicular to v. Lift is given by $L = \frac{1}{2}pAv^2C_L$ (Mestre 105) which is very similar to the drag force equation. The lift coefficient ranges from 0 to .9 for a discus and is also dependent on the shape and orientation of the object.



Similarily to the drag force, by looking at the figure above we see that the x-component of L is given by $L_x = -\frac{1}{2}pAv^2C_L\sin\theta$ and the y-component is given by $L_y = \frac{1}{2}pAv^2C_L\cos\theta$. Notice that only the x-component is negative due to the negative direction in the figure. We can now add the lift force to our main differential equations to obtain,

$$m\frac{d^2x}{dt^2} = -\frac{1}{2}pAv^2C_D\cos\theta - \frac{1}{2}pAv^2C_L\sin\theta \quad (7)$$
$$m\frac{d^2y}{dt^2} = -mg - \frac{1}{2}pAv^2C_D\sin\theta + \frac{1}{2}pAv^2C_L\cos\theta \quad (8)$$

and by factorization we finally have

$$m\frac{d^2x}{dt^2} = -\frac{1}{2}pAv^2(C_D\cos\theta + C_L\sin\theta) \quad (9)$$
$$m\frac{d^2y}{dt^2} = -mg + \frac{1}{2}pAv^2(C_Lcos\theta - C_D\sin\theta) \quad (10)$$

as our main differential equations format.

2.3 Converting to first order

As seen from the result of the previous section, the second order differential equations for modeling a stone(discus) in flight have numerous different values involved. To be able to implement these equations in MATLAB, they first need to be converted into a system of first order differential equations. By dividing both sides by m and converting to first order equations, we obtain

$$\frac{dx}{dt} = z \tag{11}$$

$$\frac{dy}{dt} = w \tag{12}$$

$$\frac{dz}{dt} = \frac{-pAv^2(C_D \cos\theta + C_L \sin\theta)}{-2m} \tag{13}$$

$$\frac{dw}{dt} = -g + \frac{pAv^2(C_D cos\theta + C_L \sin\theta)}{2m} \tag{14}$$

with z and w substituted as new variables.

3 Discus Adaptation

Next, we will be needing to define all the remaining variables and constants in terms appropriate for a discus and MATLAB implementation. Our constants in our equations are going to be

1. g - Acceleration due to gravity
2. p - Fluid density(air or water)
3. m - Mass of the object
4. ψ - Angle of attack

and the variables

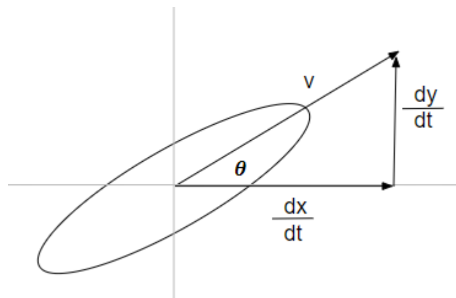
1. A - Cross-sectional area
2. v - Translational velocity
3. θ - Angle of the translational velocity with respect to the horizontal
4. C_D - Drag coefficient
5. C_L - Lift coefficient

3.1 Constants

We have the basic constants such as g(9.81 m/s^2) and p(1.225 kg/m^3 when the object is in the air and 997 kg/m^3 when the object is in contact with water). The air density is ignored when water density is involved due to a nearly negligible effect. Then we have our mass m which depends on the object's volume and density being used. Lastly, as stated before, it is assumed that the object is experiencing enough angular momentum to keep the initial angle of attack ψ constant throughout the motion.

3.2 Velocity and θ

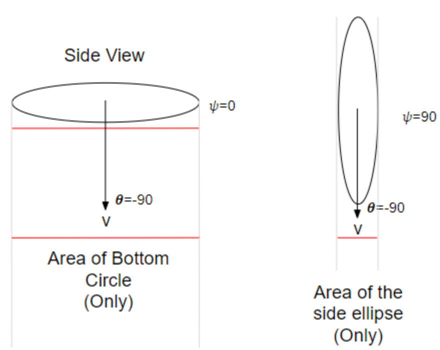
Any value that is changing throughout the motion needs to be defined in terms of $\frac{dx}{dt}(z)$ and/or $\frac{dy}{dt}(w)$ to be found at a specific time of the motion. We can obtain the velocity with the proper terms by looking at the change over time of the x and y components of the velocity.



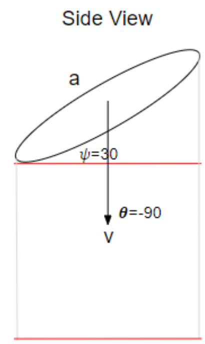
By using the Pythagorean Theorem, $v = \sqrt{(\frac{dx}{dt})^2 + (\frac{dy}{dt})^2}$. However, as stated in equations 11 and 12, we have new variables that were substituted when the equations were converted into first order. So $v = \sqrt{z^2 + w^2}$. The figure above can also be used to find θ in proper terms. $\tan\theta = \frac{w}{z}$ gives us $\theta = \tan^{-1}(\frac{w}{z})$.

3.3 Cross-Sectional Area

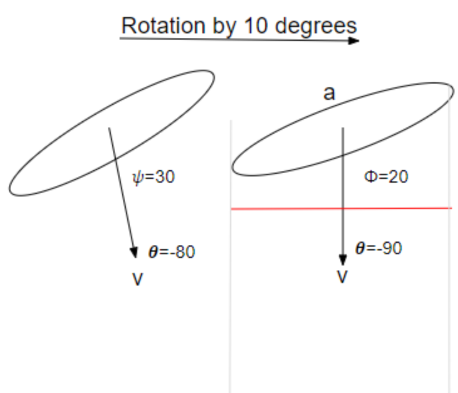
When $\theta = -90$, the cross-sectional area of the discus can be thought of as a projection of the object to the horizontal. Here are the two simplest examples of finding the cross sectional area:



At $\psi = 0$, the cross-sectional area is given just by the area of the circular bottom of the discus. At $\psi = 90$, the cross-sectional area is given just by the area of the side ellipse of the discus. As ψ increases towards $90°$, we see more of the side of the discus and less of the bottom. As ψ gets closer to 0, we see more of the bottom and less of the side. For $0 < \psi < 90$, the cross-sectional area is given by a combination of both areas depending on ψ and θ. For this, we could use



ψ to approximate how much of the circle and side ellipse is projected. For the bottom circle, we take the formula for the area of a circle and multiply it by $\cos\theta$. For the side ellipse, we take the area formula for an ellipse and multiply it by $\sin\theta$. So we have $A = \pi r^2 \cos\psi + (\frac{a}{2} \cdot \frac{b}{2})\pi \sin\psi$. This way of calculating A follows the previously stated bounds of ψ but is only an approximation. The issue here is when $\theta \neq -90$. Suppose $\psi = 30$ and $\theta = -80$.



If we calculate $a\cos\psi$, we get the area for when $\theta = -90$ but $\theta = -80$. One of the ways to work around this is by rotating the object into an orientation in which $\theta = -90$. Rotating the object clockwise by $10°$ will result with $\theta = -90$ and $\phi = 20$. We consider ψ after rotation as ϕ. ϕ is basically subtracting ψ by the distance between θ and -90(θ after rotation).

$$\phi = \psi - (-90 - \theta)$$

With this, we always have the angle of the object against the horizontal and so the area is calculated as:

$$A = \pi r^2 \cos\phi + \left(\frac{a}{2} \cdot \frac{b}{2}\right) \pi \sin\phi.$$

3.4 C_D and C_L

C_D and C_L depend on the orientation of the object with respect to the horizontal and the shape of the object. "The drag coefficient varies from 0.1 to 1.1 as the angle of attack changes from 0° to 90°, but at the same time the lift coefficient varies from zero to 0.9 (at 30°) and back to zero" (Mestre 157). Assuming that this change is linear, C_D can be modeled by

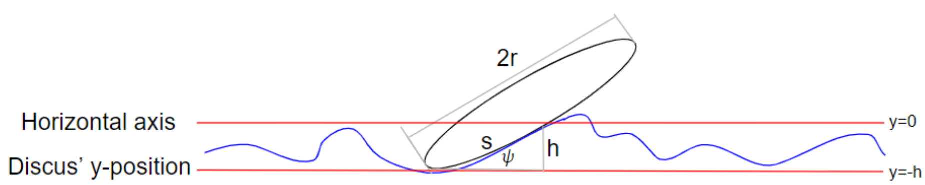
$$C_D(\psi) = \frac{\psi}{90} + .1$$

C_L increases until 30° then it starts to decrease. So for C_L, it can be modeled by

$$C_L(\psi) \begin{cases} \frac{\psi}{30}(.9), & 0 \leq \psi \leq 30 \\ .9\left(1 - \frac{\psi-30}{60}\right), & 30 < \psi \leq 90 \end{cases}$$

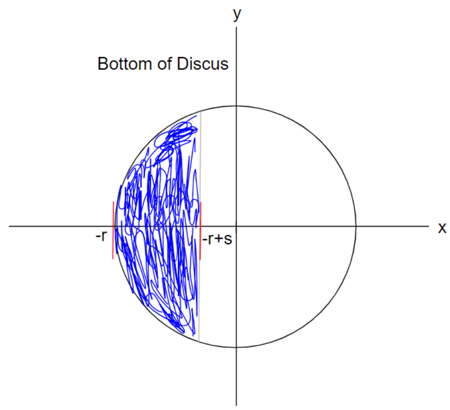
3.5 Submerged Area

As the stone hits the water at an angle, only a part of it will make contact and be submerged. This is the main event that generates the lift that causes the stone to skip on water. During this event, the effect of air resistance is negligible in comparison to the effect the water on the stone so the air resistance will be ignored when water resistance is involved. Before we are able to find the area of the bottom of the discus that is submerged, we need the length(s) of the bottom that is submerged. Notice here that the calculation of this area is different than the cross-sectional area from earlier as it does not take into account the angle of the velocity θ.



The y-position of the discus will be the very bottom of the object. MATLAB keeps track of this position at all times through the motion so this can be used to find s. The height that is submerged h is basically the vertical distance of the discus from the horizontal axis(surface of the water) so $h = |y|$ since y is always negative when any of the stone is submerged. Using the figure above,

$s \sin\psi = h \to s = \frac{h}{\sin\psi}$. The bottom of the discus is a circle, therefore we can find $A(s)$(submerged area depending on s) as a portion of the area of a circle.



This can be done using integration. The circle equation centered at the origin is given by $x^2 + y^2 = r^2$. Solving for non-negative y gives $y = \sqrt{-x^2 + r^2}$. Therefore,

$$A(s) = 2\int_{-r}^{-r+s} \sqrt{-x^2 + r^2}dx$$

gives the submerged area of the discus. The problem with using this formula in MATLAB is that it will be running thousands of integrations for one motion so the code will run for a very long time to finish executing. One way to solve this is by actually finding the result of the integral and using that as the $A(s)$ formula. So the submerged area is given by

$$A(s) = r^2 \left[\sin^{-1}\left(\frac{-r+s}{r}\right) + \frac{\pi}{2} + \sin\left(2\sin^{-1}\left(\frac{-r+s}{r}\right)\right)\right]$$

Proof. Starting off, we have

$$2\int_{-r}^{-r+s} \sqrt{-x^2 + r^2}dx$$

Trig substitution can be used here with $x = rsin\theta$ and $dx = rcos\theta d\theta$ to obtain

$$2\int_{x=-r}^{x=-r+s} \sqrt{r^2 - r^2sin^2\theta}rcos\theta d\theta = 2\int_{x=-r}^{x=-r+s} r^2cos^2\theta d\theta$$

by substitution and algebra. Next, we use the trig identity $cos^2\theta = \frac{1+cos(2\theta)}{2}$. The 2's can be canceled out and the constant r^2 can be moved to the front of the integral.

$$
\begin{aligned}
r^2 \int_{x=-r}^{x=-r+s} 1 + cos(2\theta) d\theta &= r^2 \int_{x=-r}^{x=-r+s} d\theta + r^2 \int_{x=-r}^{x=-r+s} cos(2\theta) d\theta \\
&= r^2 [\theta]_{x=-r}^{x=-r+s} + \frac{1}{2} r^2 \int_{x=-r}^{x=-r+s} cos(2\theta) 2 d\theta \\
&= r^2 [\theta]_{x=-r}^{x=-r+s} + \frac{1}{2} r^2 \left[sin(2\theta) \right]_{x=-r}^{x=-r+s}
\end{aligned}
$$

We know from earlier that $x = rsin\theta$. Therefore, $\theta = sin^{-1}\left(\frac{x}{r}\right)$ can be substituted back for the original terms.

$$
\begin{aligned}
&= r^2 \left[sin^{-1}\left(\frac{x}{r}\right) \right]_{-r}^{-r+s} + \frac{1}{2} r^2 \left[sin\left(2 sin^{-1}\left(\frac{x}{r}\right) \right) \right]_{-r}^{-r+s} \\
&= r^2 \left[\sin^{-1}\left(\frac{-r+s}{r}\right) - \sin^{-1}\left(\frac{-r}{r}\right) \right] \\
+\frac{1}{2} r^2 & \left[sin\left(2 sin^{-1}\left(\frac{-r+s}{r}\right) \right) - sin\left(2 sin^{-1}\left(\frac{-r}{r}\right) \right) \right]
\end{aligned}
$$

After simplifying, we obtain

$$
\begin{aligned}
A(s) &= r^2 \left(sin^{-1}\left(\frac{-r+s}{r}\right) + \frac{\pi}{2} \right) + \frac{1}{2} r^2 sin\left(2 sin^{-1}\left(\frac{-r+s}{r}\right) \right) \\
A(s) &= r^2 \left[sin^{-1}\left(\frac{-r+s}{r}\right) + \frac{\pi}{2} + \frac{1}{2} \sin\left(2 \sin^{-1}\left(\frac{-r+s}{r}\right) \right) \right]
\end{aligned}
$$

for the formula of the submerged bottom area. □

We will use this area formula in place of the cross-sectional area when the discus is in contact with the water.

4 MATLAB Implementation

Here, the first order differential equations

$$
\begin{aligned}
\frac{dx}{dt} &= z \\
\frac{dy}{dt} &= w \\
\frac{dz}{dt} &= \frac{-pAv^2(C_D \cos\theta + C_L \sin\theta)}{-2m} \\
\frac{dw}{dt} &= -g + \frac{pAv^2(C_D cos\theta + C_L \sin\theta)}{2m}
\end{aligned}
$$

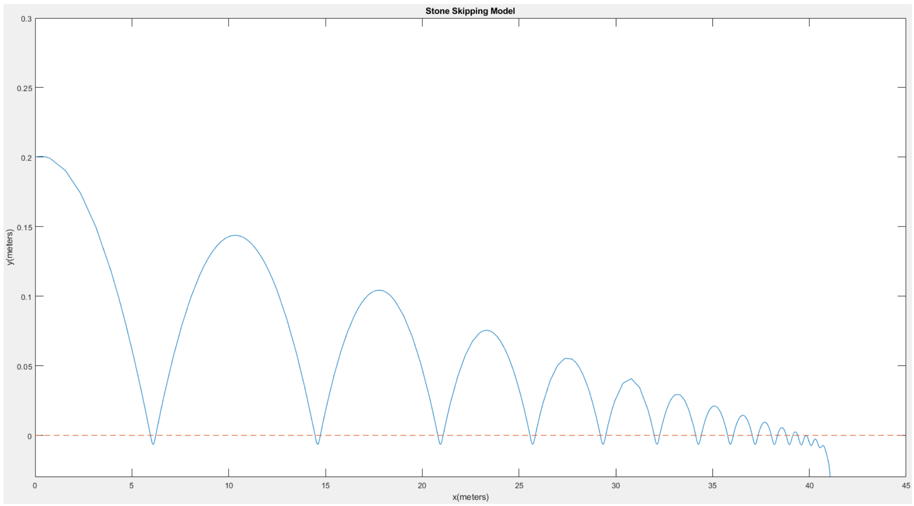
are implemented in MATLAB with the formulas from the previous sections substituted in order to simulate the motion of a stone being skipped. ODE45 is used which is a differential equation solver in MATLAB. Here are all of the initial values that were used:

1. p = 1.225 kg/m^3 (Air Density)
2. g = 9.81 m/s^2 (Acceleration due to gravity)
3. r = .025 m (Radius of the discus)
4. b = .02 m (Height of side)
5. m = .1 kg (Mass)
6. x-position = 0 m
7. y-position = .2 m
8. x-velocity = 27 m/s
9. y-velocity = .1 m/s

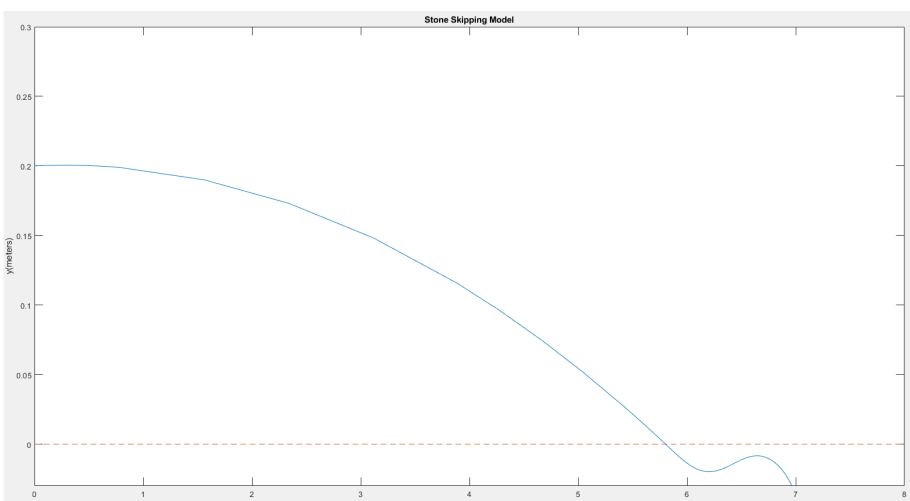
The values r, b represent a small stone with a small mass of only .1kg (100 grams). It makes sense to start the motion at $x = 0$ m with the starting point. The y-position being at $y = .2$ m assumes a person is throwing the stone .2 meters above the water level. This specific value is due to unexpected results at larger values of y but in reality, the average value of the initial y-position is a little bit higher. On average, a person can throw a baseball at around 60 miles per hour(about $27m/s$) so $x - velocity = 27$ m/s. The stone being thrown slightly above the horizontal gives $y - velocity = .1$ m/s. These are some decided initial values based on experimenting with the model. Throughout the motion of the stone, the value of p alternates between 1.225 kg/m^3 when the discus is above the water and 997 kg/m^3 when the discus is in contact with the water.

4.1 Plots and Tables

Running the 4 differential equations in MATLAB returns results of the x-positions, y-positions, x-velocities, and y-velocities all over time. According to De Mestre, when $\psi = 30$, the largest amount of lift is generated increasing the average number of skips. Using the x-positions and y-positions, a plot is made to show the motion of the stone being skipped with the values at the top of the page and $\psi = 30$.



For the model, a skip is being defined as touching the water and then reaching a height above the water level. The motion seems to follow that of a stone being skipped for the most part. The distance and height between each skip decreases due to the drag decreasing the velocity which in turn decreases the generated lift. With these initial values, the stone seems to skip 11 times with the end of the plot being ignored. The skips are only accounted up to the last time the stone reaches a height above the water(horizontal line in plot). Here it seems to cover a distance of around 33 meters after the first skip. Next, we can create a plot showing the motion with an initial angle of attack $\psi = 55$.



With an increase of 25°, the number of skips went from 11 to 0. This shows a problem with the model since in reality, the stone should still be able to skip at least once with these values. The algorithm is now run to calculate the number of skips with the same initial values but different angles of attack. Starting at $\psi = 30$ and increasing with 1 degree increments to $\psi = 55$, we obtain a table of ψ to skips. Values of ψ outside this range are not included since unpredictable results are received due to some problems with the code that have yet to be addressed.

ψ	Skips
30-31	11
32	10
33	9
34-35	8
36-37	7
38-39	6
40-41	5
42-43	4
44-46	3
47-50	2
51-54	1
55	0

As ψ increases away from 30°, the number of skips seems to decrease. This follows from De Mestre's statement that $\psi = 30$ is a good initial angle of attack to maximize the number of skips. Now, we change some of the initial values to get different results.

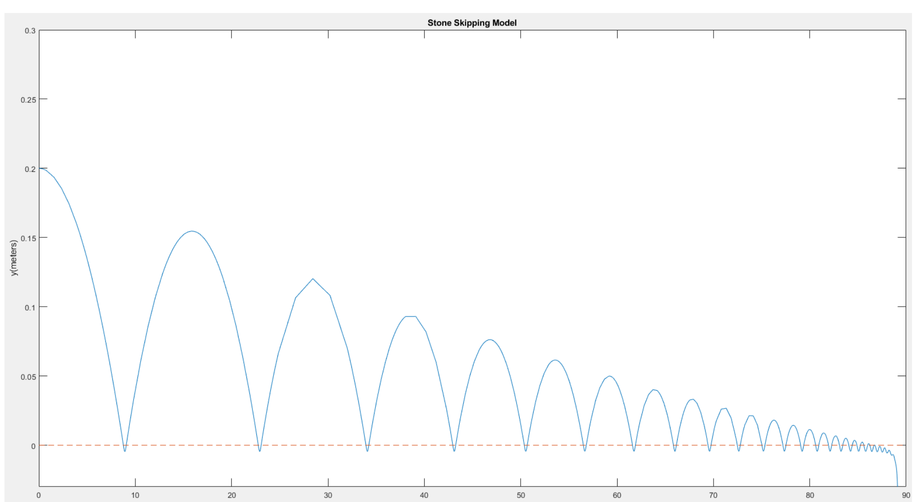
Let

$\psi = 30$

$x - velocity = 40\ m/s$

$y - velocity = 0\ m/s$

which gives the following plot:



The number of skips here increased to 19 which makes sense for an increased velocity. This seems to follow a similar pattern as the previous initial values. Now for the same range of ψ as before, the following table shows the number of skips depending on the angle of attack.

ψ	Skips
30	19
31	1
32	16
33	0
34	13
35	12
36	11
37	10
38	9
39	8
40	8
41	7
42-43	6
44	5
45-46	4
47-48	3
49-50	2
51-53	1
54-55	0

Overall, the table follows a similar pattern as the previous example. However, some problems with the code cause unexpected results such as what is shown at $\psi = 31$ and $\psi = 33$ but the main idea can be interpreted that the further ψ is from $30°$, the less skips happen.

5 Conclusion/Future Work

With more time, there are still elements of the model that need to be improved to receive better and more accurate results. We have the assumption that there is simply enough angular momentum that keeps the orientation of the stone the same as the initial angle of attack. Accounting for a specific value such as the initial angular momentum can be useful instead of the assumption. Additionally, the angle of attack is dependent on the initial angle of the velocity. When a stone is thrown, the side with the least surface area tends to be in the "front" direction of the velocity which inevitably decides the angle of attack. Implementing this as a factor will surely change the outcome of the results. Finally, the code does not give the most appropriate output depending on some of the initial values. If the mass is too high, if the angle of attack is too low/high, or if the initial y-value is too high, the results are unpredictable.

References

[1] Neville De Mestre, The Mathematics of Projectiles in Sport, *Australian Mathematical Society Lecture Series* **39, 105, 156-158**

MATLAB CODE

```
%obtaining the result for a specific angle of attack
xinitial = 0;
yinitial = .2;
xvelocity = 40;
yvelocity = 0;
attack = pi/6;
[t, v]=ode45(@(t, v) vChange(t, v, attack),[0 10],[xinitial,yinitial,xvelocity,yvelocity]);

%Getting the number of skips for that angle of attack
attack = rad2deg(attack);
disp('Angle of Attack/Number of skips:'); disp(attack);
skips = getSkips(v); disp(skips);

%Plotting x versus y for that angle of attack
plot(v(:,1),v(:,2));
hold on;
xL = get(gca, 'XLim');
plot(xL, [0 0], '--')
title('Stone Skipping Model')
ylim([-.03 .3]);
xlabel('x(meters)')
ylabel('y(meters)')
hold off;

%Obtaining a list of skips depending on the angle of attack for
%30<psi<55
xvelocity = 40;
yvelocity = 0;
degree = pi/180;
attack = pi/6;
while ((pi/6) <= attack && attack <= pi/(180/56))
    [t, v] = ode45(@(t, v) vChange(t, v, attack), [0 10], [0, .2, xvelocity, yvelocity]);
    attack = rad2deg(attack);
    disp('Angle of Attack/Number of skips:'); disp(attack);
    skips = getSkips(v); disp(skips);
    attack = deg2rad(attack);
    attack = attack + degree;
end
```

```
function skips = getSkips(v)
skips = 0;
n = 2;
rows_and_columns = size(v(:,:));
numrows = rows_and_columns(1);
while(n < numrows)
    q = n;
    if((v(n,4)*v(n-1,4))<0 && (v(n,4)<v(n-1,4)) && (v(n,2)>-.03))
        while(q < n + 500)
            if(v(q+1,2)>0)
                skips = skips + 1;
                q = n + 500;
            else
                q = q + 1;
            end
        end
    end
    n = n + 1;
end
%This next line is due to the final contact with the
%water before sinking being counted as a skip
skips = skips - 1;
end
```

```matlab
function [dRdt] = vChange(t,R, attack)
p = 1.225; %air density(kg/m^3)
g = 9.81; %gravity(m/s^2)

%defining the discus' dimensions and traits
r = .025; %radius for top/bottom circle(m)
a = 2*r; %length of side ellipse(m)
b = .02; %height of side ellipse(m)
m = .1; %mass(kg)

theta = atan(R(4)/R(3)); %angle of the translational velocity
psi = attack; %angle of attack
Cd = (psi/(pi/2))+.1; %Drag coefficient
Cl = ((pi/6)-rem(psi,(pi/6)))*(.9); %lift coefficient
phi = (psi-(pi/2))-theta; %psi after rotation(used to find Area)
Area = pi*r^2*cos(phi)+(((a*b)/4)*pi*sin(phi));%cross sectional area
v = sqrt(R(3)^2+R(4)^2); %velocity
%D/L coefficients with angles
anglesx = ((Cd/sqrt(1+(R(4)^2/R(3)^2)))+(Cl*R(4))/(R(3)*sqrt(1+(R(4)^2/R(3)^2))));
anglesy = ((Cl/sqrt(1+(R(4)^2/R(3)^2)))-(Cd*R(4))/(R(3)*sqrt(1+(R(4)^2/R(3)^2))));

%only account for the SUBMERGED area when in water
if(R(2)<0)
    p = 997; %Water density(kg/m^3)
    s = abs(R(2))/sin(psi);
    submerged = r^2*(asin((-r+s)/r)+(pi/2)+((1/2)*sin(2*asin((-r+s)/r))));
    Area = submerged;
end

dRdt(1,1) = R(3);
dRdt(2,1) = R(4);
dRdt(3,1) = (-p*Area*v^2*anglesx)/(2*m);
dRdt(4,1) = -g + (p*Area*v^2*anglesy)/(2*m);
end
```