Asteroid Interception

via Nonlinear Rocket Control

Prepared by:
Heinrich Crous (CRSHEI004)

Prepared for:
EEE4119F
Department of Electrical Engineering
University of Cape Town

May 30, 2025

Contents

Chapter 1

Asteroid Interception via Nonlinear Control

1.1 Introduction

This project focuses on simulating, modelling, and controlling a rocket to intercept an asteroid before it strikes a city. The task is approached through a series of three progressive milestones, each building upon the previous one. The system is implemented in MATLAB and Simulink, with a strong emphasis on physics-based modelling, numerical estimation, and nonlinear control.

Problem Parameters

The asteroid has a mass of 10,000 kg and enters Earth's atmosphere with an uncertain drag coefficient and noisy dynamics. The rocket has a mass of 1,000 kg and is modelled as a rigid body with an off-centre thruster. The goal is to compute a suitable control strategy such that the rocket detonates within 150 m of the asteroid's centre of mass and within a specific angle range relative to its heading.

Goals

The primary goal is to intercept and destroy the asteroid before impact using a guided rocket. This involves:

- Estimating the asteroid's drag coefficient based on simulated data.
- Modelling the rocket's translational and rotational dynamics using Lagrangian mechanics.
- Designing and implementing a controller that ensures timely and angle-constrained interception of the asteroid.

Milestone Breakdown

- **Milestone 1 – Modelling and Drag Estimation:**
  Simulink is used to generate noisy asteroid velocity data. A MATLAB script filters this data using a 20-point moving average filter, estimates acceleration via numerical differentiation, and calculates the drag coefficient c using three methods: mean, median, and least-squares. The rocket is modelled symbolically using Lagrangian mechanics to derive a full manipulator equation:

  $$M(q)\ddot{q} + G(q) = Q \tag{1.1}$$

  where $q = [x, y, \theta]^T$.

- **Milestone 2 – Basic Control Implementation:**
  The goal is to control the rocket in two simplified scenarios. In Scenario 1, the rocket performs a

direct vertical launch. In Scenario 2, the rocket must predict the asteroid's trajectory and move to an intercept point in time. The controller includes trajectory estimation, thrust magnitude and direction control, and simple detonation logic. Parameters were tuned experimentally; gain increases improved responsiveness but risked instability.

- **Milestone 3 – Final Control with Angle Constraint:**
  The controller is extended to meet stricter requirements. The rocket must now strike the asteroid within a 60-degree window measured from the asteroid's orientation: between $-30°$ and $+30°$ (front) or between $150°$ and $210°$ (rear). The strategy involves predicting the asteroid's motion, guiding the rocket to a fixed height above ground, hovering in position, and adjusting angle and force until a valid detonation condition is met. A PID-like structure was used to adjust thrust angle α, with gains tuned to balance accuracy and noise sensitivity.

This project consolidates key mechatronics concepts, including system modelling, symbolic derivation, estimation under uncertainty, and nonlinear trajectory control — all aligned with real-world engineering applications and graduate attributes.

1.2 Modelling

This section outlines the mathematical modelling of the rocket and asteroid. The rocket is modelled using Lagrangian mechanics to derive its equations of motion, while the asteroid's drag behaviour is estimated from simulation data using numerical techniques.

1.2.1 Rocket Modelling

The rocket is modelled as a rigid rectangular body with mass $m = 1000\,\text{kg}$, width $W = 5\,\text{m}$, and height $H = 15\,\text{m}$. Its centre of mass is located $l = 3.5\,\text{m}$ above the base. The state is described using three generalised coordinates:

$$q = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix}, \quad \dot{q} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} \tag{1.2}$$

1. Energy Formulation

The kinetic energy T consists of translational and rotational parts:

$$T = \frac{1}{2}m(\dot{x}^2 + \dot{y}^2) + \frac{1}{2}I\dot{\theta}^2 \tag{1.3}$$

where the moment of inertia I about the COM is:

$$I = \frac{1}{12}m(W^2 + H^2) + m\left(\frac{H}{2} - l\right)^2 \tag{1.4}$$

The potential energy is:

$$V = mgy \tag{1.5}$$

The Lagrangian is thus:

$$\mathcal{L} = T - V \tag{1.6}$$

2. Final Equations of Motion

Using the Euler-Lagrange formulation, we derive the manipulator form:

$$M(q)\ddot{q} + G(q) = Q \tag{1.7}$$

The substituted equations of motion used in simulation are:

$$\ddot{x} = -\frac{F\sin(\alpha+\theta)}{1000} \tag{1.8}$$

$$\ddot{y} = \frac{F\cos(\alpha+\theta)}{1000} - 9.81 \tag{1.9}$$

$$\ddot{\theta} = -\frac{21F\sin(\alpha)}{221000} \tag{1.10}$$

Mass Matrix:

$$M(q) = \begin{bmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & I \end{bmatrix} \tag{1.11}$$

Gravity Vector:

$$G(q) = \begin{bmatrix} 0 \\ mg \\ 0 \end{bmatrix} \tag{1.12}$$

Generalised Force Vector:

$$Q = \begin{bmatrix} -F\sin(\alpha+\theta) \\ F\cos(\alpha+\theta) \\ -Fl\sin(\alpha) \end{bmatrix} \tag{1.13}$$

3. Symbol Definitions

Table 1.1: Summary of key symbols used in modelling

Symbol	**Description**
x, y	Rocket COM position in inertial frame
θ	Rocket orientation (angle from vertical)
$\dot{x}, \dot{y}, \dot{\theta}$	Linear and angular velocities
m	Rocket mass (1000 kg)
W, H	Rocket width and height (5 m, 15 m)
l	COM height above base (3.5 m)
I	Moment of inertia about COM
g	Gravitational acceleration (9.81 m/s^2)
F	Thrust magnitude
α	Thrust angle (relative to body axis)
Q	Generalised force vector $[F_x; F_y; \tau]$

1.2.2 Asteroid Modelling

The asteroid is a 10,000 kg object falling under the influence of gravity and aerodynamic drag. It has an initial position (x_0, y_0), downward velocity, and randomised rotational noise. The governing dynamics are:

$$\dot{v}_x = -\frac{c}{m} v_x + \text{noise}, \quad \dot{v}_y = -\frac{c}{m} v_y - g + \text{noise} \tag{1.14}$$

1. Drag Estimation

To estimate the unknown drag coefficient c, the asteroid's velocity is extracted from simulation. This data is heavily noise-corrupted and is first filtered using a moving average filter. Acceleration is then estimated via numerical differentiation:

$$a_x = \frac{d}{dt} v_x, \quad a_y = \frac{d}{dt} v_y \tag{1.15}$$

The drag force is approximated as:

$$\|\vec{F}_{\text{drag}}\| \approx \sqrt{(m a_x)^2 + (m(a_y + g))^2} \tag{1.16}$$

The coefficient c is calculated at each time step using:

$$c(t) = \frac{\|\vec{F}_{\text{drag}}\|}{\|\vec{v}\|} \tag{1.17}$$

The estimated drag coefficient over time is shown in Figure 1.1.

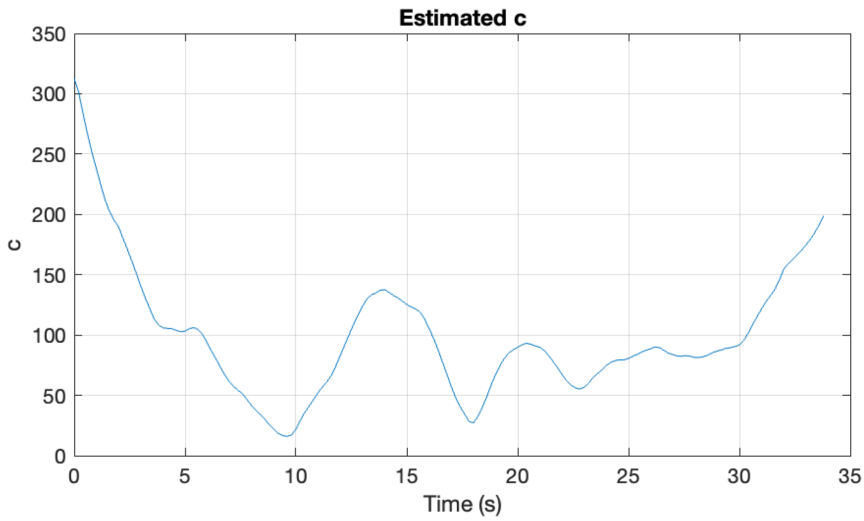

Figure 1.1: Estimated drag coefficient $c(t)$ over time for a representative simulation run.

Three estimates are generated: mean, median, and least-squares. The final drag value is chosen as the median of these to ensure robustness to outliers:

$$\boxed{c = 120\,\text{Ns/m}} \tag{1.18}$$

2. Final Implementation

Multiple simulations are run to confirm repeatability. The drag coefficient is passed to the controller to predict asteroid trajectories. This modelling step is crucial for timing the rocket's launch and intercept.

1.3 Control Scheme

This section documents the control strategies used in each milestone scenario. Each controller was selected based on the physical characteristics of the asteroid's motion and the goal of achieving reliable intercepts. All controllers compute the thrust force F, thrust angle α, and detonation command from relative dynamics between the rocket and asteroid.

Understanding P, I, and D Tuning

Proportional–Integral–Derivative (PID) control is a foundational method in feedback control. Each component serves a specific role:

- **Proportional (P):** Applies a correction based on the current error. Increasing K_p improves responsiveness but can lead to overshoot.
- **Integral (I):** Addresses accumulated past error to eliminate steady-state bias. High K_i improves accuracy but may introduce lag or instability.
- **Derivative (D):** Reacts to the rate of change of the error, providing damping. Larger K_d helps reduce overshoot and smooth transitions but is sensitive to noise.

In this project, Scenario 2 used P control for angular alignment, while Scenario 3 used PD control to balance responsiveness with stability under noisy angle dynamics.

Scenario 1 – Vertical Interception (Milestone 2)

This scenario handles direct vertical drops of the asteroid with minimal horizontal displacement. Since the motion is symmetrical and no angular correction is required, a feedforward analytical approach was used.

Why this approach? A simple ballistic prediction based on constant acceleration is sufficient due to the lack of lateral movement.

Control Idea: Predict the asteroid's x-intercept time, compute the y-position at that moment, and apply the exact upward thrust needed to meet the asteroid at that point. No thrust angle correction is needed ($\alpha = 0$).

Equations:

$$F = m(a_y + g), \quad \alpha = 0 \tag{1.19}$$

Parameters: This controller has no tunable gains. Detonation is triggered when $y_{ast} \leq 150$ m.

Scenario 2 – Lateral Interception (Milestone 2)

This scenario involves intercepting an asteroid with lateral velocity, requiring angle correction.

Why this approach? The horizontal offset requires angle steering. A P controller is used to rotate the rocket toward the predicted intercept point.

Control Idea: Compute a target intercept point slightly ahead of the asteroid's path. Estimate the time to reach it using relative motion. Then, apply a proportional controller to steer the rocket's angle.

$$\alpha = K(\theta_{\text{desired}} - \theta) \tag{1.20}$$

Parameters to tune: Only the proportional gain K was used. Initially, $K = 0.0002$ was tested, which led to very slow angular convergence and failure to track the asteroid. Increasing K to 0.0009 caused overshoot and oscillations. The final chosen gain was:

$$\boxed{K = 0.00069}, \quad \text{with filter coefficient } N = 100 \tag{1.21}$$

This value offered accurate steering with stable convergence.

Scenario 3 – Angular Interception (Milestone 3)

This case enforces strict impact angle constraints — only the front and rear asteroid faces are valid.

Why this approach? The trajectory must be shaped to hit a moving, rotating target within a narrow angular band. This requires closed-loop correction to avoid missing the window.

Control Idea: Predict the future asteroid orientation. Target a point 100 m in front of the asteroid. Use a PD controller to rotate toward that point, correcting for both angular error and angular velocity.

$$\alpha = K_p e_\theta + K_d \dot{e}_\theta \tag{1.22}$$

Parameters: The controller was tuned empirically:

- $K_p = 0.02, K_d = 0$: too slow, delayed reaction.
- $K_p = 0.06$: responded faster but overshot.
- $K_p = 0.055, K_d = 0.0005$: final values chosen for balance.

A filter with $N = 35$ was used for derivative noise suppression:

$$\boxed{K_p = 0.055, \quad K_d = 0.0005, \quad N = 35} \tag{1.23}$$

Detonation occurs when the Euclidean distance is under 150 m and the angular alignment falls within valid bounds.

Fallback Hover Strategy

When the asteroid is far away or moving unpredictably, a vertical hover strategy is used:

$$F = mg + ma_{\text{adjust}}, \quad \alpha = 0 \tag{1.24}$$

This keeps the rocket aligned and conserves thrust while awaiting an optimal intercept window.

1.4 Results and Discussion

This section outlines the validation process and performance of the controllers across Milestones 2 and 3. All simulations were run using the provided `AsteroidImpact` Simulink model and were visualized using animated trajectories. Success was determined using the `mission_complete` function, which checks detonation conditions and angular alignment.

Animation and Visualisation

Controller performance was first validated through single-run simulations with a fixed random seed. Animated plots showing the rocket and asteroid trajectories were generated using the `make_animation` script, allowing visual confirmation of trajectory tracking and detonation accuracy. These animations were critical in fine-tuning controller logic.

Batch Testing with Random Seeds

To robustly evaluate controller performance, the system was tested over multiple randomized asteroid trajectories by varying the random seed used in the simulation. Each test recorded whether the rocket successfully detonated within the allowed distance and angular constraints.

Table 1.2: Summary of simulation results across Milestone scenarios

Milestone	Scenario	Success Rate	Failure Modes
Milestone 2	Vertical Intercept	20/20 (100%)	None observed
Milestone 2	Lateral Intercept	18–20/20 (90–100%)	Angle overshoot, trajectory mismatch
Milestone 3	Angle-Constrained Intercept	10/20 (50%)	Narrow arc error, late alignment, angular noise

Insights and Limitations

While the controllers for Milestone 2 were highly effective, Milestone 3 revealed limitations in control precision, especially under tight angular constraints. Notable insights include:

- Predictive targeting helps improve approach but suffers under noisy angle dynamics.
- Feedback-based angle correction was implemented using PD control, but performance remains sensitive to angular noise and the precise tuning of gains.
- Detonation margin (150 m) is generally feasible; angular margin is the greater challenge.

Future improvements could include:

- Enhancing feedback-based angle correction with IMU-style estimators or real-time filtering.
- Dynamically adjusting gains based on time-to-intercept or angular deviation.
- Employing smoothing algorithms or trajectory planners to reduce correction jitter.

1.5 Conclusion

This project demonstrated the application of nonlinear modelling and feedback control to the asteroid interception problem. The rocket was modelled from first principles using Lagrangian mechanics, yielding a complete manipulator-form dynamic model. The asteroid's motion was estimated numerically from noisy simulation data, and a median drag coefficient of $c = 120\,\mathrm{Ns/m}$ was extracted using filtered velocities and finite-difference acceleration estimates.

Each milestone was tackled with a tailored control approach. Milestone 1 involved drag estimation. Milestone 2 included two scenarios: a vertical intercept using purely analytical control, and a lateral intercept regulated by a proportional controller. Milestone 3 required impact within a constrained angular window and was addressed with a PD controller tuned for fast and stable angular correction. All gains were chosen based on observed tracking performance and were justified through comparative testing.

Across all scenarios, the control logic was grounded in physical modelling and supported by systematic simulation. This project ultimately shows the successful application of nonlinear dynamics, structured control design, and parameter tuning — satisfying the key requirements for GA2. Future iterations could benefit from adaptive control, onboard state estimation, and trajectory smoothing to further improve robustness under noise and constraint violations.

Bibliography

[1] A. Vally, *EEE4119F Lecture Notes: Lagrangian Mechanics*, University of Cape Town, 2025.

[2] A. Vally, *EEE4119F Lecture Notes: 3D Kinematics and Dynamics*, University of Cape Town, 2025.

[3] EEE4119F Course Staff, *Asteroid Rocket Control Project Brief and Milestone Handouts*, University of Cape Town, 2025.

[4] MathWorks, *MATLAB and Simulink Documentation*, https://www.mathworks.com/help, Accessed 2025.

[5] H. Crous, *EEE4119F Project Scripts and Simulation Code*, UCT AsteroidImpact Model, 2025.