Author: Adrian Szklarski, 25.06.2022.

Technological level: Intermediate

Programming level: Mid

Technologies used: Python 3.10 | PyCharm | Linux | OpenOffice

Engineering methods for missile design:

Flight dynamics of missiles and bombs*: Motion projectile in a vacuum: Part 2 of 4*

A fundamental issue in rocket technology: the flight of a projectile in a vacuum.

Technology used: Python 3.10 | PyCharm | Linux – Ubuntu | Flask

Part 2: Study of the projectile path equation

In this case, the equation of the track represents a second degree curve whose general equation is of the form: $A\,x^2 + 2Bxy + C\,y^2 + 2\,Dx + 2\,Ey + F = 0$, so we solve the quadratic equation according to the formula:

$$x^2 + px + q = 0 \rightarrow x_{1,2} = \frac{-p}{2} \pm \sqrt{\frac{p^2}{4} - q} \quad , \quad ax^2 + bx + b = 0 \rightarrow x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4\,ac}}{2\,a} \quad ,$$

$$\frac{-g}{2\,v_0^2 * \cos^2\theta_0} * x^2 + x * tg\theta_0 - 1 * y = 0$$

$$A = \frac{-g}{2\,v_0^2 * \cos^2\theta_0} * x^2 \quad ; \quad 2\,B = 0 \quad ; \quad C = 0 \quad ; \quad 2\,D = tg\theta_0 \quad ; \quad 2\,E = -1 \quad ; \quad F = 0$$

Because $B = C = 0$ therefore the discriminant $B^2 - 4\,AC = 0$ this means that the curve is a parabola and its general form is as follows:

$$Ax^2 + \cancel{2Bxy} + \cancel{Cy^2} + 2Dx + 2Ey + \cancel{F} = 0 \quad \rightarrow \qquad \boldsymbol{Ax^2 + 2Dx + 2Ey = 0}$$

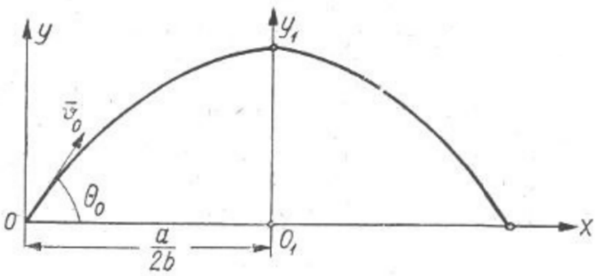
substituting we get: $-\frac{g}{2 \cdot v_0{}^2 \cdot cos^2\theta_0} \cdot x^2 + x \cdot tg\theta_0 - y = 0$

or in classical form the same formula: $-ax^2 + bx - y = 0 \rightarrow y = -ax^2 + bx$

looking for the elements (the point where the projectile hits the ground) we equate y from zero:

$$0 = -ax^2 + bx \quad \rightarrow \quad x(-ax + b) = 0$$

hence the elements: $x_1=0$ and
$x_2=b/a=X$,



x_1 - is the starting point;

x_2 - is the horizontal lift in a vacuum or the point of fall.

Let us now move the center of the system in the direction x of by the segment equal to half of the loudness. Moving to new coordinates one can write: $y = y_1;\quad x = \frac{b}{2a} + x_1$

Substituting this now into the equation: $y = -ax^2 + bx$

we get:

$$y_1 = -a\left(\frac{b}{2a} + x_1\right)^2 + b\left(\frac{b}{2a} + x_1\right),$$

and after transformations:

$$y_1 = -a\left[\left(\frac{b}{2a}\right)^2 + 2\cdot\frac{b}{2a}\cdot x_1 + {x_1}^2\right] + \frac{b\cdot b}{2a} + b\cdot x_1 = -a\left[\frac{b^2}{4a^2} + \frac{b}{a}\cdot x_1 + {x_1}^2\right] + \frac{b^2}{2a} + b\cdot x_1$$

$$y_1 = -\left[\frac{a\cdot b^2}{4a^2} + a\cdot\frac{b}{a}\cdot x_1 + a\cdot {x_1}^2\right] + \frac{b^2}{2a} + b\cdot x_1 = -\left[\frac{b^2}{4a} + b\cdot x_1 + a\cdot {x_1}^2\right] + \frac{b^2}{2a} + b\cdot x_1$$

$$y_1 = -\frac{b^2}{4a} - \cancel{b\cdot x_1} - a\cdot {x_1}^2 + \frac{b^2}{2a} + \cancel{b\cdot x_1} = -\frac{b^2}{4a} - a\cdot {x_1}^2 + \frac{b^2}{2a}$$

$$y_1 = \frac{b^2}{a}\left(\frac{1}{2} - \frac{1}{4}\right) - a\cdot {x_1}^2 = \frac{b^2}{a}\left(\frac{2-1}{4}\right) - a\cdot {x_1}^2 = \frac{1}{4}\cdot\frac{b^2}{a} - a\cdot {x_1}^2$$

$$\boldsymbol{y_1 = \frac{1}{4}\cdot\frac{b^2}{a} - a\cdot {x_1}^2}$$

This is the equation of a parabola in which the odd powers are missing x. The new axis is the new axis of symmetry, because any two values of $x = k$ and $x = -k$ always corresponds to the same value Conclusion:

A track in a vacuum has a vertical axis of symmetry, distant from the origin by a distance equal to half its lift.

In this function, we equate y to zero, looking for the points where the projectile falls.

$$\frac{-g}{2 v_0^2 \cos^2\theta_0} * x^2 + x * tg\,\theta_0 - y = 0$$

Then, after converting to a product, we look for the roots of the equation:

$$x\cdot\left(-\frac{g}{2\cdot {v_0}^2\cdot cos^2\theta_0}\cdot x + tg\ {}_0\right) = 0$$

From this it follows that the first root $\boldsymbol{x_1 = 0}$ is the point of the starting location, while the second is:

$$-\frac{g}{2\cdot {v_0}^2\cdot cos^2\theta_0}\cdot x + tg\theta_0 = 0 \quad\rightarrow\quad -\frac{g}{2\cdot {v_0}^2\cdot cos^2\theta_0}\cdot x = -tg\theta_0$$

$$\boldsymbol{x_2 = \frac{tg\theta_0\cdot 2\cdot {v_0}^2\cdot cos^2\theta_0}{g}}$$

Because: $tg\theta_0 = \frac{sin\theta_0}{cos\theta_0}$,

and $\sin 2\theta_0 = 2 \cdot sin\theta_0 \cdot cos\theta_0$

so let's transform the formula further:

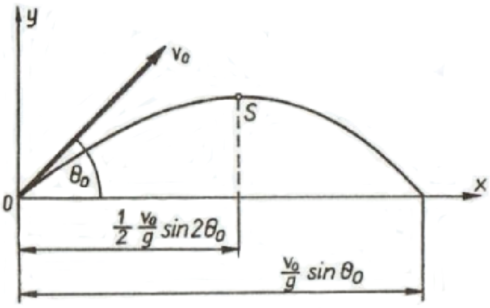
$$x_2 = \frac{sin\theta_0 \cdot 2 \cdot v_0{}^2 \cdot cos^{\cancel{2}}\theta_0}{\cancel{cos\theta_0} \cdot g} = \frac{sin\theta_0 \cdot 2 \cdot v_0{}^2 \cdot cos\theta_0}{g} = \frac{v_0{}^2}{g} \cdot 2 \cdot sin\theta_0 \cdot cos\theta_0 = \frac{v_0{}^2}{g} \cdot sin2\theta_0$$

$$\boldsymbol{x_2 = \frac{v_0{}^2}{g} \cdot sin2}\ _{\mathbf{0}}$$

So, in the end, we have the formula for the bullet velocity for a projectile fired with velocity v_0 in a vacuum, at an angle θ_0 to the earth's surface.

The highest projectile velocity is obtained for $sin2\theta_0 = 1$ $\theta_0 = 45°$

so: $$\boldsymbol{x_2 = \frac{v_0{}^2}{g}}$$



Conclusions from trajectory analysis:

- The motion of a projectile in a vacuum does not depend on the mass of the projectile;

- The trajectory of flight is a symmetrical parabola;

- The intersection of the parabola with the ground is the point of takeoff and fall (described above);

- Angle-dependent (throwing angle) θ_0 projectile lift has arbitrary values

$$x_2 \in \left(0 \div \frac{v_0{}^2}{g}\right).$$

- From conclusion two it follows that the track will reach its maximum height at a point in the interval $x_2 = 0.5\frac{v_0^2}{g}\sin 2\theta_0$, i.e. the parabola has a vertical axis of symmetry lying from the origin in the middle of the lift.

To calculate the velocity of a projectile at any point on the track, we start from the equation of mechanical energy. According to conservation of energy, a projectile flying upwards must lose velocity, <u>and the Kinetic Energy at any point on the trajectory must equal the Potential Energy of the projectile.</u>

Kinetic energy at any height: $E_K = \frac{m \cdot v_0{}^2}{2} - \frac{m \cdot v^2}{2}$

or

$$E_K = \frac{Q}{2g} \cdot v_0{}^2 - \frac{Q}{2g} \cdot v^2$$

m - projectile mass;

v_0 - initial speed;

Q - siła ciężkości pocisku;

v - speed at the height under consideration.

A projectile acquires Potential Energy by reaching a given height E_p która w zależności od masy i uzyskanej wysokości wynosi: $E_p = mgy$ therefore, from the equality of energies $E_k = E_p$ results:

$$\frac{m \cdot v_0^2}{2} - \frac{m \cdot v^2}{2} = m \cdot g \cdot y \quad |: m \quad \rightarrow \quad \frac{v_0^2}{2} - \frac{v^2}{2} = g \cdot y$$

Therefore, the velocity is: $\boldsymbol{v = \sqrt{v_0^2 - 2 \cdot g \cdot y}}$

Subsequent findings:

- The projectile velocities at two different points at the same height are equal to each other;

- Initial velocity equals final velocity (initial velocity equals final speed);

- The projectile at the highest point of the track has the lowest velocity.

Flight time can be determined from the equation: $t = \frac{x}{v_0 \cdot cos\theta_0}$

substituting the projectile penetration value for the current coordinate value:

$$x_2 = \frac{v_0^2}{g} \cdot sin2\theta_0; \quad (x = x_2):$$

$$t = \frac{\frac{v_0^2}{g} \cdot sin2\theta_0}{v_0 \cdot cos\theta_0} = \frac{v_0^2 \cdot sin2\theta_0}{g \cdot v_0 \cdot cos\theta_0} = \frac{v_0^{\cancel{2}} \cdot 2 \cdot sin\theta_0 \cdot \cancel{cos\theta_0}}{g \cdot \cancel{v_0} \cdot \cancel{cos\theta_0}}$$

So in the end $t = T$ where T is the time of flight of a full range projectile for v_0 and θ_0 :

$$\boldsymbol{t = \frac{2 \cdot v_0 \cdot sin\theta_0}{g}}$$

(Code part 2)

…..

```
    def flight_path_envelope_for_constant_speed(self):
   self.tab_y_en = []
   for x in range(0, int(self.sin_2_teta) + 1):
      y_en = round(math.pow(self.initial_velocity, 2) / (2 * G) - (G / (2 * math.pow(self.initial_velocity, 2))) *
math.pow(x, 2), 2)
      self.tab_y_en.append(y_en)
      x += 1
   return self.tab_y_en
```

```
    def axis_of_symmetry(self):
        x_symmetry = round(math.pow(self.initial_velocity, 2) * self.tangens_teta0 / (G *
((math.pow(self.tangens_teta0, 2) + 1))), 2)
        tab_sym = []
        for x in range(0, int(self.sin_2_teta) + 1):
            tab_sym.append(x_symmetry)
            x += 1
        return tab_sym

    def velocity_of_flight(self):
        self.vel = []
        for y in self.tab_y:
            velocity = math.sqrt(round(math.pow(self.initial_velocity, 2) - 2 * G * y, 2))
            self.vel.append(velocity)
            y += 1
        return self.vel

    def elements_of_path(self):
        self.x_start = 0
        self.x_end = self.sin_2_teta
        self.time_of_flight = 2 * self.initial_velocity * math.sin(self.firing_angle) / G
        self.top_of_trajectory = math.pow(self.initial_velocity, 2) * (math.sin(self.firing_angle) ** 2) / (2 * G)
        self.time_of_flightTOP = self.time_of_flight / 2
        self.average_flight_altitude = (2/3)*max(self.tab_y)

        # flight of a projectile through the layers of the atmosphere
        layer1 = self.time_of_flight*(math.sqrt(4-1+1)-math.sqrt(4-1))/math.sqrt(4)
        layer2 = self.time_of_flight*(math.sqrt(4-2+1)-math.sqrt(4-2))/math.sqrt(4)
        layer3 = self.time_of_flight*(math.sqrt(4-3+1)-math.sqrt(4-4))/math.sqrt(4)
        layer4 = self.time_of_flight*(math.sqrt(4-4+1)-math.sqrt(4-4))/math.sqrt(4)

        # average time of flight of the projectile through the layers of the atmosphere
        self.average_time = (layer1+layer2+layer3+layer4)/4
```

```
# average projectile range with air drag

if 0 < self.firing_angle <= 0.261799:
      self.x_end_angle = self.x_end /8.2
elif 0.261799 < self.firing_angle <= 0.698132:
    self.x_end_angle = self.x_end / 3.6
elif 0.698132 < self.firing_angle <= 0.785398:
    self.x_end_angle = self.x_end / 1.05
return self.x_start, self.x_end, self.time_of_flight, self.top_of_trajectory, \
       self.time_of_flightTOP, self.average_flight_altitude, self.average_time, self.x_end_angle
```