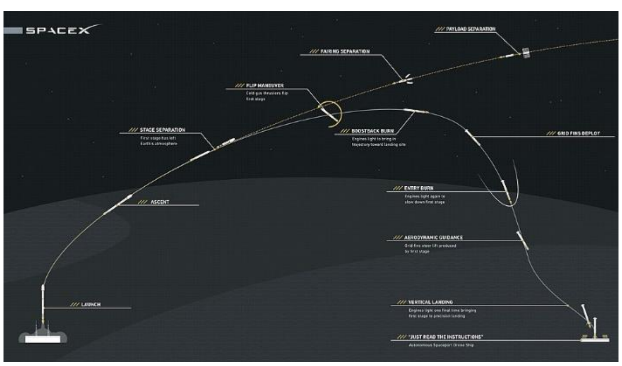
Reusable Rocket Launch and Landing Dynamic Simulation

Samir Yunus

Introduction

This project models the ascent, gravity-turn maneuver, ballistic trajectory, and controlled descent of a rocket using MATLAB and Simulink. The simulation incorporates thrust generation with fuel-mass depletion, aerodynamic drag computed using the ISA atmosphere model, full rocket equations of motion, and gravity-turn guidance driven by a custom pitch-command logic block and a first-order pitch response model for stable tracking. Real-time 3D visualization is achieved through the FlightGear simulator using a UDP external FDM interface. Together, these components approximate the major phases of a Falcon 9–style reusable booster trajectory, including ascent, the flip maneuver, boost-back and entry phases, and controlled landing..

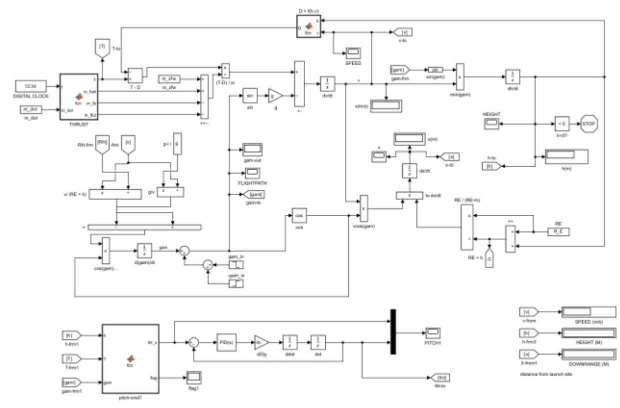
Rocket Flight Profile



1. **SpaceX Trajectory Reference Image** *Reference Flight Profile [1]*

This diagram, from SpaceX [1], illustrates the major phases of a reusable rocket booster mission, including ascent, stage separation, the flip maneuver, boost-back and entry burns, aerodynamic guidance during descent, and vertical landing. This reference profile served as conceptual guidance for designing the rocket's control logic and sequencing within the simulation, ensuring that the modeled behavior reflects the key operational stages of modern reusable launch vehicles.

Rocket Dynamics Model



2. **Simulink Model Overview** *System Architecture*

The simulation uses the standard 2D point-mass rocket dynamics.

Velocity

$$\frac{dv}{dt} = \frac{T - D}{m} - g\sin(\gamma)$$

Altitude

$$\frac{dh}{dt} = v\sin(\gamma)$$

Downrange

$$\frac{dx}{dt} = \left(\frac{R_E}{R_E + h}\right)v\cos(\gamma)$$

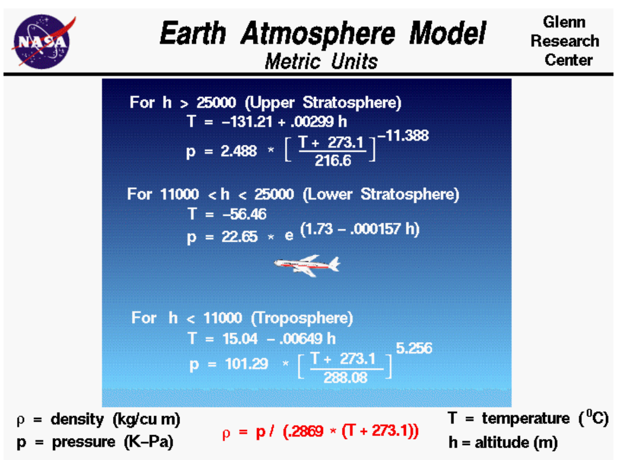
Flight Path Angle

$$\frac{d\gamma}{dt} = \left(\frac{v}{R_E + h} - \frac{g}{v}\right)\cos(\gamma)$$

These models account for gravity, planetary curvature, thrust, aerodynamic drag, and the kinematics of motion in the vertical plane, allowing the simulation to capture the essential physical forces influencing a rocket's trajectory.

Aerodynamic Drag Model (ISA)

The ISA model provides temperature and pressure curves across atmospheric layers



3. NASA ISA Atmosphere Reference Image
Atmospheric temperature and pressure lapse rates used for density and drag modeling [2]

Density is computed as:

```
rho = p / (0.2869 * (T + 273.1));

D = 0.5 * rho * v^2 * A * CD;
```

Three atmospheric regions are modeled in the simulation: the troposphere (h < 11 km), the lower stratosphere (11–25 km), and the upper stratosphere (h > 25 km).

Thrust Model & Fuel Mass Burn

The thrust block uses persistent variables to track mass consumed during ascent and landing

```
persistent mfb mfb2

if mfb <= 0.65 * mfuel          % First 65%
for ascent

    THR = 346961.2;

    mfb = t * m_dot;

else

    THR = 0;

end


% Landing burn between 125-295 s

if (125 < t) && (t < 295)

    THR = -110000;

    mfb2 = (t - 125) * (m_dot / 3);

end
```

Phase 1: Takeoff

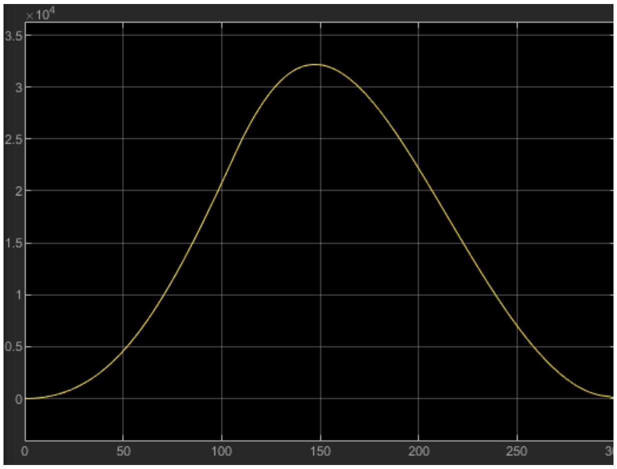
Uses 65% of total fuel | Maximum thrust

Phase 2: Landing

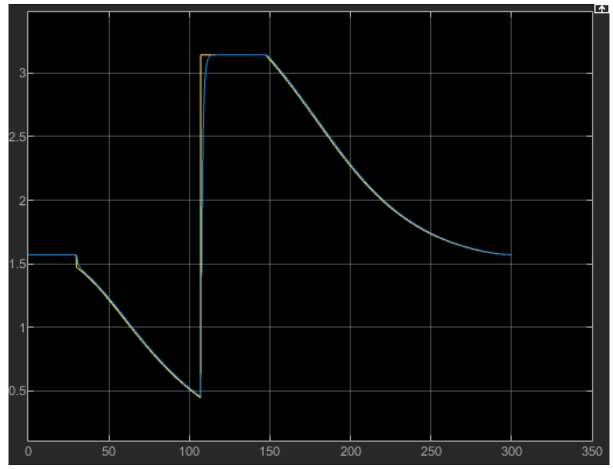
Remaining 35% | 1/3 mass flow rate | Retro-burn and landing burn

Gravity Turn & Pitch Command Logic

The pitch command function models several key behaviors throughout the rocket's flight. During vertical ascent, the pitch angle remains aligned with the flight path angle ($\gamma \approx 90°$). As the vehicle accelerates downrange, the guidance initiates a gravity turn, gradually decreasing the pitch angle to optimize the trajectory. When thrust reaches zero at main engine cutoff, the controller commands a flip maneuver to reorient the rocket for the return phase. Finally, during descent, the pitch command adjusts the vehicle's attitude to achieve a stable landing orientation suitable for entry, aerodynamic control, and terminal descent.



4. Height Plot *Altitude vs Time*



5. Pitch Tracking Plot *Pitch Response During Gravity Turn*

```
if T <= 0
    t_c = gam + pi;         % flip rocket 180
deg
    if t_c > pi
        t_c = pi;
    end
elseif T<=0 && h < 1000
    t_c = pi/2;            % vertical landing
posture
else
    t_c = gam;                % track flight
path
end
```

Pitch Control System

The pitch dynamics are linearized about the vehicle's body y-axis as

$$\frac{d^2\theta}{dt^2} = \frac{T_L\, d_2}{J_y},$$

so in the Laplace domain the plant from control torque $T_L(s)$ to pitch angle $\Theta(s)$ is

$$\frac{\Theta(s)}{T_L(s)} = \frac{d_2/J_y}{s^2}.$$

Using the parameters $d_2 = 5$ m and $J_y = 200$ kg\m^{2}gives

$$\frac{d_2}{J_y} = \frac{5}{200} = 0.025.$$

This is implemented in Simulink with the d2/Jy gain block set to 5/200. The effective open loop transfer function from commanded torque $T(s)$ to pitch angle $\Theta(s)$ is then approximated as

$$\frac{\Theta(s)}{T(s)} \approx \frac{0.04}{s^2}.$$

To shape the response, a pure D-controller is used:

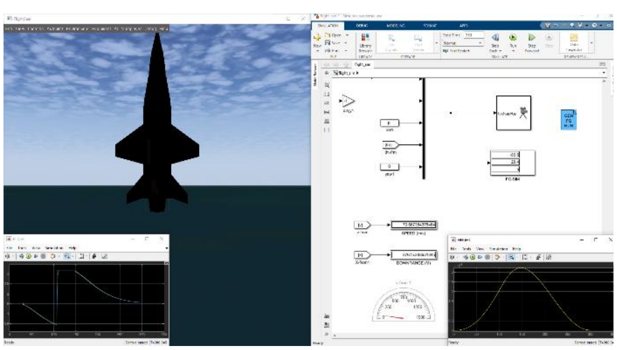
$$C(s) = 25s.$$

In Simulink this is realized with a PID block where $P = 0$, $I = 0$, $D = 200/5$, and a separate gain of $5/200$, giving the same overall controller $C(s) = 25s$.

With unity feedback, the resulting closed-loop transfer function from pitch command $\Theta_c(s)$ to actual pitch $\Theta(s)$ becomes

$$\frac{\Theta(s)}{\Theta_c(s)} = \frac{C(s)\,\Theta(s)/T(s)}{1 + C(s)\,\Theta(s)/T(s)} = \frac{1}{s+1}$$

This is a first-order system with time constant 1 s and essentially no overshoot, which matches the smooth pitch-tracking behavior seen in the PITCH scope.

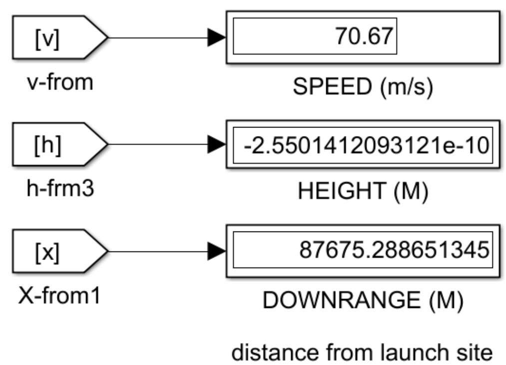


6. FlightGear Visualization *Real-Time Visualization (FlightGear)*

The simulation streams position and attitude to FlightGear using a UDP external FDM interface:

- Position: x, z
- Attitude: roll, pitch, yaw
- Altitude: converted to North-East-Down (NED) frame

This enables a complete real-time 3D visualization of the rocket's trajectory.



7. Speed, Height, and Downrange Indicators *Real-time outputs from the simulation showing vehicle speed, altitude, and horizontal distance traveled from the launch site.*

Conclusion

This project models the ascent, gravity-turn maneuver, ballistic trajectory, and controlled descent of a rocket using MATLAB and Simulink. The simulation incorporates thrust generation with fuel-mass depletion, aerodynamic drag based on the International Standard Atmosphere (ISA) model, and full rocket equations of motion. Guidance is implemented through a gravity-turn strategy supported by a custom pitch-command logic block and a first-order pitch response model that ensures stable tracking. Real-time 3D visualization is achieved through the FlightGear simulator using a UDP external FDM interface. Together, these elements approximate the major phases of a Falcon 9-style reusable booster, including ascent, stage-flip maneuver, boost-back and entry behavior, and controlled landing.

References

[1] SpaceX. (n.d.). *Falcon 9 autonomous droneship landing flight profile* [Image]. Retrieved December 4, 2025, from https://www.spacex.com/assets/images/mission/F9_AUTONOMOUS_DRONESHIP_DESKTOP.jpg

[2] NASA Glenn Research Center. (n.d.). *Earth atmosphere model*. NASA. Retrieved December 4, 2025, from https://www.grc.nasa.gov/WWW/BGH/atmosmet.html

[3] VDEngineering. (n.d.). *Home* [YouTube channel]. YouTube. Retrieved December 4, 2025, from https://www.youtube.com/@VDEngineering