doi: 10.82124/CEAS-GNC-2026-035 CEAS-GNC-2026-035

Augmented Pursuit Guidance Based Optimal Landing Trajectory Tracking for Reusable VTVL Rocket

Da-In Jeong M.S. Candidate, Korea Advanced Institute of Science and Technology, Department of Aerospace Engineering, 34141, Daejeon, South Korea. dainj0124@kaist.ac.kr

Min-Jea Tahk Professor Emeritus, Korea Advanced Institute of Science and Technology, Department of Aerospace Engineering, 34141, Daejeon, South Korea. mj-tahk317@gmail.com

Jae-Il Jang Ph.D. Student, Korea Advanced Institute of Science and Technology, Department of Aerospace Engineering, 34141, Daejeon, South Korea. jji249@kaist.ac.kr

Chang-Hun Lee Associate Professor, Korea Advanced Institute of Science and Technology, Department of Aerospace Engineering, 34141, Daejeon, South Korea. lckdgns@kaist.ac.kr

ABSTRACT

This study addresses the trajectory tracking problem for vertical landing of reusable launch vehicles. To overcome the computational complexity of real-time optimization methods and the instability of time-based reference tracking, an Augmented Pursuit Guidance (APG)-based guidance and control method is proposed. It improves trajectory alignment through a modified APG, and removes residual errors by applying a velocity controller and a longitudinal PD controller in the low-altitude phase. For validation, an optimal trajectory is precomputed, and the proposed method is then applied to track it in simulations. The results confirm that the proposed method achieves stable and fuel-efficient landings under both nominal and disturbed conditions. In conclusion, this work shows that sufficient trajectory tracking performance and landing accuracy can be achieved with a simple algorithm.

Keywords: Vertical Takeoff and Landing(VTVL), Trajectory Tracking, Optimal Trajectory, Augmented Pursuit Guidance(APG), Terminal Control

1 Introduction

Vertical Takeoff and Landing (VTVL) reusable launch vehicles have gained attention as a key technology for reducing launch costs and improving mission efficiency. The landing problem of such vehicles is a highly challenging task, as it requires minimizing fuel consumption and control effort while satisfying strict terminal conditions on altitude, velocity, and attitude [1]. In real flights, precomputed optimal trajectories are difficult to follow due to mass variation, thrust and angle-of-attack constraints, and model uncertainties. Therefore, the development of guidance and control algorithms that can generate trajectories and track them stably has emerged as a critical technical challenge.

In this regard, real-time optimization techniques have been actively studied. In particular, convex optimization approaches have been widely investigated because they can satisfy state and input constraints while guaranteeing global or near-optimal solutions [2]. Sequential Convex Programming (SCP) has also been proposed to update references in real time and track them, and it has been shown to work effectively

even in complex flight scenarios [3], [4], [5]. However, convex optimization approaches are based on iterative computations, which incur high computational costs and may produce infeasible solutions during approximation, leading to limitations in onboard applicability [6], [7].

As an alternative to real-time optimization, guidance and control methods that track offline precomputed optimal trajectories have also been studied. This approach is computationally efficient, but most references have been defined as functions of time or other predefined variables such as energy [8], [9]. When actual flight conditions deviate from the planned profile, such as due to thrust performance errors or unexpected disturbances, the vehicle may follow incorrect points on the reference trajectory. This can lead to performance degradation and instability [10], [11]. To address this problem, a free-terminal-time approach has recently been proposed, in which the final time is not fixed in advance [12]. However, this method still requires onboard optimization and has a complex structure.

In this study, a lightweight trajectory-tracking guidance and control method based on Augmented Pursuit Guidance (APG) is proposed as a computationally simple and robust alternative. APG is known as a position-reference-based method with a simple structure. To further improve alignment, a modified APG is also employed. However, APG alone leaves residual errors in the terminal phase. To overcome these issues, a velocity controller and a low-altitude lateral PD controller are additionally designed. In addition, by applying a realistic acceleration allocation that considers actuator saturation, stable trajectory tracking can be achieved in real flight environments. Here, the reference trajectory tracked by APG is not a simple geometric path but an optimal trajectory precomputed to minimize fuel consumption and control effort. Thus, the proposed method follows a reference solution that guarantees optimality, but unlike real-time optimization, it does not recompute trajectories onboard and instead reduces trajectory errors progressively through controllers.

As a result, this study demonstrates that accurate and fuel-efficient optimal trajectory tracking can be stably achieved using a simple algorithm without relying on complex optimization solvers. Simulation results also show that the proposed method achieves successful landing with stable convergence even under disturbed initial conditions. This indicates that computational efficiency can be secured while incorporating the physical validity and fuel efficiency of the optimal trajectory into the actual landing control process.

This paper is organized as follows: Section 2 defines the optimal trajectory generation problem and derives the reference trajectory. The APG-based trajectory-tracking method and its modification are described in Section 3. Supplementary controllers for velocity and terminal precision are introduced in Section 4. Section 5 outlines the integrated control logic for each flight phase. Simulation results of the proposed method are presented in Section 6, followed by conclusions in Section 7.

2 Optimal Reference Trajectory Generation

2.1 Model and mission overview for optimization

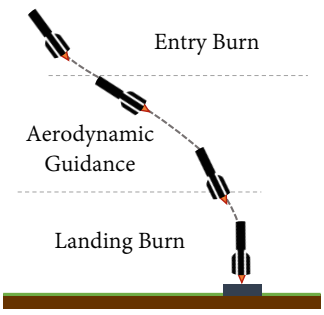


Fig. 1 Phases of the VTVL landing process

This study considers the Return to Launch Site (RTLS) mission segment of a Falcon 9–class reusable first stage [13]. The landing sequence, illustrated in Fig. 1 has three phases. Phase 1 is the Entry Burn (EB) during which three engines reduce orbital energy. Phase 2 is the Aerodynamic Guidance (AG) during which the vehicle descends unpowered and refines the path using lift and drag. Phase 3 is the Landing Burn (LB) during which one engine sets the final landing conditions. The mission geometry is planar in the downrange–altitude plane. Vehicle properties and operational limits are based on the data in [3].

The flight environment uses a standard atmosphere. Air density, speed of sound, and pressure p_{atm} are calculated as functions of altitude y. Gravity is set to a constant value of $g = 9.807\text{m}/s^2$. The aerodynamic forces of the launch vehicle are obtained by precomputed aerodynamic data as follows:

$$F_L = qSC_L(\alpha, M), \quad F_D = qSC_D(\alpha, M) \tag{1}$$

where α is angle of attack, M is Mach number, q is dynamic pressure and S is reference area. The effective thrust of the vehicle is calculated as follows:

$$F_T = n_{\text{eng,i}} \left(\delta_T \, T_{\text{max,vac}} - p_{\text{atm}}(y) A_e \right) \tag{2}$$

where $n_{\text{eng,i}} = \{3, 0, 1\}$ is the number of active engines in each phase, δ_T is throttle, $T_{\text{max,vac}}$ is the vacuum-rated max thrust per engine, and A_e is the nozzle exit area.

2.2 Formulation of the Optimal Control Problem

In this section, a three-phase multi-segment Optimal Control Problem (OCP) is formulated based on the vehicle environment and mission phase models described in Section 2.1. The objective is to obtain a reference trajectory that satisfies the constraints while minimizing the angle of attack and thrust usage.

2.2.1 Dynamics Model

The state and control inputs of the vehicle model are defined as $\mathbf{x} = [x, y, V, \gamma, m]^T$ and $\mathbf{u} = [\alpha, \delta_T]^T$. x is the downrange position, y is altitude (positive upward), V is velocity of the vehicle, γ is flight path angle measured from the vertical axis of the NED frame, and m is mass of the vehicle. The corresponding two-dimensional equations of motion are described as follows:

$$\begin{aligned}
\dot{x} &= V \sin\gamma, \qquad \dot{y} = -V \cos\gamma \\
\dot{V} &= \frac{-F_T \cos\alpha - F_D}{m} + g \cos\gamma \\
\dot{\gamma} &= \frac{F_L - F_T \sin\alpha}{mV} - \frac{g \sin\gamma}{V} \\
\dot{m} &= -\frac{F_T}{I_{sp,vac} g_0}
\end{aligned} \tag{3}$$

2.2.2 Cost Function

The following cost function is defined to minimize the squared sum of the control inputs for each Phase i = $\{1, 2, 3\}$ in order to reduce the maneuvering cost and fuel consumption.

$$J = \sum_{i=1}^{3} \int_{t_{0,i}}^{t_{f,i}} \left(w_{\alpha,i} \alpha^2 + w_{\delta_T,i} \, \delta_T^2 \right) dt \tag{4}$$

2.2.3 Constraints in the Optimization

The constraints used in the optimization process are as follows:

$$\dot{\mathbf{x}}_i = f_i(\mathbf{x}_i, \mathbf{u}_i), \quad i = 1, 2, 3 \tag{5}$$

$$\mathbf{s}_i := \left[x_i,\ y_i,\ V_i,\ \gamma_i\right]^\top, \quad \hat{\mathbf{s}}_{i,0} := \left[\hat{x}_{i,0},\ \hat{y}_{i,0},\ \hat{V}_{i,0},\ \hat{\gamma}_{i,0}\right]^\top$$
$$\mathbf{s}_{i,0} = \hat{\mathbf{s}}_{i,0} \tag{6}$$

$$u_{\min,i} \ \leq \ u_i(t) \ \leq \ u_{\max,i} \tag{7}$$

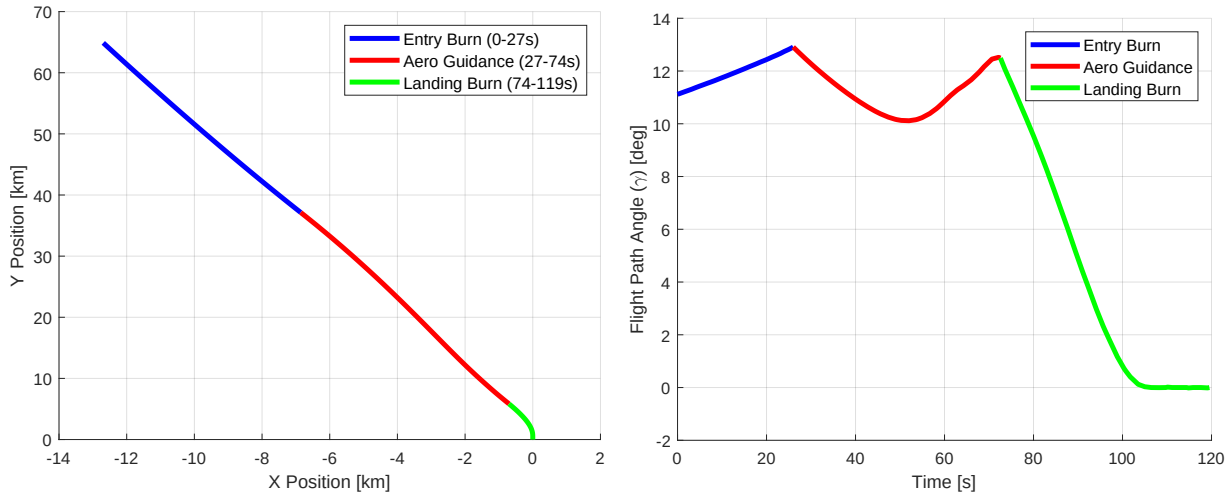
$$x_3(t_{f,3}) = 0, \quad y_3(t_{f,3}) = 0, \quad V_3(t_{f,3}) = 0, \quad \gamma_3(t_{f,3}) = 0 \tag{8}$$

$$\begin{aligned} \mathbf{x}_2(t_{0,2}) &= \mathbf{x}_1(t_{f,1}), \qquad t_{0,2} = t_{f,1} \\ \mathbf{x}_3(t_{0,3}) &= \mathbf{x}_2(t_{f,2}), \qquad t_{0,3} = t_{f,2} \end{aligned} \tag{9}$$

Equation (5) represents the dynamic constraints. Equation (6) constrains the initial state of each Phase, $\mathbf{s}_{i,0}$, to the actual flight data $\hat{\mathbf{s}}_{i,0}$. The flight data is extracted from reconstructed telemetry (video-based trajectory) of the actual Falcon 9[1], NROL-76. The temporal and geometric features of the real vehicle are reflected in the optimization. Equation (7) imposes bound constraints on the angle of attack and throttle for each Phase, in order to reflect operational limits of the real vehicle and safety margins. Equation (8) constrains the final altitude, velocity, and attitude to match the landing conditions, so that the trajectory terminates in a state that enables touchdown. And Eq. (9) enforces state and time continuity across phases. Thus, these exclude physically infeasible minima and ensure that the optimization produces reference inputs that are practically trackable.

2.2.4 Optimal Reference Trajectory

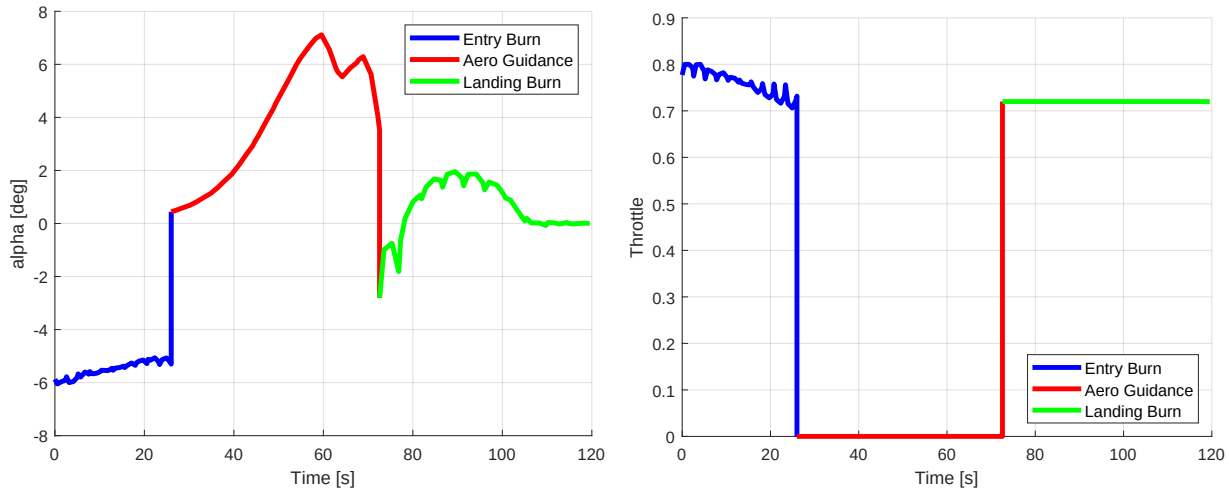
Under these constraints, solving the OCP yields the control inputs and the corresponding state $\mathbf{x}$ that minimize J while satisfying all bounds and constraints. The optimization is performed using GPOPS-II. The results are shown in Fig. 2 and Fig. 3 below. These optimal trajectory and state values are used as the reference for tracking control in Sections 3-5.



(a) Optimal reference trajectory **(b) Optimal flight path angle**

Fig. 2 Optimal trajectory and flight path angle

[1]Telemetry-Data repository, available at `https://github.com/shahar603/Telemetry-Data`, accessed on 25 Sep 2025.



(a) Optimal angle of attack profile **(b) Optimal throttle profile**

Fig. 3 Optimal control inputs: angle of attack and throttle

3 Augmented Pursuit Guidance Based Trajectory Following Method

3.1 Baseline of APG

Augmented Pursuit Guidance(APG) is an algorithm that guarantees target interception by tracking the lead angle. Its basic equation is given as follows:

$$a_{\mathrm{APG}} = NV\left(\lambda_{\mathrm{ref}} - \lambda\right) \tag{10}$$

where a_{APG} acts as the normal acceleration perpendicular to the velocity vector, inducing trajectory rotation without changing the magnitude of the velocity. N is the guidance gain, and V is the vehicle velocity. In this mission, γ and the line-of-sight (LOS) angle are defined as shown in Fig. 4. And the lead angle is defined from these angles as follows:

$$\lambda = \gamma - \sigma, \quad \lambda_{\mathrm{ref}} = \gamma_{\mathrm{ref}} - \sigma_{\mathrm{ref}} \tag{11}$$

According to previous studies [14], the lead angle tracking error remained nearly constant after the initial transient and diverged only near the terminal phase. The terminal error was shown to remain within a bounded region and at the level of only a few meters. In addition, γ_{ref} and σ_{ref} were generated using downrange-based references, because when a time-based reference is used, the final position may shift if the actual velocity profile deviates from the model prediction.

However, in missions such as Falcon 9 landing, the use of downrange-based references can cause instability in the definition of the reference because monotonicity is not guaranteed during the vertical descent phase. Therefore, this study adopts an altitude-based reference to overcome the limitation of the downrange-based method under the characteristics of the landing mission. Since altitude decreases monotonically during the landing process, the current altitude is set as the independent variable, and $\gamma_{\mathrm{ref}}(y)$ and $\sigma_{\mathrm{ref}}(y)$ are defined accordingly. This allows the reference values corresponding to the current altitude to be retrieved in a stable manner even if the flight time changes or the velocity profile varies, and it provides numerical stability in the final landing phase while directly satisfying the terminal altitude condition.

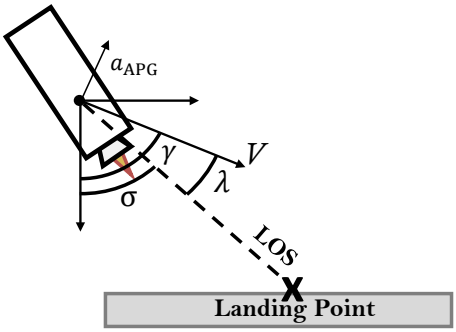


Fig. 4 Geometric relation of flight path, LOS, and lead angle

3.2 Modified APG

APG guarantees target interception, but it does not ensure precise tracking of the reference trajectory during the intermediate phase. However, in an RTLS mission, it is important to reduce the trajectory alignment error (LOS error) first, because the trajectory must accurately match the target point before vertical landing. To address this, the study modifies the conventional APG, referred to as M-APG, by adding a weighting factor to the lead angle error term as follows:

$$\lambda_{\text{err}} = k_\gamma \left(\gamma_{\text{ref}} - \gamma\right) - k_\sigma \left(\sigma_{\text{ref}} - \sigma\right) \tag{12}$$

$k_\sigma > k_\gamma$ is set so that a larger weight is assigned to the LOS error, and using this λ_{err}, the APG acceleration is calculated as $a_{\text{APG}} = NV\lambda_{\text{err}}$. This approach reduces the trajectory tracking error by decreasing the LOS error.

4 Speed and Path Controllers

4.1 Speed Control

The method presented in this section is a supplementary controller designed to regulate the velocity magnitude, while APG is mainly a rotational control for trajectory alignment through lead angle tracking.

In the optimal trajectory described in Section 2, the trajectory and control inputs are designed to minimize fuel consumption and control cost. They drive both altitude and velocity to zero at the landing point. However, in actual flight, it is difficult to satisfy these final ideal conditions due to disturbances, model uncertainties, and thrust limitations. In particular, residual velocity or altitude mismatch may occur in the final landing phase. To address this issue, this study divides Phase 3, the Landing Burn segment, into two parts: Phase 3A maintains reference trajectory tracking, and Phase 3B, the low-altitude landing phase, precisely controls the velocity just before touchdown. And a control loop that directly regulates the velocity magnitude is used. It generates acceleration aligned with the velocity vector and the corresponding thrust is calculated and applied.

4.1.1 Control Algorithm

The velocity control law is derived in the form of a feedback control that expresses acceleration as a function of the state variables. First, the acceleration is assumed as follows:

$$A(t) = a + bt^n \tag{13}$$

According to the terminal condition that the acceleration must be zero, the following relation with respect to the final time t_f can be obtained:

$$A(t_f) = a + bt_f^n = 0 \tag{14}$$

Through time integration, the final conditions of velocity and altitude are expressed as follows:

$$v_f = v_0 + at_f + \frac{b}{n+1}t_f^{n+1} \tag{15}$$

$$h_f = h_0 + v_0 t_f + \tfrac{1}{2}at_f^2 + \frac{b}{(n+1)(n+2)}t_f^{n+2} \tag{16}$$

From Eqs. (14), (15) and (16), the variations are obtained as follows:

$$\Delta v = v_f - v_0 = at_f - \frac{a}{n+1}t_f = \frac{n}{n+1}at_f \tag{17}$$

$$\Delta h = h_f - h_0 - v_0 t_f = \tfrac{1}{2}at_f^2 - \frac{a}{(n+1)(n+2)}t_f^2 = \frac{n(n+3)}{2(n+1)(n+2)}at_f^2 \tag{18}$$

By eliminating a from Eqs. (17) and (18), we obtain the following:

$$\Delta h = \left(v_0 + \frac{n+3}{2(n+2)}\Delta v\right)t_f \tag{19}$$

This is a relation that satisfies both the initial state and the terminal constraints. In addition, to describe the case for an arbitrary starting point, the initial state variables h_0, v_0, t_f are generalized to h, v, t_{go} and from Eq. (19) the following equation is obtained.

$$t_{go} = \frac{2(n+2)\Delta h}{(n+3)v + (n+1)\Delta v} \tag{20}$$

Rewriting Eq. (17) from Eq. (20) gives:

$$a = \frac{n+1}{n}\frac{\Delta v}{t_{go}} \tag{21}$$

Substituting Eq. (20) into Eq. (21) yields the following:

$$a = \frac{n+1}{n}\frac{\Delta v}{\Delta h}\left(v + \frac{n+3}{2(n+2)}\Delta v\right) \tag{22}$$

From Eq. (22), the required acceleration a is directly expressed in terms of the current state variables v, h and the differences Δv, Δh with respect to the target state. In practical implementation, at each time step the current state is taken as the initial condition, and the acceleration required to reach the target state is computed. Thus, the derived a becomes a feedback solution that reflects the state at every instant. As a result, the acceleration command is continuously updated to drive the vehicle from the current state to the target point, and the desired landing conditions $v_f = 0, h_f = 0$ can ultimately be achieved.

Also, t_{go} becomes shorter as n increases, so a larger n results in a shorter flight time and reduced propellant consumption. In addition, as n increases, the acceleration gradually decreases and then rapidly converges to zero just before landing, and in the limit $n \to \infty$ the trajectory converges to a constant acceleration profile with $A(t) = \mathrm{const}$.

4.1.2 Speed Profile Selection

Using the above velocity control algorithm, the required accelerations for velocity control are computed by varying n for each phase. In Phase 1 and Phase 3A, a constant acceleration is simply assumed. Since strict satisfaction of the terminal conditions is not required in the initial and mid-descent phases, the constant acceleration assumption is sufficient. In this case, the required acceleration corresponds to the case of $n \to \infty$, and it is given as follows:

$$a_{\text{spd}} = \frac{0.5\left(v_{\text{target}}^2 - v_i^2\right)}{y_{\text{target}} - y_i} + g \tag{23}$$

where v_{target} and y_{target} are applied as the state values at the end of each phase. By adding the gravitational acceleration g and multiplying by the current mass, the required thrust can be calculated.

Phase 3B applies a segmented polynomial acceleration profile to enable precise velocity control just before landing. In this phase, $n = 2$ is used so that the acceleration takes a quadratic form and converges to zero at touchdown, thereby removing the residual velocity. At this time, the required acceleration is given as follows:

$$a_{\text{spd}} = \frac{3}{16}\frac{(v_{\text{target}} - v_i)(5v_{\text{target}} + 3v_i)}{y_{\text{target}} - y_i} + g \tag{24}$$

Both v_{target} and y_{target} are zero. This choice of n enables a gradual decrease in acceleration while avoiding unnecessary extension of the landing time that would increase the propellant consumption rate. In addition, during Phase 2 which is the Aerodynamic Guidance segment, only lateral trajectory control is performed and no velocity control is applied.

4.2 Terminal Path Control

This section addresses a precise control method introduced to complement the limitation of APG in the low-altitude landing phase.

APG is a proportional linear trajectory shaping method that tracks the lead angle and can guarantee trajectory stability in the mid- and long-range phases. However, in the terminal phase, the lead angle error diverges and the miss distance remains within a bounded value. Since the RTLS mission requires a very strict tolerance on the terminal position, it is not appropriate to guide the entire flight using APG alone. Therefore, a more direct error-correction-based controller is required in the terminal phase and in this study the lateral acceleration command is computed using a PD controller. In other words, APG is applied in Phases 1 to 3A, while the PD controller is applied in Phase 3B, the low-altitude landing phase. The acceleration command can be expressed as:

$$a_{\text{PD}} = K_p(x_{\text{ref}} - x) + K_d(\dot{x}_{\text{ref}} - \dot{x}) \tag{25}$$

where x_{ref} and $\dot{x}_{\text{ref}}$ are defined using the altitude-based reference, so the system can operate stably even if the actual flight time differs from the plan. The proposed PD controller acts directly on the position and velocity errors, thereby providing more robust performance in the terminal phase than APG.

5 Guidance and Control Implementation

5.1 Allocation of Control Input under Saturation

In the previous section, the required accelerations $a_\perp(a_{\text{APG}}, a_{\text{PD}})$ and a_{spd} were computed by the controllers. But these are ideal values that do not account for the actual thrust limits. For practical

implementation, the longitudinal and lateral components must be properly allocated within the available maximum thrust, and the result must be converted into the actual control inputs: angle of attack(α) and throttle(δ). This section describes the allocation procedure and the computation of these α and δ. In the powered phases, where thrust dominates aerodynamic forces, the control input allocation is based on thrust, with aerodynamic effects treated implicitly in the dynamics.

Acceleration allocation is performed in the velocity coordinate system. First, the required longitudinal acceleration (a_{spd}) is secured to satisfy the velocity profile. Then, within the remaining thrust capacity, the lateral component ($a_{\perp}$) is assigned. The detailed algorithm is as follows. First, the required acceleration is given as follows:

$$a_{\mathrm{req}} = \sqrt{a_{\mathrm{spd}}^2 + a_{\perp}^2}, \qquad a_{\perp} \in \{a_{\mathrm{APG}},\, a_{\mathrm{PD}}\} \tag{26}$$

If the a_{req} does not exceed $a_{\max} = F_T/m$, then $a_{\mathrm{cmd}} = a_{\mathrm{req}}$ and the throttle is set to $\delta = a_{\mathrm{cmd}}/a_{\max}$. If a_{req} exceeds $a_{\max}$, the longitudinal component is preserved while the lateral component is clipped as follows:

$$a_{\perp} = \sqrt{a_{\max}^2 - a_{\mathrm{spd}}^2}\ \mathrm{sign}(a_{\perp,0}) \tag{27}$$

In this case, $a_{\mathrm{cmd}} = a_{\max}$ and the throttle is set to max throttle ($\delta_{\max}$).

The actual input of the vehicle is the angle of attack α. For simplicity, in this paper thrust vector control is not considered, and the thrust is assumed to be always aligned with the body axis. Under these assumptions, in Phases 1, 3A, and 3B, α is calculated from the geometric relation in Fig. 5.

$$\alpha_{\mathrm{cmd}} = -\arctan\left(\frac{a_{\perp}}{|a_{\mathrm{spd}}|}\right) \tag{28}$$

During Phase 2, the Aerodynamic Guidance phase, thrust is unavailable and only lift generates the lateral acceleration. Approximating lift as $L \approx L_\alpha\, \alpha$ the corresponding angle of attack is:

$$\alpha_{\mathrm{cmd}} = \frac{m\, a_{\mathrm{APG}}}{L_\alpha} \tag{29}$$

The obtained α_{cmd} is saturated within the allowable range $[-\alpha_{\max}, \alpha_{\max}]$. The above logic, i.e., allocating the required acceleration under the thrust constraint and converting it into the control inputs α and δ, is summarized in Algorithm 1.

Algorithm 1 Control input allocation

Input: $a_{\mathrm{spd}}, a_{\perp 0}, a_{\max}, a_{\mathrm{req}}, \mathrm{phase}, L_\alpha, m, \alpha_{\max}, \delta_{\max}$
Output: $\alpha_{\mathrm{cmd}},\ \delta$
if $|a_{\mathrm{spd}}| > a_{\max}$ **then**
    $(a_{\mathrm{spd}}, a_{\perp}, a_{\mathrm{cmd}}, \delta) \leftarrow (\mathrm{sign}(a_{\mathrm{spd}}) \cdot a_{\max},\ 0,\ a_{\max},\ \delta_{\max})$
else if $a_{\mathrm{req}} > a_{\max}$ **then**
    $(a_{\mathrm{spd}},\ a_{\perp}, a_{\mathrm{cmd}}, \delta) \leftarrow (a_{\mathrm{spd}},\ \mathrm{clipping}(a_{\perp 0}),\ a_{\max},\ \delta_{\max})$ use Eq. (27) from main text
else
    $(a_{\mathrm{spd}},\ a_{\perp}, a_{\mathrm{cmd}}, \delta) \leftarrow (a_{\mathrm{spd}},\ a_{\perp 0},\ a_{\mathrm{req}},\ (a_{\mathrm{cmd}}/a_{\max}))$
end if
if phase = $Engine\ OFF$ **then**
    $\alpha_{\mathrm{cmd}} \leftarrow f(a_{\perp}, m, L_\alpha)$ use Eq. (29) from main text
else
    $\alpha_{\mathrm{cmd}} \leftarrow f(a_{\perp}, a_{\mathrm{spd}})$ use Eq. (28) from main text
end if
$\alpha_{\mathrm{cmd}} \in [-\alpha_{\max}, \alpha_{\max}]$

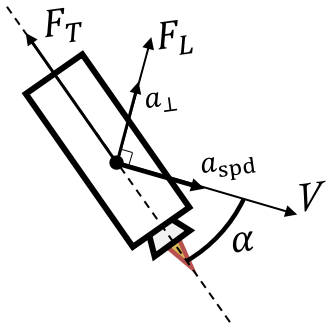


Fig. 5 Geometric relation between acceleration components and aerodynamic forces

5.2 Guidance and Control Logic

In this section, the application of the M-APG lateral control algorithm, the velocity control algorithm, the PD controller, and the acceleration allocation logic to each phase is described.

The control methods for each phase are summarized in Tab. 1. In the Entry Burn and the initial part of the Landing Burn (Phase 3A), lateral control is performed using the M-APG algorithm, and velocity control is based on the constant-acceleration assumption. The required acceleration is allocated within the thrust limits, and constraints are satisfied by clipping when necessary. In the Aerodynamic Guidance phase, the engines are shut down, so only lift generates the lateral acceleration; thus, velocity control is not performed, and the control input is converted into the angle of attack. In the low-altitude (Phase 3B), a PD controller is applied for lateral control, and velocity control is based on a polynomial acceleration profile with $n = 2$, ensuring that the terminal velocity converges to zero. In all phases, the final control inputs are converted into angle-of-attack and throttle commands based on the accelerations computed in each phase. This structure enables phase-specific control that reflects the dynamics and constraints of each phase, ultimately achieving a stable and fuel-efficient landing.

Table 1 Guidance and control logic applied in each phase

	Entry Burn	**Aero Guidance**	**Landing Burn**	
	Phase1	Phase2	Phase3A	Phase3B
Condition	$y > y_{1,0}$	$y_{2,f} < y < y_{2,0}$	300m $< y <$	$y <$ 300m
N_{engine}	3	0	1	1
$a_\perp$	M-APG	M-APG	M-APG	PD
a_{spd}	$n = \infty$	–	$n = \infty$	$n = 2$
a_{cmd}	a_{cmd} allocation	–	a_{cmd} allocation	a_{cmd} allocation
α	Eq. (28)	Eq. (29)	Eq. (28)	Eq. (28)
δ	$\min(\delta_{\max}, a_{\mathrm{cmd}}/a_{\max})$ (all phases)			

6 Simulation Results

In this section, the performance of the proposed algorithm is validated through simulations. All simulations use the optimal trajectory obtained in Section 2 as the reference trajectory, and the controller is designed to track this trajectory. The guidance and control logic applied in each phase are as described in the previous section, and saturation limits are imposed on angle of attack and throttle to reflect practical implementation.

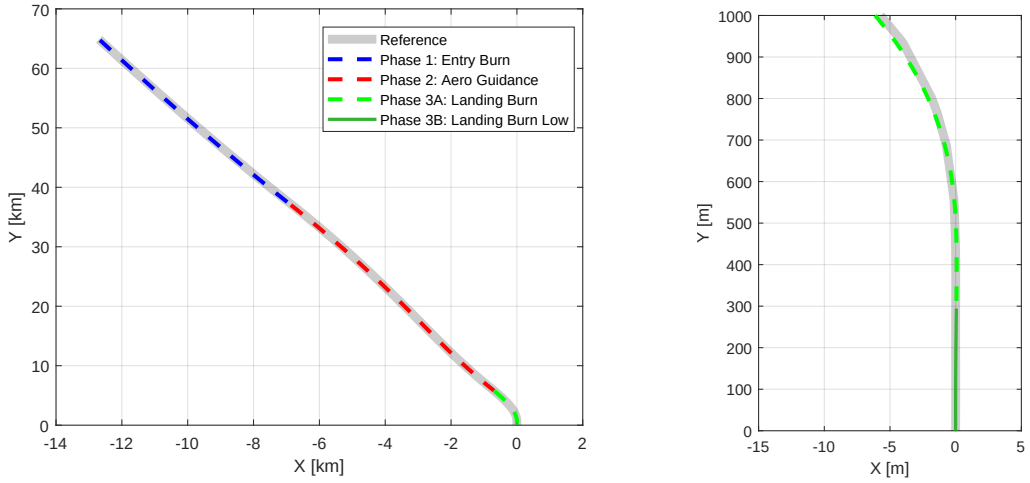
The main gains and constraints for each phase are summarized in Tab. 2. Regarding the M-APG gains, k_σ is set relatively larger to prioritize LOS alignment. The $\delta_{\max}$ is reduced in the final landing phase to leave a thrust margin, thereby ensuring flight stability and control authority. The maximum angle of attack for each phase is determined considering the aerodynamic stability of the actual vehicle and stall prevention: 10° in the EB and LB phases where engine thrust dominates, and 20° in the AG phase where aerodynamic maneuvering is dominant. Through the following simulations, it is examined whether the proposed algorithm can stably track the optimal trajectory and accomplish the vertical landing mission.

Table 2 Control gains and saturation limits for each phase

Phase	N	k_γ	k_σ	K_p	K_d	$\alpha_{\max}$ (deg)	$\delta_{\max}$
EB	3	0.5	1.0	–	–	10	1.00
AG	3	0.5	1.5	–	–	20	0.00
LB-A	3	0.5	1.0	–	–	10	0.85
LB-B	–	–	–	0.1	0.005	10	0.85

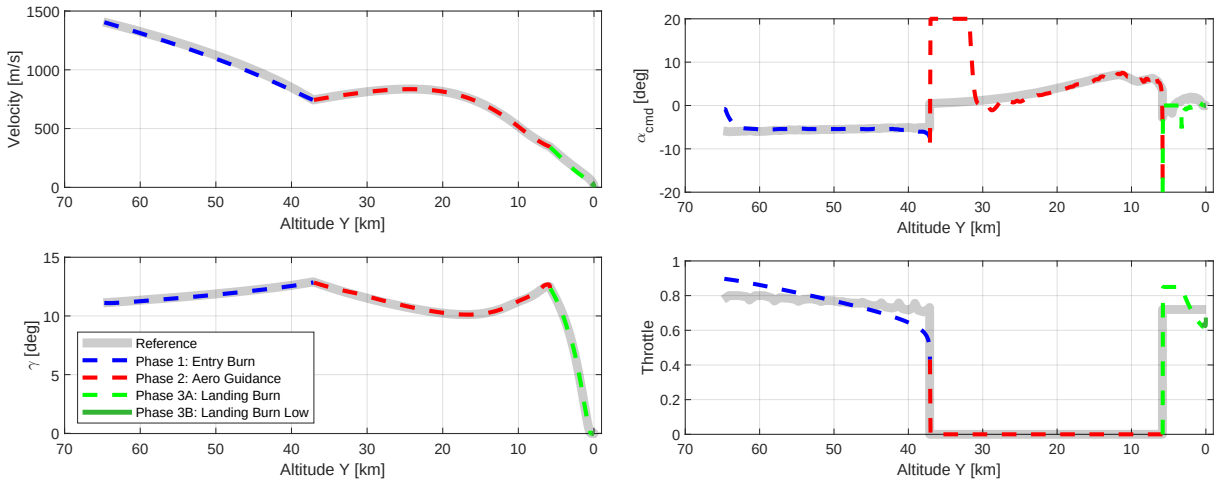
6.1 Scenario with Nominal Conditions

In the nominal scenario, the proposed guidance and control algorithm is applied with the same initial conditions as the reference trajectory. Figure 6 shows the overall trajectory and the zoomed-in landing segment, confirming that the proposed algorithm stably tracks the optimal trajectory and reaches the target landing point. The simulation ended at 130.5 s, with final state of $x = 0.0028$m and $y = 0.066$m, indicating virtually error-free landing performance. The final velocity is 0.72 m/s, and the flight path angle error is 0.0016° both satisfying the terminal landing conditions. Figure 7 presents the variations in flight states and control inputs. Both results closely match the reference trajectory, and the control inputs are well allocated within the saturation limits. In Fig. 7b, angle of attack temporarily reaches saturation at phase transitions, due to abrupt changes in the required acceleration and thrust mode switching. But this remains within the constraints and ensures stable trajectory convergence without significantly affecting the overall performance. Compared with the optimal trajectory, approximately 9.1×10^2kg additional propellant (about 2.2% increase) is consumed. This is mainly due to a quadratic acceleration profile in the final landing phase, which ensures gradual velocity convergence to zero rather than strictly minimizing fuel consumption.



(a) Full trajectory **(b) Zoomed-in trajectory**

Fig. 6 Simulated trajectory in downrange–altitude plane



(a) Velocity and flight path angle vs. altitude **(b) Angle of attack and throttle vs. altitude**

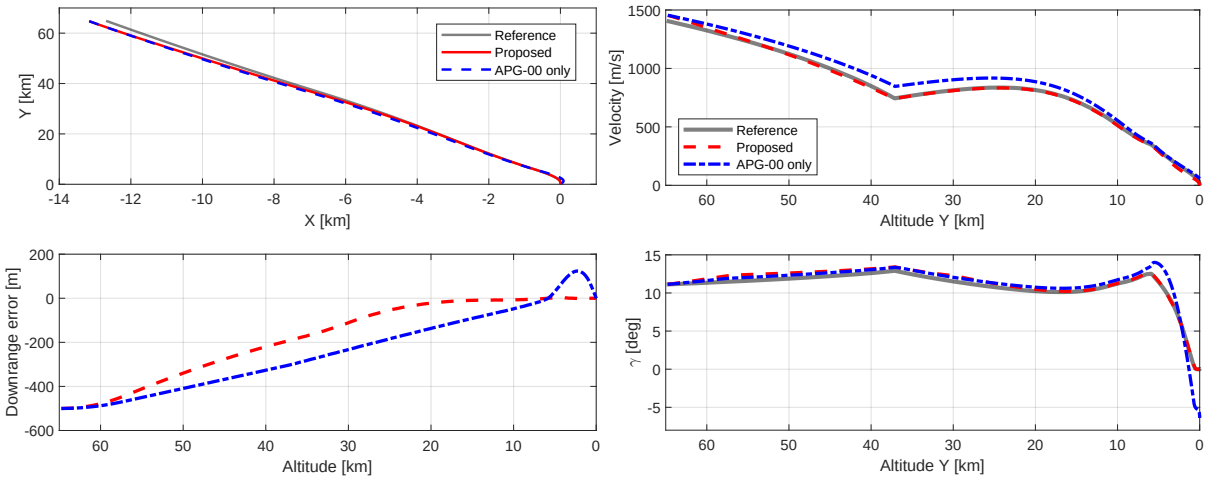
Fig. 7 Velocity and flight path angle profiles

6.2 Scenario with Entry Condition Error

In this scenario, disturbances of 500 m in the initial downrange position and 50 m/s in the initial velocity were introduced to evaluate the proposed algorithm and compare it with the conventional APG. The comparison with APG was conducted under identical initial conditions, reference trajectory, constraints, and input saturation limits, with the APG-only case used as the baseline without the proposed supplementary controllers. Figure 8 shows the trajectory and state profiles, where the downrange error is defined as the difference between the vehicle and reference downrange positions at the same altitude.

When only APG is applied, it guarantees terminal convergence but performs trajectory alignment slowly. As a result, the remaining downrange error cannot be sufficiently corrected in the terminal phase due to reduced control authority at low velocity, leading to overshoot and poor convergence of the flight path angle. The final state in the APG case was $x = 34.72$ m and $\gamma = -8°$, which does not satisfy the terminal conditions.

In contrast, the proposed method rapidly reduces the downrange error through the modified APG and achieves effective alignment with the reference trajectory. The additional velocity and terminal control provide sufficient control authority in the low-altitude phase, enabling precise trajectory tracking. The final state was $x = -0.0034$ m and $\gamma = 0.0019°$, demonstrating accurate landing performance even under large initial disturbances.



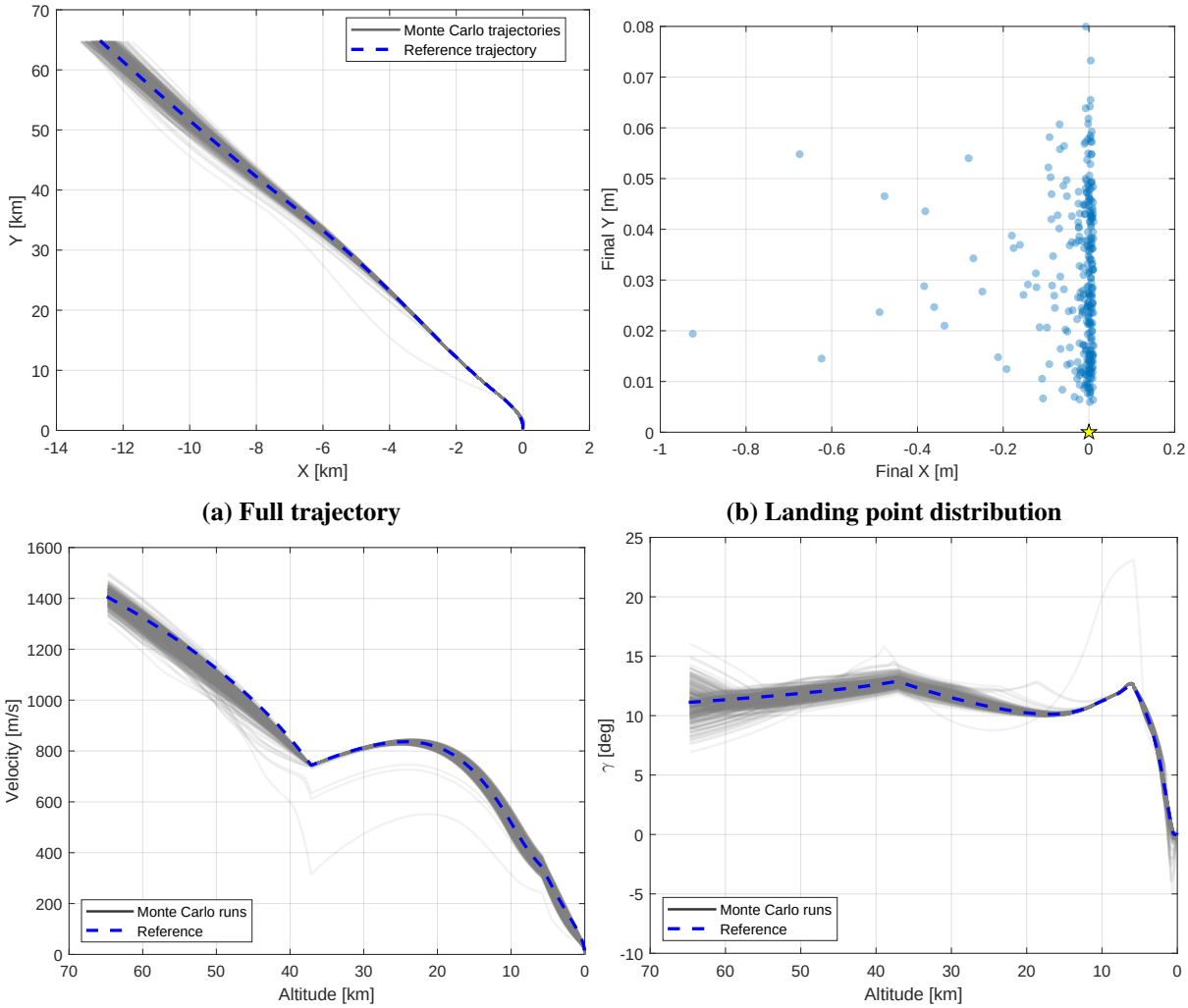
(a) Trajectory and downrange error **(b) Velocity and flight path angle vs. altitude**

Fig. 8 Comparison of MAPG and APG under entry condition errors

6.3 Monte Carlo Simulation with Uncertainties

Monte Carlo simulation was conducted to evaluate the robustness of the proposed guidance algorithm under initial errors, aerodynamic uncertainties, and wind disturbances. A total of 300 simulation runs were performed. The initial downrange position, velocity magnitude, and flight path angle were perturbed using zero-mean Gaussian distributions with standard deviations of 300 m, 30 m/s, and 1.5°, respectively, and were bounded within $\pm 3\sigma$. The horizontal wind was modeled as a zero-mean Gaussian disturbance with a standard deviation of 15 m/s. Aerodynamic uncertainties were introduced as multiplicative errors, with $k_{CL} \sim \mathcal{N}(1, 0.1^2)$ and $k_{CD} \sim \mathcal{N}(1, 0.08^2)$. A successful landing is defined as the case where the final altitude is less than 1.0 m and the final velocity magnitude is below 1 m/s.

The Monte Carlo results show that 299 out of 300 runs successfully reached the terminal region, corresponding to a success rate of 99.67%. Figure 9 shows the Monte Carlo trajectories and other states distributions. The terminal states converged to $x_f = -0.033 \pm 0.101$ m, $y_f = 0.030 \pm 0.015$ m, $\gamma_f = 0.018 \pm 0.057°$, and $V_f = 0.461 \pm 0.115$ m/s. The state dispersion tends to increase as the control authority decreases with decreasing altitude. Nevertheless, the proposed method rapidly aligns the trajectory and maintains stable convergence in the terminal state through additional control actions applied in the terminal phase.



(a) Full trajectory **(b) Landing point distribution**

(c) Velocity vs. altitude **(d) Flight path angle vs. altitude**

Fig. 9 Monte-Carlo simulation results under state uncertainties and wind disturbances

7 Conclusion

This study proposes an integrated guidance and control method for the RTLS mission of a reusable launch vehicle. First, a multi-phase optimal control problem is formulated using GPOPS-II, and an optimal trajectory is obtained by considering constraints and phase continuity. This trajectory is used as the reference trajectory for the proposed guidance algorithm. Next, APG is introduced to reduce the lead angle error with a position-based reference for stable trajectory tracking. An altitude-based reference is used, and a modified APG (M-APG) with weighting is proposed to overcome the limitation of reducing the LOS error in the conventional APG, which improved alignment with the landing target point. And two additional control methods are also designed to apply APG, which is suitable for mid-course guidance, to the RTLS landing mission: (i) a velocity controller based on a polynomial acceleration profile that gradually decreases the velocity to precisely control the final landing velocity, (ii) a longitudinal PD controller that directly corrects position and velocity errors in the terminal phase. In addition, an integrated acceleration allocation algorithm is used to distribute longitudinal and lateral accelerations within the thrust limit. These are then converted into angle-of-attack and throttle commands, enabling realistic control inputs in all phases.

The proposed method was then validated through simulations. Under nominal conditions, a stable landing was achieved with terminal errors converging to nearly zero. Even in the presence of initial condition disturbances, the proposed method rapidly aligned the trajectory and achieved improved terminal accuracy and stability compared to the conventional APG. Furthermore, Monte Carlo simulation results confirmed a high success rate and stable convergence of terminal states under various disturbances and model uncertainties, demonstrating the robustness of the proposed method.

In conclusion, this study demonstrated that stable and fuel-efficient vertical landing can be achieved by integrating optimal trajectory generation, APG-based tracking, supplementary controllers, and input allocation within a two-dimensional high-fidelity dynamic environment that considers multi-phase flight. This result is meaningful in that stable trajectory convergence and terminal accuracy can be achieved while maintaining a simple APG-based structure, without relying on real-time iterative optimization.

However, the present study is limited to a two-dimensional framework, and further analysis is required to investigate the effects of attitude dynamics and axis coupling on performance. Future work will extend the guidance and control structure to a 3-DOF model and validate its performance under more realistic flight conditions.

Acknowledgement

This research is part of the results of the "Development of Guidance and Control Simulation Program for Reusable Launch Vehicles" project supported by Hanwha Aerospace. This work is supported by the InnoCORE program of the Ministry of Science and ICT(N10250155). We would like to express our gratitude for these support.

Declaration of Use of Artificial Intelligence

Artificial intelligence tools (OpenAI ChatGPT, GPT-5, September 2025 version) were used only for grammar checking and typo correction. No AI tool was used in the research work, analysis, or generation of results.

References

[1] Xinfu Liu. Fuel-optimal rocket landing with aerodynamic controls. *Journal of Guidance, Control, and Dynamics*, 42(1):65–77, 2019.

[2] Behcet Acikmese and Scott R Ploen. Convex programming approach to powered descent guidance for mars landing. *Journal of Guidance, Control, and Dynamics*, 30(5):1353–1366, 2007.

[3] Ki-Wook Jung, Sang-Don Lee, Cheol-Goo Jung, and Chang-Hun Lee. Model predictive guidance for fuel-optimal landing of reusable launch vehicles. *arXiv preprint arXiv:2405.01264*, 2024.

[4] Michael Szmuk, Behcet Acikmese, and Andrew W Berning. Successive convexification for fuel-optimal powered landing with aerodynamic drag and non-convex constraints. In *AIAA Guidance, Navigation, and Control Conference*, page 0378, 2016.

[5] Michael Szmuk and Behcet Acikmese. Successive convexification for 6-dof mars rocket powered landing with free-final-time. In *2018 AIAA Guidance, Navigation, and Control Conference*, page 0617, 2018.

[6] Nicolas Berend and Christophe Talbot. Overview of some optimal control methods adapted to expendable and reusable launch vehicle trajectories. *Aerospace science and technology*, 10(3):222–232, 2006.

[7] Cheol-Goo Jung, Chang-Hun Lee, and Min-Jea Tahk. Convex programming-based optimal three-dimensional mid-course guidance with lossless convexification. In *33rd Congress of the International Council of the Aeronautical Sciences, ICAS 2022*. International Council of the Aeronautical Sciences (ICAS), 2022.

[8] Jon C Harpold and Donald E Gavert. Space shuttle entry guidance performance results. *Journal of Guidance, Control, and Dynamics*, 6(6):442–447, 1983.

[9] Allan R Klumpp. Apollo lunar descent guidance. *Automatica*, 10(2):133–146, 1974.

[10] Dequan Zeng, Shicong Pan, Yinquan Yu, Yiming Hu, Jinwen Yang, Peizhi Zhang, Lu Xiong, Giuseppe Carbone, and Letian Gao. A comparative study on trajectory tracking control methods for automated vehicles. *Scientific Reports*, 15(1):17073, 2025.

[11] Larasmoyo Nugroho, Ammar Abdurrauf, Salma Sonia Jneina Sagiri, and Anggriawan Rayzadmiko. Minimizing rocket landing's trajectory following error via visual guided wind compensator. In *2024 IEEE International Conference on Aerospace Electronics and Remote Sensing Technology (ICARES)*, pages 1–7. IEEE, 2024.

[12] Marco Sagliano, Ansgar Heidecker, José Macés Hernández, Stefano Farì, Markus Schlotterer, Svenja Woicke, David Seelbinder, and Etienne Dumont. Onboard guidance for reusable rockets: Aerodynamic descent and powered landing. In *AIAA Scitech 2021 Forum*, page 0862, 2021.

[13] Lance A Davis. First stage recovery. *Engineering*, 2(2):152–153, 2016.

[14] Min-Jea Tahk, Jong-Chan Park, Boseok Kim, Heekun Rho, and Chang-Hun Lee. Augmented pursuit guidance for flight trajectory shaping. In *Joint 10th EUCASS-9th CEAS Conference*. European Conference for AeroSpace Sciences (EUCASS)-Council of European . . . , 2023.