

Weapon Systems Homework 3

Aidan Sharpe

February 17th, 2025

1. Rocket Motor Equation

```
def specific_impulse(v_burnout, w_launch, w_burnout):
    return v_burnout / (g*np.log(w_launch/w_burnout))

def main():
    v_burnout = 1000
    w_rocket = 300
    w_propellant_0 = 800

    w_launch = w_rocket + w_propellant_0

    I_sp = specific_impulse(v_burnout, w_launch, w_rocket)
    v_exit = I_sp * g

    print("I_sp:", I_sp)
    print("V_e:", v_exit)

    # Assume constant weight flow rate
    a_burnout = 18*g
    weight_flow_rate = a_burnout*w_rocket/v_exit

    dt = 0.1
    t = np.arange(0, 12, dt)

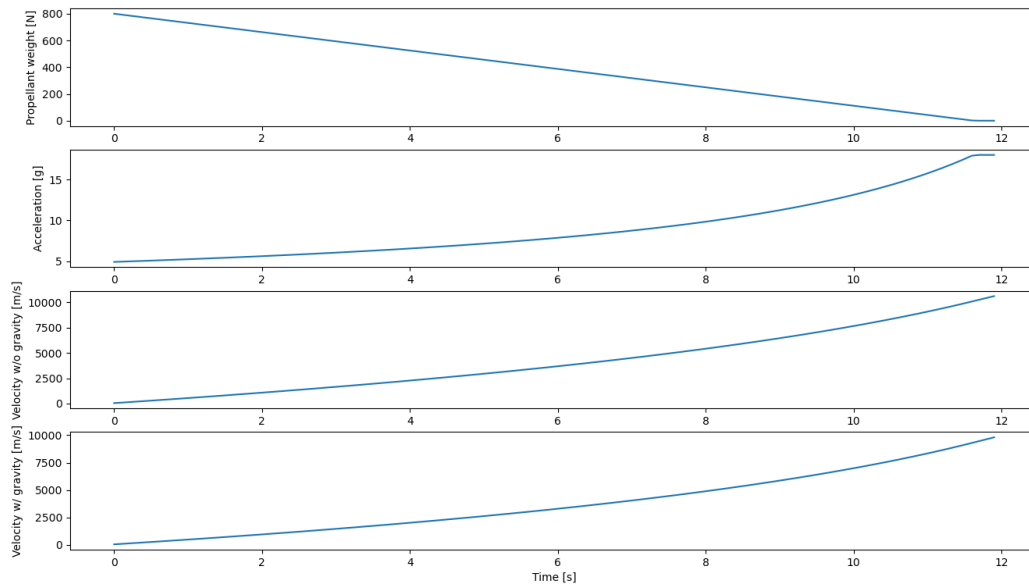
    w_propellant = w_propellant_0 - weight_flow_rate*t
    w_propellant = np.maximum(w_propellant, 0)

    thrust = weight_flow_rate*v_exit/g
    w_total = w_rocket + w_propellant
    # Acceleration in g is force/weight
    acceleration_g = thrust/w_total

    v_no_gravity = np.cumsum(acceleration_g*g)
    v_gravity_45_x = v_no_gravity*np.cos(np.pi/4)
    v_gravity_45_y = v_no_gravity*np.sin(np.pi/4) - np.cumsum(g*np.ones_like(acceleration_g))
    v_gravity_45 = np.sqrt(v_gravity_45_x**2 + v_gravity_45_y**2)
```

$$I_{SP} = 78.46$$

$$V_e = 769.66$$



2. Projectile with Attack Angle

I have been re-writing the math from the cannonball exercise to make it easier to expand in the future. For this reason, I am still working on my simulation routine:

```
def simulate(mass, position, velocity, acceleration, attack_angle):
    altitude = position[1]
    speed = np.linalg.norm(velocity, 2)

    Mach = atmosphere.Mach(altitude, speed)
    axial_drag = np.array([-axial_drag.CA(Mach), 0])

    dynamic_pressure = atmosphere.dynamic_pressure(altitude, velocity)

    elevation = np.atan(velocity[1]/velocity[0]) + attack_angle

    gravity = mass*g*np.array([np.cos(-angle-np.pi/2), np.sin(-angle-np.pi/2)])

    normal_force = np.zeros(2)
    normal_force[1] = TARGET_ACCEL - np.linalg.norm(axial_drag + gravity, 2)
```