aerospace

MDPI

Article

Safety-Guaranteed Go-Around Decision Framework Based on Funnel for Automatic Carrier Landing Systems

Zhen Liu , Jianjun Luo, Yunzhao Liu and Weihua Ma *

National Key Laboratory of Aerospace Flight Dynamics, Northwestern Polytechnical University, Xi'an 710072, China
* Correspondence: whma_npu@nwpu.edu.cn

Abstract

Existing go-around decision methods for Automatic Carrier Landing Systems (ACLSs) are mainly based on altitude deviation. The safe region implied by these methods is therefore effectively single-dimensional, which can lead to overly conservative and incomplete go-around decisions because the remaining state deviations are not evaluated. This paper proposes a safety-guaranteed go-around decision framework based on a funnel that explicitly characterizes the time-varying safe region in the multi-dimensional state space throughout the landing process, thereby enabling safe and precise decisions. A six-dimensional funnel is then computed backward from the terminal set using sum-of-squares programming while accounting for nonlinear dynamics, actuator saturation constraints, and prescribed bounded uncertainties. The resulting funnel guarantees that all closed-loop trajectories initialized within it remain within the subsequent funnel and reach the terminal set at the final time. In the online stage, the online go-around decision module evaluates at each sampling instant whether the complete six-dimensional state deviation lies inside the corresponding precomputed funnel and outputs either the continue-landing decision or the go-around decision. Simulation results show that incorporating the range state reduces the mean, maximum, and standard deviation of the terminal altitude deviations by approximately 68%, 64%, and 66%, respectively, and the corresponding statistics of the terminal range deviations by approximately 99%. Under a representative deck-motion disturbance, the framework remains effective after the six-dimensional funnel is recomputed offline. The proposed method also avoids unnecessary go-around decisions when the altitude component exceeds the conventional threshold but the complete six-dimensional state deviation remains inside the funnel.

Keywords: carrier landing; funnel; go-around decision; safe region; sum-of-squares programming

Academic Editor: Mostafa Hassanalian

Received: 1 August 2026
Revised: 8 September 2026
Accepted: 8 September 2026
Published: 16 September 2026

1. Introduction

Carrier landings are considerably more hazardous than conventional land-based landings. Their failure rate is substantially higher, and landing-related accidents account for the majority of total mishaps in carrier aviation, underscoring the critical need for safe automatic carrier landing to support future naval aviation [1,2].

Making safe and precise go-around decisions requires an explicitly characterized safe region, whereas existing methods are mainly based on altitude deviation. In these methods, a nominal trajectory is precomputed, and the current altitude deviation is compared with a prescribed altitude-deviation threshold to determine whether a go-around should be

 https://doi.org/10.3390/aerospace13090842

initiated during landing [3]. Such a criterion is straightforward to implement online because it requires only a single state deviation and a prescribed threshold. However, this simplicity comes at the cost of not directly evaluating the coupled effects of the remaining state deviations in the go-around decision criterion. Specifically, an altitude deviation outside the prescribed threshold does not necessarily imply that the complete six-dimensional state deviation lies outside the safe region, whereas an altitude deviation within the threshold does not guarantee that the remaining state deviations are compatible with the actuator saturation constraints and terminal landing requirements. Therefore, a single-dimensional criterion cannot fully determine whether all landing-task requirements can still be satisfied and may lead either to unnecessary go-around decisions or to continue-landing decisions without a complete-state safety basis.

In this paper, the safe region is characterized as the set of complete six-dimensional state deviations from which, under the closed-loop dynamics and prescribed uncertainty bounds, the subsequent closed-loop trajectory is guaranteed to remain within the corresponding time-varying funnel throughout the landing process, satisfy the actuator saturation constraints, and reach the terminal set reflecting the required landing precision at the final time. Environmental disturbances are treated as one type of uncertainty within this framework. Conventional carrier-landing go-around decision methods are commonly formulated using simplified longitudinal aircraft dynamics. In particular, representative methods based on small-disturbance dynamics and terminal-state prediction estimate the aircraft altitude at the carrier stern and determine the go-around decision according to the prescribed go-around boundary [3]. Consistent with this established modeling practice, this study adopts a nonlinear longitudinal model and extends the decision basis from altitude deviation alone to the complete six-dimensional longitudinal state. By jointly evaluating the airspeed, flight-path angle, pitch angle, pitch rate, altitude, and range deviations, the proposed six-dimensional rule accounts for the coupled effects of the complete longitudinal state while retaining a simple scalar membership test during online implementation. Therefore, a more appropriate go-around decision rule should evaluate the complete six-dimensional state deviation rather than the altitude deviation alone.

In addition, making safe and precise go-around decisions in ACLSs still faces three major challenges [2]: (1) Complex nonlinear dynamics. Unlike rotorcraft [4], fixed-wing carrier aircraft must handle strongly coupled nonlinear aerodynamic effects and complex flight dynamics. (2) Stringent required landing precision. The desired landing area—approximately ± 22.65 feet longitudinally and ± 20 feet laterally [5]—is substantially more limited than the conventional land-based landing area. (3) Severe environmental disturbances. Atmospheric turbulence and deck motion introduce significant time-varying disturbances that challenge flight stability and control robustness. To address these challenges, existing ACLS control methods can be broadly classified according to their design mechanisms. Classical control approaches include PID control [6] and robust control [7]. Nonlinear control methods, particularly dynamic inversion and backstepping control, have been introduced to better accommodate the nonlinear aircraft dynamics [8]. Robustness-oriented approaches, including adaptive control, adaptive sliding-mode control, active disturbance rejection control, and neural-network-based disturbance observers, have been developed to mitigate the effects of uncertainties and environmental disturbances [9–15]. Prescribed-performance control, fixed-time control, and control-barrier-function-based control have also been investigated to constrain tracking errors during carrier landing [16–19]. In practical ACLS designs, several of these techniques are often integrated into composite control architectures [7,10,14–23]. In addition, feedforward–feedback composite approaches, including preview control and model-predictive-control formulations such as output-feedback stochastic MPC and receding-horizon control, have been studied to improve landing performance and adaptability [3,24–28]. Although these methods improve trajectory-tracking accuracy and robustness, they primarily act on the guidance and control system rather than modifying the go-around decision criterion itself. Consequently, the decision may still be based on altitude deviation and does not explicitly determine whether the complete six-dimensional state deviation can satisfy the actuator saturation constraints and terminal landing requirements.

Safe and precise go-around decision-making requires an explicit characterization of the time-varying safe region in the multi-dimensional state space. To meet this requirement, this study introduces funnel theory from the field of robot motion planning, which provides a suitable means of explicitly characterizing such a region [29]. As illustrated in Figure 1, the six-dimensional funnel evolves along the nominal trajectory and explicitly characterizes the time-varying safe region in the multi-dimensional state space at each time instant. In this study, each of these safe regions is geometrically represented by a six-dimensional ellipsoid; therefore, Figure 1 provides only a schematic visualization of the six-dimensional funnel. Tedrake et al. [30] first introduced sum-of-squares (SOS) programming to address the computational challenges associated with funnel estimation. Funnel-based approaches were later successfully applied in contexts such as spacecraft rendezvous [31] and the Mars landing problem [32]. In this study, the six-dimensional funnel is constructed offline to realize the safe-region characterization defined above. Its invariance property enables it to serve as the decision boundary for online go-around decision-making. During online implementation, the online go-around decision module checks at each sampling instant whether the complete six-dimensional state deviation lies inside the precomputed funnel. If the state deviation lies inside the funnel, the module outputs the continue-landing decision; otherwise, it outputs the go-around decision.

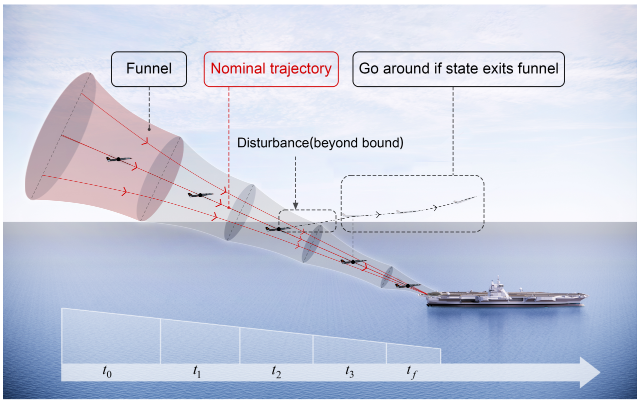


Figure 1. Safety-guaranteed go-around decision framework, in which the funnel represents the safe region.

Existing funnel-based studies have demonstrated the applicability of funnel theory to trajectory guidance problems such as powered descent [32]. However, directly applying these methods to automatic carrier-landing go-around decisions remains challenging because of the differences in the application objective and problem formulation. First, this study establishes a polynomial representation of the nonlinear longitudinal dynamics for the complete six-dimensional state, allowing the range state and the corresponding terminal requirement to be incorporated into the funnel-based go-around decision framework. Second, carrier-specific environmental effects, such as deck motion, must be incorporated into the offline computation of the corresponding decision boundary. Third, rather than using the funnel for trajectory guidance, this study uses the offline-computed six-dimensional funnel as the decision boundary for an online go-around decision rule. The online module evaluates the complete six-dimensional state deviation and outputs either the continue-landing decision or the go-around decision according to the funnel membership condition.

In summary, this study aims to establish a safety-guaranteed go-around decision framework for ACLSs by explicitly characterizing the time-varying safe region of the complete six-dimensional longitudinal state and using this region as the decision boundary for online go-around decision-making. Specifically, this study investigates how to construct such a time-varying multi-dimensional safe region while accounting for the closed-loop nonlinear dynamics, actuator saturation constraints, terminal landing requirements, and prescribed bounded uncertainties; how to use the offline-computed safe region to make online continue-landing or go-around decisions based on the complete six-dimensional state deviation; and what additional decision information and benefits can be obtained compared with the conventional single-dimensional altitude-deviation criterion. The key contributions of this study are summarized as follows:

(1) A safety-guaranteed go-around decision framework is proposed for ACLSs. In the offline stage, a six-dimensional funnel is computed while accounting for the closed-loop polynomial dynamics, actuator saturation constraints, the terminal set reflecting the required landing precision, and prescribed bounded uncertainties. During online implementation, the offline-computed funnel serves as the decision boundary, and the online go-around decision module evaluates the complete six-dimensional state deviation to output either the continue-landing decision or the go-around decision. This framework establishes a clear separation between offline decision-boundary computation and online decision execution.
(2) A complete six-dimensional formulation that includes the range state is developed for the carrier-landing problem. This formulation enables the range state and its corresponding terminal requirement to be incorporated into the feedback design, terminal set, and online membership test. Compared with the five-dimensional formulation, incorporating the range state reduces the mean, maximum, and standard deviation of the terminal altitude deviations by approximately 68%, 64%, and 66%, respectively, while reducing the corresponding statistics of the terminal range deviations by approximately 99%.
(3) The effectiveness of the proposed framework is demonstrated from the perspectives of polynomial-model accuracy, representative deck-motion disturbance, conventional altitude-deviation-based decisions, and required landing precision. The second-order closed-loop polynomial system provides a more reliable offline decision boundary than the first-order approximation. Under the representative deck-motion disturbance, the framework remains effective after the six-dimensional funnel is recomputed offline, while the online decision rule remains unchanged. The proposed method also avoids unnecessary go-around decisions when the altitude deviation exceeds the conventional threshold but the complete six-dimensional state deviation remains inside the funnel, and reveals the trade-off between the required landing precision and the safe region size.

The remainder of the paper is structured as follows: Section 2 presents the nonlinear longitudinal model and the closed-loop polynomial system. Section 3 develops the offline six-dimensional funnel estimation method using SOS programming and formulates the corresponding online go-around decision rule. Section 4 evaluates the effectiveness of the proposed go-around decision framework, analyzes how polynomial-model accuracy, the inclusion of the range state, deck motion, and the required landing precision affect the offline decision boundary and the resulting online decisions, and compares the proposed complete-state decision rule with the conventional altitude-deviation-based rule. Section 5 concludes the paper and discusses future research directions.

2. Preliminaries

As shown in Figure 2, the safety-guaranteed go-around decision framework consists of an offline computation stage and an online implementation stage for the carrier-landing scenario illustrated in Figure 1. The "Pre-computation" block in Figure 2 corresponds to offline decision-boundary computation, whereas the "In-flight Operation" block corresponds to online decision execution. In the offline stage, the nonlinear longitudinal model is established, the nominal trajectory is generated, the corresponding closed-loop polynomial system is constructed, and the six-dimensional funnel is computed while accounting for actuator saturation constraints, the terminal set reflecting the required landing precision, and prescribed uncertainty bounds. The outputs of the offline stage are the nominal state trajectory $\boldsymbol{x}_{\mathrm{op}}(t_k)$ and the precomputed funnel matrix sequence $\{\boldsymbol{P}_k\}_{k=1}^{N}$, which are stored for subsequent online use. During online implementation, the current aircraft state $\boldsymbol{x}(t_k)$ is acquired at each sampling instant. The online module retrieves the corresponding nominal state and funnel matrix, computes the complete six-dimensional state deviation as $\bar{\boldsymbol{x}}(t_k) = \boldsymbol{x}(t_k) - \boldsymbol{x}_{\mathrm{op}}(t_k)$, and evaluates the level-set function $\bar{\boldsymbol{x}}^{\top}(t_k)\boldsymbol{P}_k\bar{\boldsymbol{x}}(t_k)$. The module then outputs the continue-landing decision when the complete six-dimensional state deviation lies inside the corresponding precomputed funnel and outputs the go-around decision otherwise. Therefore, the relationship between the two stages is a one-way transfer of the offline-computed nominal trajectory and funnel matrix sequence to the online decision module; no SOS optimization or funnel recomputation is performed during online implementation. In practical ACLS implementation, the proposed method can therefore be integrated as an online go-around decision module within the existing feedback-control architecture. The existing feedback controller remains responsible for tracking the nominal landing trajectory, while the proposed module independently evaluates whether the current complete six-dimensional state deviation remains within the precomputed safe region and outputs either the continue-landing decision or the go-around decision. Thus, the proposed method supplements rather than replaces the existing flight-control system.

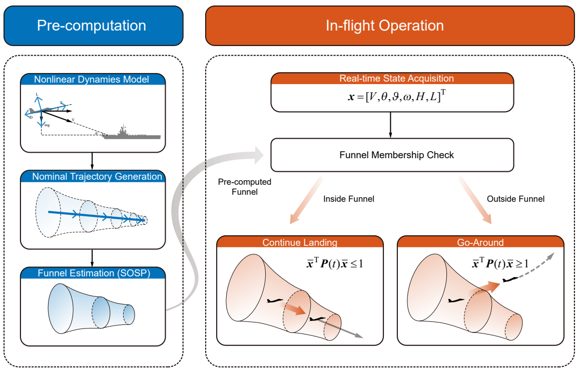


Figure 2. Workflow of the safety-guaranteed go-around decision framework. The tube-like shapes are schematic visualizations of the time-varying six-dimensional funnel, whereas the online decision is made using the complete six-dimensional level-set function.

2.1. Nonlinear Longitudinal Model

To retain the nonlinear longitudinal dynamics neglected by the small-disturbance linear models commonly adopted in conventional ACLS design, this study employs the nonlinear longitudinal model of the carrier aircraft presented in [16].

In the flight-path coordinate system, the longitudinal dynamics are expressed as follows:

$$
\begin{aligned}
\dot{V} &= \frac{P_{\max} D_p \cos\alpha - C_D\, q\, S_{\text{ref}} - mg \sin\theta}{m} \\
\dot{\theta} &= \frac{P_{\max} D_p \sin\alpha + C_L\, q\, S_{\text{ref}} - mg \cos\theta}{mV} \\
\dot{\vartheta} &= \omega \\
\dot{\omega} &= \frac{C_M\, q\, S_{\text{ref}}\, c}{I} \\
\dot{H} &= V \sin\theta \\
\dot{L} &= V \cos\theta,
\end{aligned} \tag{1}
$$

where $P_{\max}$ is the maximum thrust, D_p is the throttle setting, V is the airspeed, θ is the flight-path angle, ϑ is the pitch angle, ω is the pitch rate, H is the altitude, and L is the range. The angle of attack is given by $\alpha = \vartheta - \theta$, and the dynamic pressure is given by $q = \frac{1}{2}\rho V^2$, where ρ is the air density. In addition, g is the gravitational acceleration, m is the aircraft mass, I is the pitch moment of inertia, S_{ref} is the reference wing area, and c is the mean aerodynamic chord.

The nonlinear model in (1) represents the baseline longitudinal aircraft dynamics. Prescribed bounded environmental disturbances can be incorporated into the corresponding closed-loop model through the prescribed bounded uncertainty formulation introduced in Section 3.1.4. In the numerical study, the heave component of deck motion is explicitly incorporated into the altitude dynamics in Section 4.3.

The aerodynamic drag coefficient C_D, lift coefficient C_L, and pitching moment coefficient C_M are given by:

$$
\begin{aligned}
C_D &= C_{D0} + C_{D\alpha}\alpha + C_{D\alpha^2}\alpha^2 \\
C_L &= C_{L0} + C_{L\alpha}\alpha + C_{LD_z} D_z \\
C_M &= C_{M0} + C_{M\alpha}\alpha + C_{MD_z} D_z + C_{M\omega}\omega,
\end{aligned} \tag{2}
$$

where D_z is the elevator deflection.

The complete six-dimensional longitudinal state vector and the two-dimensional control vector of the nonlinear longitudinal model (1) are defined as:

$$
\boldsymbol{x} = \begin{bmatrix} V & \theta & \vartheta & \omega & H & L \end{bmatrix}^\top, \boldsymbol{u} = \begin{bmatrix} D_z & D_p \end{bmatrix}^\top. \tag{3}
$$

The nonlinear longitudinal model in (1) is used to generate the nominal landing trajectory. Specifically, the nominal state trajectory $\boldsymbol{x}_{\text{op}}(t)$ and the corresponding nominal control trajectory $\boldsymbol{u}_{\text{op}}(t)$ are obtained by solving the following optimal control problem:

$$
\begin{aligned}
\min_{\boldsymbol{x}_{\text{op}}(\cdot),\, \boldsymbol{u}_{\text{op}}(\cdot)} \quad & J_{\text{op}} = \int_0^{t_f} \frac{1}{2}\Big[D_{z,\text{op}}^2(t) + D_{p,\text{op}}^2(t)\Big]\mathrm{d}t, \\
\text{s.t.} \quad & \dot{\boldsymbol{x}}_{\text{op}}(t) = \boldsymbol{f}\big(\boldsymbol{x}_{\text{op}}(t), \boldsymbol{u}_{\text{op}}(t)\big), \\
& \boldsymbol{x}_{\text{op}}(0) = \boldsymbol{x}_0, \\
& \boldsymbol{x}_{\text{op}}(t_f) \in \mathcal{X}_{\text{op},f}, \\
& \boldsymbol{x}_{\min} \le \boldsymbol{x}_{\text{op}}(t) \le \boldsymbol{x}_{\max}, \\
& \boldsymbol{u}_{\min} \le \boldsymbol{u}_{\text{op}}(t) \le \boldsymbol{u}_{\max}, \qquad t \in [0, t_f].
\end{aligned} \tag{4}
$$

Here, $\boldsymbol{f}$ denotes the right-hand side of the nonlinear dynamics in (1), and t_f is the final time. The nominal state trajectory $\boldsymbol{x}_{\text{op}}(t)$ and nominal control trajectory $\boldsymbol{u}_{\text{op}}(t)$ are the decision trajectories, while $\boldsymbol{x}_0$ is the prescribed initial state. The set $\mathcal{X}_{\text{op},f}$ specifies the terminal constraints used for nominal trajectory generation and constrains the nominal terminal state $\boldsymbol{x}_{\text{op}}(t_f)$ in the original state space. The vectors $\boldsymbol{x}_{\min}$, $\boldsymbol{x}_{\max}$, $\boldsymbol{u}_{\min}$, and $\boldsymbol{u}_{\max}$ denote the prescribed state and control bounds, which are enforced throughout the landing process. The optimal control problem is transcribed using GPOPS-II and solved using SNOPT.

2.2. Closed-Loop Polynomial System

To enable funnel estimation by SOS programming, the dynamics are rewritten in terms of the deviation from the nominal trajectory. Specifically, the state is separated into a nominal component and a deviation component,

$$\boldsymbol{x}(t) = \boldsymbol{x}_{\text{op}}(t) + \bar{\boldsymbol{x}}(t), \tag{5}$$

and the control input is expressed in a similar manner,

$$\boldsymbol{u}(t) = \boldsymbol{u}_{\text{op}}(t) + \bar{\boldsymbol{u}}(t). \tag{6}$$

Here, $\boldsymbol{x}(t) \in \mathbb{R}^6$ and $\boldsymbol{u}(t) \in \mathbb{R}^2$ are the same state and control vectors defined in (3), respectively. Specifically, $\boldsymbol{u}(t) = [D_z(t)\ D_p(t)]^\top$ consists of the elevator deflection and throttle setting. The quantities $\boldsymbol{x}_{\text{op}}(t) \in \mathbb{R}^6$ and $\boldsymbol{u}_{\text{op}}(t) \in \mathbb{R}^2$ denote the nominal state and control trajectories obtained through offline trajectory planning, while $\bar{\boldsymbol{x}}(t) \in \mathbb{R}^6$ and $\bar{\boldsymbol{u}}(t) \in \mathbb{R}^2$ denote the corresponding deviations from the nominal trajectories.

The nonlinear longitudinal model is first linearized along the nominal trajectory. Substituting (5) and (6) into the nonlinear longitudinal model and retaining the first-order terms yields the following linear time-varying deviation model:

$$\dot{\bar{\boldsymbol{x}}}(t) = \boldsymbol{A}(t)\bar{\boldsymbol{x}}(t) + \boldsymbol{B}(t)\bar{\boldsymbol{u}}(t). \tag{7}$$

The state and control matrices are obtained by linearizing the nonlinear longitudinal model around the nominal state and control trajectories:

$$\begin{aligned} \boldsymbol{A}(t) &= \left.\frac{\partial \boldsymbol{f}(\boldsymbol{x},\boldsymbol{u})}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{x}_{\text{op}}(t),\,\boldsymbol{u}=\boldsymbol{u}_{\text{op}}(t)}, \\ \boldsymbol{B}(t) &= \left.\frac{\partial \boldsymbol{f}(\boldsymbol{x},\boldsymbol{u})}{\partial \boldsymbol{u}}\right|_{\boldsymbol{x}=\boldsymbol{x}_{\text{op}}(t),\,\boldsymbol{u}=\boldsymbol{u}_{\text{op}}(t)}. \end{aligned} \tag{8}$$

Here, $\boldsymbol{A}(t) \in \mathbb{R}^{6\times 6}$ and $\boldsymbol{B}(t) \in \mathbb{R}^{6\times 2}$. Since the nominal state and control trajectories vary throughout the landing process, the nonlinear longitudinal model is linearized along the entire nominal trajectory rather than at a single fixed operating point. In the numerical implementation, $\boldsymbol{A}(t_k)$ and $\boldsymbol{B}(t_k)$ are evaluated at each discretization point t_k using the corresponding nominal state $\boldsymbol{x}_{\text{op}}(t_k)$, nominal control $\boldsymbol{u}_{\text{op}}(t_k)$, the nonlinear dynamics in (1), and the aircraft parameters listed in Table 1. The linear time-varying deviation model in (7) is used to design the LQR controller. The subsequent funnel estimation is performed using a polynomial approximation of the resulting closed-loop nonlinear error dynamics.

The linear-quadratic regulator (LQR) is employed to obtain the deviation feedback control law $\bar{\boldsymbol{u}}(t)$. Given the weighting matrices $\boldsymbol{Q}_f$, $\boldsymbol{Q}(t)$, and $\boldsymbol{R}(t)$, the corresponding Riccati differential equation is

$$\begin{aligned} \dot{\boldsymbol{S}}(t) &= -\boldsymbol{Q}(t) - \boldsymbol{A}(t)^\top \boldsymbol{S}(t) - \boldsymbol{S}(t)\boldsymbol{A}(t) \\ &\quad + \boldsymbol{S}(t)\boldsymbol{B}(t)\boldsymbol{R}(t)^{-1}\boldsymbol{B}(t)^\top \boldsymbol{S}(t), \\ \boldsymbol{S}(t_f) &= \boldsymbol{Q}_f. \end{aligned} \tag{9}$$

Table 1. Parameters of carrier-based aircraft.

Parameter	Value
Mass m	16,374 kg
Pitch moment of inertia I	239,720 kg $\cdot$ m^2
Reference wing area S_{ref}	37.16 m^2
Mean aerodynamic chord c	3.51 m
Drag coefficient C_{D0}	0.1423
Lift coefficient C_{L0}	0.732
Drag coefficient $C_{D\alpha}$	-0.00438
Lift coefficient $C_{L\alpha}$	0.0751
Drag coefficient $C_{D\alpha^2}$	0.0013
Lift coefficient C_{LD_z}	0.0144
Moment coefficient C_{M0}	-0.1885
Moment coefficient $C_{M\alpha}$	-0.00437
Moment coefficient C_{MD_z}	-0.0196
Moment coefficient $C_{M\omega}$	-0.123

Solving (9) yields the positive-definite Riccati matrix $\boldsymbol{S}(t)$. The resulting time-varying LQR feedback gain and deviation feedback control law are given by

$$\begin{aligned} \bar{\boldsymbol{u}}(t) &= -\boldsymbol{K}(t)\bar{\boldsymbol{x}}(t), \\ \boldsymbol{K}(t) &\triangleq \boldsymbol{R}(t)^{-1}\boldsymbol{B}(t)^{\top}\boldsymbol{S}(t). \end{aligned} \tag{10}$$

After substituting the closed-loop control law $\boldsymbol{u}(t) = \boldsymbol{u}_{\text{op}}(t) - \boldsymbol{K}(t)\bar{\boldsymbol{x}}(t)$ into the nonlinear longitudinal model (1), the exact closed-loop error dynamics are expressed as

$$\begin{aligned} \dot{\bar{\boldsymbol{x}}}(t) &= \boldsymbol{f}_{\text{cl}}(\bar{\boldsymbol{x}}(t), t) \\ &\triangleq \boldsymbol{f}\big(\boldsymbol{x}_{\text{op}}(t) + \bar{\boldsymbol{x}}(t), \boldsymbol{u}_{\text{op}}(t) - \boldsymbol{K}(t)\bar{\boldsymbol{x}}(t)\big) \\ &\quad - \boldsymbol{f}\big(\boldsymbol{x}_{\text{op}}(t), \boldsymbol{u}_{\text{op}}(t)\big). \end{aligned} \tag{11}$$

For subsequent funnel estimation via SOS programming, the exact closed-loop error dynamics in (11) are approximated by a polynomial in the state deviation at each time instant along the nominal trajectory. Specifically, a Taylor expansion about $\bar{x} = 0$ is performed with the selected truncation order, yielding the closed-loop polynomial system

$$\dot{\bar{\boldsymbol{x}}}(t) \approx \boldsymbol{p}(\bar{\boldsymbol{x}}(t), t). \tag{12}$$

Based on the closed-loop polynomial system in (12), the funnel estimation problem can be formulated using sum-of-squares programming. Specifically, the polynomial nonnegativity conditions associated with funnel invariance and the actuator saturation constraints are converted into SOS constraints, which are subsequently represented as semidefinite programming subproblems. Since the resulting formulation contains bilinear terms between the funnel variables and SOS multipliers, it is solved using the backward point-by-point alternating strategy detailed in Section 3.

3. Offline Funnel Computation and Online Go-Around Decision Rule

This section establishes the mathematical basis of the proposed go-around decision framework through three parts. First, the six-dimensional funnel is formulated offline as a time-varying safe region for the closed-loop polynomial system using SOS programming [30]. The terminal-set inclusion, invariance, actuator saturation, and prescribed bounded uncertainty conditions give the funnel membership condition a direct decision meaning: a complete six-dimensional state deviation inside the corresponding funnel is associated with a closed-loop trajectory that remains within the subsequent funnel, satisfies the actuator saturation constraints, and reaches the terminal set under the prescribed uncertainty bounds. Second, a backward point-by-point computation strategy is employed to obtain the funnel sequence and the corresponding time-varying decision boundary efficiently [32]. Third, during online implementation, the current complete six-dimensional state deviation is evaluated against the corresponding precomputed funnel at each sampling instant, and the online go-around decision module outputs either the continue-landing decision or the go-around decision.

3.1. Offline Backward Funnel Formulation

The six-dimensional funnel used as the online decision boundary is defined along the nominal trajectory over the interval $[0, t_f]$ as a time-varying ellipsoidal set in $\mathbb{R}^n$:

$$F(t) = \left\{ \bar{\boldsymbol{x}} \;\middle|\; V(\bar{\boldsymbol{x}}, t) \triangleq \bar{\boldsymbol{x}}^\top \boldsymbol{P}(t) \bar{\boldsymbol{x}} \le 1 \right\}. \tag{13}$$

Here, $V(\bar{\boldsymbol{x}}, t)$ is the level-set function, and $\boldsymbol{P}(t)$ is a symmetric positive-definite matrix that determines the shape of the funnel ellipsoid. For the complete six-dimensional longitudinal state considered in this study, $n = 6$ and $\boldsymbol{P}(t) \in \mathbb{S}^6_{++}$. To enable numerical computation, the time interval $[0, t_f]$ is discretized into N time points t_k $(k = 1, \ldots, N)$, where $t_1 = 0$ and $t_N = t_f$. At each discrete time point, the corresponding six-dimensional funnel is represented as

$$F(t_k) = \left\{ \bar{\boldsymbol{x}} \;\middle|\; V_k(\bar{\boldsymbol{x}}) \triangleq \bar{\boldsymbol{x}}^\top \boldsymbol{P}_k \bar{\boldsymbol{x}} \le 1 \right\}. \tag{14}$$

Accordingly, the offline funnel computation determines the sequence of shape matrices $\{\boldsymbol{P}_k\}_{k=1}^{N}$ subject to the terminal-set inclusion condition, funnel invariance condition, actuator saturation constraints, and prescribed bounded uncertainty constraints introduced below. These conditions ensure that the resulting funnel sequence provides a valid time-varying safe region and decision boundary for the subsequent online membership test.

3.1.1. Funnel Terminal-Set Inclusion Condition

To ensure that the online continue-landing decision is consistent with the required landing precision, the terminal set is specified as the following ellipsoid:

$$\mathcal{X}_f = \left\{ \bar{\boldsymbol{x}} \;\middle|\; \bar{\boldsymbol{x}}^\top \boldsymbol{E} \bar{\boldsymbol{x}} \le 1 \right\}. \tag{15}$$

Here, $\boldsymbol{E} \in \mathbb{S}^6_{++}$ is the shape matrix of the terminal ellipsoid. The terminal set encodes the required landing precision together with any additional terminal-state requirements prescribed for the considered formulation. In the six-dimensional formulation examined in this study, the range state and its corresponding terminal requirement are additionally incorporated into the terminal set. To ensure that the terminal funnel is contained within $\mathcal{X}_f$, the following inclusion condition is imposed:

$$\boldsymbol{P}_N - \boldsymbol{E} \succeq \boldsymbol{0}. \tag{16}$$

This condition guarantees that $F(t_f) \subseteq \mathcal{X}_f$ and therefore ensures that the offline decision boundary is consistent with the required landing precision at the final time. The funnel invariance condition introduced next establishes the connection between current funnel membership and eventual arrival at this terminal set.

3.1.2. Funnel Invariance Condition

To ensure that the funnel membership condition provides a valid basis for the online go-around decision, an invariance condition is imposed on the funnel boundary. Specifically, a complete six-dimensional state deviation classified as lying inside the funnel at time t_k must be associated with a closed-loop trajectory that remains within the subsequent funnel. Therefore, the online membership test reflects the predicted evolution of the closed-loop trajectory over the landing process rather than only an instantaneous geometric relation. This requirement is enforced through the following boundary implication:

$$V_k(\bar{x}) = 1 \;\Rightarrow\; -\dot{V}_k(\bar{x}) \geq 0, \qquad k = 1, \ldots, N-1. \tag{17}$$

The derivative of the level-set function at t_k is approximated as

$$\begin{aligned} \dot{V}_k(\bar{x}) &= 2\bar{x}^\top \boldsymbol{P}_k \dot{\bar{x}} + \bar{x}^\top \dot{\boldsymbol{P}}_k \bar{x} \\ &\approx 2\bar{x}^\top \boldsymbol{P}_k \boldsymbol{p}(\bar{x}, t_k) + \frac{V_{k+1}(\bar{x}) - V_k(\bar{x})}{t_{k+1} - t_k}. \end{aligned} \tag{18}$$

Using the S-procedure, the implication in (17) is imposed through the following SOS constraint:

$$-\dot{V}_k(\bar{x}) - L_k(\bar{x})\big(1 - V_k(\bar{x})\big) \text{ is SOS}. \tag{19}$$

Here, $L_k(\bar{x})$ is a multiplier polynomial. Together with the funnel terminal-set inclusion condition, the invariance condition gives the online membership test a predictive and terminal meaning for the deterministic closed-loop polynomial system. Specifically, if the current complete six-dimensional state deviation lies inside the corresponding funnel, the associated closed-loop trajectory is required to remain within the subsequent funnel and eventually enter the terminal set. Therefore, the continue-landing decision is based on the predicted closed-loop evolution over the landing process rather than on a single instantaneous state threshold. The extension of this condition to prescribed bounded uncertainties is introduced in Section 3.1.4.

3.1.3. Actuator Saturation Constraints

To ensure that a continue-landing decision based on funnel membership is compatible with the physical control capabilities of the carrier aircraft, the elevator deflection and throttle setting must satisfy their actuator saturation constraints. For the jth control component, these constraints are expressed as

$$u_{\min,j} \leq u_j(t) \leq u_{\max,j}, \qquad j = 1, 2. \tag{20}$$

At the discrete time point t_k, the corresponding closed-loop control input is given by

$$u_{j,k}(\bar{x}) \triangleq u_{\mathrm{op},j}(t_k) - \boldsymbol{K}_j(t_k)\bar{x}, \tag{21}$$

where $\boldsymbol{K}_j(t_k)$ denotes the jth row of the feedback gain matrix $\boldsymbol{K}(t_k)$. To ensure that every complete six-dimensional state deviation inside the funnel is associated with admissible control inputs, the following implication is imposed for $j = 1, 2$ and $k = 1, \ldots, N-1$:

$$V_k(\bar{x}) \leq 1 \;\Rightarrow\; \begin{cases} u_{j,k}(\bar{x}) - u_{\min,j} \geq 0, \\ u_{\max,j} - u_{j,k}(\bar{x}) \geq 0. \end{cases} \tag{22}$$

Using the S-procedure, the lower-bound constraint is formulated as

$$u_{j,k}(\bar{x}) - u_{\min,j} - L_{Lj,k}(\bar{x})\big(1 - V_k(\bar{x})\big) \text{ is SOS}, \tag{23}$$

$$L_{Lj,k}(\bar{x}) \text{ is SOS}, \tag{24}$$

and the upper-bound constraint is formulated as

$$u_{\max,j} - u_{j,k}(\bar{\boldsymbol{x}}) - L_{Uj,k}(\bar{\boldsymbol{x}})\big(1 - V_k(\bar{\boldsymbol{x}})\big) \text{ is SOS,} \tag{25}$$

$$L_{Uj,k}(\bar{\boldsymbol{x}}) \text{ is SOS,} \tag{26}$$

where $L_{Lj,k}(\bar{\boldsymbol{x}})$ and $L_{Uj,k}(\bar{\boldsymbol{x}})$ are SOS multiplier polynomials for $j = 1, 2$ and $k = 1, \ldots, N-1$. These constraints are imposed separately on the elevator deflection and throttle setting at each discrete time point. Together with the funnel invariance condition, they ensure that a complete six-dimensional state deviation classified as lying inside the funnel is associated with closed-loop control inputs that remain within the prescribed actuator limits throughout the landing process. Therefore, the continue-landing decision is supported not only by terminal reachability but also by the physical feasibility of the required control inputs.

3.1.4. Prescribed Bounded Uncertainty Constraints

To ensure that the offline-computed decision boundary remains valid under the prescribed bounded effects, a bounded uncertainty variable is incorporated into the closed-loop polynomial system. Environmental disturbances, modeling errors, and Taylor truncation errors can be accommodated through this unified representation when their corresponding bounds are prescribed [32]. Accordingly, the closed-loop polynomial system in (12) is extended as

$$\dot{\bar{\boldsymbol{x}}}(t) \approx \boldsymbol{p}(\bar{\boldsymbol{x}}(t), t, \boldsymbol{w}), \qquad \boldsymbol{w} \in \mathcal{W}, \tag{27}$$

where $\boldsymbol{w}$ denotes the bounded term introduced for the considered uncertainty, and $\mathcal{W}$ denotes the corresponding prescribed bounded set. For different bounded effects, $\boldsymbol{w}$ and $\mathcal{W}$ are specified accordingly. The bounded set is described by polynomial inequalities as

$$\mathcal{W} = \{\boldsymbol{w} \mid g_i(\boldsymbol{w}) \geq 0, \quad i = 1, \ldots, n_w\}, \tag{28}$$

where $g_i(\boldsymbol{w})$ is the polynomial function defining the ith bound, and n_w is the number of polynomial inequalities used to describe $\mathcal{W}$.

To ensure that the funnel invariance condition holds for all bounded effects contained in $\mathcal{W}$, the boundary implication in (17) is extended as

$$V_k(\bar{\boldsymbol{x}}) = 1, \quad \boldsymbol{w} \in \mathcal{W} \;\Rightarrow\; -\dot{V}_k(\bar{\boldsymbol{x}}, \boldsymbol{w}) \geq 0, \tag{29}$$

for $k = 1, \ldots, N-1$. Under the prescribed bounded uncertainty, the derivative of the level-set function is approximated as

$$\begin{aligned} \dot{V}_k(\bar{\boldsymbol{x}}, \boldsymbol{w}) &= 2\bar{\boldsymbol{x}}^\top \boldsymbol{P}_k \dot{\bar{\boldsymbol{x}}} + \bar{\boldsymbol{x}}^\top \dot{\boldsymbol{P}}_k \bar{\boldsymbol{x}} \\ &\approx 2\bar{\boldsymbol{x}}^\top \boldsymbol{P}_k \boldsymbol{p}(\bar{\boldsymbol{x}}, t_k, \boldsymbol{w}) \\ &\quad + \frac{V_{k+1}(\bar{\boldsymbol{x}}) - V_k(\bar{\boldsymbol{x}})}{t_{k+1} - t_k}. \end{aligned} \tag{30}$$

Using the S-procedure, the implication in (29) is imposed through the following SOS constraint:

$$\begin{aligned} &-\dot{V}_k(\bar{\boldsymbol{x}}, \boldsymbol{w}) - L_k(\bar{\boldsymbol{x}}, \boldsymbol{w})\big(1 - V_k(\bar{\boldsymbol{x}})\big) \\ &- \sum_{i=1}^{n_w} L_{Wi,k}(\bar{\boldsymbol{x}}, \boldsymbol{w}) g_i(\boldsymbol{w}) \text{ is SOS.} \end{aligned} \tag{31}$$

Here, $L_k(\bar{\boldsymbol{x}}, \boldsymbol{w})$ is a polynomial multiplier associated with the funnel boundary condition. The multiplier polynomials associated with the prescribed bounded set satisfy

$$L_{Wi,k}(\bar{\boldsymbol{x}}, \boldsymbol{w}) \text{ is SOS}, \qquad i = 1, \ldots, n_w. \tag{32}$$

For the funnel computation under prescribed bounded uncertainties, the robust SOS constraint in (31) is used in place of the deterministic invariance constraint in (19). It requires the funnel invariance condition to hold for every bounded effect contained in $\mathcal{W}$. When no uncertainty is considered, the robust formulation reduces to the deterministic invariance condition. In the numerical study, the heave component of deck motion is treated as a representative bounded environmental disturbance, and its effect is incorporated into the longitudinal dynamics before the corresponding closed-loop polynomial system and funnel are recomputed offline. Other bounded environmental effects, such as atmospheric disturbances, can be incorporated through the same formulation when their corresponding models and admissible bounds are specified.

Together with the funnel terminal-set inclusion condition and actuator saturation constraints, the prescribed bounded uncertainty condition ensures that a complete six-dimensional state deviation inside the corresponding funnel is associated with a closed-loop trajectory that remains within the subsequent funnel, satisfies the actuator saturation constraints, and reaches the terminal set under the prescribed uncertainty bounds. The prescribed bounded set affects the shape matrices $\boldsymbol{P}_k$ computed offline and hence the resulting time-varying decision boundary, but it does not change the go-around decision rule. If the actual uncertainty exceeds the prescribed bounds, this guarantee is no longer established. If the resulting complete six-dimensional state deviation leaves the funnel, the online go-around decision module detects the membership violation and outputs the go-around decision.

3.1.5. Offline Decision-Boundary Optimization Formulation

To reduce conservatism in online go-around decision-making, the offline computation seeks to enlarge the six-dimensional funnel while satisfying the terminal-set inclusion condition, funnel invariance condition, actuator saturation constraints, and prescribed bounded uncertainty constraints. Since the funnel defines the continue-landing decision region and its boundary separates the continue-landing and go-around decisions, enlarging the funnel increases the range of complete six-dimensional state deviations for which the continue-landing decision can be supported by the offline-computed conditions.

The volume of the funnel ellipsoid at t_k, up to a dimension-dependent constant, is proportional to $(\det \boldsymbol{P}_k)^{-1/2}$ [33]. Directly minimizing $\log\det(\boldsymbol{P}_k)$ is nonconvex. Therefore, a fixed positive-definite scaling matrix $\boldsymbol{M}_k$ and a positive scalar σ_k are introduced through the following constraint:

$$\sigma_k \boldsymbol{M}_k \succeq \boldsymbol{P}_k, \qquad k = 1, \ldots, N-1. \tag{33}$$

This condition guarantees that the ellipsoid defined by $\sigma_k \boldsymbol{M}_k$ is contained within the funnel $F(t_k)$. For a fixed $\boldsymbol{M}_k$, minimizing σ_k enlarges this inner ellipsoid and therefore promotes enlargement of the corresponding funnel and its continue-landing decision region. In this study, $\boldsymbol{M}_k$ is chosen as the identity matrix $\boldsymbol{I}_n$ and remains fixed during the optimization, where $n = 6$ for the complete six-dimensional formulation.

For the backward computation, the terminal shape matrix is fixed as $\boldsymbol{P}_N = \boldsymbol{E}$, which satisfies the terminal-set inclusion condition in (16) with equality. Based on the preceding conditions, the overall offline decision-boundary computation problem is formulated as follows:

P1

given $\quad \boldsymbol{P}_N = \boldsymbol{E}$ and $\boldsymbol{M}_k = \boldsymbol{I}_n \ (k = 1, \ldots, N-1)$

find $\quad \boldsymbol{P}_k,\ \sigma_k,\ L_k(\bar{\boldsymbol{x}}, \boldsymbol{w}) \ (k = 1, \ldots, N-1)$

$\quad L_{Wi,k}(\bar{\boldsymbol{x}}, \boldsymbol{w}) \ (i = 1, \ldots, n_w;\ k = 1, \ldots, N-1)$

$\quad L_{Lj,k}(\bar{\boldsymbol{x}}),\ L_{Uj,k}(\bar{\boldsymbol{x}}) \ (j = 1, 2;\ k = 1, \ldots, N-1)$

min $\quad \sum_{k=1}^{N-1} \sigma_k$

s.t. $\quad \boldsymbol{P}_k \in \mathbb{S}^n_{++},\ \sigma_k > 0 \ (k = 1, \ldots, N-1)$

(31), (32), (23), (24), (25), (26), and (33).

The solution of P1 determines the sequence of shape matrices $\{\boldsymbol{P}_k\}_{k=1}^N$, with $\boldsymbol{P}_N = \boldsymbol{E}$, that defines the offline six-dimensional funnel and its time-varying decision boundary. This precomputed matrix sequence is subsequently used by the online go-around decision module without solving the SOS optimization problem during online operation.

3.2. Backward Point-by-Point Decision-Boundary Computation

Solving P1 for all shape matrices $\boldsymbol{P}_k$ simultaneously introduces a large number of decision variables and constraints, makes the design of feasible initial guesses difficult, and leads to low computational efficiency. Therefore, a backward point-by-point computation strategy is adopted [32]. Because the terminal shape matrix is fixed as $\boldsymbol{P}_N = \boldsymbol{E}$, the remaining $N-1$ shape matrices can be computed successively for $k = N-1, N-2, \ldots, 1$. At each time point, $\boldsymbol{P}_{k+1}$ is treated as known, and the following local problem is solved:

P1(k)

given $\quad \boldsymbol{P}_{k+1}$ and $\boldsymbol{M}_k = \boldsymbol{I}_n$

find $\quad \boldsymbol{P}_k,\ \sigma_k,\ L_k(\bar{\boldsymbol{x}}, \boldsymbol{w})$

$\quad L_{Wi,k}(\bar{\boldsymbol{x}}, \boldsymbol{w}) \ (i = 1, \ldots, n_w)$

$\quad L_{Lj,k}(\bar{\boldsymbol{x}}),\ L_{Uj,k}(\bar{\boldsymbol{x}}) \ (j = 1, 2)$

min $\quad \sigma_k$

s.t. $\quad \boldsymbol{P}_k \in \mathbb{S}^n_{++},\ \sigma_k > 0$

(31), (32), (23), (24), (25), (26), and (33).

Problem P1(k) contains bilinear terms between the shape matrix and multiplier polynomials, such as $L_k(\bar{\boldsymbol{x}}, \boldsymbol{w})V_k(\bar{\boldsymbol{x}})$. Therefore, the shape matrix and multiplier polynomials cannot be optimized simultaneously within a standard SOS program. Following the bilinear alternating-search strategy in [31], the multiplier polynomials are first optimized for a fixed candidate $\boldsymbol{P}_k$, after which $\boldsymbol{P}_k$ is updated with the multipliers fixed.

For a fixed candidate $\boldsymbol{P}_k$, the invariance multiplier subproblem is formulated as P1$(k)_L^I$

given $\quad \boldsymbol{P}_{k+1}$ and $\boldsymbol{P}_k$

find $\quad L_k(\bar{\boldsymbol{x}}, \boldsymbol{w}),\ L_{Wi,k}(\bar{\boldsymbol{x}}, \boldsymbol{w}) \ (i = 1, \ldots, n_w)$, and γ_I

max $\quad \gamma_I$

s.t. $\quad -\dot{V}_k(\bar{\boldsymbol{x}}, \boldsymbol{w}) - L_k(\bar{\boldsymbol{x}}, \boldsymbol{w})\big(1 - V_k(\bar{\boldsymbol{x}})\big)$

$\quad - \sum_{i=1}^{n_w} L_{Wi,k}(\bar{\boldsymbol{x}}, \boldsymbol{w}) g_i(\boldsymbol{w}) - \gamma_I$ is SOS

$\quad L_{Wi,k}(\bar{\boldsymbol{x}}, \boldsymbol{w})$ is SOS $(i = 1, \ldots, n_w)$.

For the lower actuator bound of the jth control component, the corresponding multiplier subproblem is

$\mathrm{P1}(k)_L^{Lj}$
given $\boldsymbol{P}_k$
find $L_{Lj,k}(\bar{\boldsymbol{x}})$ and γ_{Lj}
max γ_{Lj}
s.t. $u_{j,k}(\bar{\boldsymbol{x}}) - u_{\min,j} - L_{Lj,k}(\bar{\boldsymbol{x}})\big(1 - V_k(\bar{\boldsymbol{x}})\big) - \gamma_{Lj}$ is SOS
$L_{Lj,k}(\bar{\boldsymbol{x}})$ is SOS, $\quad j = 1, 2.$

For the upper actuator bound of the jth control component, the corresponding multiplier subproblem is

$\mathrm{P1}(k)_L^{Uj}$
given $\boldsymbol{P}_k$
find $L_{Uj,k}(\bar{\boldsymbol{x}})$ and γ_{Uj}
max γ_{Uj}
s.t. $u_{\max,j} - u_{j,k}(\bar{\boldsymbol{x}}) - L_{Uj,k}(\bar{\boldsymbol{x}})\big(1 - V_k(\bar{\boldsymbol{x}})\big) - \gamma_{Uj}$ is SOS
$L_{Uj,k}(\bar{\boldsymbol{x}})$ is SOS, $\quad j = 1, 2.$

The three types of multiplier subproblems are collectively denoted by $\mathrm{P1}(k)_L$ and can be solved independently. A candidate $\boldsymbol{P}_k$ is feasible if $\gamma_I \geq 0$, $\gamma_{Lj} \geq 0$, and $\gamma_{Uj} \geq 0$ for both control components.

With all multiplier polynomials fixed, the shape-matrix update subproblem is formulated as

$\mathrm{P1}(k)_P$
given $\boldsymbol{P}_{k+1}$, $L_k(\bar{\boldsymbol{x}}, \boldsymbol{w})$, $L_{Wi,k}(\bar{\boldsymbol{x}}, \boldsymbol{w})$ $(i = 1, \ldots, n_w)$,
$L_{Lj,k}(\bar{\boldsymbol{x}})$ and $L_{Uj,k}(\bar{\boldsymbol{x}})$ $(j = 1, 2)$
find $\boldsymbol{P}_k^{\mathrm{new}}$ and σ_k
min σ_k
s.t. $\boldsymbol{P}_k^{\mathrm{new}} \in \mathbb{S}_{++}^n$, $\sigma_k > 0$
(31), (23), (25), and (33).

In $\mathrm{P1}(k)_P$, the occurrences of $\boldsymbol{P}_k$ in the listed constraints are replaced by the new decision variable $\boldsymbol{P}_k^{\mathrm{new}}$. The iterative procedure used to solve P1(k) is summarized in Algorithm 1.

Algorithm 1 Backward bilinear search for P1(k)

Step0	Initialize the point-by-point computation Set the convergence tolerance $\varepsilon > 0$, choose the initial candidate $\boldsymbol{P}_k = \boldsymbol{P}_{k+1}$, and specify a roll-back factor $b > 1$.
Step1	Obtain a feasible shape matrix Substitute $\boldsymbol{P}_{k+1}$ and the current $\boldsymbol{P}_k$ into $\mathrm{P1}(k)_L^I$, $\mathrm{P1}(k)_L^{Lj}$, and $\mathrm{P1}(k)_L^{Uj}$ and solve the multiplier subproblems. If $\gamma_I \geq 0$, $\gamma_{Lj} \geq 0$, and $\gamma_{Uj} \geq 0$ for $j = 1, 2$, the current $\boldsymbol{P}_k$ is feasible and the algorithm proceeds to Step2. Otherwise, update $\boldsymbol{P}_k = b\boldsymbol{P}_k$ and repeat Step1.
Step2	Refine the shape matrix Substitute $\boldsymbol{P}_{k+1}$ and the multiplier polynomials obtained in Step1 into $\mathrm{P1}(k)_P$ to obtain $\boldsymbol{P}_k^{\mathrm{new}}$.
Step3	Check convergence Compute the ellipsoid volumes before and after Step2, up to the common dimension-dependent constant, using $\mathrm{vol} = (\det \boldsymbol{P})^{-1/2}$. If $(\mathrm{vol}^{\mathrm{new}} - \mathrm{vol}^{\mathrm{pre}})/\lvert\mathrm{vol}^{\mathrm{pre}}\rvert \leq \varepsilon$, set $\boldsymbol{P}_k = \boldsymbol{P}_k^{\mathrm{new}}$ and terminate the computation of P1(k). Otherwise, proceed to Step4.
Step4	Update the multiplier polynomials Set $\boldsymbol{P}_k = \boldsymbol{P}_k^{\mathrm{new}}$, reinsert $\boldsymbol{P}_{k+1}$ and $\boldsymbol{P}_k$ into the multiplier subproblems, and return to Step2.

Starting from $\boldsymbol{P}_N = \boldsymbol{E}$ and applying this procedure for $k = N-1, N-2, \ldots, 1$ yields the complete matrix sequence $\{\boldsymbol{P}_k\}_{k=1}^{N}$. The entire SOS computation is performed offline. The resulting sequence defines the time-varying six-dimensional decision boundary subsequently used by the online go-around decision module.

3.3. Online Go-Around Decision Rule

During online implementation, no SOS problem is solved and the funnel is not recomputed. At each sampling instant t_k, the current complete six-dimensional state deviation is calculated as

$$\bar{\boldsymbol{x}}(t_k) = \boldsymbol{x}(t_k) - \boldsymbol{x}_{\mathrm{op}}(t_k). \tag{34}$$

The online go-around decision module then evaluates the corresponding precomputed level-set function $V_k(\bar{\boldsymbol{x}}(t_k))$ and applies the following rule:

$$\begin{cases} V_k(\bar{\boldsymbol{x}}(t_k)) \leq 1, & \text{continue-landing decision,} \\ V_k(\bar{\boldsymbol{x}}(t_k)) > 1, & \text{go-around decision.} \end{cases} \tag{35}$$

If $V_k(\bar{\boldsymbol{x}}(t_k)) \leq 1$ and the actual bounded effects remain within the prescribed uncertainty bounds used in the offline computation, the terminal-set inclusion condition, funnel invariance condition, and actuator saturation constraints guarantee that the corresponding closed-loop trajectory remains within the subsequent funnel, satisfies the actuator saturation constraints, and reaches the terminal set reflecting the required landing precision. The online module therefore outputs the continue-landing decision. If $V_k(\bar{\boldsymbol{x}}(t_k)) > 1$, this guarantee is no longer available, and the online module outputs the go-around decision. If the actual uncertainty exceeds the prescribed bounds, the offline guarantee is no longer established; any resulting violation of the funnel membership condition is detected by the same online rule.

The decision in (35) is made using the complete six-dimensional state deviation. Any lower-dimensional funnel projection or slice is used only for visualization and does not replace the complete six-dimensional membership test performed during online implementation.

4. Simulation Results and Analysis

This section evaluates the proposed safety-guaranteed go-around decision framework through numerical simulations. The numerical studies are organized to validate the main elements of the framework in the same order as they are introduced in Sections 2 and 3. Section 4.1 examines the polynomial approximation introduced in Section 2.2 and evaluates whether the resulting offline-computed funnel provides a reliable decision boundary for the original closed-loop nonlinear system. Section 4.2 examines the complete six-dimensional state representation introduced in Section 2.1 and its use in the online membership test in Section 3.3, with particular attention to the incorporation of the range state. Section 4.3 examines the treatment of prescribed bounded environmental disturbances introduced in Section 3.1.4 by recomputing the six-dimensional funnel under deck motion. Section 4.4 compares the proposed complete-state go-around decision rule with the conventional altitude-deviation-based rule using the six-dimensional funnel recomputed under the prescribed deck-motion condition. Section 4.5 examines the terminal-set condition introduced in Section 3.1.1 and analyzes how the required landing precision affects the offline-computed decision boundary and the resulting online decisions.

For each formulation or operating condition considered in this section, the corresponding funnel is computed or recomputed offline, whereas the online go-around decision module applies the same go-around decision rule in (35) to the corresponding precomputed funnel. Changes in the polynomial approximation, state dimension, prescribed disturbance condition, or required landing precision affect the offline-computed decision boundary but do not change the online decision rule. Except for the deliberately reduced five-dimensional formulation examined in Section 4.1, the online decision module evaluates the complete six-dimensional state deviation. All lower-dimensional visualizations of the six-dimensional funnel are used only to show its projections or slices and do not replace the complete six-dimensional membership test performed during online implementation.

The main parameters of the F-18 aircraft are listed in Table 1 [34]. All numerical experiments are conducted in MATLAB R2020b. GPOPS-II with SNOPT is used for offline trajectory planning, whereas YALMIP and MOSEK are used to formulate and solve the SOS-based funnel-estimation problems, respectively [35,36]. The initial and terminal conditions for offline trajectory planning are provided in Table 2. The resulting nominal state and control trajectories, obtained using GPOPS-II, are shown in Figure 3 and Figure 4, respectively, with a total duration of 25 s. Relevant parameters for funnel computation are listed in Table 3.

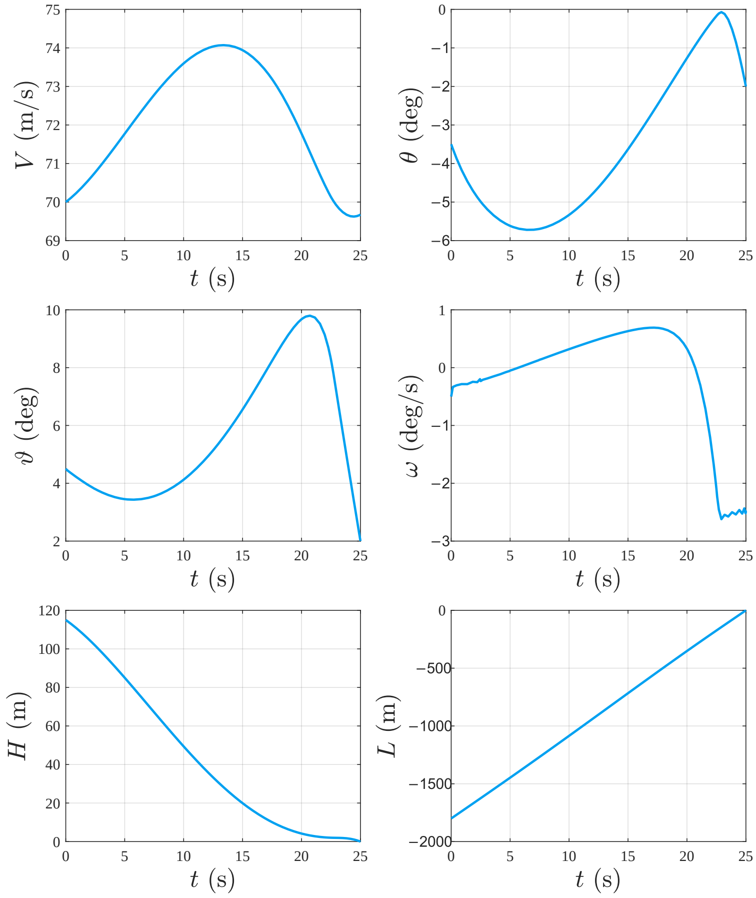


Figure 3. Nominal state trajectory generated through offline trajectory planning.

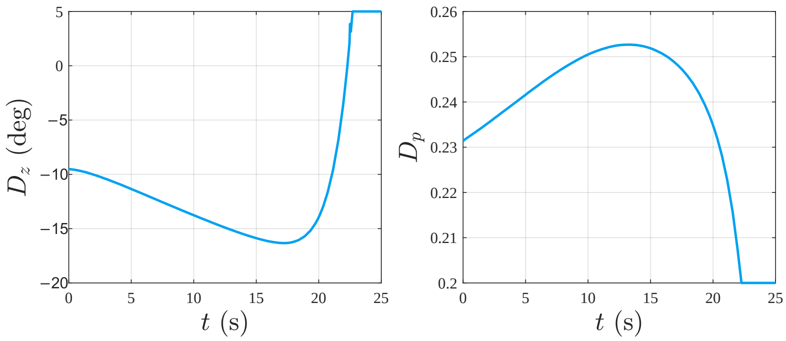


Figure 4. Nominal control trajectory generated through offline trajectory planning.

Table 2. Initial and final states for the nominal trajectory.

State Variable	Initial Value	Terminal Constraint
Airspeed V_0	$70\,\mathrm{m \cdot s^{-1}}$	$[68\,\mathrm{m \cdot s^{-1}}, 72\,\mathrm{m \cdot s^{-1}}]$
Flight path angle θ_0	$-3.5°$	$[-2°, 2°]$
Pitch angle ϑ_0	$4.5°$	$[-2°, 2°]$
Pitch rate ω_0	$-0.5\,° \cdot \mathrm{s^{-1}}$	–
Altitude H_0	115 m	0 m
Range L_0	−1800 m	0 m

Table 3. Parameter settings for funnel computation.

Parameter	Value (Range)
Number of discretization points N	51
Closed-loop elevator deflection constraint, D_z	$[-20, 10]°$
Closed-loop throttle setting constraint, D_p	[0.1, 1]
Roll-back factor b	1.5
Convergence threshold for shape matrix ε	0.02

The nominal state and control trajectories satisfy the initial and terminal constraints specified in Table 2. In particular, the altitude decreases to zero and the range approaches zero at the final time, while the terminal airspeed, flight-path angle, and pitch angle remain within their prescribed intervals. The state trajectories evolve continuously throughout the landing process, and the nominal elevator deflection and throttle setting remain within the control bounds imposed during trajectory planning. The relatively larger state and control variations near the terminal phase result from the simultaneous enforcement of the terminal position and attitude constraints.

The nominal state and control trajectories provide the reference centerline for all subsequent simulations. Specifically, they are used to construct the linear time-varying deviation model, compute the time-varying LQR gain, establish the closed-loop polynomial error dynamics, and define the nominal trajectory along which the five-dimensional and six-dimensional funnels are computed.

4.1. Effect of Polynomial-Model Accuracy on Go-Around Decisions

Conventional ACLS studies still commonly employ small-disturbance or locally linearized aircraft models, although nonlinear models have increasingly been adopted to better capture the nonlinear dynamics of carrier landing. In the present funnel computation, the first-order polynomial approximation is equivalent to such a local linearization, whereas the second-order approximation additionally retains the leading nonlinear terms. Therefore, this subsection investigates whether the accuracy of the linearized model is sufficient for constructing a reliable offline funnel decision boundary, or whether a second-order nonlinear approximation is necessary for the ACLS go-around decision problem considered in this study. To isolate the effect of polynomial-model accuracy from the effect of state dimensionality, both cases considered in this subsection use the same reduced five-dimensional formulation, in which the range state L is omitted. A first-order polynomial approximation and a second-order polynomial approximation are used to construct two five-dimensional funnels. The reliability of the resulting offline decision boundaries is then evaluated using trajectories propagated by the original closed-loop nonlinear system.

To provide a reduced benchmark consistent with conventional ACLS modeling practice, the five-state longitudinal formulation in [2] is adopted only for the comparative analysis in this subsection. The selected state variables are the scaled airspeed $v = V/V_0$, flight-path angle θ, pitch angle ϑ, pitch rate ω, and scaled altitude $h = H/V_0$. The control inputs are the elevator deflection D_z and throttle setting D_p. Here, V_0 denotes the initial airspeed of the nominal trajectory and is used as the scaling factor. The weighting matrices for the LQR controller are specified as

$$\boldsymbol{Q}_f = \boldsymbol{Q} = \mathrm{diag}([1\ 1\ 1\ 1\ 1]), \qquad \boldsymbol{R} = \mathrm{diag}([1\ 1]).$$

The corresponding LQR controller is used to construct the closed-loop error dynamics. For the reduced five-dimensional formulation, the shape matrix of the terminal set is denoted by $\boldsymbol{E}_5$ and specified as

$$\boldsymbol{E}_5 = \mathrm{diag}\left(\left[\frac{V_0^2}{4}\ \ 1\ \ 1\ \ \frac{1}{4}\ \ V_0^2\right]\right).$$

Accordingly, the terminal deviations relative to the nominal trajectory are required to remain within 2 m/s in airspeed, 1° in flight-path angle, 1° in pitch angle, 2°/s in pitch rate, and 1 m in altitude. These terminal bounds are selected as physically reasonable reference values to define the required landing precision for the numerical study. In practical applications, the terminal-set matrix can be adjusted according to the actual landing requirements, aircraft characteristics, and operational specifications.

First-order and second-order polynomial approximations of the closed-loop error dynamics with respect to the state deviation are constructed using the same LQR controller. The time-varying sizes of the resulting five-dimensional funnels are shown in Figure 5. Since the volume of a funnel ellipsoid is proportional to $(\det \boldsymbol{P}_k)^{-1/2}$, a larger value of $-\log\det \boldsymbol{P}_k$ represents a larger funnel at the corresponding time instant. Both funnels generally contract as the landing process approaches the prescribed terminal set. The funnel constructed using the first-order polynomial approximation is generally larger than that constructed using the second-order approximation. However, a larger funnel does not necessarily provide a more reliable decision boundary, because the neglected nonlinear terms may cause the first-order approximation to overestimate the safe region of the original closed-loop nonlinear system. This issue is further evaluated using the nonlinear closed-loop trajectories shown in Figure 6. At the entrance time $t_1 = 0$, 100 initial perturbation vectors are uniformly sampled within the five-dimensional funnel for each approximation case. Because these perturbation vectors lie in a five-dimensional state space, they are not directly displayed in Figure 5, which only presents the scalar quantity used to represent the funnel size. Starting from these initial states, the corresponding trajectories are propagated by numerically integrating the original closed-loop nonlinear system rather than using either polynomial approximation. Each blue curve in Figure 6 represents the level-set function history associated with one of the sampled initial perturbation vectors.

During each closed-loop simulation, the reduced five-dimensional version of the online go-around decision module operates at every sampling instant. Specifically, the module evaluates the current five-dimensional state deviation against the corresponding precomputed five-dimensional funnel and outputs either the continue-landing decision or the go-around decision according to the same membership rule as that in (35). Therefore, the level-set histories in Figure 6 directly evaluate whether the offline decision boundaries constructed from the two polynomial approximations remain valid when applied to the original closed-loop nonlinear system.

For the funnel constructed using the second-order closed-loop polynomial system, none of the level-set function values in Figure 6b exceed the funnel boundary $V_k = 1$. Consequently, the online decision module continuously outputs the continue-landing decision throughout the simulation. All trajectories subsequently reach the prescribed five-dimensional terminal set at the final time, supporting the validity of these continue-landing decisions with respect to the reduced five-dimensional formulation. The corresponding elevator deflection and throttle setting are shown in Figure 7a and Figure 7b, respectively, and both remain within the saturation limits specified in Table 3.

For the funnel constructed using the first-order polynomial approximation, several trajectories of the original closed-loop nonlinear system in Figure 6a exceed the funnel boundary $V_k = 1$. According to the go-around decision rule in (35), the online decision module outputs the go-around decision for each of these trajectories at the first sampling instant for which $V_k(\bar{x}(t_k)) > 1$, because the guarantee that the trajectory will remain within the subsequent funnel and reach the prescribed terminal set is no longer available. Therefore, Figure 6a explicitly demonstrates the online transition from the continue-landing decision to the go-around decision. The trajectories are propagated beyond the triggering instant only to evaluate the discrepancy between the first-order closed-loop polynomial system and the original closed-loop nonlinear system. This continued propagation does not represent execution of the post-trigger go-around maneuver, which is outside the scope of this study. The first-order approximation produces a funnel that extends into regions farther from the nominal trajectory, where the nonlinear terms neglected by the approximation become significant. Consequently, the resulting funnel does not provide a reliable offline decision boundary for the original closed-loop nonlinear system.

Although the second-order polynomial model provides a reliable offline decision boundary with respect to the five modeled states, the reduced formulation does not evaluate the range state. Figure 8 and Table 4 summarize the terminal altitude and range dispersions for the five-dimensional formulation considered in this subsection and the six-dimensional formulation introduced in Section 4.2. The present subsection focuses on the five-dimensional results. Although the terminal altitude remains within the corresponding terminal requirement, the terminal range exhibits large deviations, with some values exceeding 20 m. This occurs because the range state L is excluded from the five-dimensional formulation. Therefore, the continue-landing decisions based on the five-dimensional funnel are valid only with respect to the five modeled states and cannot guarantee satisfaction of the additional terminal range requirement considered in this study. This limitation motivates the incorporation of the range state and the construction of the six-dimensional funnel in the following subsection.

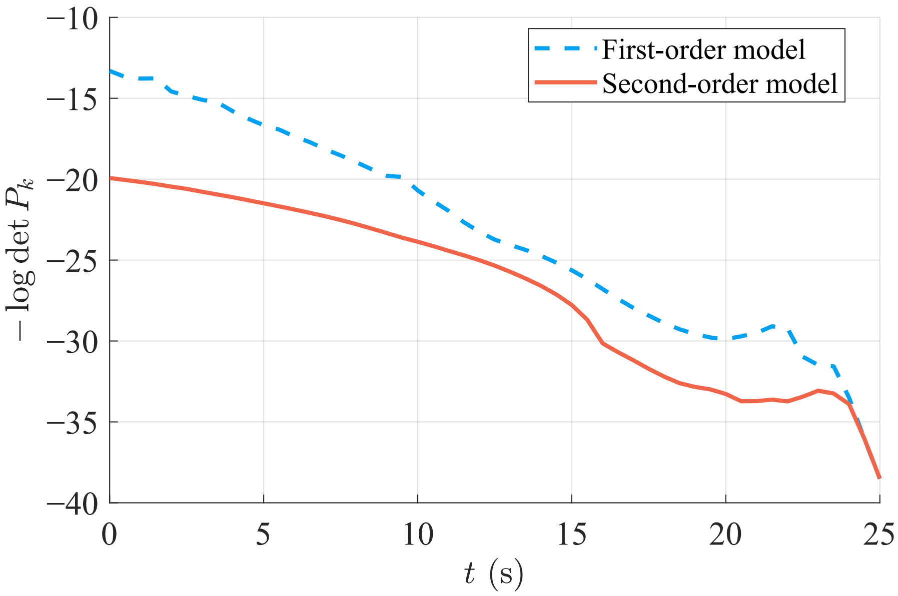
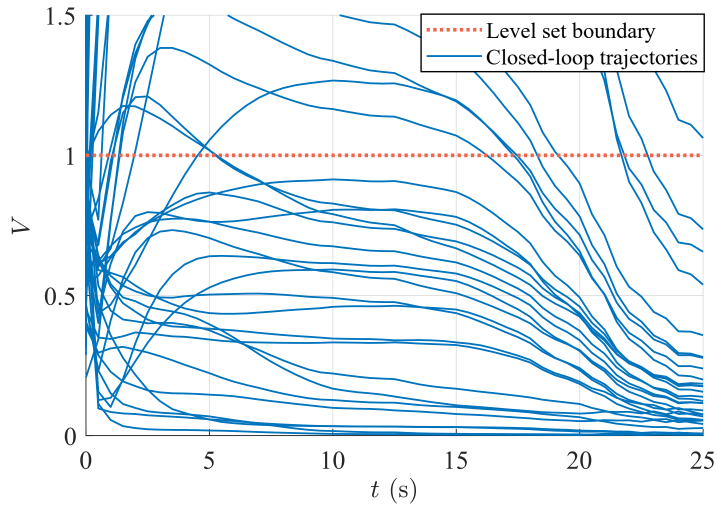
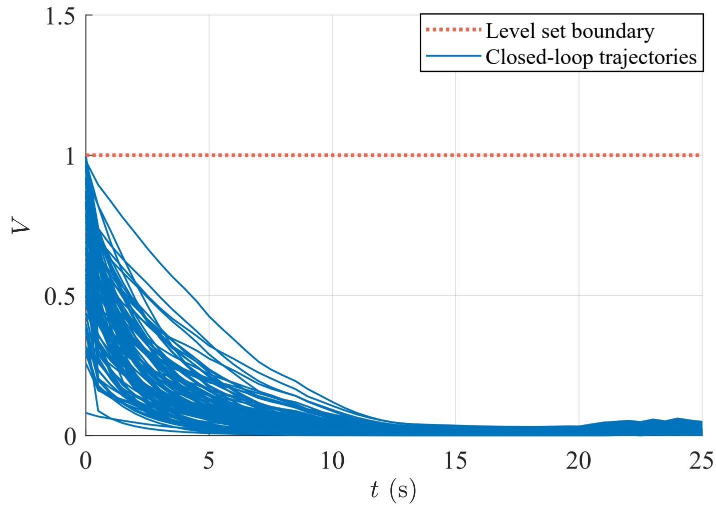


Figure 5. Time-varying sizes of the five-dimensional funnels constructed using the first-order and second-order closed-loop polynomial systems ($-\log\det \boldsymbol{P}_k$ is used to quantify the funnel size at each time instant).

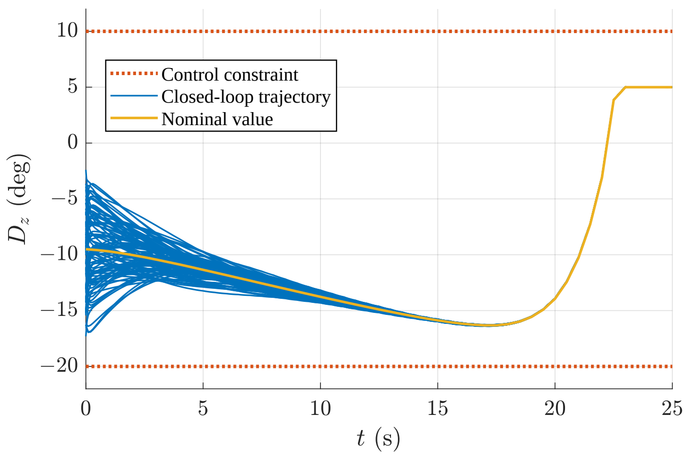


(**a**)



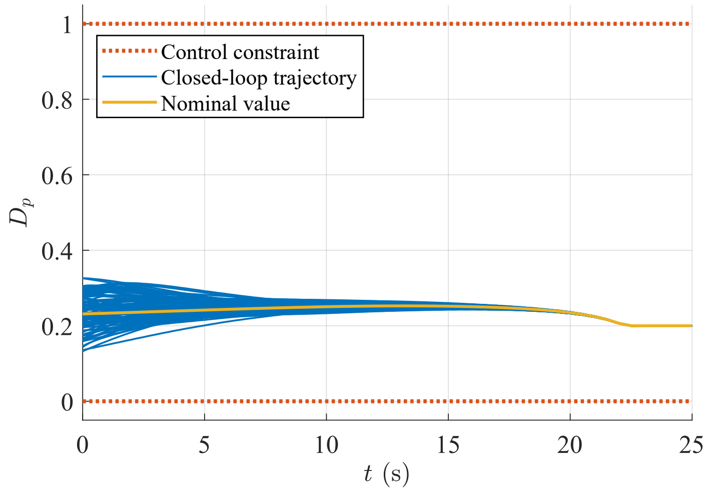
(**b**)

Figure 6. Level-set function values evaluated along trajectories of the original closed-loop nonlinear system. (**a**) Using the funnel constructed from the first-order closed-loop polynomial system. (**b**) Using the funnel constructed from the second-order closed-loop polynomial system.



(**a**) Elevator deflection.

Figure 7. *Cont.*



(**b**) Throttle setting.

Figure 7. Control inputs along trajectories of the original closed-loop nonlinear system initialized from perturbations uniformly sampled within the funnel constructed from the second-order closed-loop polynomial system.

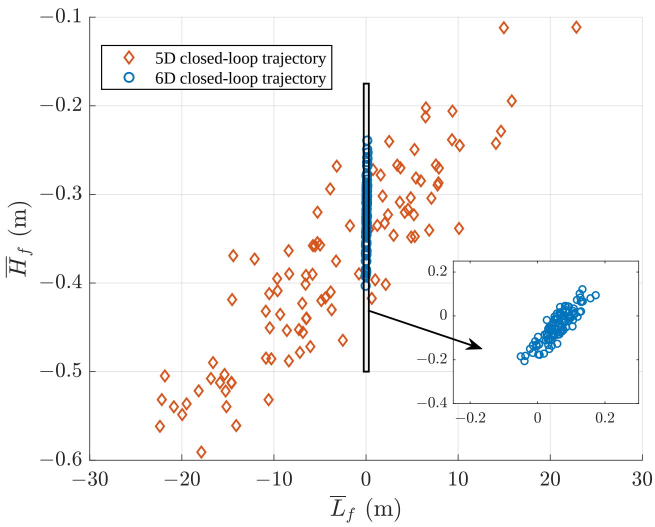


Figure 8. Terminal altitude and range dispersions under the five-dimensional and six-dimensional formulations.

Table 4. Terminal altitude and range dispersion statistics of the closed-loop trajectories (m).

Formulation	State	Mean	Max.	Standard Deviation
5D	Altitude H	0.213	0.582	0.154
5D	Range L	8.486	22.841	5.765
6D	Altitude H	0.069	0.207	0.052
6D	Range L	0.065	0.173	0.039

4.2. Effect of the Range State on Go-Around Decisions

The results in Section 4.1 show that a reliable five-dimensional decision boundary remains insufficient for evaluating the complete terminal set considered in this study because the range state is omitted. Therefore, this subsection incorporates the range state L into the state-feedback design, terminal set, and online membership test. The resulting six-dimensional funnel enables the online go-around decision module to evaluate the complete six-dimensional state deviation and determine whether the corresponding closed-loop trajectory is guaranteed to remain within the subsequent funnel and reach the prescribed terminal set, including the additional terminal range requirement considered in this study.

Consistent with the scaling adopted in Section 4.1, the range state is scaled as $l = L/V_0$. The six-dimensional state used for the funnel computation is therefore composed of the scaled airspeed $v = V/V_0$, flight-path angle θ, pitch angle ϑ, pitch rate ω, scaled altitude $h = H/V_0$, and scaled range $l = L/V_0$. The weighting matrices for the LQR controller are specified as

$$\boldsymbol{Q}_f = \boldsymbol{Q} = \mathrm{diag}([1\ 1\ 1\ 1\ 1\ 1]), \qquad \boldsymbol{R} = \mathrm{diag}([1\ 1]).$$

Based on the findings in Section 4.1, a second-order polynomial approximation of the resulting closed-loop error dynamics is constructed and used for the offline computation of the six-dimensional funnel. The shape matrix of the corresponding six-dimensional terminal set is specified as

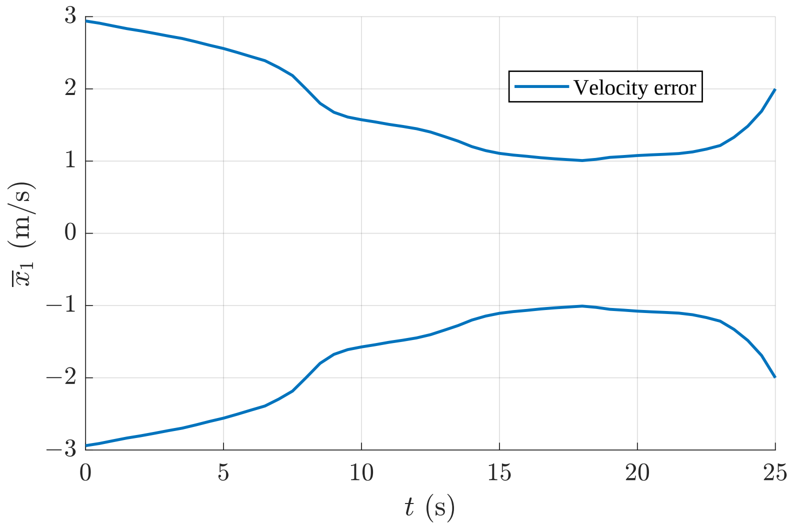
$$\boldsymbol{E} = \mathrm{diag}\left(\begin{bmatrix} \frac{V_0^2}{4} & 1 & 1 & \frac{1}{4} & V_0^2 & \frac{V_0^2}{4} \end{bmatrix}\right).$$

Accordingly, the terminal deviations relative to the nominal trajectory are required to remain within 2 m/s in airspeed, 1° in flight-path angle, 1° in pitch angle, 2°/s in pitch rate, 1 m in altitude, and 2 m in range.

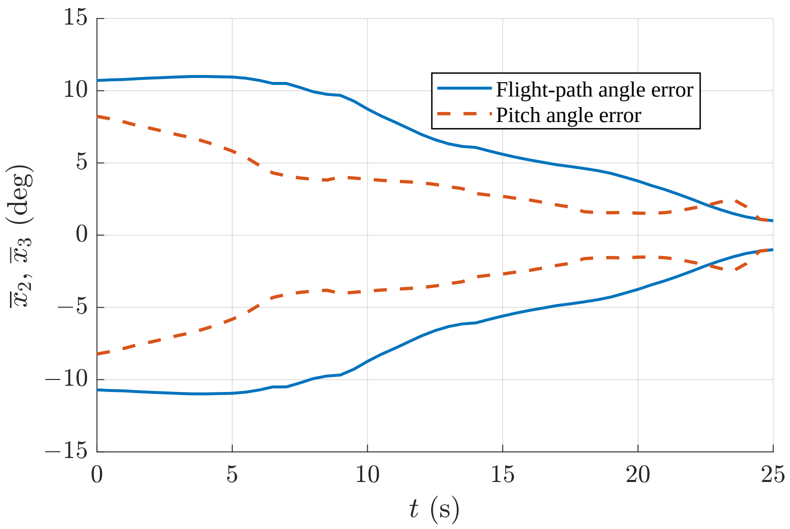
Figure 9 shows the one-dimensional projections of the resulting six-dimensional funnel onto selected state axes. The projections indicate that the offline-computed funnel entrance accommodates deviations of approximately 3 m/s in airspeed, 30 m in altitude, and 10 m in range along the corresponding coordinate directions. These values characterize the projections of the six-dimensional safe region and should not be interpreted as independent componentwise thresholds. As in Section 4.1, 100 initial perturbation vectors are uniformly sampled within the six-dimensional funnel at the entrance time $t_1 = 0$. Starting from these initial states, the corresponding trajectories are propagated using the original closed-loop nonlinear system. At every sampling instant, the online go-around decision module evaluates the complete six-dimensional state deviation according to (35).

Note 1. At a given time, a state component may lie within the corresponding one-dimensional funnel projection even though the complete six-dimensional state deviation lies outside the funnel. This is analogous to a point whose individual coordinates lie within the coordinate projections of a tilted ellipse, while the complete point still lies outside the ellipse. Therefore, the online go-around decision is made using the level-set function evaluated with the complete six-dimensional state deviation rather than any individual funnel projection.

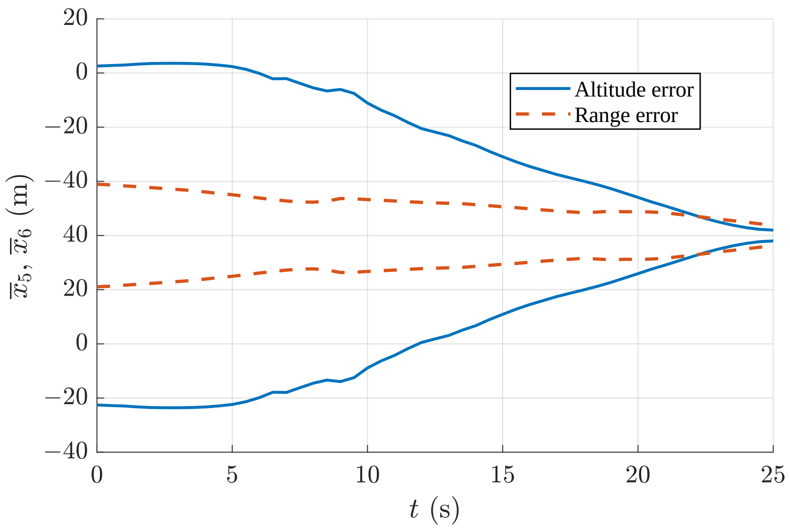
Throughout the simulations, none of the sampled trajectories exceeds the six-dimensional funnel boundary. The online go-around decision module therefore continuously outputs the continue-landing decision. Figure 8 and Table 4 show that the trajectories subsequently satisfy both the terminal altitude and terminal range requirements. The statistics in Table 4 are used only to evaluate the terminal performance of the closed-loop trajectories and are not involved in the online go-around decision, which is made exclusively according to the six-dimensional funnel membership condition in (35). Moreover, all six terminal state deviations remain within the prescribed six-dimensional terminal set. These results support the validity of the continue-landing decisions under the considered conditions: the corresponding trajectories remain within the six-dimensional funnel and reach the prescribed terminal set, which encodes the required landing precision together with the additional terminal range requirement considered in this study.



(**a**) Airspeed-error projection.



(**b**) Flight-path-angle and pitch-angle projections.



(**c**) Altitude-error and range-error projections.

Figure 9. Selected one-dimensional projections of the six-dimensional funnel.

For this five-dimensional versus six-dimensional comparison, the aircraft parameters, nominal trajectory, second-order polynomial approximation, actuator saturation constraints, terminal requirements for the five common states, and numerical settings are kept consistent. The formulation change is the incorporation of the range state and its corresponding terminal requirement into the state-feedback design, terminal set, and online membership test. Under this controlled comparison, incorporating the range state substantially reduces the terminal altitude and range dispersions. For altitude H, the mean, maximum, and standard deviation decrease from 0.213 m, 0.582 m, and 0.154 m to 0.069 m, 0.207 m, and 0.052 m, corresponding to reductions of approximately 68%, 64%, and 66%, respectively. For range L, the corresponding values decrease from 8.486 m, 22.841 m, and 5.765 m to 0.065 m, 0.173 m, and 0.039 m, representing reductions of approximately 99% in the mean, maximum, and standard deviation.

The five-dimensional benchmark adopted in Section 4.1 follows a conventional longitudinal tracking formulation centered on airspeed, flight-path and attitude variables, and altitude, in which range is not included as an independent feedback state. The present controlled comparison shows that, when terminal range accuracy is explicitly included in the landing requirements, incorporating the range state into the state-feedback design, terminal set, and online membership test can substantially improve terminal landing precision. Moreover, omission of the range state can result in a continue-landing decision even though the terminal range requirement is not satisfied. Consequently, the offline-computed six-dimensional funnel not only improves the terminal altitude and range performance under the considered simulation settings, but also enables the online go-around decision to account directly for the additional terminal range requirement.

4.3. Effect of Deck Motion on Go-Around Decisions

Based on the six-dimensional formulation established in Section 4.2, this subsection investigates the effect of deck motion on the offline-computed decision boundary and the corresponding online go-around decisions. Deck motion is treated as a representative carrier-specific bounded environmental disturbance within the prescribed bounded uncertainty framework established in Section 3.1.4 [34]. Its heave component directly changes the relative altitude between the aircraft and the carrier deck during the terminal landing phase. Therefore, deck motion is selected in this subsection to illustrate how a prescribed environmental disturbance affects the offline-computed six-dimensional funnel and the corresponding online go-around decisions. The prescribed heave component of deck motion is described as

$$\dot{h}_{\text{deck}} = 1.219 \sin\left(0.6t + \frac{\pi}{2}\right) + \frac{0.707 \times 90}{57.3} \sin(0.6t).$$

From the perspective of the relative motion between the aircraft and the carrier deck, the heave component of deck motion is incorporated into the nonlinear longitudinal model. Accordingly, the altitude dynamics in (1) are modified as

$$\dot{H} = V \sin\theta - \dot{h}_{\text{deck}}.$$

For the deck-motion case, the shape matrix of the six-dimensional terminal set is denoted by $\boldsymbol{E}_{\text{deck}}$ and specified as

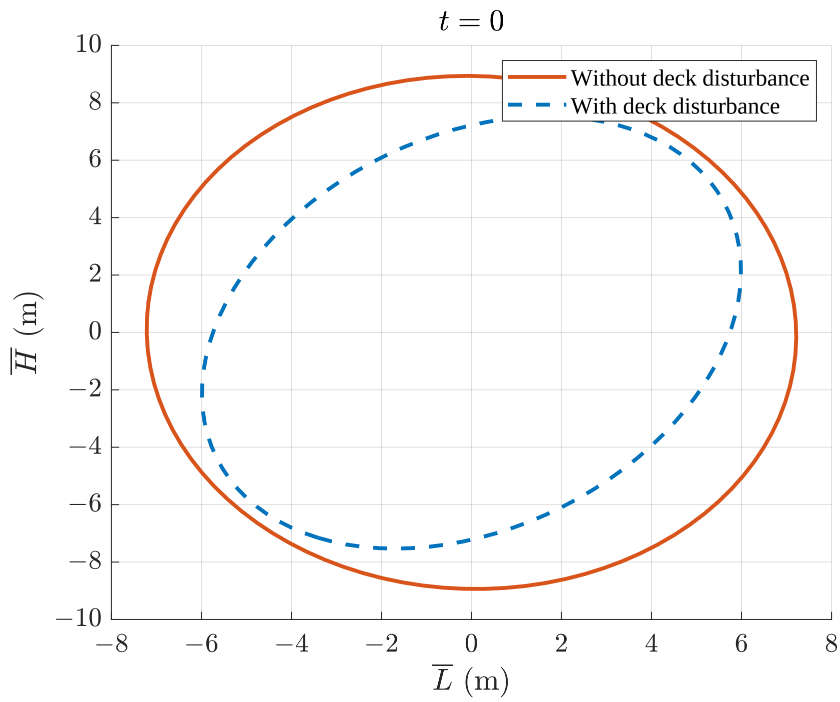
$$\boldsymbol{E}_{\text{deck}} = \operatorname{diag}\left(\begin{bmatrix} \frac{V_0^2}{4} & 1 & 1 & \frac{1}{4} & \frac{V_0^2}{16} & \frac{V_0^2}{4} \end{bmatrix}\right).$$

Accordingly, the terminal deviations relative to the nominal trajectory are required to remain within 2 m/s in airspeed, 1° in flight-path angle, 1° in pitch angle, 2°/s in pitch rate, 4 m in altitude, and 2 m in range. The prescribed deck motion is incorporated during the interval from 22 to 25 s. Based on the modified nonlinear longitudinal dynamics, the corresponding closed-loop polynomial system is reconstructed and the six-dimensional funnel is recomputed offline. For the comparison in Figure 10, the funnels with and without deck motion are computed using the same terminal set $\boldsymbol{E}_{\text{deck}}$, LQR controller, actuator saturation constraints, polynomial approximation order, and numerical settings. Therefore, the presence of the prescribed deck motion is the only difference between the two cases.

Figure 10 compares the altitude–range slices of the six-dimensional funnels computed with and without deck motion at $t = 0$ s and $t = 22$ s. These slices are used only to visualize the change in the offline-computed decision boundary. The actual continue-landing or go-around decision is still made by the online go-around decision module using the complete six-dimensional state deviation according to (35).

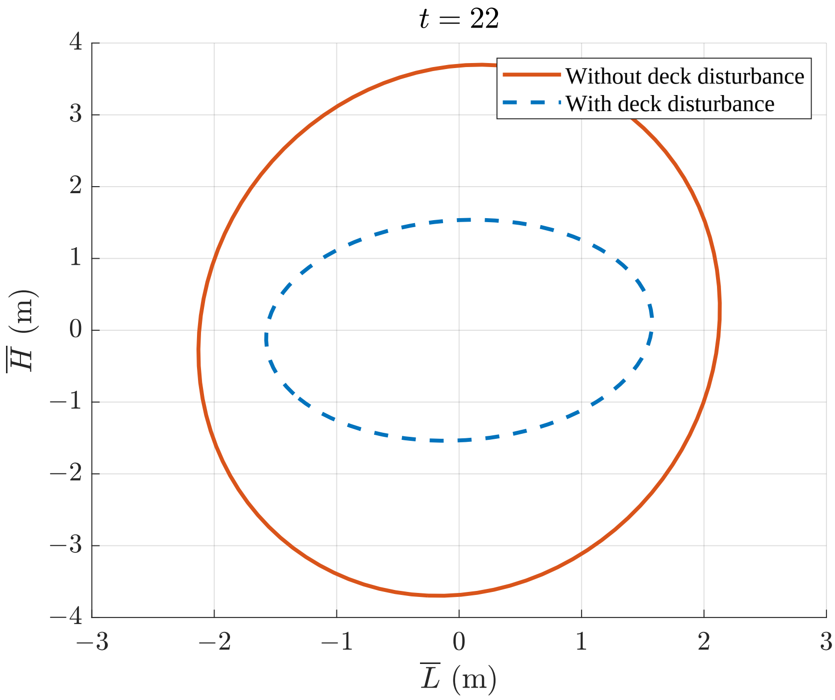
Note 2. Unlike the one-dimensional projections discussed in Section 4.2, the two-dimensional altitude–range slice in Figure 10 is obtained by setting the remaining four state deviations, namely the airspeed, flight-path-angle, pitch-angle, and pitch-rate deviations, to zero. Therefore, each point inside the slice, together with these four zero-valued state deviations, represents a complete six-dimensional state deviation inside the funnel. The slice is used only for visualization and does not replace the complete six-dimensional membership test performed by the online go-around decision module.

Figure 10 demonstrates how the proposed framework remains applicable after deck motion is incorporated into the nonlinear longitudinal model. The effect of deck motion is accounted for by recomputing the six-dimensional funnel offline, whereas the online go-around decision rule remains unchanged. As shown in Figure 10b, the six-dimensional funnel computed with deck motion is smaller at $t = 22$ s than that computed without deck motion. Because the funnel is constructed backward from the terminal set, this reduction also propagates backward to the funnel entrance, as shown in Figure 10a. Therefore, the prescribed deck motion reduces the range of complete six-dimensional state deviations for which the continue-landing guarantee is available.



(**a**) $t = 0$ s

Figure 10. *Cont.*



(**b**) $t = 22\,\mathrm{s}$

Figure 10. Comparison of the altitude–range slices of the six-dimensional funnels computed with and without deck motion.

Under the considered deck-motion condition and the other conditions imposed during the offline computation, every complete six-dimensional state deviation inside the recomputed funnel is associated with a closed-loop trajectory that remains within the subsequent funnel, satisfies the actuator saturation constraints, and reaches the prescribed terminal set. These results demonstrate that the proposed method remains applicable under the prescribed deck-motion disturbance by updating the offline-computed decision boundary. Deck motion may change the online decision corresponding to a given complete six-dimensional state deviation because the precomputed funnel changes, but it does not change the go-around decision rule itself.

If the actual deck motion exceeds the prescribed bounds, the offline guarantee is no longer established. If the resulting complete six-dimensional state deviation leaves the recomputed funnel, the online go-around decision module detects the membership violation and outputs the go-around decision according to the same rule in (35).

4.4. Comparison with Altitude-Deviation-Based Go-Around Decisions

The six-dimensional funnel used in Figure 11 is the funnel recomputed offline under the prescribed deck-motion condition examined in Section 4.3. The comparison in this subsection is intended to isolate the effect of the go-around decision criterion rather than to reproduce a specific existing ACLS implementation. Accordingly, a representative altitude-deviation-based criterion, following the conventional decision principle summarized in [3], is implemented within the same aircraft model, nominal trajectory, feedback controller, and prescribed deck-motion condition as the proposed method, so that the two decision criteria are evaluated under identical conditions. The prescribed altitude-deviation band represents the continue-landing region admitted by this altitude-deviation-based benchmark, whereas the time-varying altitude–range slices of the recomputed six-dimensional funnel represent the corresponding continue-landing region provided by the proposed method when the remaining four state deviations are set to zero.

The relationship between the two decision regions can be interpreted in three cases. First, complete six-dimensional state deviations represented inside both regions lead both decision criteria to output the continue-landing decision. Second, the portions inside the recomputed funnel but outside the prescribed altitude-deviation band correspond to complete six-dimensional state deviations whose altitude components exceed the prescribed altitude-deviation threshold while the complete-state deviations remain inside the funnel. For these states, the altitude-deviation-based benchmark outputs the go-around decision, whereas the proposed online go-around decision module outputs the continue-landing decision because the corresponding closed-loop trajectories remain covered by the offline guarantee and reach the prescribed terminal set under the considered deck-motion condition. Therefore, the go-around decisions produced by the altitude-deviation-based benchmark in these portions are unnecessary under the considered conditions. Third, complete six-dimensional state deviations outside the recomputed funnel lie outside the safety-guaranteed continue-landing region, and the proposed online go-around decision module outputs the go-around decision because the continue-landing guarantee is no longer available.

It should also be emphasized that the altitude–range slices in Figure 11 are obtained by setting the other four state deviations to zero. Therefore, the fact that the prescribed altitude-deviation band lies within the displayed slices does not imply that every complete-state deviation satisfying the altitude threshold is safe for continued landing. When nonzero deviations in airspeed, flight-path angle, pitch angle, or pitch rate cause the complete six-dimensional state deviation to lie outside the funnel, the proposed module outputs the go-around decision even if the altitude deviation remains within the prescribed altitude-deviation band. Thus, compared with the altitude-deviation-based benchmark, the proposed rule can avoid unnecessary go-around decisions caused by reliance on a single-dimensional threshold while also identifying complete-state deviations for which the continue-landing guarantee is no longer available even when the altitude deviation remains within the prescribed threshold.

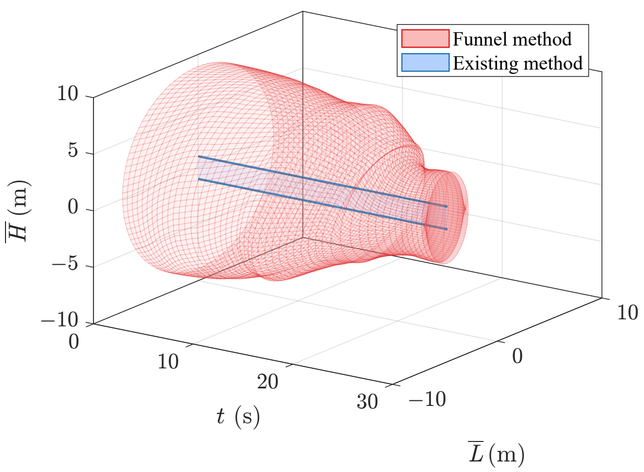


Figure 11. Decision-region comparison under the prescribed deck-motion condition between the time-varying altitude–range slices of the recomputed six-dimensional funnel and the prescribed altitude-deviation band of the existing method.

4.5. Effect of Required Landing Precision on Go-Around Decisions

This subsection investigates how the required landing precision affects the offline-computed six-dimensional decision boundary and the corresponding online go-around decisions. Three terminal sets are considered: the baseline terminal set defined by $\boldsymbol{E}$, a stricter terminal set defined by $2\boldsymbol{E}$, and a looser terminal set defined by $0.5\boldsymbol{E}$. According to the terminal-set definition in (15), multiplying the terminal-set shape matrix by a positive scalar α scales each coordinate-axis radius of the terminal ellipsoid by $1/\sqrt{\alpha}$. For the baseline terminal set $\boldsymbol{E}$ defined in Section 4.2, the corresponding coordinate-axis terminal deviation limits are 2 m/s in airspeed, 1° in flight-path angle, 1° in pitch angle, 2°/s in pitch rate, 1 m in altitude, and 2 m in range. Therefore, the stricter terminal set $2\boldsymbol{E}$ reduces these limits to approximately 1.414 m/s, 0.707°, 0.707°, 1.414°/s, 0.707 m, and 1.414 m, respectively. Conversely, the looser terminal set $0.5\boldsymbol{E}$ increases them to approximately 2.828 m/s, 1.414°, 1.414°, 2.828°/s, 1.414 m, and 2.828 m, respectively.

For each terminal set, the terminal funnel shape matrix is set to the corresponding terminal-set shape matrix, and the six-dimensional funnel is recomputed offline. The six-dimensional formulation, second-order closed-loop polynomial system, LQR controller, and actuator saturation constraints are kept unchanged from those used in Section 4.2. During online implementation, the online go-around decision module applies the same go-around decision rule in (35) to the corresponding precomputed funnel. Therefore, changing the required landing precision changes the offline-computed decision boundary but does not change the go-around decision rule itself.

Figure 12 shows the time-varying sizes of the six-dimensional funnels corresponding to the three explicitly specified levels of required landing precision. Because the closed-loop dynamics, LQR controller, actuator saturation constraints, and prescribed uncertainty conditions are identical in the three cases, the differences among the curves are caused solely by the different terminal sets. The funnel computed for the looser terminal set $0.5\boldsymbol{E}$, whose coordinate-axis terminal deviation limits are $\sqrt{2}$ times those of the baseline terminal set, has the largest size. In contrast, the funnel computed for the stricter terminal set $2\boldsymbol{E}$, whose coordinate-axis terminal deviation limits are $1/\sqrt{2}$ times the baseline values, has the smallest size. The baseline funnel corresponding to $\boldsymbol{E}$ lies between these two curves. The differences among the three funnels are relatively small during the initial interval of approximately 0–5 s, suggesting that the actuator saturation constraints on the elevator deflection and throttle setting have a stronger influence on the funnel size during this stage. After approximately 5 s, the effect of the selected terminal set becomes increasingly significant, and the ordering among the three curves becomes more pronounced. Thus, Figure 12 directly demonstrates that stricter required landing precision produces a smaller six-dimensional safe region, whereas looser required landing precision produces a larger one.

Under the conditions considered in the offline computation, the six-dimensional funnel contains the complete six-dimensional state deviations whose corresponding closed-loop trajectories remain within the subsequent funnel, satisfy the actuator saturation constraints, and reach the selected terminal set. Consequently, imposing stricter required landing precision reduces the size of the safe region for which the continue-landing guarantee is available, whereas imposing looser required landing precision enlarges this region.

At each sampling instant, the online go-around decision module evaluates the complete six-dimensional state deviation using the funnel computed for the selected terminal set. A given complete six-dimensional state deviation may lie inside the funnel computed for the looser terminal set $0.5\boldsymbol{E}$ but outside the funnel computed for the stricter terminal set $2\boldsymbol{E}$. In the former case, the online go-around decision module outputs the continue-landing decision because the corresponding closed-loop trajectory remains covered by the guarantee for reaching the looser terminal set. In the latter case, the guarantee for reaching the stricter terminal set is no longer available, and the online go-around decision module outputs the go-around decision. Thus, the same complete six-dimensional state deviation may produce different online decisions under different levels of required landing precision, even though the online go-around decision rule remains unchanged.

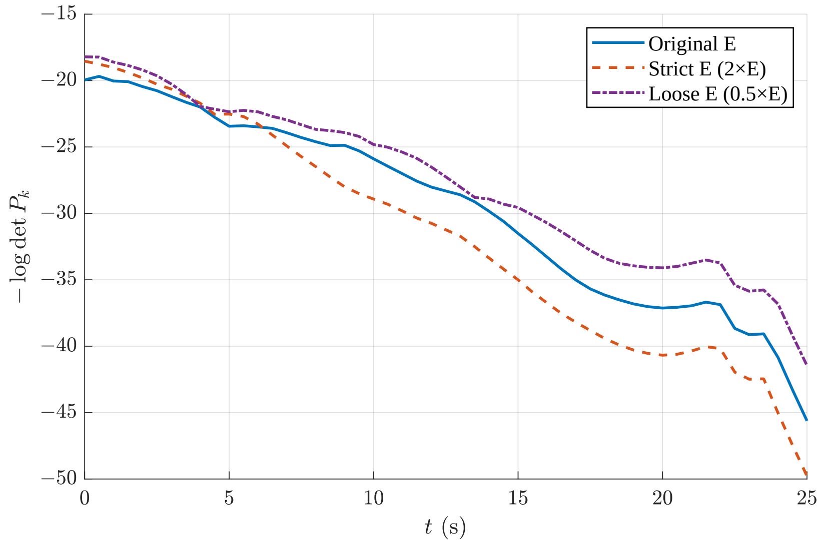


Figure 12. Time-varying sizes of the six-dimensional funnels under different levels of required landing precision.

These results demonstrate that the proposed funnel-based method explicitly captures the trade-off between the required landing precision and the safe region size of the control system. A looser terminal set produces a larger six-dimensional funnel and allows the continue-landing guarantee to remain available for a wider range of complete six-dimensional state deviations. In contrast, a stricter terminal set produces a smaller six-dimensional funnel and reduces the range of complete six-dimensional state deviations for which the continue-landing guarantee is available. Therefore, the proposed method not only provides a quantitative tool for assessing the safe region size of the control system under a given required landing precision, but also provides the corresponding offline decision boundary required for implementing the go-around decision rule online.

5. Conclusions

This paper proposes a safety-guaranteed go-around decision framework for ACLSs based on an offline-computed six-dimensional funnel along the nominal landing trajectory. The results provide several findings regarding how the go-around decision can be formulated using the complete-state safe region and how the corresponding decision boundary is affected by the modeled dynamics, operating conditions, and landing requirements. The main conclusions are as follows:

(1) The results show that the go-around decision can be formulated as a complete-state membership problem in a time-varying safe region rather than as a comparison between a single altitude deviation and a prescribed threshold. The offline-computed six-dimensional funnel incorporates the closed-loop dynamics, actuator saturation constraints, terminal landing requirements, and prescribed bounded uncertainties into the decision boundary. During online implementation, only the complete six-dimensional state deviation and the corresponding quadratic level-set function need to be evaluated. This provides a practical means of introducing a complete-state go-around decision criterion into an existing ACLS without requiring online SOS optimization or funnel recomputation.

(2) The controlled comparison between the five-dimensional and six-dimensional formulations shows that explicitly incorporating the range state can substantially improve terminal landing precision. In the proposed six-dimensional formulation, the range state and its corresponding terminal requirement are incorporated into the state-feedback design, terminal set, and online membership test. Under the considered

simulation settings, the mean, maximum, and standard deviation of the terminal altitude deviations are reduced by approximately 68%, 64%, and 66%, respectively, while the corresponding terminal range-deviation statistics are reduced by approximately 99%. These results demonstrate the benefit of explicitly including the range state when terminal range accuracy is incorporated into the complete-state go-around decision formulation.

(3) The results further show that the go-around decision boundary is not a fixed geometric threshold but depends on the fidelity of the closed-loop model, the prescribed disturbance condition, and the required landing precision. The first-order approximation can overestimate the valid continue-landing region of the original closed-loop nonlinear system, whereas the second-order approximation provides a more reliable boundary under the considered conditions. Deck motion reduces the available continue-landing region and can be incorporated by recomputing the funnel offline, while stricter terminal requirements also reduce the funnel size. Compared with the representative altitude-deviation-based criterion considered in this study, the complete-state funnel provides additional decision information: it can avoid unnecessary go-around decisions when the altitude deviation exceeds the prescribed threshold while the complete state remains inside the funnel, and it can also identify complete-state deviations for which the continue-landing guarantee is no longer available even when the altitude deviation remains within the prescribed threshold.

These findings should be interpreted within the scope of the formulated guarantee. The safety guarantee is established for the adopted closed-loop longitudinal dynamics, actuator saturation constraints, terminal landing requirements, and prescribed bounded uncertainty set. The present numerical study uses deck motion as a representative carrier-specific environmental disturbance, whereas other bounded environmental effects, such as atmospheric disturbances, can be incorporated through the same prescribed bounded uncertainty formulation when their corresponding models and admissible bounds are specified. The reliability of the offline-computed decision boundary also depends on the accuracy of the polynomial approximation, which is why the first- and second-order approximations are explicitly evaluated against the original closed-loop nonlinear system. If the actual uncertainty exceeds the prescribed bounds, the guarantee is no longer established. Therefore, the proposed framework provides an explicit complete-state continue-landing region under specified modeling and operating conditions rather than an unconditional safety guarantee beyond these conditions. Joint optimization of the feedback controller and the funnel can further enlarge the safe region of the closed-loop system and remains a direction for subsequent development.

Author Contributions: Methodology, Y.L.; writing—original draft preparation, Z.L.; writing—review and editing, W.M.; supervision, J.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the Fundamental and Interdisciplinary Disciplines Breakthrough Plan of the Ministry of Education of China (Grant No. JYB2025XDXM116).

Data Availability Statement: The original contributions presented in this study are included in the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Zhen, Z.; Wang, X.; Jiang, J.; Yang, Y. Research progress in guidance and control of automatic carrier landing of carrier-based aircraft. *Acta Aeronaut. Astronaut. Sin.* **2017**, *38*, 020435. [CrossRef]

2. Zhen, Z. Research development in autonomous carrier-landing/ship-recovery guidance and control of unmanned aerial vehicles. *Acta Autom. Sin.* **2019**, *45*, 669–681. [CrossRef]
3. Zhang, Z.; Zhen, Z. Research progress on guidance and control of fixed-wing manned and unmanned carrier-based aircraft landing. *Acta Aeronaut. Astronaut. Sin.* **2025**, *46*, 532336. [CrossRef]
4. Pravitra, J.; Clarke, J.-P.B.; Johnson, E.N. Landing a fixed-wing UAV on a moving platform: A pseudospectral optimal control approach. In Proceedings of the AIAA Aviation 2019 Forum, Dallas, TX, USA, 17–21 June 2019. [CrossRef]
5. Denison, N.A. Automated Carrier Landing of an Unmanned Combat Aerial Vehicle Using Dynamic Inversion. Master's Thesis, Air Force Institute of Technology, Wright–Patterson AFB, OH, USA, 2007.
6. Urnes, J.M.; Hess, R.K. Development of the F/A-18A automatic carrier landing system. *J. Guid. Control Dyn.* **1985**, *8*, 289–295. [CrossRef]
7. Reiner, J.; Balas, G.J.; Garrard, W.L. Robust dynamic inversion for control of highly maneuverable aircraft. *J. Guid. Control Dyn.* **1995**, *18*, 18–24. [CrossRef]
8. Singh, S.; Padhi, R. Automatic path planning and control design for autonomous landing of UAVs using dynamic inversion. In Proceedings of the 2009 American Control Conference, St. Louis, MO, USA, 10–12 June 2009; pp. 2409–2414. [CrossRef]
9. Reed, S.; Steck, J.E. Adaptive control for fault tolerant autonomous carrier recovery. In Proceedings of the AIAA Guidance, Navigation, and Control Conference, Kissimmee, FL, USA, 8–12 January 2018. [CrossRef]
10. Lu, K.; Liu, C. A L1 adaptive control scheme for UAV carrier landing using nonlinear dynamic inversion. *Int. J. Aerosp. Eng.* **2019**, *2019*, 6917393. [CrossRef]
11. Chen, M.; Zou, Q.; Jiang, C.; Wu, Q. Dynamical inversion flight control based on neural network disturbance observer. *Control Decis.* **2008**, *23*, 283–287.
12. Hu, W.; Wan, W.; Chen, M. Neural network and disturbance observer based control for automatic carrier landing of UAV. *Acta Aeronaut. Astronaut. Sin.* **2022**, *43*, 726963. . [CrossRef]
13. Ma, Y.; Guan, Z.; Zheng, Z. Nonlinear control for automatic carrier landing with deck motion compensation. In Proceedings of the 37th Chinese Control Conference (CCC), Wuhan, China, 25–27 July 2018; pp. 9883–9888. [CrossRef]
14. Zhen, Z.; Tao, G.; Yu, C.; Xue, Y. A multivariable adaptive control scheme for automatic carrier landing of UAV. *Aerosp. Sci. Technol.* **2019**, *92*, 714–721. [CrossRef]
15. Zhu, Q.; Yang, Z. Dynamic recurrent fuzzy neural network-based adaptive sliding control for longitudinal automatic carrier landing system. *J. Intell. Fuzzy Syst.* **2019**, *37*, 53–62. [CrossRef]
16. Yu, Y.; Wang, H.; Li, N.; Su, Z.; Wu, J. Automatic carrier landing system based on active disturbance rejection control with a novel parameters optimizer. *Aerosp. Sci. Technol.* **2017**, *69*, 149–160. [CrossRef]
17. Guan, Z.; Ma, Y.; Zheng, Z. Moving path following with prescribed performance and its application on automatic carrier landing. *IEEE Trans. Aerosp. Electron. Syst.* **2020**, *56*, 2576–2590. [CrossRef]
18. Zheng, Z.; Jin, Z.; Sun, L.; Zhu, M. Adaptive sliding mode relative motion control for autonomous carrier landing of fixed-wing unmanned aerial vehicles. *IEEE Access* **2017**, *5*, 5556–5565. [CrossRef]
19. Zhang, Y.; Wang, S.; Chang, B.; Wu, W. Adaptive constrained backstepping controller with prescribed performance methodology for carrier-based UAV. *Aerosp. Sci. Technol.* **2019**, *92*, 55–65. [CrossRef]
20. Guan, Z.; Ma, Y.; Zheng, Z.; Guo, N. Prescribed performance control for automatic carrier landing with disturbance. *Nonlinear Dyn.* **2018**, *94*, 1335–1349. [CrossRef]
21. Guan, Z.; Liu, H.; Zheng, Z.; Lungu, M.; Ma, Y. Fixed-time control for automatic carrier landing with disturbance. *Aerosp. Sci. Technol.* **2021**, *108*, 106403. [CrossRef]
22. Duan, H.; Yuan, Y.; Zeng, Z. Automatic carrier landing system with fixed time control. *IEEE Trans. Aerosp. Electron. Syst.* **2022**, *58*, 3586–3600. [CrossRef]
23. Yuan, Y.; Duan, H.; Zeng, Z. Automatic carrier landing control with external disturbance and input constraint. *IEEE Trans. Aerosp. Electron. Syst.* **2023**, *59*, 1426–1438. [CrossRef]
24. Zhang, Z.; Zhen, Z.; Jiang, J.; Xue, Y. Review on development in guidance and control of automatic carrier landing of carrier-based aircraft. *J. Nanjing Univ. Aeronaut. Astronaut.* **2018**, *50*, 734–744. [CrossRef]
25. Zhou, H.T.; Zheng, Z.W.; Guan, Z.Y.; Ma, Y.P. Control barrier function based nonlinear controller for automatic carrier landing. In Proceedings of the 16th International Conference on Control, Automation, Robotics and Vision (ICARCV), Shenzhen, China, 13–15 December 2020; pp. 584–589. [CrossRef]
26. Zhen, Z.; Jiang, S.; Ma, K. Automatic carrier landing control for unmanned aerial vehicles based on preview control and particle filtering. *Aerosp. Sci. Technol.* **2018**, *81*, 99–107. [CrossRef]
27. Zhen, Z.; Jiang, S.; Jiang, J. Preview control and particle filtering for automatic carrier landing. *IEEE Trans. Aerosp. Electron. Syst.* **2018**, *54*, 2662–2674. [CrossRef]
28. Zhen, Z.; Peng, M.; Xue, Y.; Jiang, S. Robust preview control and autoregressive prediction for aircraft automatic carrier landing. *IEEE Access* **2019**, *7*, 181273–181283. [CrossRef]

29. Tang, Y.; Zhu, Z.; Zhang, H.; Luo, J.; Yuan, J. A tutorial and review on robot motion planning. *Acta Aeronaut. Astronaut. Sin.* **2023**, *44*, 26495. [CrossRef]
30. Tedrake, R.; Manchester, I.R.; Tobenkin, M.; Roberts, J.W. LQR-trees: Feedback motion planning via sums-of-squares verification. *Int. J. Robot. Res.* **2010**, *29*, 1038–1052. [CrossRef]
31. Gao, D. Research on the Computation of Reachable Sets and the Feedback Motion Planning for Nonlinear Systems. Ph.D. Thesis, Northwestern Polytechnical University, Xi'an, China, 2020.
32. Liu, Y.; Wang, M.; Luo, J.; Li, J.; Cheng, X.; Ding, J.; Sun, K. A tracking guidance method with funnel scheduling for Mars landing. *Aerosp. Sci. Technol.* **2024**, *155*, 109738. [CrossRef]
33. Magnani, A.; Lall, S.; Boyd, S. Tractable fitting with convex polynomials via sum-of-squares. In Proceedings of the 44th IEEE Conference on Decision and Control and the European Control Conference 2005 (CDC-ECC'05), Seville, Spain, 12–15 December 2005; pp. 1672–1677. [CrossRef]
34. Bian, Q. Research on Flight Control Technology for Carrier-Based Aircraft Landing. Ph.D. Thesis, Northwestern Polytechnical University, Xi'an, China, 2019.
35. Löfberg, J. Pre- and post-processing sum-of-squares programs in practice. *IEEE Trans. Autom. Control* **2009**, *54*, 1007–1011. [CrossRef]
36. Andersen, E.D.; Roos, C.; Terlaky, T. On implementing a primal-dual interior-point method for conic quadratic optimization. *Math. Program.* **2003**, *95*, 249–277. [CrossRef]