**The report and documentation of ‘For Rocket’**

Donghun Kim

This documentation is written to explain and guide for utilize of ‘For Rocket’ program. This program is basically designed and made to calculate the launch of rockets and simulate the launch itself including plenty of situations, information, and knowledge such as rocket mechanics, mathematics, physics, and astrophysics with ‘Python’ and ‘Tkinter’ as main and central tools. I totally want that this program can provide and allow tons of students, undergraduates, graduates, and other general people as well as scientists and engineers the invaluable and priceless opportunities to enjoy more over the space science and to open their myriad and infinite possibilities. I will account for a number of equations, terms, and symbols with the ones which are employed in mathematics, physics, and programing in common.

* The basic rocket equation

First of all, this chapter requires readers to understand some knowledge with regard to the rocket mechanics from basic things to profound things as an essential requisite as well as python language.

* Projectile motion

The projectile motion chapter is that examining and researching the object’s motions which are launched with some degree. Furthermore, we can make plots and tables(charts) with ‘matplotlib’ and ‘pandas’.

* Arrangement of terms in ‘Projectile motion equation’

> v0 : Initial velocity. [m/s]

> deg : Degree(has radian value) -> I will utilize this constant as ’θ’

> g : Gravity of Earth. The value is 9.81 [m/s^2]

> v\_x = v0 \* cos(θ) [m/s] : The velocity of x-axis at initial time, which is comparable with the velocity of x-axis over time when other obstacles or resistances are not existed. This is because the velocity of x-axis is not affected by other accelerations, which means that the velocity of x-axis is not changed whenever the time flows.

(1) velocity at x-axis

v = v0 x cos(θ)

> v\_y = v0 \* sin(θ) - g \* t [m/s] : The velocity of y-axis over time. The element 'v\_y' is composed with the multiplication of **1)** velocity of y-axis at initial time - v0 \* sin(θ), and **2)** gravitional restriction per time - g \* t. Then, subtract **2)** from **1)**

(2) velocity at y-axis

v = v0 x sin(θ) (at initial time)

v = v0 x sin(θ) - g x t (over time)

This means that, in contrast with the velocity of x-axis, the velocity of y-axis is influenced by gravitional restriction. Therefore, the latter is converting when the time flows, but the former is not.

> t\_h = v0 \* sin(θ) / g [s] : The time when the object reaches at the maximum height. This equation is made the multiplication with initial velocity(v0) and sin(θ), and this value is divided with gravity of Earth(g). The velocity at the maximum height is 0. This is because gravitional restriction makes the highest point. In short, while any restrictions are not existed, the objects are moving to myriad point, but the restrictions are existing, the objects have their own limited points. The time when the object reaches maximum height is

(3)

v22= v12 - 2a(Δx)

v2 = v1 + at

th = v0sin(θ) / g

(in above equation, v1 = v0 x sin(θ), v2 = 0, a = -g)

> h = (v0 \* sin(θ) \* t\_h) – (0.5 \* g \*(t\_h)\*\*2) [m] : Maximum height which the object can reach. This equation is composed with **1)** the multiplication of the v0(initial velocity), sin(θ), and t\_h(the time when the object reaches at the maximum height) and **2)** 0.5, g(gravity of Earth), and the square of t\_h. Then subtracting the equation **2)** from **1).**

(4) maximum height

H = v0 x sin(θ) x th - 0.5 x g x th2

> t\_r = 2 \* t\_h [m/s]: The time when the object reaches at maximum distance(at x-axis). This time is exactly comparable with the double of the time when the object reaches at maximum height(t\_h). In this suituation, the resistances such as air, drag coefficent, lift, any frictions, and rebounding are neglected.

(5) the time when the object reaches at maximum distance

tr = 2th

th = v0sin(θ), so,

tr = 2v0sin(θ) / g

> r = v\_x \* t\_r [m] : The maximum distance where the object can reach at x-axis. The equation of distance is given as D(distance) = v(velocity) x t(time). The element 'v\_x' is the velocity at x-axis(both at initail time and over time) and element 't\_r' is that the time when the object reaches at maximum distnace. In this equation, the restrictions are not existed, so the velocity at x-axis is immutable.

(6) maximum distance

R = vx x tr

(vx = v0cos(θ), tr = 2v0sin(θ) / g)

R = 2v02sin(θ)cos(θ) / g

= v02sin(2θ) / g

> x = v0cos(θ) \* t : The location(distance) of object over time at x-axis.

> h\_t = (v0 \* sin(θ) \* t) - 1/2 \* g \* t\*\*2 : The height over time(location of y-axis over time). Do not confuse the element 't(juts time)' with 't\_h(the time when the object reache maximum height)' in the above equation of 'h'. Those two equations ' x' and 'h\_t' are basically originated from

(7) distance equation

x = x0 + v0 x t + 0.5at2

(at equation 'x' : x0 = 0, v0 = v0cos(θ), a = 0)

(at equation 'h\_t(or 'y')' : x0 = 0, v0 = v0sin(θ), a = -g)

* Setting the table and graph

> pandas : Import pandas to draw the table which is expressed the time, height, distance.

> matplotblib : Import maplotlib to draw the graph which is expressed the route of object moving over time.

>

* 3
* 4
* 5
* 6
* 7
* 8
* 9
* Geostationary orbit
* Escape speed(velocity)
* 1
* 2
* 3
* 4
* 5