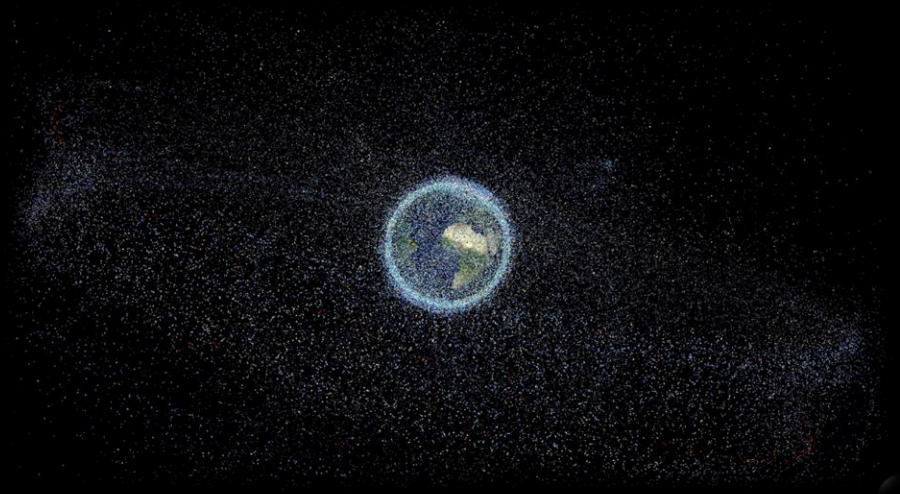
PYTHON SIMULATION PACKAGE FOR SPACE FLIGHT PROFILES

UNIVERSIDAD DE ANTIOQUIA

Astra

Sebastián A. Zapata
Julián M. Arenas
Jorge I. Zuluaga

Universidad de Texas
Sounding Rockets



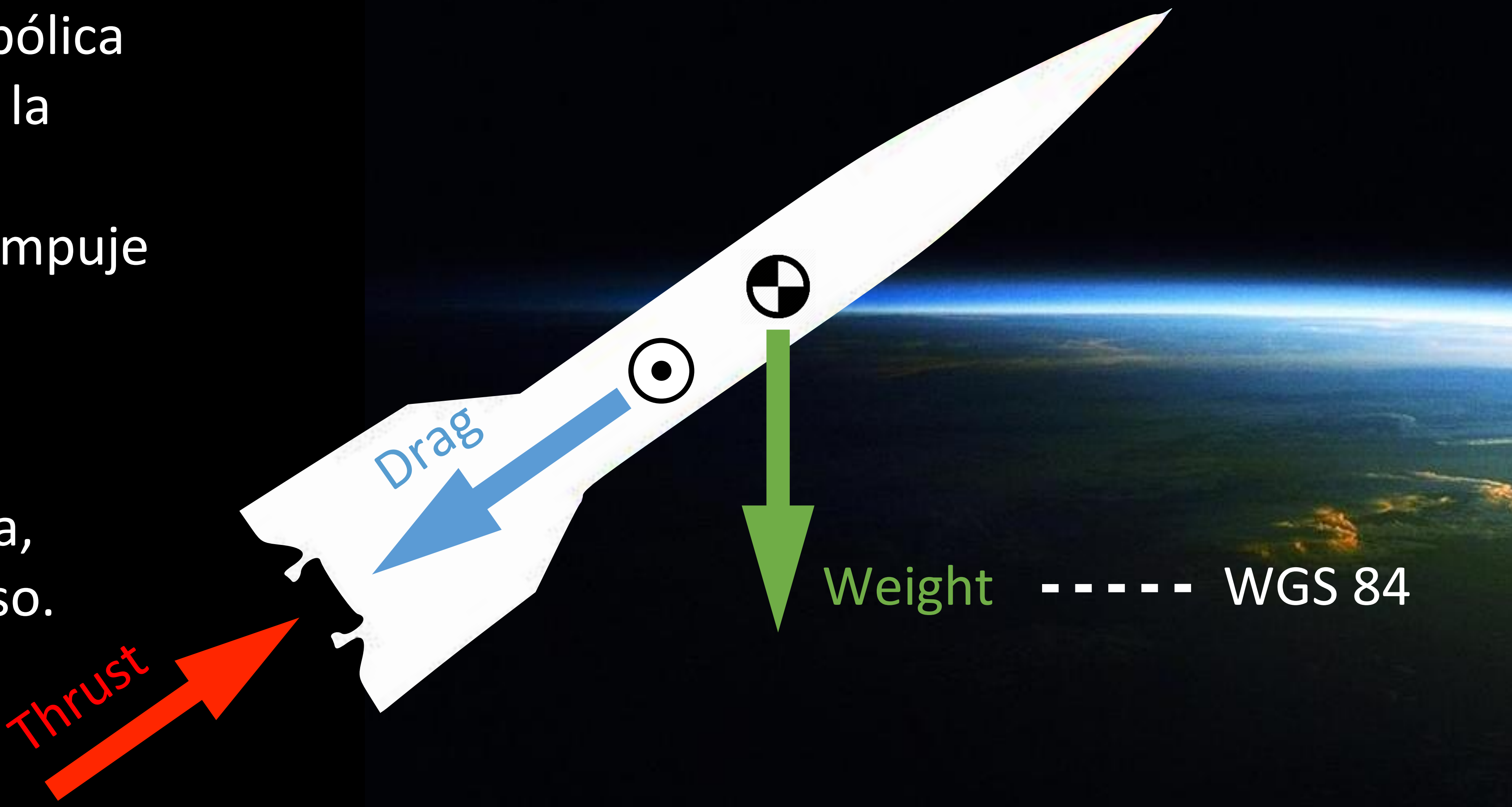
CARACTERÍSTICAS

- Python 3 → Open Source
- Licencia GNU 3.0
- Modular
- Perfecto para trabajar en Jupyter Notebooks
- Base de datos → Debrisk

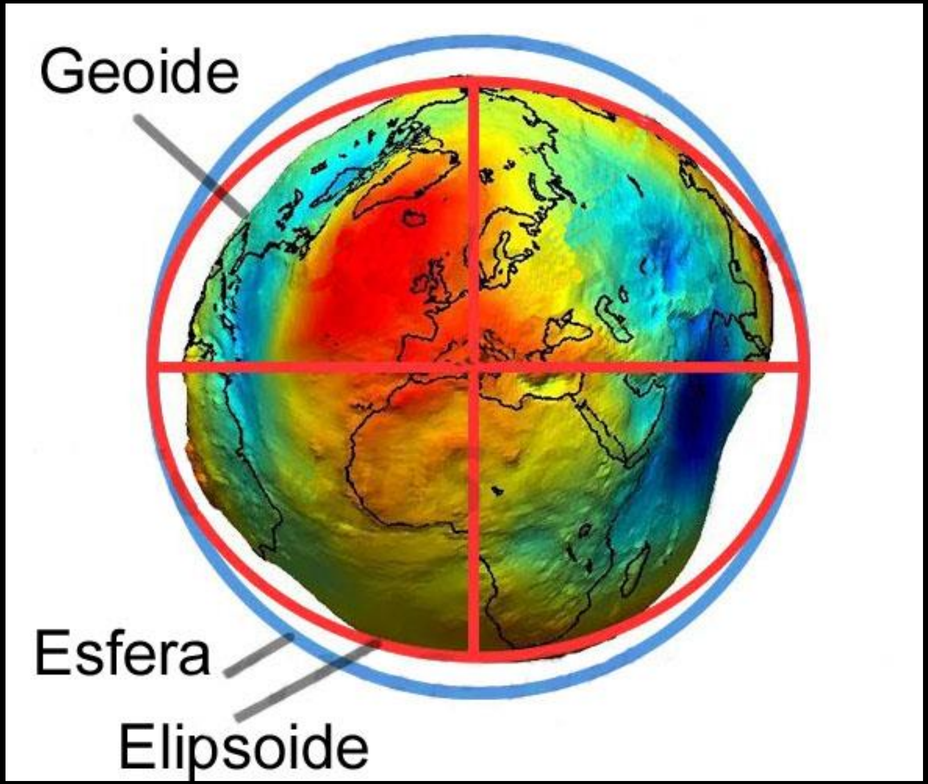
- Vuelo suborbital.

- Trayectoria parabólica determinada por la atracción gravitacional,el empuje y el arrastre aerodinamico.

- Fases: propulsada, balística, reingreso.



World Geodetic System (WGS 84)



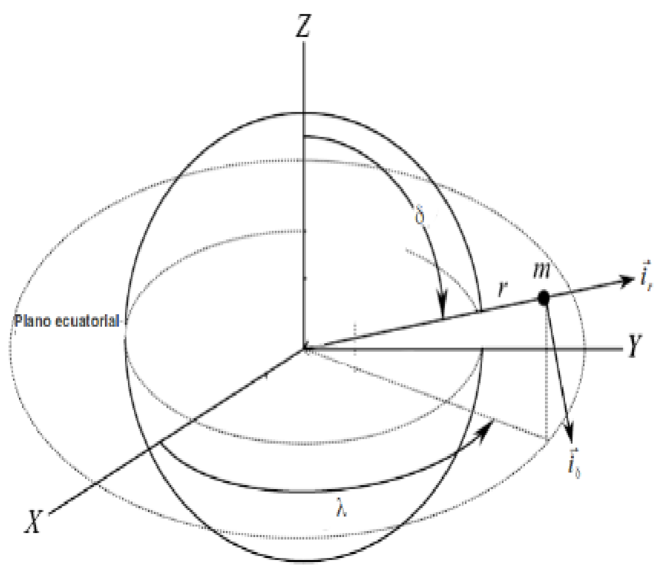
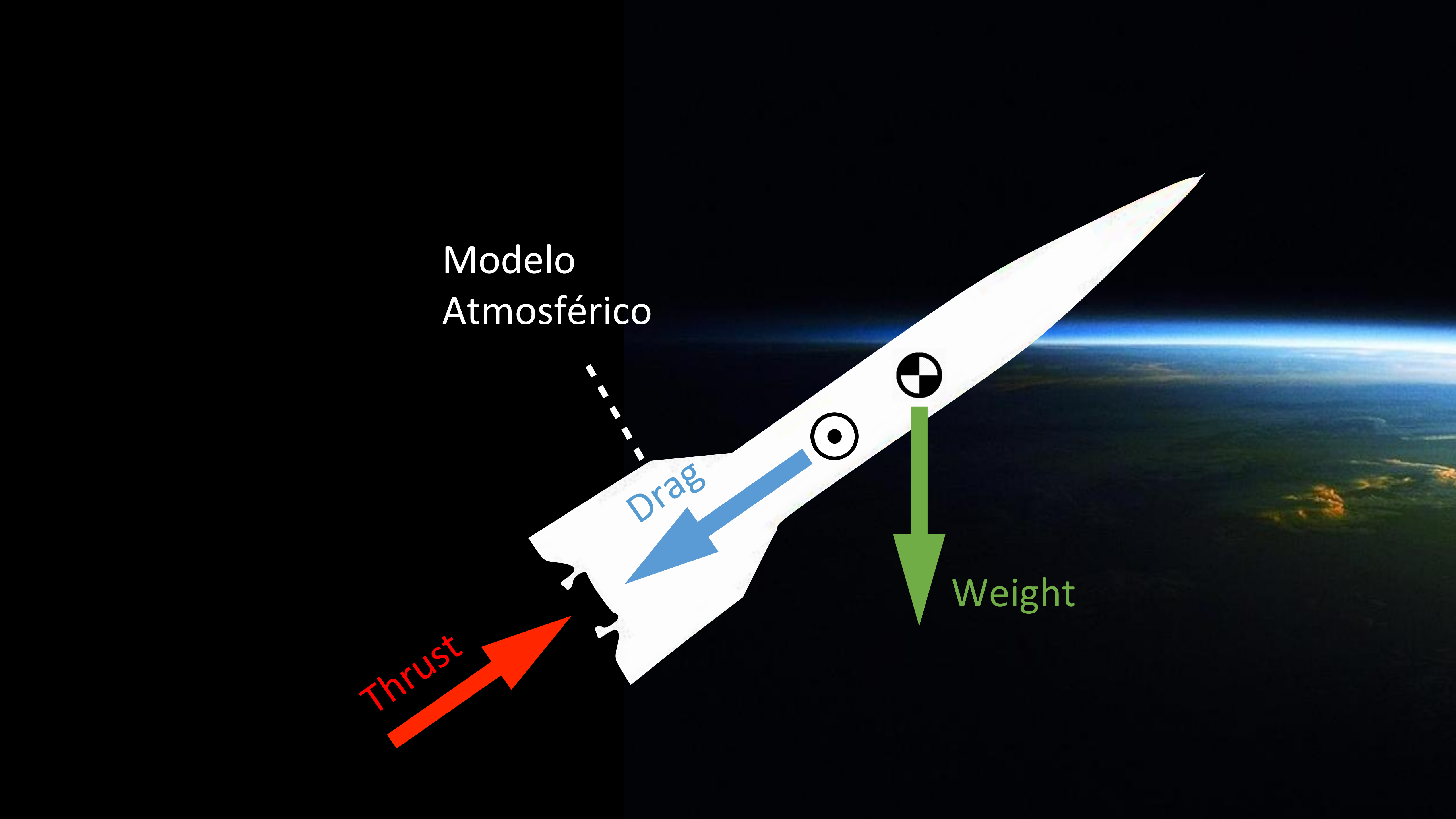


Figura 5.1. Dirección de las componentes de la gravedad

```
def gravity(r,latitude):

    phi=pi/2-latitude

    g_center=-mu*(1-1.5*J2*(3*(cos(phi)**2)-1)*(Re/r)**2-2*J3*cos(phi)*(5*cos(phi)**2-3)*
                    (Re/r)**3-(5/8)*J4*(35*cos(phi)**4 -30*cos(phi)**2+3)*(Re/r)**4)/r**2

    g_north=3*mu*sin(phi)*cos(phi)*(Re/r)*(Re/r) *(J2+0.5*J3*(5*cos(phi)**2-1)*
                        (Re/r)/cos(phi)+(5/6)*J4*(7*cos(phi)**2-1)*(Re/r)**2)/r**2

    return [g_center, g_north ]
```

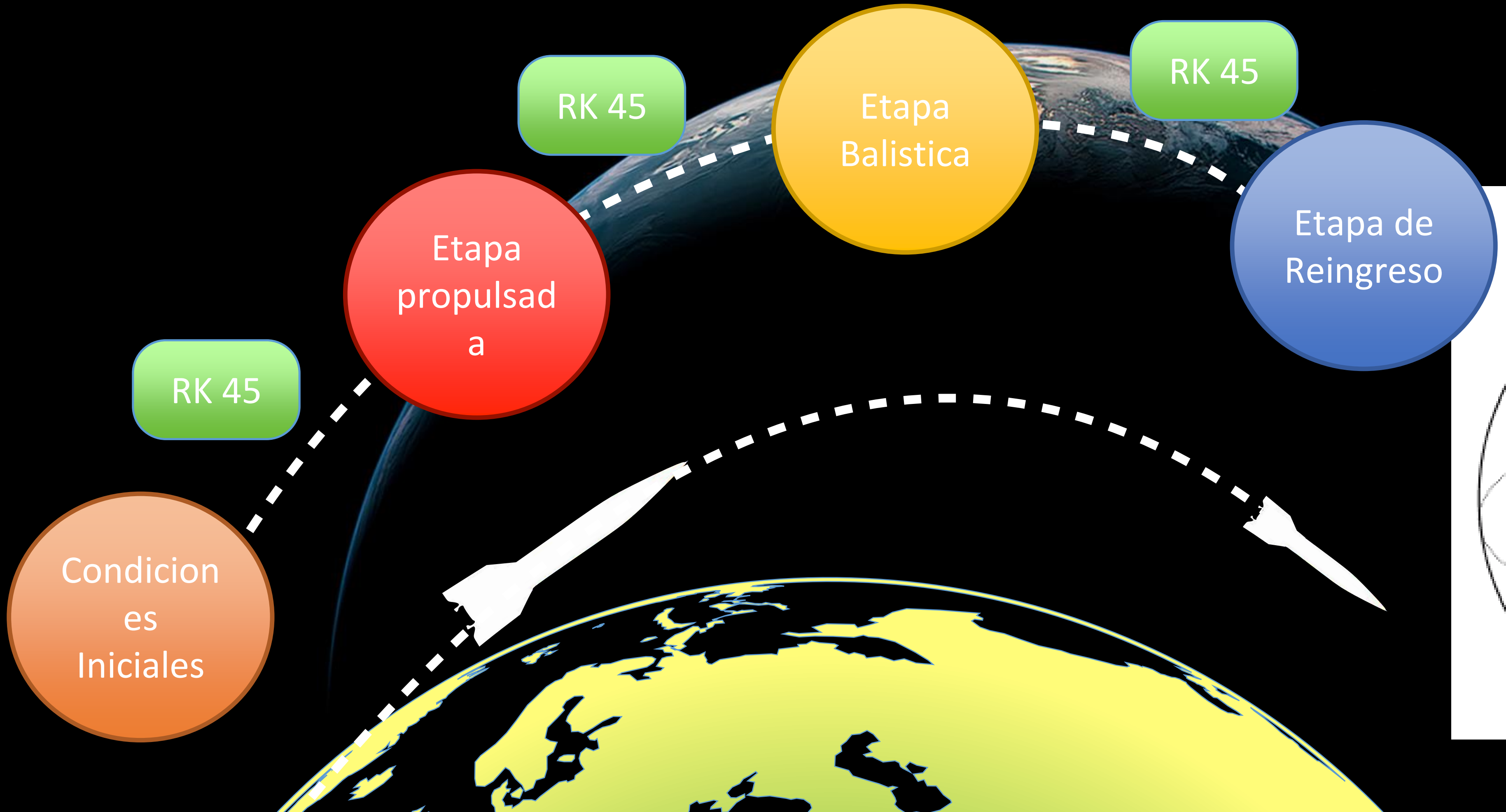

Modelo
Atmosférico
Drag
Weight
Thrust

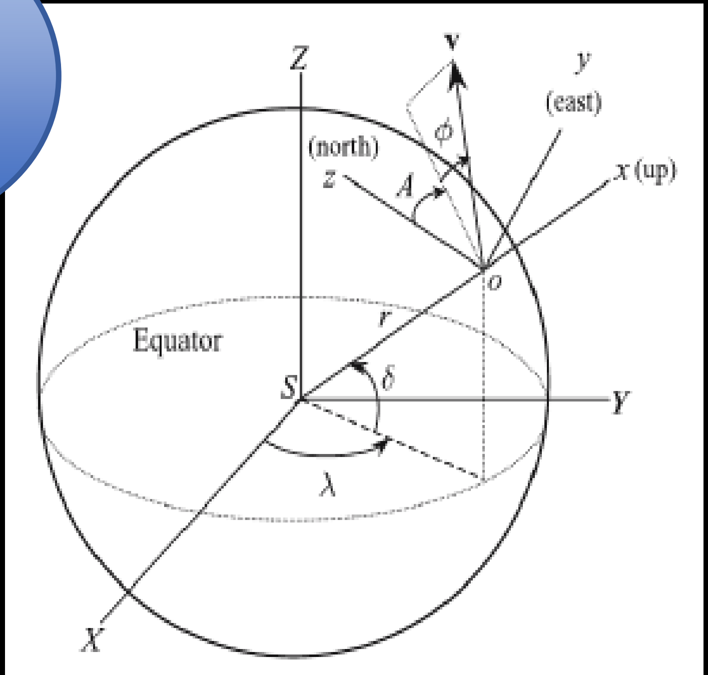

```
Lc = Lc_fustrum
h = r-R
Y = atmosphere(h,Lc)
rho = Y[0]
v_sound = Y[1]
Knudsen = Y[2]
Ma = v/v_sound
s = Ma*sqrt(gamma/2)
continuum_flow = 0.01
molecular_flow = 10

if Ma < subs:
    Cd_cont = a
elif Ma < trans:
        Cd_cont = -b+c*Ma
else:
    Cd_cont = d+e/sqrt(Ma**2-f)

if Knudsen <= continuum_flow:
    Cd = Cd_cont
else:
    Cd_freemol = 1.75+sqrt(pi)/(2*s)
    if Knudsen > molecular_flow:
        Cd = Cd_freemol
    else:
        Cd = Cd_cont+(Cd_freemol-Cd_cont)*(0.333*log10(Knudsen/sin(pi/6))+0.5113)

D = (1/2)*rho*(v**2)*A_fustrum*Cd
```

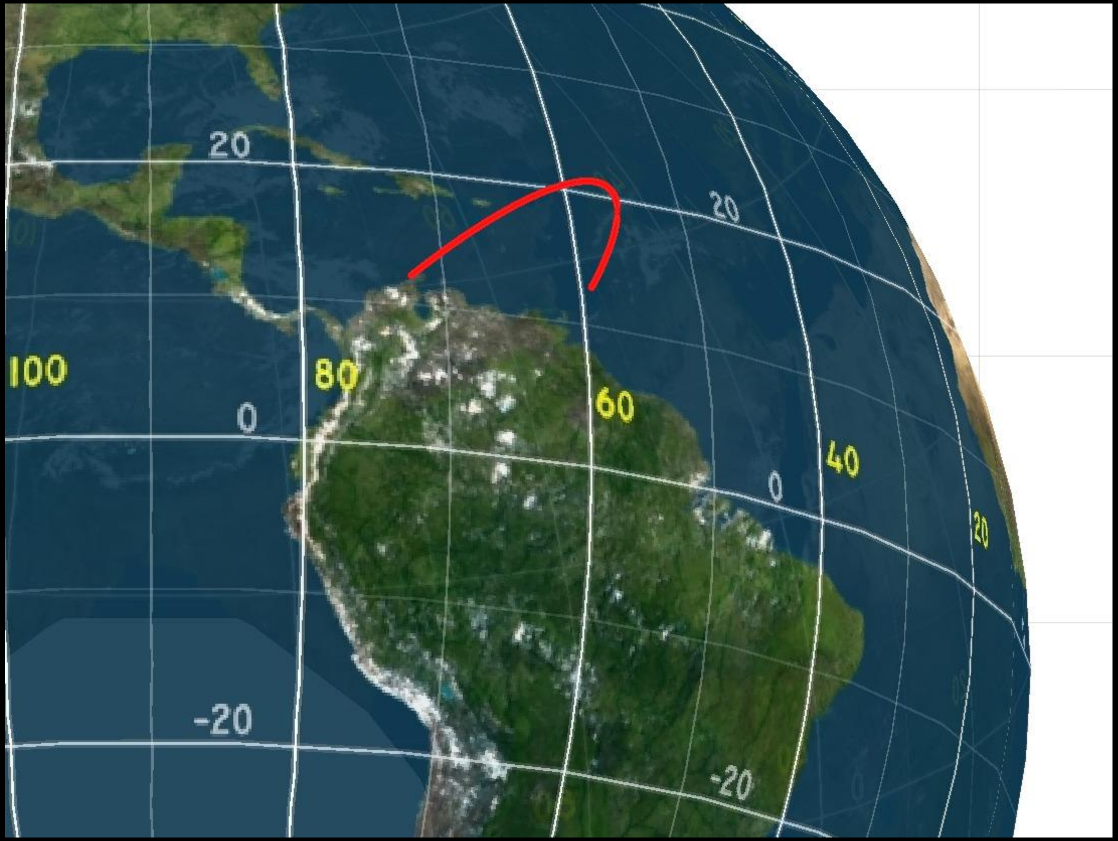
Runge Kutta 45

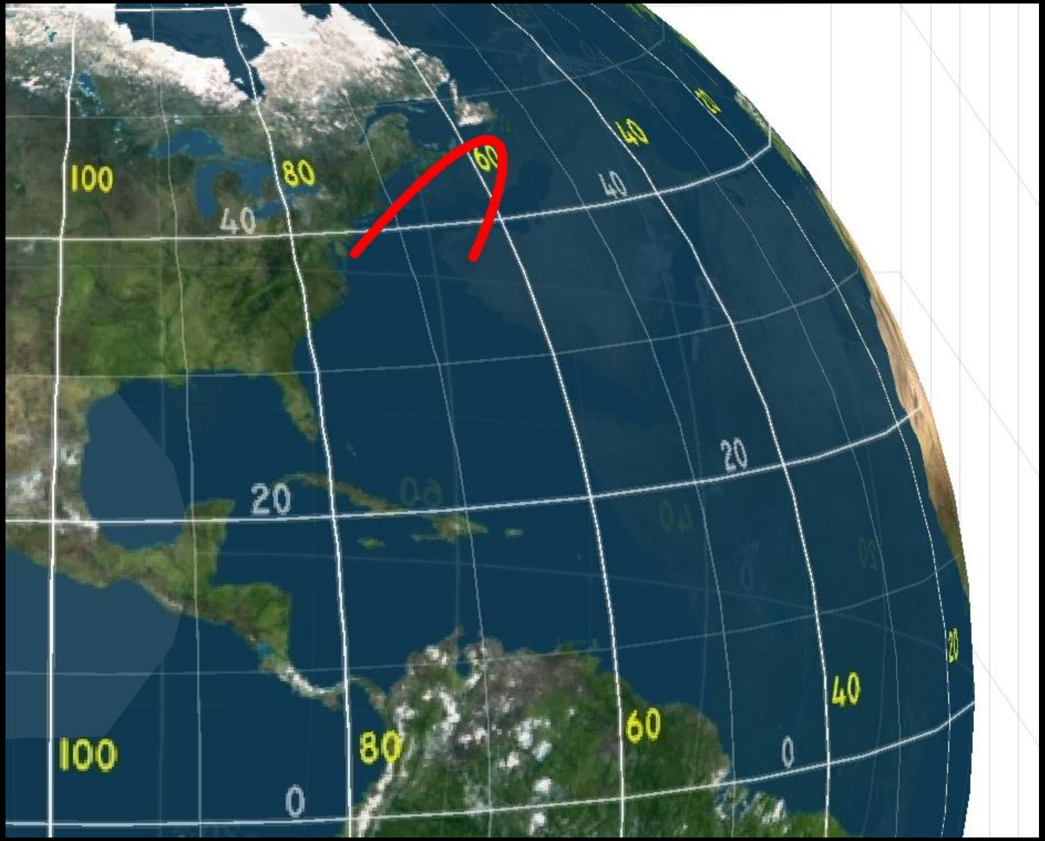
(Método de integración para ODE´s)

$$[\, v(t)\, , A(t)\, ,\ \varphi(t)\, , r(t)\, ,\ \lambda(t)\, , \delta(t)\,]$$









Latitud= 11.71°
Longitud = -72.27°

Latitud = 37.940194°
Longitud = -75.466389°

DEBRISK