doi: 10.82124/CEAS-GNC-2026-018 CEAS-GNC-2026-018

Robustness Analysis and Comparison of Linear and Nonlinear Control Approaches: an Application to Launch Vehicles

João T. Diz PhD Candidate/GNC Engineer, TU Delft/Indra Deimos, Control & Simulation/GNC Team, 2628 CD/1070-061, Delft/Lisbon, The Netherlands/Portugal. jtmarques@indracompany.com

Paulo Rosa Head of Flight Segment, Indra Deimos, Avenida Columbano Bordalo Pinheiro 85, 1070-061 Lisbon, Portugal. parosa@indracompany.com

Spilios Theodoulis Associate Professor, Technical University of Delft, 2628 CD Delft, Netherlands. s.theodoulis@tudelft.nl

Pedro Simplício GNC Engineer, European Space Agency, Keplerlaan 1, 2201 AZ Noordwijk, The Netherlands. pedro.simplicio@esa.int

Paul Acquatella GNC Engineer, European Space Agency, Keplerlaan 1, 2201 AZ Noordwijk, The Netherlands. paul.acquatella@esa.int

ABSTRACT

This paper presents a robustness assessment and a comparison of three control techniques: Nonlinear Dynamic Inversion (NDI), Incremental Nonlinear Dynamic Inversion (INDI), and $\mathcal{H}_\infty$ Loop Shaping (LS). The methods are evaluated in the context of launch vehicle (LV) attitude control, making use of linear analysis tools to assess performance and robustness. The study proceeds in two stages. First, the linear versions of NDI and INDI are used to invert the launch vehicle system. The input-output singular values of four systems are then compared: the two inverted systems, a shaped plant from an $\mathcal{H}_\infty$ LS controller, and the open-loop system. The results show that the inversion strategies naturally achieve singular value patterns similar to the $\mathcal{H}_\infty$ design with minimal designer intervention, although inherent limitations are identified. Second, an outer-loop controller is added to the linear INDI system and tuned by shaping the closed-loop singular values. The resulting controller is compared, in terms of performance and robustness, to an $\mathcal{H}_\infty$ LS controller and to a controller employing the LV state-of-the-art architecture. The analyses confirm the superiority of the $\mathcal{H}_\infty$ LS controller, and that the limitations inherent to the inversion strategies cannot be fully mitigated by outer-loop control. Nevertheless, the tuned inversion-based controller meets all design requirements and surpasses the performance of the state-of-the-art approach. Overall, this paper illustrates that NDI and INDI, when applied to LV attitude control, can achieve most — though not all — of the performance and robustness of linear robust control approaches. This is achieved with lower design complexity, highlighting the practical use of these techniques for flight control system design.

Keywords: Nonlinear Dynamic Inversion, Incremental Nonlinear Dynamic Inversion, Launch Vehicle, Robust Control, Nonlinear Control

1 Introduction

Driven by satellite constellations, commercial ventures, and international collaborations, the growing demand for launch vehicles (LVs) has intensified the need for reliable and cost-effective access to space [1]. While advances in manufacturing and agile development have reduced production costs, innovation in Guidance, Navigation, and Control (GNC) systems has lagged behind. In the context of control system design, many recent government-led LV programs, such as Ares I and VEGA, continue to rely on classical linear control approaches [2, 3]. This involves partitioning the flight envelope, designing linear controllers for each point, and applying gain scheduling [2, 4]. Although effective, these methods demand extensive validation and verification (V&V), resulting in time- and resource-intensive development cycles.

In recent years, interest in applying artificial intelligence and machine learning to LV GNC systems has grown [1, 5–7]. Despite this, industrial adoption of such data-driven or model-free approaches remains limited, primarily due to the challenges of algorithm training, validation, and certification across the full flight envelope.

In this context, nonlinear control techniques—particularly those based on Nonlinear Dynamic Inversion (NDI)—offer a practical intermediate solution. These techniques have been successfully applied in a variety of aerospace vehicles, including flexible aircraft [8], helicopters [9], spacecraft [10], and fighter jets such as the Lockheed Martin F-35 Lightning II production aircraft [11], among others. By reducing reliance on gain scheduling while preserving desired performance and robustness, NDI-based methods can be used to simplify the design of launch vehicle Thrust Vector Control (TVC) systems. Furthermore, the successful implementation of NDI-based control techniques may facilitate the eventual adoption of data-driven or model-free approaches in LV guidance, navigation, and control design.

Previous work formalized NDI and INDI control laws for launch vehicles and demonstrated their theoretical feasibility [1]. The present study extends this investigation by addressing practical aspects of TVC design through a two-stage methodology. In the first stage, the linearized NDI and INDI laws are employed to achieve dynamic inversion of an LV model. The input–output singular values of the inverted systems are then compared with those of the shaped plant obtained from an $\mathcal{H}_\infty$ loop shaping (LS) controller, as well as with those of the open-loop system. This analysis demonstrates the intrinsic loop-shaping benefits of NDI and INDI, even without any outer-loop controller, while also identifying limitations imposed by feedback linearization. In the second stage, an outer-loop controller is tuned for the linearized INDI system via closed-loop singular value shaping. The resulting controller is then benchmarked against an $\mathcal{H}_\infty$ LS controller and a controller based on the state-of-the-art LV attitude control architecture [3] in terms of performance and robustness. This two-stage assessment provides insight into the practical capabilities and limitations of NDI/INDI, and offers guidance for future controller design.

The rest of the paper is organized as follows. Section 2 describes the launch vehicle model used throughout the study. Section 3 presents a derivation of a generic NDI law and its linearized version as applied to an LV. Section 4 provides the same treatment for INDI. Section 5 compares the input-output singular values of the inverted systems, the open-loop system, and the $\mathcal{H}_\infty$ shaped system. Section 6 presents the design of the outer-loop controller for the IDI systems and evaluates its performance and robustness. Finally, Section 7 summarizes the findings and outlines directions for future work.

2 Launch Vehicle Model

For axisymmetric vehicles with quasi-zero roll rate, pitch and yaw dynamics can be assumed uncoupled and the task of control design performed for each axis separately. The relevant quantities for the pitch axis of a launch vehicle are illustrated in Figure 1. For the sake of simplicity, structural bending dynamics are not considered in this paper, but an analysis of the impact of these dynamical effects has been formalized in [12]. The nonlinear equations of motion governing the drift (z) and the pitch angle

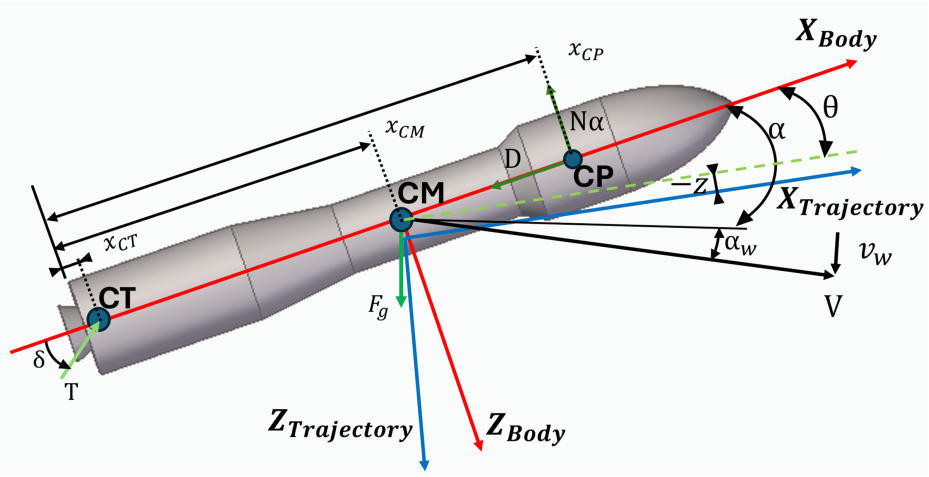


Fig. 1 Pitch plane diagram of launch vehicle.

(θ) of a launch vehicle are given by [1]:

$$\begin{aligned} m\ddot{z} &= F_\alpha + F_c + F_n - mg\sin(\theta) \\ J\ddot{\theta} &= M_\alpha + M_c + M_n \end{aligned} \tag{1}$$

where m and J are the vehicle's mass and lateral moment of inertia, respectively; while the terms F_x and M_x denote forces and moments, with subscripts α, c, and n referring to aerodynamic, TVC, and tail-wags-dog (TWD) effects, respectively.

Applying the gravity-turn assumption[1] [13] and linearizing Eq. 1 about the local trajectory reference frame following [14], yields the linearized equations of motion:

$$\begin{aligned} m\ddot{z} =& -(T-D)\theta - N_\alpha\left(\theta + \frac{\dot{z}}{V} - \frac{v_w}{V} - \frac{\dot{\theta}}{V}(x_{CP}-x_{CM})\right) - T\delta - m_n l_n \ddot{\delta} \\ J\ddot{\theta} =& N_\alpha(x_{CP}-x_{CM})\left(\theta + \frac{\dot{z}}{V} - \frac{v_w}{V} - \frac{\dot{\theta}}{V}(x_{CP}-x_{CM})\right) - T\delta(x_{CM}-x_{CT}) - (m_n l_n(x_{CM}-x_{CT}) + I_n)\ddot{\delta} \end{aligned} \tag{2}$$

where T is the thrust force, D the aerodynamic drag, and N_α the aerodynamic force gradient with respect to the angle of attack; the variables $\dot{z}$ and $\dot{\theta}$ are the drift and pitch rates; V is the ground velocity, v_w the wind disturbance; and finally, δ is the nozzle deflection. The parameters m_n and l_n are the nozzle mass and distance between the nozzle's center of mass and the thrust vector's application point. x_{CM}, x_{CT}, and x_{CP} represent the longitudinal positions of the centers of mass, thrust, and pressure, respectively.

To express the dynamics in state-space form, it is convenient to define $l_c = x_{CM} - x_{CT}$ and $l_\alpha = x_{CP} - x_{CM}$, leading to the following coefficients [15]:

$$\begin{aligned} \mu_\alpha &= l_\alpha \frac{N_\alpha}{J}, & n_\alpha &= \frac{N_\alpha}{m}, \\ \mu_c &= l_c \frac{T}{J}, & n_c &= \frac{T}{m}, \\ \mu_n &= \frac{m_n l_n l_c + J_n}{J}, & n_n &= \frac{m_n l_n}{m} \end{aligned} \tag{3}$$

[1]The gravity turn assumption states that the commanded pitch reference is set such that the centripetal acceleration of the launch vehicle equalizes the gravitational force [13].

Additionally, the contributions of the Inertial Measurement Unit (IMU) displacement are given by:

$$\begin{aligned} \theta_{IMU} &= \theta \\ \dot{\theta}_{IMU} &= \dot{\theta} \\ z_{IMU} &= z - (x_{IMU} - x_{CM})\theta \\ \dot{z}_{IMU} &= \dot{z} - (x_{IMU} - x_{CM})\dot{\theta} \end{aligned} \tag{4}$$

where x_{IMU} represents the position of the Inertial Measurement Unit. Combining Eqs. 2 – 4 yields the following Linear Time-Invariant (LTI) representation of the system[2]:

$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \\ \dot{z} \\ \ddot{z} \end{bmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ \mu_\alpha & -\frac{l_\alpha \mu_\alpha}{V} & 0 & \frac{\mu_\alpha}{V} \\ 0 & 0 & 0 & 1 \\ -n_\alpha - \frac{T-D}{m} & \frac{l_\alpha n_\alpha}{V} & 0 & \frac{-n_\alpha}{V} \end{pmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ z \\ \dot{z} \end{bmatrix} + \begin{pmatrix} 0 & 0 \\ -\mu_c & -\mu_n \\ 0 & 0 \\ -n_c & -n_n \end{pmatrix} \begin{bmatrix} \delta \\ \ddot{\delta} \end{bmatrix} + \begin{pmatrix} 0 \\ -\frac{\mu_\alpha}{V} \\ 0 \\ \frac{n_\alpha}{V} \end{pmatrix} \begin{bmatrix} v_w \end{bmatrix}$$

$$\begin{bmatrix} \alpha \\ z_{IMU} \\ \dot{z}_{IMU} \\ \theta_{IMU} \\ \dot{\theta}_{IMU} \end{bmatrix} = \begin{pmatrix} 1 & 0 & 0 & \frac{1}{V} \\ -(x_{IMU} - x_{CM}) & 0 & 1 & 0 \\ 0 & -(x_{IMU} - x_{CM}) & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ z \\ \dot{z} \end{bmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{bmatrix} \delta \\ \ddot{\delta} \end{bmatrix} + \begin{pmatrix} -\frac{1}{V} \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \begin{bmatrix} v_w \end{bmatrix} \tag{5}$$

where the angle of attack is denoted by α and is given as a function of the other states by:

$$\alpha = \theta + \frac{\dot{z}}{V} - \frac{v_w}{V} \tag{6}$$

The product of the angle of attack and the instantaneous dynamic pressure, $Q = \frac{1}{2}\rho V^2$, is commonly used in launch vehicle literature as an indicator of the aerodynamic loads acting on the vehicle structure. Here, ρ denotes the air density at the vehicle's current altitude.

Following the approach described in [16], the open-loop launch vehicle model also accounts for actuator dynamics and on-board computer (OBC) delays. Specifically, the model includes a second-order representation of the actuator dynamics (G_{TVC}) as well as a time-delay model implemented through a second-order Padé approximation (G_{Delay}). The actuator and time-delay dynamics together form the

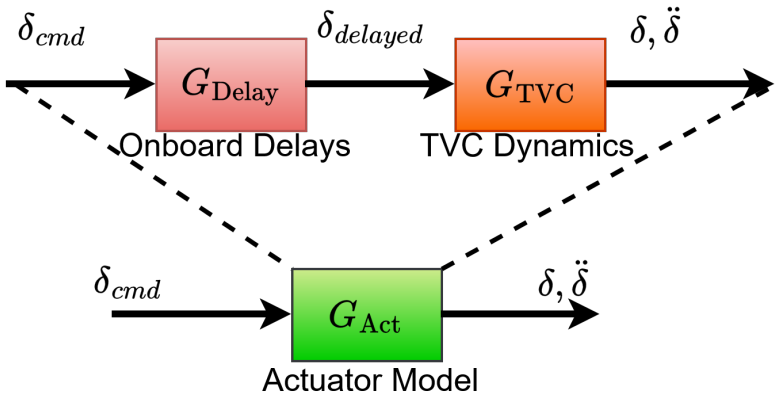


Fig. 2 Actuator Model.

[2]Note that the inputs δ and $\ddot{\delta}$ are not independent, as they both result from the actuator dynamics.

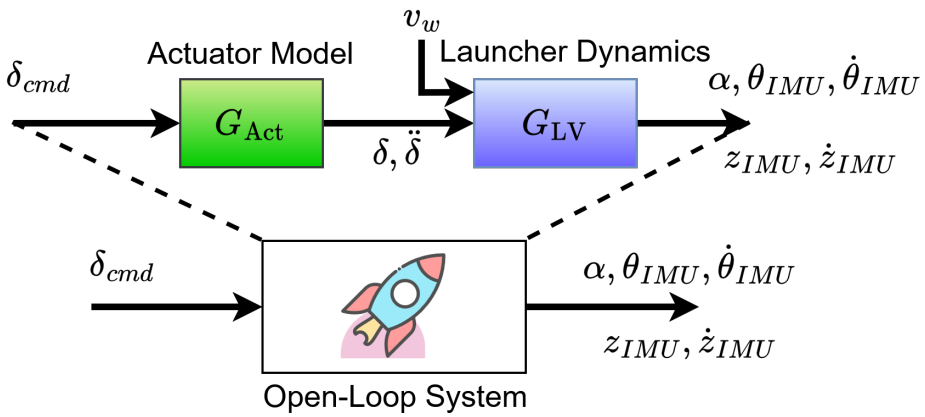


Fig. 3 Open Loop System.

block G_{Act}, as shown in Fig. 2. This block is then combined with the vehicle equations of motion (G_{LV}) to form the complete open-loop system, as illustrated in Fig. 3.

3 Nonlinear Dynamic Inversion

The underlying idea of (Incremental) Nonlinear Dynamic Inversion (NDI/INDI) is that, through appropriately designed feedback, the system nonlinearities are canceled out, yielding a closed-loop system with linear behavior (under the assumption of perfect inversion). In this linearized framework, a single linear controller can be used to specify the desired performance across the entire flight envelope, thereby eliminating—or at least significantly reducing [4]—the need for gain-scheduling.

3.1 Fundamentals of Nonlinear Dynamic Inversion

Without loss of generality, consider a nonlinear, input affine, multi-input-multi-output system, with m inputs, n states, and the same number of inputs and outputs:

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{aligned} \tag{7}$$

where $u \in \mathbb{R}^m$ represents the control input vector, $x \in \mathbb{R}^n$ represents the state vector, and $y \in \mathbb{R}^m$ is the output vector. $g(x) \in \mathbb{R}^{n \times m}$ is a matrix whose columns are assumed to be smooth vector fields, while, $f(x) \in \mathbb{R}^n$ and $h(x) \in \mathbb{R}^m$ are also assumed to be smooth vector fields. The system's relative degree vector, $\rho = [\rho_1, \ldots, \rho_m]^T$, indicates the number of times each output must be differentiated before the control input explicitly appears into the system dynamics [17, 18]. Assuming the system outputs coincide with the states themselves:

$$y = x \tag{8}$$

so that each output has relative degree one (and, consequently, the system is full-state feedback-linearizable; see [17, 18]); a single differentiation of the outputs directly yields:

$$\dot{y} = \dot{x} = f(x) + g(x)u \tag{9}$$

which shows that the control input appears explicitly after a single differentiation. Recall that the objective of NDI is to cancel the nonlinear terms so that the resulting closed-loop dynamics are linear; this is achieved by selecting u such that the effects of the terms $f(x)$ and $g(x)$ are canceled, resulting in:

$$\dot{y} = \nu \tag{10}$$

where ν is called the virtual control input, and will be the output of an outer-loop controller.

Substituting Eq. 9 into Eq. 10, and solving for u gives the NDI control law:

$$u_{\text{NDI}} = g(x)^{-1}[\nu - f(x)] \tag{11}$$

In practice, $f(x)$ and $g(x)$ are replaced by their corresponding on-board model (OBM) representations. Consequently, the quality of the inversion and the robustness of the system is highly dependent on the accuracy of the OBM. Under the assumption of no model uncertainty or external perturbations, applying the previous control law to the system in Eq. 7 yields a linear input-output relation:

$$\frac{y}{\nu} = \frac{1}{s} \tag{12}$$

A notable property of this formulation is that it not only linearizes the system, but also decouples the input-output channels [17, 18]. Consequently, each virtual input ν_i influences only the corresponding output y_i, allowing independent control design for each channel.

3.2 Dynamic Inversion for Launch Vehicles

For the linearized launch vehicle dynamics considered in this work, the NDI strategy reduces to Dynamic Inversion (DI). With this in mind, the pitch angle measured by the IMU, θ_{IMU}, is selected as the control variable. Differentiating θ_{IMU} until the control input δ appears explicitly yields:

$$\ddot{\theta}_{IMU} = \mu_\alpha \alpha - \mu_c \delta - \mu_n \ddot{\delta} \tag{13}$$

Then, the DI control law is derived by algebraically solving for the control input without accounting for the engine inertia terms. These terms are not considered as they depend on $\ddot{\delta}$, which is typically not measurable. Following this process yields:

$$\delta_{\text{DI}} = u_{\text{DI}} = -\frac{1}{\mu_c}[\nu - \mu_\alpha \alpha] = -\frac{1}{\mu_c}[\nu - \xi_{\text{DI}}] \tag{14}$$

where ξ_{DI} is the inversion path's signal. Applying this control law to the model in Eq. 5 leads to:

$$\ddot{\theta}_{IMU} = \nu - \mu_n \ddot{\delta} \tag{15}$$

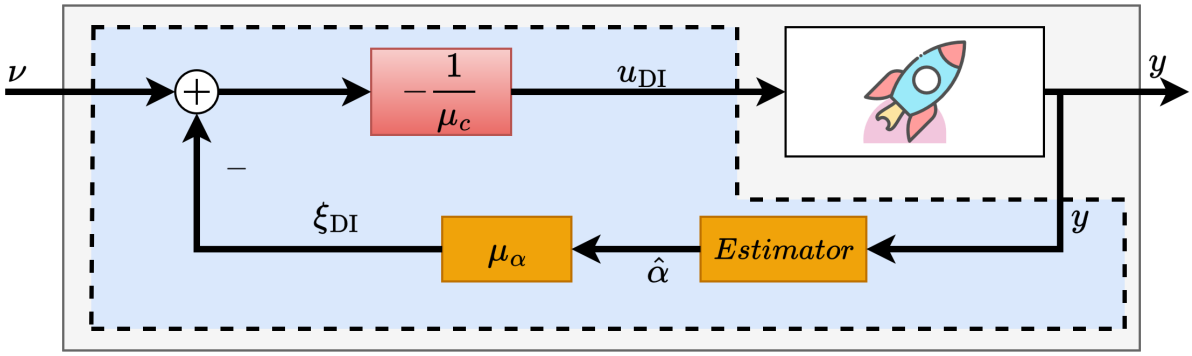


Fig. 4 Dynamic Inversion (DI) block-diagram for the LV.

which, if the engine inertia is disregarded, gives the expected double integrator relation[3]. A schematic representation of the DI implementation for the launch vehicle is shown in Figure 4. The figure highlights that the angle of attack is typically not measured directly and is instead provided by the *Estimator* block. In the present work, however, perfect measurement of the angle of attack is assumed.

4 Incremental Nonlinear Dynamic Inversion

Incremental Nonlinear Dynamic Inversion (INDI) can be regarded as the application of NDI to a system expressed in incremental form [9, 19]. While INDI is often described as inherently more robust than NDI, this is only true at low frequencies [4]. At low frequencies, its reduced reliance on the on-board model (OBM) makes the control law less sensitive to modeling errors and parameter variations, improving robustness. At higher frequencies, INDI is less robust because its inherently higher loop gain increases sensitivity to sensor noise and unmodeled dynamics, as per classical control theory (see, e.g., [20], p. 45).

From an implementation standpoint, INDI is generally less demanding than NDI, as it requires only a simplified OBM and does not rely on full state feedback. However, it requires accurate measurements or estimations of actuator deflections and angular accelerations. This shifts the design challenge from maintaining a detailed OBM to ensuring high-quality sensing and filtering, which can introduce practical complications [21].

4.1 Fundamentals of Incremental Nonlinear Dynamic Inversion

Starting from the initial nonlinear dynamics of Eq. 7, the control variables are chosen to coincide with the system states, for the same reasons as before. An incremental representation of the system can then be derived by introducing a small time delay, λ, as in [22]. Define the delayed quantities:

$$\begin{aligned} \dot{x}_0 &= \dot{x}(t-\lambda) \\ x_0 &= x(t-\lambda) \\ u_0 &= u(t-\lambda) \end{aligned} \tag{16}$$

The corresponding increments are then given by $\Delta\dot{x} = \dot{x} - \dot{x}_0$, $\Delta x = x - x_0$, and $\Delta u = u - u_0$, which represent the incremental state derivative, incremental state, and incremental control input, respectively. A first-order Taylor series expansion of $\dot{x}$ with respect to the delay λ is then performed to express the system dynamics in incremental form [1, 10, 19, 23–25]:

$$\begin{aligned} \dot{x} &= \dot{x}_0 + \frac{\partial}{\partial x}\left[f(x) + g(x)u\right]\Big|_{\substack{x=x_0 \\ u=u_0}} \Delta x + g(x_0)\Delta u + H.O.T \\ &= \dot{x}_0 + g(x_0)\Delta u + N(x,\lambda) \end{aligned} \tag{17}$$

where

$$N(x,\lambda) = \frac{\partial}{\partial x}\left[f(x) + g(x)u\right]\Big|_{\substack{x=x_0 \\ u=u_0}} \Delta x + H.O.T \tag{18}$$

which represents a residual term that includes both the Jacobian linearization of the on-board model and the higher-order terms (H.O.T.) arising from the Taylor series expansion.

Assuming that the time delay λ is sufficiently small, the time-scale separation assumption can be applied. This assumption states that the control input u varies much more rapidly than the system states

[3]Since the control variable was differentiated twice, the resulting relation between the virtual input and θ should be that of a double integrator [17].

x [1, 10, 19, 23–25]. In practice, this means there is a significant frequency separation between actuator dynamics and vehicle dynamics, which generally holds for aerospace systems [1]. Under this assumption,

$$\Delta x = x - x_0 \approx 0 \tag{19}$$

and, by neglecting the higher-order terms in Eq. 17, the incremental system reduces to [1]:

$$\dot{x} = \dot{x}_0 + g(x_0)\Delta u + \underbrace{N(x,\lambda)}_{\approx 0} \implies \Delta\dot{x} \approx g(x_0)\Delta u \tag{20}$$

Finally, assuming that $g(x_0)$ is invertible in the relevant domain, NDI can be applied to the incremental system. That is, the incremental control input Δu is computed to cancel the effects of $\dot{x}_0$ and $g(x_0)$ in Eq. 20:

$$\Delta u = g(x_0)^{-1}\left[\nu - \dot{x}_0\right] \tag{21}$$

The full INDI control input is then obtained by adding the previous actuation signal:

$$u_{\text{INDI}} = u_0 + g(x_0)^{-1}\left[\nu - \dot{x}_0\right] \tag{22}$$

In contrast to the NDI control law (Eq. 3), the INDI formulation no longer requires knowledge of the full system dynamics $f(x)$. This significantly reduces sensitivity to modeling errors and facilitates implementation. The resulting control input instead relies on the previous actuation command u_0, the control effectiveness matrix $g(x_0)$, and the measured or estimated state derivative $\dot{x}_0$. Assuming perfect measurement or estimation of these quantities, and that each control variable has relative degree one, the application of the INDI control law to the system in Eq. 20 yields:

$$\frac{y}{\nu} = \frac{1}{s} \tag{23}$$

where ν is the virtual control input. As with NDI, INDI also achieves the decoupling of the input-output channels.

4.2 Incremental Dynamic Inversion for Launch Vehicles

For the linearized launch vehicle dynamics considered in this work, the INDI strategy reduces to Incremental Dynamic Inversion (IDI). Choosing the pitch angle as sensed in the IMU, θ_{IMU}, as the control variable and substituting the corresponding launch-vehicle quantities into Eq. 22 gives:

$$\delta_{\text{IDI}} = u_{\text{IDI}} = \delta_0 - \frac{1}{\mu_c}(\nu - \ddot{\theta}_0) = \delta_0 - \frac{1}{\mu_c}(\nu - \xi_{\text{IDI}}) \tag{24}$$

where ξ_{IDI} is the inversion path's signal, and quantities with the subscript 0 denote λ-time-delayed variables. In practice, angular acceleration is typically measured or estimated only up to frequencies below the engine-inertia dynamics. As a result, high-frequency contributions from engine inertia are not canceled. Under this assumption, applying the control law to the system of Eq. 5 gives:

$$\ddot{\theta}_{IMU} = \nu - \mu_n\ddot{\delta} \tag{25}$$

If engine inertia is neglected, the standard input–output double integrator relation is recovered, as with NDI.

A schematic representation of the IDI implementation for the launch vehicle is shown in Figure 5. Two practical aspects of the implementation are highlighted. First, angular acceleration is not directly measured and must therefore be obtained using some form of filtering or estimation. In this work, a

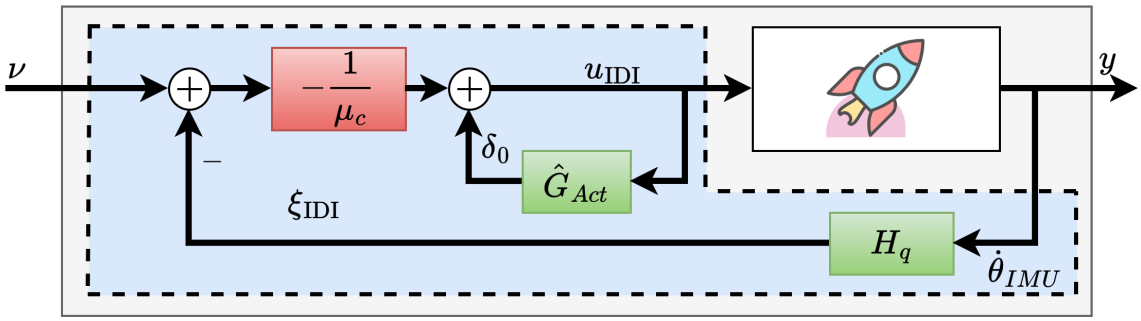


Fig. 5 Incremental Dynamic Inversion (IDI) block-diagram for the LV.

simple first-order derivative filter, denoted H_q, is employed. The filter's cutoff frequency is chosen to ensure that the inversion quality — *i.e.*, the degree to which the inverted channel approximates a chain of integrators — matches that reported in [1]. In other words, the cutoff is chosen so that the double-integrator input–output relationship is maintained over the same frequency range.

Second, the previous actuator command δ_0 is not directly measured and must be reconstructed. Here, it is obtained by filtering the IDI command through the nominal model of the actuator dynamics and time delay ($\hat{G}_{Act}$). This actuation feedback effectively introduces integral action into the control law, as discussed in [26]. While this approach provides good robustness, it reintroduces some model dependence. As an alternative, a simple low-pass filter could be used in the actuation feedback loop, which could then be tuned together with the outer-loop controller.

Although NDI and INDI are nonlinear control laws, the comparison that follows uses their linear counterparts. The linearization provides a simplified representation of the nonlinear system. Consequently, limitations identified in the linearized analysis are also expected to occur — and potentially be amplified — in the nonlinear system.

5 Loop Gain Analysis

The goal of this section is to assess the performance that can be achieved by directly applying DI/IDI to the launch vehicle, as well as to identify the limitations inherent to these methods. No outer-loop controller is designed for both the DI and IDI cases; instead, the objective is to illustrate the intrinsic loop-shaping properties of these inversion strategies.

5.1 Setup

For the subsequent analysis, the open-loop model is populated with the same nominal parameter data as studied in [14], and parametric uncertainty ranges are defined following [13]. This allows for the use of the nominal $\mathcal{H}_\infty$ Loop Shaping (LS) controller from [14] as a reference for performance comparison.

The nominal $\mathcal{H}_\infty$ LS controller was designed using the standard two-step procedure [27]. First, the open-loop system is shaped using weighting filters to meet desired performance objectives. Second, the shaped plant is robustified against Normalized Coprime Factor (NCF) uncertainty. This is accomplished by computing a stabilizing controller that minimizes the $\mathcal{H}_\infty$ norm of a specific four-block structure. Provided that the robustness margin attained in the second step is sufficiently large, the robustification largely preserves the specified loop shape and provides formal robustness guarantees. Consequently, the shaped plant can be used as a reliable baseline to evaluate the performance directly attained by DI and IDI control laws.

With this reference established, the systems considered for the input–output singular value comparison are defined as follows:

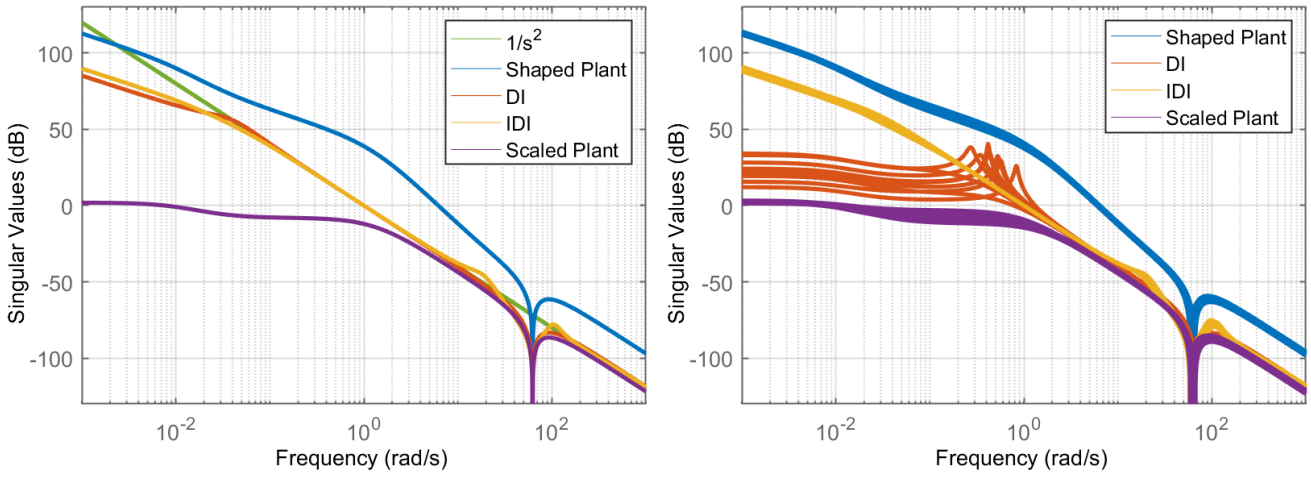
- Shaped Plant: The shaped plant of the nominal $\mathcal{H}_\infty$ Loop Shaping controller from [14]. This controller performed successfully in all time-domain and stability tests, so the shaped plant serves as a baseline for evaluating DI and IDI performance.
- Dynamic Inversion (DI): The DI control law from Eq. 14 is applied to the launch vehicle model. The input-output channels of the resulting closed-loop system (see Figure 4) are evaluated.
- Incremental Dynamic Inversion (IDI): The IDI control law from Eq. 22 is applied to the launch vehicle model (see Figure 5). Analysis is performed in the same manner as for DI.
- Scaled Plant: The open-loop model is scaled by dividing all inputs and outputs by their maximum expected value/error, as in [20]. The resulting scaled plant serves as a baseline to highlight the modifications induced by the other methods.

Finally, the subsequent analysis builds on the launch vehicle requirements, corresponding open-loop specifications and relevant frequency ranges presented in [14]. Nonetheless, it is worth noting that, as a rule of thumb, desirable loop shaping corresponds to high gain at low frequencies, low gain at high frequencies, and a slope of approximately -20 dB/dec near the crossover frequency.

5.2 Analysis Results

As a first step, Figure 6 shows the singular values (SVs) of the four systems for the transfer function (TF) from ν or δ_{cmd} to θ_{IMU}, with the nominal configuration on the left-hand side and 10 dispersed cases on the right-hand side. For the nominal case, the singular values of a double integrator ($\frac{1}{s^2}$) are also plotted, highlighting the frequency range over which 'perfect' inversion is achieved.

Several observations can be drawn from the nominal case (Figure 6a). First, regarding the inversion quality — that is, how closely the system's singular values approximate those of a double integrator — accuracy is highest in the mid-frequency range. At low frequencies, two distinct inversion errors appear. For DI (red line), the error stems from the fact that the drift dynamics—which dominate the low-frequency behavior—are not included in the inversion model. For IDI (yellow line), the low-frequency error stems from the time-scale separation assumption, which requires actuator dynamics to be much faster than the vehicle dynamics. Since real actuators are not infinitely fast, a residual inversion error is unavoidable.



(a) Nominal. **(b) Dispersed conditions.**

Fig. 6 Singular values of the transfer function from the control input to θ_{IMU}.

This can be verified by increasing the actuator model's cutoff frequency, which progressively reduces the low-frequency error and can eventually eliminate it.

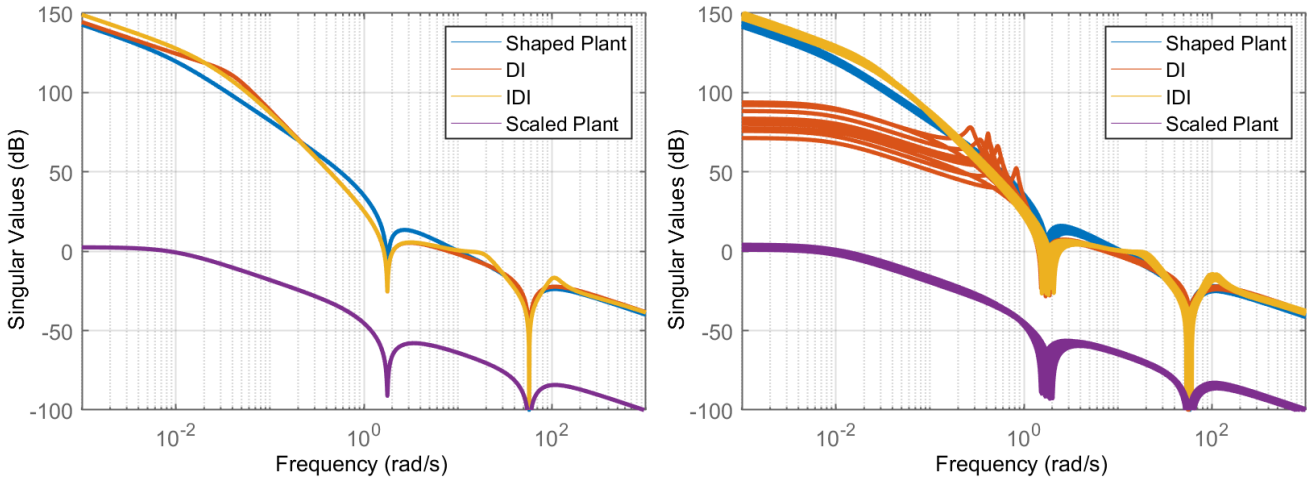
At high frequencies, the closed-loop response appears to match the double integrator (green line) almost perfectly. However, this apparent match is misleading—it merely reflects that the system's natural roll-off rate matches that of a double integrator. In reality, an inversion error is present, as indicated by the dip just below 100 rad/s, which arises from a non-minimum-phase (NMP) zero introduced by TWD effects.

Beyond inversion accuracy, further insights can be drawn from the SVs. Compared to the Scaled Plant (purple line), the DI and IDI responses have higher SVs and a steeper slope across the low- to mid-frequency range, indicating *improved shaping without requiring any manual tuning*. Compared to the Shaped Plant (blue line), they show a similar slope at low frequencies but a noticeably slower slope in the mid-frequency range. This slower slope will ultimately limit achievable aerodynamic load relief, which is a driving requirement in launcher attitude control, as will be discussed later.

The dispersed results, shown in Figure 6b, highlight the robustness of the different strategies. Both the IDI system and the Shaped Plant remain largely unaffected by parameter uncertainty, whereas DI exhibits significant sensitivity especially in the lower frequencies. Under uncertain conditions, the DI loop gain at low frequencies drops well below its nominal value, degrading the inversion seen in the nominal case. This behavior is consistent across all output channels (see Figures 7b and 8b). In essence, the IDI controller demonstrates significantly higher invariance to parameter variations compared to DI, which is known to be the main advantage of the incremental variant of DI laws [19].

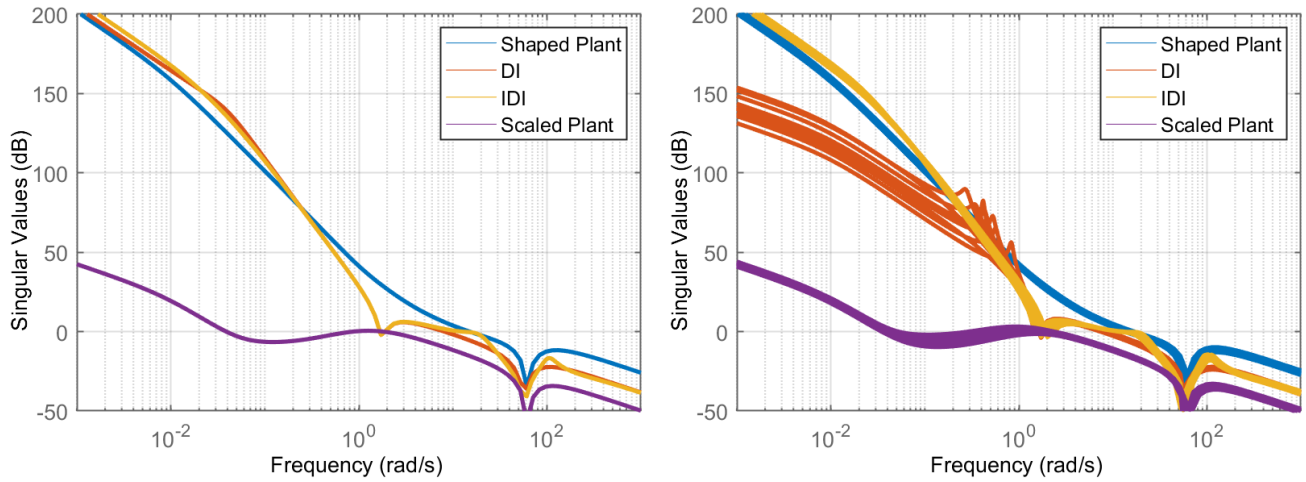
Figure 7 shows the comparison between the four systems for the transfer function from the input to $\dot{z}_{IMU}$. For the nominal case of the drift rate output (Fig. 7a), it is clear that applying the inversion alone already achieves performance close to that of the shaped plant. This is evident in both the gain increase and the change in slope at lower frequencies. It should also be noted that DI and IDI produce nearly identical results.

However, some limitations of the inversion become apparent. For both inverted systems, the slope preceding the NMP zero around 2 rad/s is steeper than that of the shaped plant, resulting in lower gain around the NMP zero. As discussed in [14], load relief is particularly important in the frequency range around the NMP zero and depends on maintaining high loop gain in the drift rate and pitch angle outputs within this range.



(a) Nominal. **(b) Dispersed conditions.**

Fig. 7 Singular values of the transfer function from the control input to lateral drift rate $\dot{z}_{IMU}$.



(a) Nominal. **(b) Dispersed conditions.**

Fig. 8 Singular values of the transfer function from the control input to all outputs.

On one hand, the drift rate channel exhibits lower loop gain in the relevant frequency range, which cannot be increased without delaying the crossover and compromising robustness. On the other hand, the pitch angle channel cannot compensate for this lower gain either, as its mid-frequency slope is smaller than that of the shaped plant (recall Fig. 6a). Increasing its gain would similarly delay crossover and reduce robustness.

A second limitation, specific to the IDI strategy, arises from the derivative filter H_q (recall Fig. 5). Figure 7a shows a peak at approximately 20 rad/s, introduced by that filter, which limits robust stability at high frequencies and makes the system more sensitive to sensor noise.

Finally, consider Figure 8a, which presents the singular values of the four systems from the input to all outputs. This figure allows for a holistic assessment of the attained system properties. Up to frequencies around 0.2 rad/s, the DI and IDI strategies closely match the response of the shaped plant. As noted in [14], this indicates that drift rate disturbance rejection will be very effective (and the same conclusion holds for the drift channel).

At slightly higher frequencies, where load relief is typically more relevant [14], the inversion strategies begin to falter. The dip observed in the loop gain arises from two factors. First, the higher slope, and consequently lower loop gain, near the NMP zero of the drift rate channel (Figure 7a). Second, the lower loop gain in the pitch angle channel (Figure 6a) and, to a lesser extent, the pitch rate channel. One might expect this could be corrected with an outer-loop controller; however, as previously discussed, the slower slope in the pitch angle channel limits outer-loop compensation. Increasing the gain via pitch angle feedback to match the desired singular value shape would require a higher crossover frequency, thus reducing robust stability. This constitutes the *main limitation of the inversion strategies* for the presented case. Moving closer to the crossover frequency, the behavior of the DI/IDI systems aligns more closely with that of the Shaped Plant. The only notable feature is the peak in the IDI gain caused by the derivative filter; however preliminary investigations suggest that higher-order derivative filters could mitigate this effect.

6 Outer-Loop Controller Design and Closed-Loop Analysis

As discussed in Section 5.2, DI is highly sensitive to parameter variations, exhibiting significant performance degradation under dispersed conditions. While DI can often be made robust through a properly designed outer-loop controller [4], IDI is inherently less sensitive to modeling errors and parameter variations. The previous section also demonstrated that, for the nominal case, both DI and IDI produce similar performance. With this in mind, subsequent analysis excludes DI and focuses exclusively on IDI. Its reduced sensitivity to modeling errors allows for a simpler outer-loop controller design. Moreover, because both approaches attained similar performance and limitations, IDI results are expected to be representative of both and sufficient to demonstrate that the limitations identified in Section 5.2 cannot be addressed through outer-loop compensation. Accordingly, Section 6.1 presents the design of the outer-loop controller for the IDI system, *i.e.*, the controller responsible for computing the virtual input ν. Section 6.2 then proceeds with a detailed comparative closed-loop analysis of different control approaches.

6.1 Design of Outer-loop Controller for the IDI System

The controller follows the standard two-PD structure on pitch angle and drift, commonly used in launch vehicle control [2, 3, 15]. The outer-loop controller, denoted as K_{Out}, is illustrated in the control tuning scheme shown in Figure 9. To tune the outer-loop controller, the methodology described in [14] is adopted. The open-loop system is initialized with the parameter values from that work, and the inputs and outputs of the tuning scheme (see Figure 9) are scaled accordingly. $\mathcal{H}_\infty$ constraints are then imposed on five transfer functions to enforce the main launch vehicle performance and robustness objectives (see [16] for a detailed description): load relief, drift and drift-rate rejection, pitch angle tracking, actuation limiting, and input stability margins. The outer-loop controller is subsequently tuned via $\mathcal{H}_\infty$ Closed Loop Shaping (CLS) to satisfy all selected constraints, as summarized in Table 1.

The numerical values of the constraints were selected following [14]. The input stability margins constraint, the tracking overshoot constraint ($||T_{\theta_{cmd}\to e_\theta}||_\infty$), the aerodynamic load constraint ($||T_{v_w\to\alpha}||_\infty$), and the TVC actuation constraint ($||T_{\theta_{cmd}\to\delta}||_\infty$) were used to reproduce the behavior of the $\mathcal{H}_\infty$ LS controller. Subsequently, the drift motion constraint ($||T_{v_w\to z_{IMU}}||_\infty$) was tightened as much as possible, while ensuring that all other optimization constraints were still satisfied, *i.e.*, that all γ values remained very close (but below) 1.

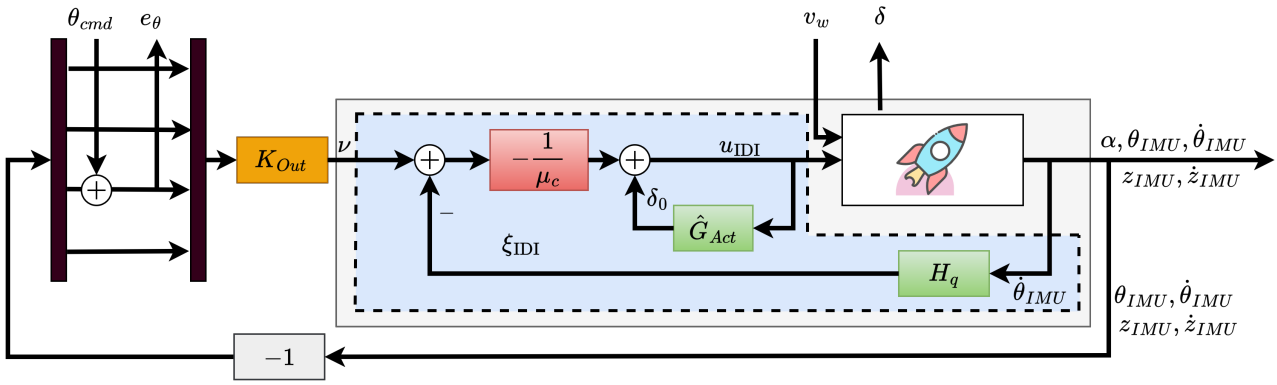


Fig. 9 Closed-loop interconnections used to design the outer-loop controller (K_{Out}) for the IDI system.

Table 1 Requirements used to design the outer-loop controller for the IDI system.

Requirement	Constraint
Max. aerodynamic load	$\|\|T_{v_w \to Q\alpha}\|\|_\infty < 2.05 \cdot 10^5$
Max. drift motion	$\|\|T_{v_w \to z_{IMU}}\|\|_\infty < 4$
Max. TVC actuation	$\|\|T_{\theta_{cmd} \to \delta}\|\|_\infty < 2.3$
Max. tracking overshoot	$\|\|T_{\theta_{cmd} \to e_\theta}\|\|_\infty < 1.37$
Min. input stability margins	Disk gain margin > 5.41 dB Disk phase margin > 32.2 deg

6.2 Comparative Analysis

In this section, three controllers are compared:

- K_{IDI-PD}: IDI with two outer-loop PDs tuned via $\mathcal{H}_\infty$ CLS (designed in Section 6.1).
- K_{PI-LL}: Nominal $\mathcal{H}_\infty$ LS controller from [14], consisting of a Proportional–Integral (PI) term, two lead–lag compensators, and two proportional gains.
- K_{PD}: Nominal $\mathcal{H}_\infty$ CLS controller from [14], composed of two PD controllers. This architecture represents the current state of the art for LV ascent attitude control [3, 14] and is used here as a baseline.

It should be noted that, although the controllers are tuned using different methods ($\mathcal{H}_\infty$ LS versus $\mathcal{H}_\infty$ CLS), [14] demonstrates that, when properly applied, both methods yield equivalent controllers with identical performance and robustness. Therefore, the comparison presented here focuses on controller architecture rather than tuning methodology.

Two assessments are performed. First, in Section 6.2.1, the input stability margins of all controllers are evaluated for both the nominal and the worst-sampled case scenarios. Second, in section 6.2.2, the linear closed-loop systems are subjected to a low-frequency, high-amplitude wind gust, using a profile commonly applied in LV controller testing [13, 16]. The wind gust has a maximum amplitude of 40 m/s, followed by a constant value of 35 m/s.

6.2.1 Robust Stability

Figure 10 shows the input Nichols plots for all three controllers. In the figure, the blue disk represents the nominal disk margin, while the blue line corresponds to the nominal Nichols plot. The red disk indicates the worst-sampled case disk margin, defining the exclusion zone based on 10,000 dispersed cases. The red line shows the Nichols plot of this worst-sampled case system, and the light blue shaded region represents the area encompassing all sampled Nichols curves. Although the figure illustrates disk margins, it is important to note that launch vehicle stability requirements are typically expressed in terms of classical gain and phase margins [3].

All controllers achieve worst-sampled case disk margins well above the required thresholds (*i.e.*, significantly higher than 3 dB and 20° [14]), with K_{PD} (right-hand side plot) showing the best margins by a small difference. This difference should not be interpreted as K_{PD} trading performance for robustness. Unlike K_{PI-LL} and K_{IDI-PD}, where further performance improvements would inevitably reduce robustness, the limiting factor for K_{PD} is actuation. Any attempt to further improve load relief or drift rejection would increase actuation, which is already substantially higher than that of the other two controllers, as will be shown in Section 6.2.2.

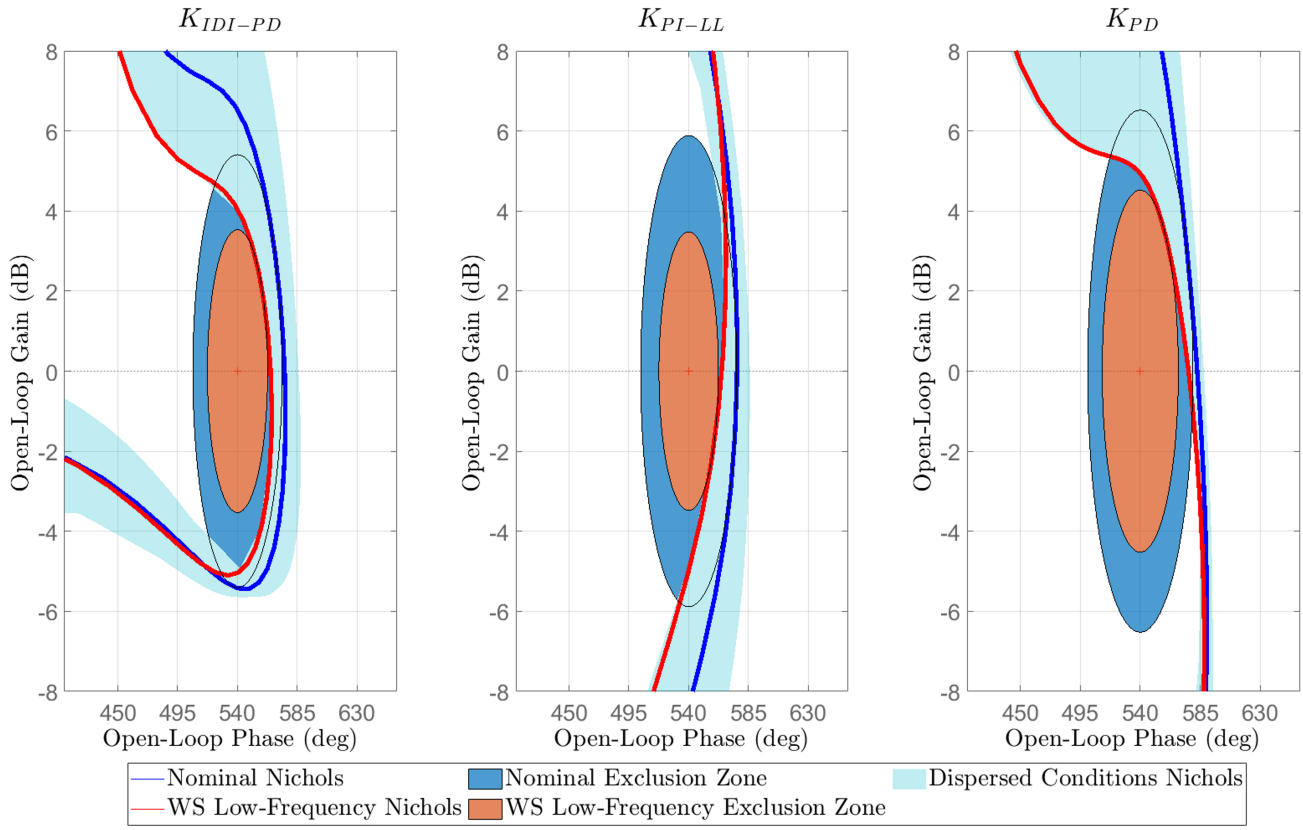


Fig. 10 Plant input Nichols plot with guaranteed disk margins.

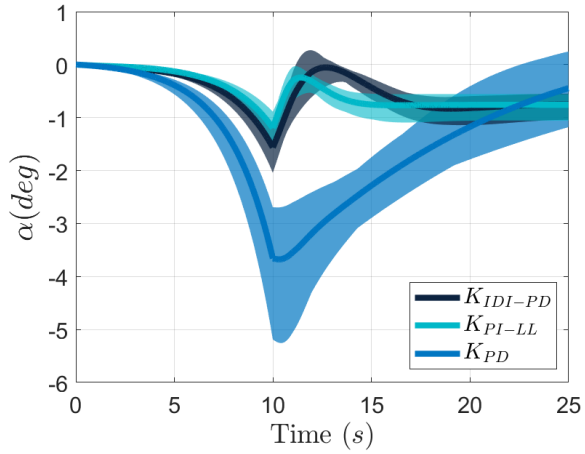
Regarding the nominal margins, K_{PD} again achieves the highest values, followed by K_{PI-LL} (middle plot) and K_{IDI-PD} (left-hand side plot). Both K_{PD} and K_{PI-LL} meet all requirements. The IDI controller meets all but one requirement: the gain margin at the second crossover of 540°, which is short by approximately 0.5 dB. Since the worst-sampled case gain margin at this frequency is quite high, and the Nichols plot shows minimal change across simulated conditions, this is not expected to pose an issue.

For the IDI controller, the gain peak associated to the derivative filter that was identified in Section 5.2 leads to a relatively small gain margin (2.53 dB under worst-sampled case conditions) at higher frequencies, outside the range shown in the Nichols plot. While this is not problematic in the present case, it should be considered in future designs, especially if flexible-body dynamics are included. In such cases, a slight reduction of high-frequency gain may be necessary to maintain sufficient margins.

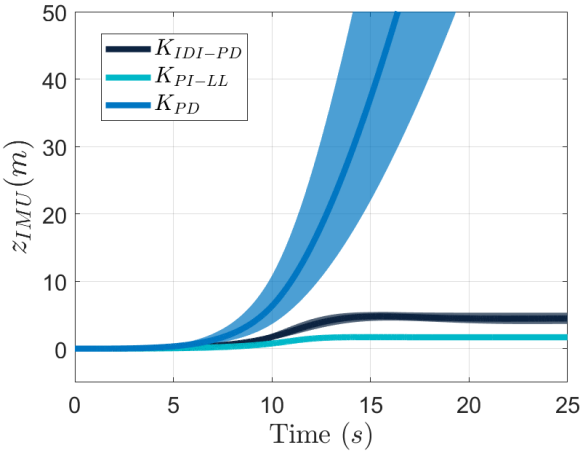
6.2.2 Robust Performance

The closed-loop performance of the controllers under the wind gust disturbance is shown in Figure 11. In this figure, thick lines correspond to the nominal response, while the shaded regions represent the results of 300 Monte Carlo simulations. All controllers satisfy the design requirements under both nominal and uncertain conditions. However, the superiority of the more sophisticated architectures, K_{IDI-PD} (in black) and K_{PI-LL} (in cyan), compared to the state-of-the-art baseline, K_{PD} (in blue), is evident. The former two controllers provide improved load relief (Fig. 11a), drastically reduced drift (Fig. 11b), and enhanced actuator performance (Fig. 11d). These improvements are primarily due to the integral action and are consistent with the results reported in [1] (for K_{IDI-PD}) and [14] (for K_{PI-LL}). Taken together, these findings strongly support revising the current state-of-practice in LV ascent attitude control.

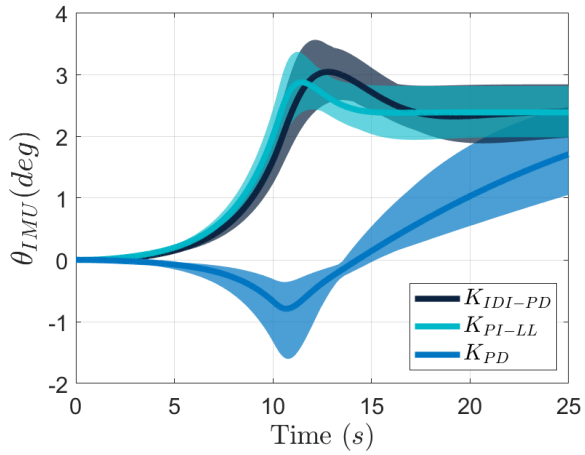
Comparing K_{IDI-PD} with K_{PI-LL}, the former shows slightly worse nominal and dispersed performance since the peak values of the evaluated LV quantities are slightly higher with K_{IDI-PD}. This



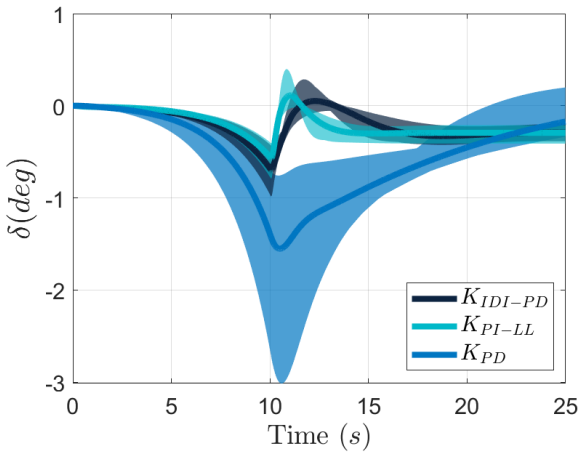
(a) Angle of Attack.



(b) Drift.



(c) Pitch Angle.



(d) TVC Deflection.

Fig. 11 Time-domain responses to a low-frequency, high-amplitude wind gust (v_w).

observation aligns with the findings Section 5.2: the outer-loop controller cannot fully compensate for the limitations imposed by the inversion strategy. These limitations arise from the minimum loop gain introduced by IDI, which sets a baseline that cannot be reduced by the outer loop. Consequently, the outer-loop controller can only "add" gain on top of this baseline. If the inversion already produces a moderately high crossover frequency, further increasing the outer-loop gain will inevitably raise the crossover frequency even more, potentially degrading high-frequency robustness.

One could argue that the performance gap could be narrowed—or even reversed—by increasing the outer-loop order of the IDI-based controller, potentially allowing K_{IDI-PD} to surpass K_{PI-LL}. While this is theoretically possible, such a comparison would be unfair because the controllers would differ in order and complexity. To preserve a fair basis for comparison, both controllers would need to be increased to the same order. However, even if both controllers are increased to the same higher order, the $\mathcal{H}_\infty$ LS controller is expected to maintain superior performance because the structural baseline gain inherent to IDI fundamentally limits the outer loop's ability to fully shape the closed-loop dynamics. While it is possible to modify the inversion forward path—for instance, by including a low-pass filter—to decrease the baseline gain, such modifications would complicate the system and the extension of the IDI control law across the entire flight envelope.

Nevertheless, the IDI controller achieves results very close to those of the $\mathcal{H}_\infty$ LS controller, with only a modest reduction in performance. Notably, based on these results, all requirements are still expected

to be fulfilled by a large margin. In essence, the inversion strategy provides a simplified approach to achieving the desired performance and robustness, offering a practical trade-off: a small reduction in performance in exchange for an easier and more structured control design.

7 Conclusion

The practical implications of Nonlinear and Incremental Nonlinear Dynamic Inversion (NDI and INDI), explored through their linear counterparts DI and IDI, are highlighted in this study. As explained in Section 4.2, the limitations imposed in the linear domain are expected to hold in the nonlinear one. With this in mind, the conclusions of this work are presented.

By directly inverting the system, DI and IDI significantly improve the system's singular value shape, achieving results that approach those of an $\mathcal{H}_\infty$ Loop Shaping (LS) controller. However, both introduce inherent performance/robustness limitations during the inversion process. While these limitations may appear minor or subtle, they can have meaningful practical effects. This is demonstrated by an outer-loop controller designed for IDI to evaluate how much these limitations could be mitigated. Results show that even with additional outer-loop compensation, the constraints imposed by the inversion cannot be fully overcomed; as a result, although the IDI controller achieves very good performance and robustness, it remains slightly below that of a (well designed) $\mathcal{H}_\infty$ LS controller.

The underlying reason lies in the design philosophy: the $\mathcal{H}_\infty$ LS controller is designed from a "white canvas," constrained only by the system and designer choices. This provides maximal freedom in control design but offers little inherent structure, making the control design and analysis process harder. In contrast, inversion-based techniques such as DI and IDI impose structure and a baseline loop gain from the outset. This baseline provides a reliable starting point, simplifying the design process, at the expense of limiting the achievable performance and robustness. Consequently, the main advantage of inversion-based strategies - at least for systems that are not highly nonlinear - is not superior performance or robustness, but the *simpler, more structured* framework they provide.

The analyses provided in this paper are focused on a single flight/operation point. However, the simpler and more structured framework offered by inversion-based techniques becomes particularly advantageous when designing controllers for the full flight envelope of intended operation. Follow-up work will therefore extend the results of this paper toward the design and analysis of a full-flight-envelope (I)NDI-based launcher vehicle control system.

Acknowledgments

This study was funded by the European Space Agency (ESA) through contract No. 4000147243 "Bridging the gap between robust and data-driven control prototype for launchers (BRAVE).

Declaration of Use of Artificial Intelligence

Artificial Intelligence was utilized to proofread the work.

References

[1] P. Simplício, P. Acquatella, and S. Bennani. Design and Analysis of a Launcher Flight Control System Based on Incremental Nonlinear Dynamic Inversion. *Aerospace*, 12(4), 2025. doi: 10.3390/aerospace12040296.

[2] A. Marcos, D. Navarro-Tapia, P. Simplício, and S. Bennani. Robust Control for Launchers: VEGA Study Case. *Journal of The Society of Instrument and Control Engineers*, 59(3):192–202, 2020. doi: 10.11499/sicejl.59.192.

[3] D. Navarro-Tapia. Robust and Adaptive TVC Control Design Approaches for the VEGA Launcher. PhD thesis, University of Bristol, 2019.

[4] T. Pollack. Advances in Dynamic Inversion-based Flight Control Law Design: Multivariable Analysis and Synthesis of Robust and Multi-Objective Design Solutions. PhD thesis, Technische Universiteit Delft, 2024. doi: 10.4233/uuid:28617ba0-461d-48ef-8437-de2aa41034ea.

[5] P. Rosa, J. Vasconcelos, N. Somma, A. Botelho, G. Tofanelli, J. Bravo, R. Hinz, J. Belhadj, M. Casasco, and S. Bennani. Deep Reinforcement Learning based Integrated Guidance and Control for a Launcher Landing Problem. In *ESA GNC and ICATT 2023*, 2023. doi: 10.5270/esa-gnc-icatt-2023-145.

[6] S. Xue, H. Bai, D. Zhao, and J. Zhou. Research on Intelligent Control Method of Launch Vehicle Landing Based on Deep Reinforcement Learning. *Mathematics*, 11(20), 2023. doi: 10.3390/math11204276.

[7] S. Xue, Z. Wang, H. Bai, C. Yu, and Z. Li. Research on Self-Learning Control Method of Reusable Launch Vehicle Based on Neural Network Architecture Search. *Aerospace*, 11(9), 2024. doi: 10.3390/aerospace11090774.

[8] I. Gregory. Dynamic Inversion to Control Large Flexible Transport Aircraft. In *Guidance, Navigation, and Control Conference and Exhibit*, page 4323, 1998. doi: 10.2514/6.1998-4323.

[9] P. Simplício, M. Pavel, E. Van Kampen, and Q. P. Chu. An Acceleration Measurements-based Approach for Helicopter Nonlinear Flight Control using Incremental Nonlinear Dynamic Inversion. *Control Engineering Practice*, 21(8):1065–1077, 2013. doi: 10.1016/j.conengprac.2013.03.009.

[10] P. Acquatella. Robust Nonlinear Attitude Control of Aerospace Vehicles: An Incremental Nonlinear Control Approach. PhD thesis, Delft University of Technology, 2020. doi: 10.4233/uuid:99d82992-080c-4c5d-8d40-4e62e62285c0.

[11] D. Canin, J. McConnell, and P. James. F-35 High Angle of Attack Flight Control Development and Flight Test Results. In *AIAA aviation 2019 forum*, page 3227, 2019. doi: 10.2514/6.2019-3227.

[12] J. T. Diz, P. Rosa, S. Theodoulis, P. Simplício, and P. Acquatella. (Incremental) Nonlinear Dynamic Inversion Control for Launch Vehicles: Internal Dynamics and Stability Analysis. *Journal of Guidance, Control, and Dynamics*, 2026. Under submission.

[13] P. Simplício. Guidance and Control Elements for Improved Access to Space from Planetary Landers to Reusable Launchers. PhD thesis, University of Bristol, 2019.

[14] J. T. Diz. Thrust Vector Control for Flexible Launchers: an $\mathcal{H}_\infty$ Open Loop Shaping Approach. Master's thesis, Technische Universiteit Delft, 2025. `https://resolver.tudelft.nl/uuid:2d7cb930-bc4c-4470-8582-2846c7e313fe`.

[15] P. Simplício, S. Bennani, A. Marcos, C. Roux, and X. Lefort. Structured Singular-value Analysis of the VEGA Launcher in Atmospheric Flight. *Journal of Guidance, Control, and Dynamics*, 39(6):1342–1355, 2016. doi: 10.2514/1.G000335.

[16] J. T. Diz, P Simplício, and S. Theodoulis. $\mathcal{H}_\infty$ Loop Shaping for Robust Attitude Control of Launch Vehicles. In *Proceedings ROCOND'2025/LPVS'2025*, 2025.

[17] H. Khalil and J. Grizzle. Nonlinear Systems, volume 3. Prentice Hall Upper Saddle River, NJ, 2002. ISBN: 978-0130673893.

[18] J.J.E. Slotine and W. Li. Applied Nonlinear Control. Prentice Hall, Englewood Cliffs, NJ, 1991.

[19] S. Sieberling, Q. P. Chu, and J. Mulder. Robust Flight Control using Incremental Nonlinear Dynamic Inversion and Angular Acceleration Prediction. *Journal of Guidance, Control, and Dynamics*, 33(6):1732–1742, 2010. doi: 10.2514/1.49978.

[20] S. Skogestad and I. Postlethwaite. Multivariable Feedback Control: Analysis and Design. John Wiley & Sons, 2005. ISBN: 978-0470011683.

[21] C. Vlaar and G. Looye. Incremental Nonlinear Dynamic Inversion Flight Control. Master's thesis, Delft University of Technology, 2014.

[22] P. Acquatella, W. van Ekeren, and Q. P. Chu. PI(D) Tuning for Flight Control Systems via Incremental Nonlinear Dynamic Inversion. *IFAC-PapersOnLine*, 50(1):8175–8180, 2017. 20th IFAC World Congress. doi: 10.1016/j.ifacol.2017.08.1265.

[23] B. Bacon, A. Ostroff, and S. Joshi. Nonlinear Dynamic Inversion Reconfigurable Controller utilizing a Fault Tolerant Accelerometer. In *19th DASC. 19th Digital Avionics Systems Conference. Proceedings (Cat. No. 00CH37126)*, 2000. doi: 10.1109/DASC.2000.884920.

[24] B. Bacon, A. Ostroff, and S. Joshi. Reconfigurable NDI Controller using Inertial Sensor Failure Detection & Isolation. *IEEE Transactions on Aerospace and Electronic Systems*, 37(4):1373–1383, 2001. doi: 10.2514/6.2000-4565.

[25] X. Wang, E. van Kampen, Q. Chu, and P. Lu. Stability Analysis for Incremental Nonlinear Dynamic Inversion Control. *Journal of Guidance, Control, and Dynamics*, 42(5):1116–1129, 2019. doi: 10.2514/1.G003791.

[26] R. Steffensen, A. Steinert, and F. Holzapfel. Incremental Control as an Enhanced and Robust Implementation of Gain Scheduled Controllers avoiding Hidden Coupling Terms. *Aerospace Science and Technology*, 141:108500, 2023. ISSN: 1270-9638. doi: 10.1016/j.ast.2023.108500.

[27] D. McFarlane and K. Glover. A Loop Shaping Design Procedure using $\mathcal{H}_\infty$ Synthesis. *IEEE Transactions on Automatic Control*, 37(6):759–769, 1992. doi: 10.1109/9.256330.