LAUNCHER ATTITUDE CONTROL BASED ON INCREMENTAL NONLINEAR DYNAMIC INVERSION: A FEASIBILITY STUDY TOWARDS FAST AND ROBUST DESIGN APPROACHES

Pedro Simplício$^{*(1)}$**, Paul Acquatella**$^{(2)}$**, Samir Bennani**$^{(3)}$

$^{(1)}$ *Aurora Technology for the European Space Agency, Noordwijk, The Netherlands*
$^{(2)}$ *DLR, German Aerospace Center, Oberpfaffenhofen, Germany*
$^{(3)}$ *European Space Agency, Noordwijk, The Netherlands*

ABSTRACT

The so-called ”New Space era” has seen a disruptive change in the business models and manufacturing technologies of launch vehicle companies. However, limited consideration has been given to the benefits that innovation in control theory can bring; not only in terms of increasing the limits of performance but also reducing mission preparation or “missionisation” efforts. Moreover, there is a gap between the current state-of-practice that still relies on linear controls and other modern control techniques that could bring relevant improvements in launcher attitude control; this is the case for nonlinear control algorithms, especially those based on Nonlinear Dynamic Inversion (NDI). NDI is a technique that basically ‘cancels’ the nonlinearities of a class of nonlinear systems, allowing for a single linear control law to be applied without the need for gain-scheduling across different operational points. Incremental NDI (INDI) is a variation of NDI that generates incremental commands and employs acceleration feedback to reduce model dependency, making it easier to design, and results in being more robust in closed-loop. While INDI has been applied successfully to several aerospace applications, its applicability to launch vehicles has not yet been adequately investigated. The objective of this paper is therefore to introduce and raise awareness of the INDI method among the launcher guidance, navigation, and control (GNC) community, showcasing its implementation on a representative launch ascent application scenario which highlights INDI’s strengths and challenges. We present a new, practical approach for stability analysis of INDI for attitude control, and compare INDI with scheduled PD controllers with- and without angular acceleration estimates. Results show that, while INDI controllers are generally more sensitive to sensor noise and actuator delay than linear controllers, their potential benefits outweigh these limitations in terms of robustness and performance.

1 INTRODUCTION

1.1 Background and Motivation

The space industry has undergone significant changes in recent years with the advent of the “New Space era” marked by disruptive changes in the business models, manufacturing technologies, and agile practices of launch vehicle companies; all aimed at minimising their production and operating costs in an ever more competitive market. However, limited attention has been given to the benefits of control theory innovation in this context despite the potential for such innovations to increase performance limits and reduce mission preparation (or “missionisation”) efforts. Moreover, government-led developments of recent launchers such as Ares I and VEGA still use the same design approach of the Saturn V, i.e. linear controllers [1]. This approach relies on single channel-at-a-time tuning and ad–hoc gain-scheduling followed by extensive validation and verification (V&V); these are in fact quite time- and cost-consuming processes.

[*] Corresponding author, email: `pedro.simplicio@ext.esa.int`

In contrast to the approach presented above, the past few years have seen a growing interest in the application of artificial intelligence and machine learning methods for launcher GNC, but the industrial use of such data-driven/model-free methods remains limited by well-known issues related to training and certification of the algorithms on the full flight envelope of intended operation. In that sense, there is a clear gap between these strategies and the current state-of-practice, in which other techniques could bring relevant improvements; this is the case for nonlinear control algorithms, especially those based on Nonlinear Dynamic Inversion (NDI). On one hand, agile practices of New Space companies provide the ideal opportunity to explore the benefits of this type of design approach. On the other hand, a successful adoption of nonlinear launcher control will likely facilitate the augmentation with and transition to data-driven methods in the future. This is therefore our motivation and aim for this paper, to start bridging the gap between these two approaches while presenting a potential alternative based on incremental nonlinear control.

1.2 Related Work

In this paper we introduce briefly and focus on (Incremental) Nonlinear Dynamic Inversion (NDI) which is a control design method based on *feedback linearisation* [2, 3]; it basically consists on a nonlinear (state feedback) transformation that linearises the nominal system dynamics, and a linear part that imposes the desired closed-loop dynamics. Actually, NDI is a very well known and applied (nonlinear) control technique in the aerospace field, especially in aeronautics for various flight control applications [4–7]. Successful implementation of NDI requires a match between the onboard model and the system model, and accurate knowledge of all nonlinearities, which is often not the case in reality; this results in poor robustness properties because they rely on exact availability of the system dynamics. This highlights the need for robustness in these methodologies, as the inner-loop of the control system is critical and can be compromised by model and sensor uncertainties, potentially affecting stability and performance. In this regard, alternative methods involving robustness and improvements of the method for NDI-based flight control applications were considered, among many others, in [8–12].

A successful technique that became popular in the recent years for aerospace applications is Incremental Nonlinear Dynamics Inversion (INDI). The concept using incremental nonlinear control was first developed in the late nineties and was initially focused on the '*implicit*' dynamic inversion for DI-based flight control. The works of Smith, Bacon, and others laid the foundation for these developments [5, 10], for which the term '*incremental*' is now more commonly used to describe this methodology as it better reflects the nature of these control laws [13–15]. Those early studies further developed the incremental approach and, since then, it has been further elaborated theoretically and successfully applied in various high-performance systems including fault-tolerant control of aircraft subjected to sensor and actuator faults [16, 17], in practice for quadrotors using adaptive control [18, 19], in real flight tests of small (unmanned) and business jet (Cessna Citation II, PH-LAB) aircraft [20–22], but also for spacecraft attitude control [23–25]. However, its applicability to launch and re-entry vehicles has not been fully investigated but only considered in [26–28], and planned to be flight-tested in the upcoming '*Reusability Flight Experiment (ReFEx)*' by DLR [29]. These related works have demonstrated INDI's performance and robustness against aerodynamic model uncertainties and disturbance rejection for several aerospace vehicles; hence, the potential benefits of INDI are quite relevant for reusable launchers which have much tighter dynamical couplings between online-generated trajectory and attitude control during descent flight. Moreover, due to the nonlinear nature of INDI, it has been proven difficult to attain an analytical proof of stability which has been derived in [30]. With this paper we aim for further close this gap towards the application of INDI for launchers with special focus on the ascent of a TVC-controlled launcher and also aim to present a new, practical approach for stability analysis of such INDI control laws applied for attitude control.

1.3 Objectives and Outline

It is therefore the objective of this study to introduce and raise awareness of the INDI technique among the launcher GNC community, to showcase its implementation on a representative application scenario, and to highlight its strengths and challenges in the face of the industrial state-of-practice. To achieve this, the paper provides a concise description of the NDI and INDI approach, followed by the detailed design and comparison of different control laws: linear, linear with angular acceleration feedback and INDI-based. Furthermore, the paper is also aimed to address the (mainly) two well-known challenges associated with the practical implementation of INDI-based control:

- **Sensitivity to sensor noise and actuator delay**. By relying on angular acceleration and control input measurements/estimates, INDI controllers are generally more sensitive to sensor noise and actuator delay than classical controllers. To assess the severity of this challenge, the paper shows a comprehensive nonlinear simulation campaign with wind disturbances, uncertainties, as well as different levels of sensor noise and actuator delay. These simulations serve as a basis to analyse the sensitivity to sensor noise and actuator delay in comparison to more classical approaches and we showcase how to remediate or tackle these issues properly.

- **Nonlinear stability analysis**. The second challenge of INDI is that, due to its nonlinear nature, attaining an analytical proof of stability is not trivial [30]. For this second challenge, the paper proposes a simple yet insightful linearisation-based approach to evaluate stability degradation related to an inexact feedback linearisation and to deviations from the control tuning conditions. This method provides a new way to analyse and evaluate stability analysis of the nonlinear controller using linear control techniques; since INDI is designed from the theory of feedback linearisation, this approach is very intuitive in the sense it provides a measure of degradation with respect to the feedback linearised plant and linear stability analysis can be performed.

To demonstrate the benefits and challenges of the INDI approach, we showcase the method within an application scenario consisting of a launcher model during ascent flight while featuring attitude and lateral drift degrees-of-freedom, actuator dynamics, and moving-mass effects. All the controllers and filters are implemented at a sampling frequency that is compatible with current onboard capabilities (25 Hz).

The outline of this paper is as follows. A brief introduction to Nonlinear Dynamic Inversion (NDI) and Incremental NDI is presented in Sec. 2. Section 3 presents the modelling aspects of the launcher application in consideration and describes the simulator used for the attitude control design and testing. Launcher attitude control designs including angular acceleration feedback are presented in Sec. 4. Time-domain robust performance results and analysis of the obtained simulations comparing the controllers studied are presented in Sec. 5, while Sec. 6 presents the frequency-domain stability results and analysis. Conclusions are finally presented in Sec. 7.

2 BASIC PRINCIPLES OF (INCREMENTAL) NONLINEAR DYNAMIC INVERSION

2.1 Nonlinear Dynamic Inversion (NDI)

Without loss of generality, we consider a multiple-input and multiple-output (MIMO) system whose number of inputs are equal to the number of outputs in order to avoid control allocation and internal dynamics problems. Let's also assume momentarily that the nonlinear system can be described affine in the inputs as:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})\boldsymbol{u} \tag{1a}$$

$$\boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x}) \tag{1b}$$

where $\boldsymbol{x} \in \mathcal{R}^n$ is the state vector, $\boldsymbol{u} \in \mathcal{R}^m$ is the control input vector, and $\boldsymbol{y} \in \mathcal{R}^m$ is the system output vector, the functions $\boldsymbol{f}(\boldsymbol{x})$ and $\boldsymbol{h}(\boldsymbol{x})$ are assumed to be smooth vector fields on $\mathcal{R}^n$, and $\boldsymbol{g}(\boldsymbol{x}) \in \mathcal{R}^{n\times m}$

is a matrix whose columns are also assumed as smooth vector fields $\boldsymbol{g}_j$. For these systems, the vector of relative degree represents the number of differentiations of each output y_i, $i = 1, \ldots, m$, that are needed for the input to appear [2, 3]. In this brief introduction to NDI we consider $\boldsymbol{y} = \boldsymbol{x}$ so that the relative degree of each of the outputs y_i is one; for a detailed explanation of NDI for higher relative degrees including the transformation of the nonlinear system into a normal form decomposed into an external (input–output) part and an internal (unobservable) part, the reader is referred to [25, 30].
Nonlinear Dynamic Inversion (NDI) is a technique that aims to eliminate the nonlinearities present in a given nonlinear system, resulting in closed-loop dynamics that can be expressed in a linear form. To achieve this, the nonlinear system is inverted into a linear structure using state feedback, making it possible to apply conventional linear controllers. However, NDI has a significant disadvantage in that it relies on the fundamental assumption that the system model is known exactly, making it vulnerable to uncertainties. Additionally, NDI assumes that the system state is fully and accurately known, which can be challenging to achieve in practice. NDI involves applying the following input transformation [2, 3]:

$$\boldsymbol{u}_{\text{cmd}} = \boldsymbol{g}^{-1}(\boldsymbol{x})\left(\boldsymbol{\nu} - \boldsymbol{f}(\boldsymbol{x})\right) \tag{2}$$

which cancels all nonlinearities in closed-loop, and a simple linear input-output relationship between the new virtual control input $\boldsymbol{\nu}$ and the output $\boldsymbol{y}$ is obtained:

$$\dot{\boldsymbol{y}} = \boldsymbol{\nu} \tag{3}$$

In addition to being linear, an interesting feature of this relationship is that it is also decoupled, meaning that the input ν_i only affects the output y_i. This property gives rise to the so-called “decoupling control law” to describe the input transformation in (2), and the resulting linear system in (3) is referred to as a “single-integrator” form. By utilising appropriate (linear, robust) control techniques, the single-integrator form in (3) can be rendered exponentially stable. For instance, the single-integrator can be made exponentially stable through the use of:

$$\boldsymbol{\nu} = \dot{\boldsymbol{y}}_{\text{des}} = \dot{\boldsymbol{y}}_{\text{cmd}} + \boldsymbol{K}_P\,\boldsymbol{e} \tag{4}$$

where $\boldsymbol{\nu} = \dot{\boldsymbol{y}}_{\text{des}}$ defines the desired dynamics for the output vector or control variables. The feedforward term for tracking is given by $\dot{\boldsymbol{y}}_{\text{cmd}}$, while $\boldsymbol{e} = \boldsymbol{y}_{\text{cmd}} - \boldsymbol{y}$ represents the error vector. Here, $\boldsymbol{y}_{\text{cmd}}$ denotes the (smooth) desired output vector, which is (in this case, since relative degree is one) at least once differentiable. The gain matrix $\boldsymbol{K}_P \in \mathcal{R}^{m\times m}$ is used to ensure that the polynomials given by $s + K_{P_i}$ for $i = 1, \ldots, m$, become Hurwitz. The diagonal elements K_{P_i} of $\boldsymbol{K}_P$ are then selected accordingly. As a result of using (4), the desired error dynamics $\dot{e}_i + K_{P_i}\,e_i = 0$, become exponentially stable and decoupled, leading to $e_i(t) \to 0$ for $i = 1, \ldots, m$.

2.2 Incremental Nonlinear Dynamic Inversion (INDI)

Incremental nonlinear dynamic inversion (INDI) consists on the application of NDI to a system expressed in an incremental form [14, 15, 30]. To obtain a system in incremental form, first we introduce a *sufficiently small* time–delay λ and define the following deviation variables $\dot{\boldsymbol{x}}_0 := \dot{\boldsymbol{x}}(t-\lambda)$, $\boldsymbol{x}_0 := \boldsymbol{x}(t-\lambda)$, and $\boldsymbol{u}_0 := \boldsymbol{u}(t-\lambda)$, which are the λ–time–delayed signals of the current state derivative $\dot{\boldsymbol{x}}(t)$, state $\boldsymbol{x}(t)$, and control $\boldsymbol{u}(t)$, respectively [24]. Moreover, we will denote $\Delta\dot{\boldsymbol{x}} := \dot{\boldsymbol{x}} - \dot{\boldsymbol{x}}_0$, $\Delta\boldsymbol{x} := \boldsymbol{x} - \boldsymbol{x}_0$, and $\Delta\boldsymbol{u} := \boldsymbol{u} - \boldsymbol{u}_0$ as the incremental state derivative, the incremental state, and the so–called incremental control input, respectively. Subsequently, we consider a first-order Taylor series expansion of $\dot{\boldsymbol{x}}$, not in the geometric sense, but with respect to the newly introduced time–delay λ as [14, 15, 24, 30]:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \dot{\boldsymbol{x}}_0 + \frac{\partial}{\partial \boldsymbol{x}}\left[\boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})\boldsymbol{u}\right]\Bigg|_{\substack{\boldsymbol{x}=\boldsymbol{x}_0\\ \boldsymbol{u}=\boldsymbol{u}_0}} \Delta\boldsymbol{x} + \boldsymbol{g}(\boldsymbol{x}_0)\Delta\boldsymbol{u} + H.O.T \\ &= \dot{\boldsymbol{x}}_0 + \boldsymbol{g}(\boldsymbol{x}_0)\Delta\boldsymbol{u} + \boldsymbol{N}(\boldsymbol{x},\lambda) \end{aligned} \tag{5}$$

with:

$$\dot{\boldsymbol{x}}_0 = \boldsymbol{f}(\boldsymbol{x}_0) + \boldsymbol{g}(\boldsymbol{x}_0)\boldsymbol{u}_0 \tag{6a}$$

$$\boldsymbol{N}(\boldsymbol{x}, \lambda) = \frac{\partial}{\partial \boldsymbol{x}}\Big[\boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})\boldsymbol{u}\Big]\Big|_{\substack{\boldsymbol{x}=\boldsymbol{x}_0\\ \boldsymbol{u}=\boldsymbol{u}_0}} \Delta\boldsymbol{x} + \mathit{H.O.T} \tag{6b}$$

which represents a residual containing the Jacobian linearisation of the on-board model and the higher order terms (*H.O.T*) of the series expansion. Notice that the model–based control effectiveness $\boldsymbol{g}(\boldsymbol{x}_0)$ is sampled at the previous incremental time. This means an approximate linearisation about the $\lambda-$delayed signals is performed *incrementally*, and not with respect to a particular equilibrium or operational point of interest. Further, we consider the following *time-scale separation assumption*:

For a sufficiently small time-delay λ and for any incremental control input, it is assumed that $\Delta\boldsymbol{x}$ does not vary significantly during λ. In other words, the input rate of change is much faster than the state rate of change:

$$\epsilon_{INDI_{\mathrm{TSS}}}(t) \equiv \Delta\boldsymbol{x} := \boldsymbol{x} - \boldsymbol{x}_0 \cong 0,\ \forall\, \Delta\boldsymbol{u} \tag{7}$$

which leads to:

$$\dot{\boldsymbol{x}} \cong\ \dot{\boldsymbol{x}}_0 + \boldsymbol{g}(\boldsymbol{x}_0)\cdot(\boldsymbol{u} - \boldsymbol{u}_0) + \underbrace{\boldsymbol{N}(\boldsymbol{x}, \lambda)}_{\cong 0}$$

or simply:

$$\Delta\dot{\boldsymbol{x}} \cong \boldsymbol{g}(\boldsymbol{x}_0)\cdot\Delta\boldsymbol{u} \tag{8}$$

This assumption, corroborated by the fact that the perturbation term $\boldsymbol{N}(\boldsymbol{x}, \lambda)$ satisfies [30]:

$$\lim_{\lambda\to 0} \|\boldsymbol{N}(\boldsymbol{x}, \lambda)\|_2 \to 0,\ \forall\,\boldsymbol{x} \tag{9}$$

implies that the nonlinear system dynamics in its incremental form is approximated at each time-step by the model-based control effectiveness $\boldsymbol{g}(\boldsymbol{x}_0)$. Finally, applying NDI to the system based on the approximation (8) results in a relation between the incremental control input and the output of the system:

$$\boldsymbol{u} = \boldsymbol{u}_0 + \boldsymbol{g}(\boldsymbol{x}_0)^{-1}(\boldsymbol{\nu} - \dot{\boldsymbol{x}}_0). \tag{10}$$

and noticing that while implementing this control law it will be required the availability of $\dot{\boldsymbol{x}}_0$ and that the incremental input $\boldsymbol{u}_0$ is obtained from the output of the actuators or estimated from an actuator dynamical model; recall it has been assumed that a commanded control is achieved *sufficiently fast* in regards to the actuator dynamics. The total control command along with the obtained linearising control $\boldsymbol{u}_0 = \boldsymbol{u}(t-\lambda)$ can be rewritten as:

$$\boldsymbol{u}(t) = \boldsymbol{u}(t-\lambda) + \boldsymbol{g}(\boldsymbol{x}_0)^{-1}[\boldsymbol{\nu} - \dot{\boldsymbol{x}}(t-\lambda)]. \tag{11}$$

This improves the robustness of the closed-loop system as compared with conventional NDI since dependency on the accurate knowledge of the plant dynamics is reduced; more specifically, the dependency on accurate knowledge of the dynamic model in $\boldsymbol{f}(\boldsymbol{x})$ is largely decreased. Therefore, the INDI control law design is more dependent on accurate measurements or accurate estimates of $\dot{\boldsymbol{x}}_0$, the state derivatives, and $\boldsymbol{u}_0$, the incremental control input, respectively.

3 LAUNCHER MODEL AND SIMULATOR DESCRIPTION

The present study relies on a conventional 3 degrees-of-freedom launcher model in ascent flight featuring lateral drift z and pitch θ dynamics, as schematised in Fig. 1. These dynamics, representing the first/second time-derivatives of z as $\{w = \dot{z},\, \dot{w} = \ddot{z}\}$ and the first/second time-derivatives of θ as $\{q = \dot{\theta},\, \dot{q} = \ddot{\theta}\}$, are governed by the well-known nonlinear Newton-Euler equations:

$$m\dot{w} = F_\alpha + F_\mathrm{c} + F_\mathrm{n} - mg\sin\theta \tag{12}$$

$$J\dot{q} = M_\alpha + M_\mathrm{c} + M_\mathrm{n} \tag{13}$$

where m, J and g are the launcher's mass, lateral moment of inertia and gravity acceleration, $\{F_\alpha$, $M_\alpha\}$ are the aerodynamic force/torque, $\{F_\mathrm{c}$, $M_\mathrm{c}\}$ are the TVC-induced force/torque and $\{F_\mathrm{n}$, $M_\mathrm{n}\}$ are the nozzle moving-mass effects, also known as tail-wags-dog (TWD).

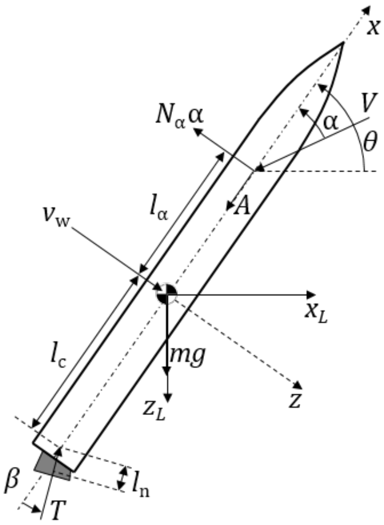


Figure 1: Launcher model diagram

The aerodynamic force and torque are computed as:

$$F_\alpha = -SC_{N_\alpha}Q\alpha \tag{14}$$

$$M_\alpha = -l_\alpha F_\alpha \tag{15}$$

where S, C_{N_α} and l_α are the reference aerodynamic area, lateral force gradient and aerodynamic arm (distance between the launcher's centres of pressure and gravity). $Q\alpha$ is the aerodynamic load indicator, defined as the product between aerodynamic pressure and angle of attack, which are respectively given by:

$$Q = \frac{1}{2}\rho V^2 \tag{16}$$

$$\alpha = \theta + \arctan\frac{w - l_\alpha q - v_\mathrm{w}}{V} \tag{17}$$

where ρ is the air density, V is the total airspeed and v_w is the lateral wind turbulence speed. The term $l_\alpha q$ is often known as aerodynamic damping.

The TVC-induced force and torque are computed as:

$$F_\mathrm{c} = -T\sin\beta \tag{18}$$

$$M_\mathrm{c} = l_\mathrm{c}F_\mathrm{c} \tag{19}$$

where T is the thrust magnitude, l_c is the TVC arm (distance between the launcher's centre of gravity and nozzle's pivot point) and β is the TVC deflection angle.
Finally, the nozzle TWD effects are computed as:

$$F_\mathrm{n} = -m_\mathrm{n} l_\mathrm{n} \ddot{\beta} \tag{20}$$

$$M_\mathrm{n} = l_\mathrm{c} F_\mathrm{n} - J_\mathrm{n} \ddot{\beta} \tag{21}$$

where m_n is the nozzle moving-mass, l_n is the moving-mass arm (distance between the nozzle's centre of gravity and pivot point), $\ddot{\beta}$ is the TVC deflection acceleration and J_n is the nozzle moment of inertia with respect to the pivot point (not to the centre of gravity).
Most of the model's parameters vary along the launcher's trajectory (this dependence was not evidenced in the previous equations for the sake of readability) and are highly uncertain. These parameters were extracted as a function of time from the simulator presented in [31] for a 80 seconds trajectory. The uncertainty levels assumed in this study are summarised in Table 1.

Table 1: Uncertainty level per type of parameter

Type of parameters	Variables	Uncertainty level
Aerodynamics	C_{N_α}, l_α, ρ, V	20%
Mass/propulsion	m, J, l_c, T	10%

Note that while mass/propulsion parameters have an explicit dependency on time, related to the way the propellant burns, aerodynamics parameters have an implicit dependency through intermediate quantities such as altitude and Mach number.
In addition to the launcher model described above, the present study considers the dynamical effects of TVC actuation and wind turbulence. Both effects are modelled as time-invariant transfer functions for the sake of simplicity without loss of generality. The TVC dynamics corresponds to a second-order system given by:

$$G_\mathrm{TVC}(\mathrm{s}) = \frac{67.8^2}{\mathrm{s}^2 + 90.9\,\mathrm{s} + 67.8^2} \tag{22}$$

where s represents the Laplace variable. The wind turbulence speed v_w is modelled by colouring a white noise signal through a first-order Dryden filter [32] given by:

$$G_\mathrm{w}(\mathrm{s}) = \frac{3.54}{\mathrm{s} + 0.32} \tag{23}$$

The launcher, TVC and wind models were put together in a simulator that allows to quickly analyse and compare several control systems, which is illustrated in Fig. 2. In this figure, different simulation rates are highlighted using different colours: black for the continuous-time dynamics, red for GNC computations ($f_\mathrm{GNC} = 25$ Hz, which is well representative of current onboard capabilities) and green for wind noise generation ($f_\mathrm{w} = 20$ Hz in this case).
For control design purposes, it is also convenient to define a linear model that fully captures the driving dynamics of Eq. (12) and (13). To do so [33], consider the following coefficients relative to the rotational motion:

$$\mu_\alpha = \frac{l_\alpha Q S C_{N_\alpha}}{J}, \qquad \mu_\mathrm{c} = \frac{l_\mathrm{c} T}{J}, \qquad \mu_\mathrm{n} = \frac{m_\mathrm{n} l_\mathrm{n} l_\mathrm{c} + J_\mathrm{n}}{J} \tag{24}$$

and to the translational motion:

$$n_\alpha = \frac{Q S C_{N_\alpha}}{m}, \qquad n_\mathrm{c} = \frac{T}{m}, \qquad n_\mathrm{n} = \frac{m_\mathrm{n} l_\mathrm{n}}{m} \tag{25}$$

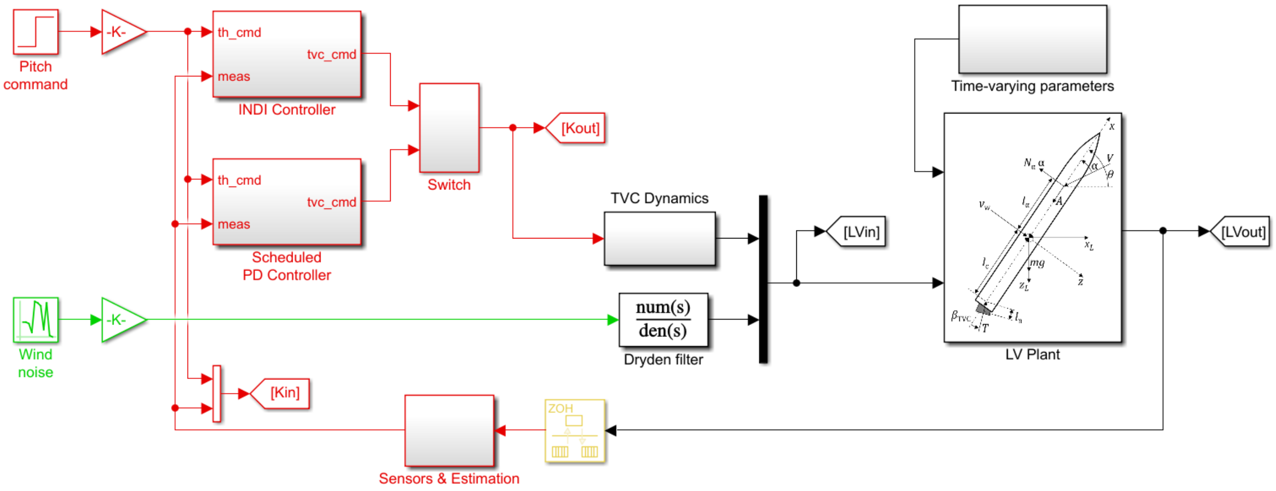


Figure 2: Launcher simulator diagram

Using these coefficients, the transfer functions $\beta(\mathrm{s}) \rightarrow \theta(\mathrm{s})$ and $\beta(\mathrm{s}) \rightarrow w(\mathrm{s})$ correspond to the solutions of the system:

$$\begin{bmatrix} \mathrm{s}^2 + l_\alpha \dfrac{\mu_\alpha}{V}\mathrm{s} - \mu_\alpha & -\dfrac{\mu_\alpha}{V} \\ -l_\alpha \dfrac{n_\alpha}{V}\mathrm{s} + n_\alpha + g \sin\theta_0 & \mathrm{s} + \dfrac{n_\alpha}{V} \end{bmatrix} \begin{bmatrix} \dfrac{\theta(\mathrm{s})}{\beta(\mathrm{s})} \\ \dfrac{w(\mathrm{s})}{\beta(\mathrm{s})} \end{bmatrix} = - \begin{bmatrix} \mu_\mathrm{n}\mathrm{s}^2 + \mu_\mathrm{c} \\ n_\mathrm{n}\mathrm{s}^2 + n_\mathrm{c} \end{bmatrix} \tag{26}$$

Furthermore, as a first approximation for attitude control design purposes, drift and TWD dynamics can be neglected and the transfer function $\beta(\mathrm{s}) \rightarrow \theta(\mathrm{s})$ simplifies into:

$$\frac{\theta(\mathrm{s})}{\beta(\mathrm{s})} \approx -\frac{\mu_\mathrm{c}}{\mathrm{s}^2 + l_\alpha \dfrac{\mu_\alpha}{V}\mathrm{s} - \mu_\alpha} \tag{27}$$

4 LAUNCHER CONTROL DESIGN USING ANGULAR ACCELERATION FEEDBACK

This section describes and justifies the four attitude control systems developed in this study.

4.1 Scheduled PD controller

The baseline controller for this study is a classic proportional-derivative (PD) controller with the following structure:

$$\beta(\mathrm{s}) = k_P\Big(\theta_\mathrm{cmd}(\mathrm{s}) - \theta(\mathrm{s})\Big) - k_D\, q(\mathrm{s}) = k_P\, \theta_\mathrm{cmd}(\mathrm{s}) - \Big(k_P + \mathrm{s}\, k_D\Big)\theta(\mathrm{s}) \tag{28}$$

Despite their simplicity, PD controllers represent the industrial state-of-practice for the vast majority of launch vehicles [1]. The gains k_P and k_D can be tuned using a multitude of methods. Here, they are selected based on pole placement of the closed-loop transfer function, which is obtained by substituting Eq. (28) in (27):

$$\frac{\theta(\mathrm{s})}{\theta_\mathrm{cmd}(\mathrm{s})} = -\frac{\mu_\mathrm{c} k_P}{\mathrm{s}^2 + \Big(l_\alpha \dfrac{\mu_\alpha}{V} - \mu_\mathrm{c} k_D\Big)\mathrm{s} - \Big(\mu_\alpha + \mu_\mathrm{c} k_P\Big)} \tag{29}$$

It is clear from this equation that k_P and k_D can be chosen so as to enforce the desired natural frequency ω_θ and damping ratio ζ (here assumed constant throughout the flight for simplicity without

loss of generality). It is also clear that this approach does not allow to specify the steady-state gain (when $\mathrm{s} \to 0$) independently of the natural frequency as they both depend on k_P only.
In order to handle the wide variation of the model's parameters during the flight, the two gains need to be scheduled throughout the trajectory. To do so, they are pre-computed for a grid of $N = 9$ points (spaced every 10 seconds along the trajectory) as:

$$k_P[i] = -\frac{1}{\mu_{\mathrm{c}}[i]}\Big(\mu_\alpha[i] + \omega_\theta^2\Big), \qquad k_D[i] = \frac{1}{\mu_{\mathrm{c}}[i]}\Big(l_\alpha[i]\frac{\mu_\alpha[i]}{V[i]} - 2\zeta\omega_\theta\Big), \qquad i = 1, ..., N \tag{30}$$

and then linearly interpolated online during the simulation. The robustness of this approach can be increased by scheduling the controller with respect to online measurements/estimates of some of the model's parameters. This is the underlying idea of LPV control [34], which is outside the scope of this paper.

4.2 INDI controller

In this section, an INDI-based control law is developed and applied to regulate the launcher's attitude channel, i.e. to:

$$y = h(\boldsymbol{x}) = q \tag{31}$$

where $\boldsymbol{x}$ represents the state vector. In order to apply the INDI technique, this equation has to be time-differentiated until an explicit dependency on the TVC input appears. The first-order derivative corresponds to Eq. (13), which can be recast as:

$$\dot{y} = \dot{q} = f(\boldsymbol{x}) + g(\boldsymbol{x})u \tag{32}$$

where $f(\boldsymbol{x})$ is the control-independent part of the model, $g(\boldsymbol{x})$ expresses the influence of the controls in the system and u is the control input. For the launcher scenario, the latter two terms correspond to:

$$g(\boldsymbol{x}) \approx -\mu_{\mathrm{c}}, \qquad u = \beta \tag{33}$$

A virtual control input can now be defined in order to transform the nonlinear system into a linear form as follows:

$$\nu = \dot{q} = \ddot{\theta} \quad \Rightarrow \quad \frac{\theta(\mathrm{s})}{\nu(\mathrm{s})} = \frac{1}{\mathrm{s}^2} \tag{34}$$

Following the procedure of Sec. 2.2, the command signal sent to the TVC actuator is given by:

$$\beta = \beta_0 - \frac{1}{\mu_{\mathrm{c}}}\Big(\nu - \dot{q}_0\Big) \tag{35}$$

where β_0 and $\dot{q}_0$ are measurements/estimates of the TVC command and angular acceleration at the current computation step, respectively. The estimate of the TVC command β_0 is obtained with a low pass filter and because angular acceleration sensors are not common in launchers today, $\dot{q}_0$ is estimated by passing the angular rate q through a derivative filter of the form:

$$H_{\dot{q}}(\mathrm{s}) = \frac{\mathrm{s}\,\omega_{\dot{q}}}{\mathrm{s} + \omega_{\dot{q}}} \tag{36}$$

where $\omega_{\dot{q}}$ represents the filter bandwidth. Note that, after the feedback linearisation of Eq. (35), there are still some degrees of internal dynamics in the system related to the drift motion and TWD effect, but these dynamics are known to be stable and can be further handled by outer control loops.
Using the virtual control and the linearised system of Eq. (34), an outer PD control law is able to enforce the desired closed-loop response as follows:

$$\nu(\mathrm{s}) = k_P\Big(\theta_{\mathrm{cmd}}(\mathrm{s}) - \theta(\mathrm{s})\Big) - k_D\, q(\mathrm{s}) \quad \Rightarrow \quad \frac{\theta(\mathrm{s})}{\theta_{\mathrm{cmd}}(\mathrm{s})} = \frac{k_P}{\mathrm{s}^2 + k_D\,\mathrm{s} + k_P} \tag{37}$$

$$k_P = \omega_\theta^2, \qquad k_D = 2\zeta\omega_\theta \tag{38}$$

Note that, in contrast with the PD controller of Sec. 4.1, k_P and k_D do not need to be scheduled as they depend on ω_θ and ζ only, but a pre-computed grid of $\mu_c[i]$ is still required to perform the feedback linearisation. This is highlighted in the blue area of Fig. 3, which illustrates the implementation of the INDI controller in the simulator. Alternatively, μ_c could be estimated based on online measurements.

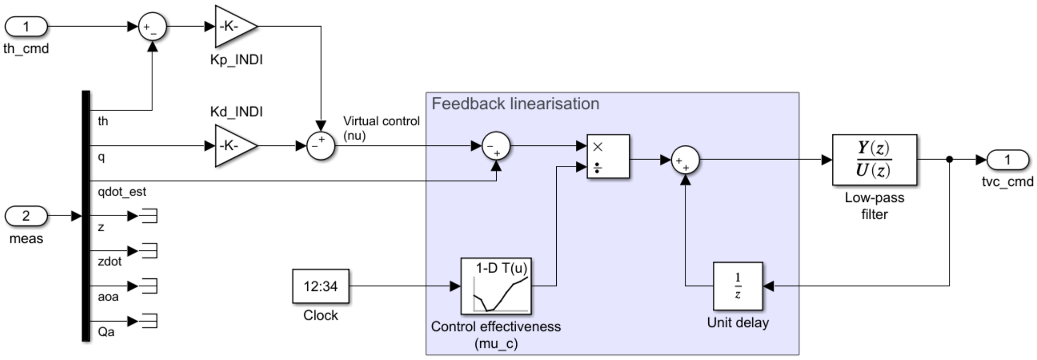


Figure 3: INDI controller implementation diagram

4.3 Scheduled PD controller with $\dot{q}$ feedback

As explained in Sec. 2.2, the INDI controller of Eq. (35) relies on $\dot{q}$ information to reduce the impact of the launcher's model on the achievable control performance. For a fair comparison of controllers, it is then pertinent to consider a linear controller where $\dot{q}$ feedback is also employed. In this case, the control law takes the form:

$$\beta(\mathrm{s}) = k_P\Big(\theta_{\mathrm{cmd}}(\mathrm{s}) - \theta(\mathrm{s})\Big) - k_D\, q(\mathrm{s}) - k_A\, \dot{q}(\mathrm{s}) \tag{39}$$

where k_A is the acceleration feedback gain. Similar to Sec. 4.1, the three gains can be tuned via pole placement of the closed-loop transfer function, which is obtained by substituting Eq. (39) in (27):

$$\frac{\theta(\mathrm{s})}{\theta_{\mathrm{cmd}}(\mathrm{s})} = -\frac{k_P}{1-\mu_c k_A}\,\frac{\mu_c}{\mathrm{s}^2 + \dfrac{l_\alpha\frac{\mu_\alpha}{V} - \mu_c k_D}{1-\mu_c k_A}\mathrm{s} - \dfrac{\mu_\alpha + \mu_c k_P}{1-\mu_c k_A}} \tag{40}$$

In contrast with the pure PD controller, the $\dot{q}$ feedback allows to minimise tracking errors because the desired steady-state gain G_0 be specified independently of ω_θ through the proportional gain as follows:

$$k_P[i] = \frac{\mu_\alpha[i]}{\mu_c[i]}\frac{G_0}{1-G_0}, \qquad i = 1, ..., N \tag{41}$$

which is scheduled along a grid of $N = 9$ points along the launcher's trajectory. The other two gains are then derived as a function of ω_θ and ζ as:

$$k_A[i] = \frac{1}{\mu_c[i]}\left(1 + \frac{\mu_\alpha[i] + \mu_c[i]k_P[i]}{\omega_\theta^2}\right), \quad k_D[i] = \frac{1}{\mu_c[i]}\left(l_\alpha[i]\frac{\mu_\alpha[i]}{V[i]} - 2\zeta\omega_\theta\left(1 - \mu_c[i]k_A[i]\right)\right) \tag{42}$$

For the estimation of $\dot{q}$ in Eq. (39), the same approach of Sec. 4.2, i.e. passing the angular rate q through the first-order derivative filter of Eq. (36), was followed. In practice, it was verified that the performance of this controller is fairly sensitive to the filter bandwidth $\omega_{\dot{q}}$. This impact is illustrated in Fig. 4, which shows root-mean-square (RMS) values of pitch error ($\theta_{\mathrm{err}} = \theta_{\mathrm{cmd}} - \theta$) vs. TVC rate ($\dot{\beta}$) for a step command in θ_{cmd} using different controllers and nominal conditions.

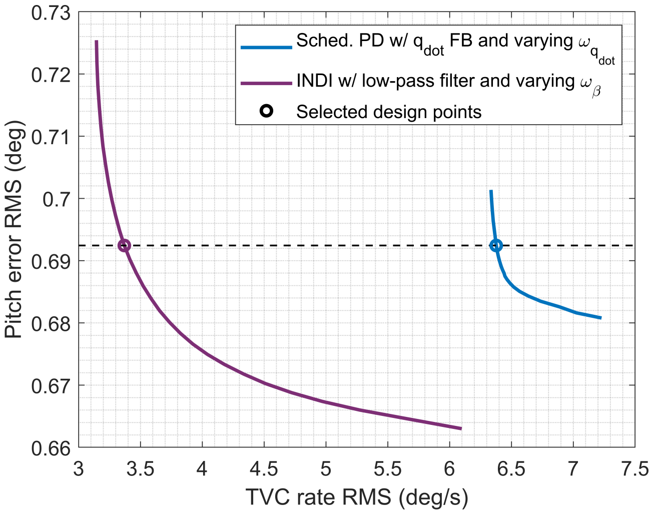


Figure 4: Tuning trade-off of angular acceleration feedback approaches

The blue line in Fig. 4 shows results using the scheduled PD controller with $\dot{q}$ feedback (FB) and varying values of the derivative filter bandwidth $\omega_{\dot{q}}$. Based on the results, the selection of $\omega_{\dot{q}}$ provides a key (and intuitive) tuning trade-off: increasing the bandwidth leads to smaller errors at the expense of more demanding TVC actuation, and vice-versa. A more favourable trade-off would likely be achieved by using a higher-order derivative filter, which is outside the scope of this paper.

4.4 INDI controller with low-pass filter

When applied to the pure INDI controller developed in Sec. 4.2, the same tuning trade-off analysis showed a much smaller sensitivity to $\omega_{\dot{q}}$ but unacceptably high TVC rates. To address this issue, the INDI controller was augmented with a low-pass filter at the output of the feedback linearisation loop, as depicted on the right-hand side of Fig. 3.
The feedback linearisation loop, outer linear gains and $\dot{q}$ estimation filter remain unchanged. The low-pass filter has bandwidth ω_β and a first-order structure as follows:

$$H_\beta(\mathrm{s}) = \frac{\omega_\beta}{\mathrm{s} + \omega_\beta} \tag{43}$$

The purple line in Fig. 4 shows the tuning trade-off using the INDI controller with low-pass filter and varying values of its bandwidth ω_β. Comparing with the PD controller with $\dot{q}$ feedback (blue line), the two controllers show a similar trend (i.e. smaller errors and larger TVC rates for higher bandwidths), yet the INDI controller leads to smaller TVC rates for the same level of error. As before, a more favourable trade-off would likely be achieved by using a higher-order low-pass filter, but this is outside the scope of the paper.

4.5 Control design summary

The four controllers in Sec. 4.1 to 4.4 have been designed so as to enforce the same closed-loop properties throughout the flight. These are:

- Natural frequency $\omega_\theta = 2.5$ rad/s;
- Damping ratio $\zeta = 0.8$;
- Steady-state error of 5%, i.e. $G_0 = 1.05$, only applicable to Sec. 4.3.

Furthermore, the bandwidth of the filters in Sec. 4.3 and 4.4, $\omega_{\dot{q}}$ and ω_{β}, has been tuned so as to provide the same pitch error in nominal conditions, as highlighted in Fig. 4. The robust performance of these controllers will then be analysed in Sec. 5.

Table 2 provides an overview of each controller's dependency on the model parameters and sensor measurements/estimates. As anticipated, from the scheduled PD controller to the INDI controller, there is a progressive reduction of model dependency and increased use of sensor information. More specifically, the INDI controller relies on measurements/estimates of $\dot{q}$ and β to fully circumvent the knowledge of the aerodynamics model.

Table 2: Dependencies per control design method

Control design method	Dependency on model parameters	Dependency on measurements/estimates
Scheduled PD	J, l_{c}, T, C_{N_α}, l_α, ρ, V	θ, q
Scheduled PD with $\dot{q}$ feedback	J, l_{c}, T, C_{N_α}, l_α, ρ, V	θ, q, $\dot{q}$
INDI with or without low-pass filter	J, l_{c}, T	θ, q, $\dot{q}, \beta$

5 TIME-DOMAIN ROBUST PERFORMANCE ANALYSIS

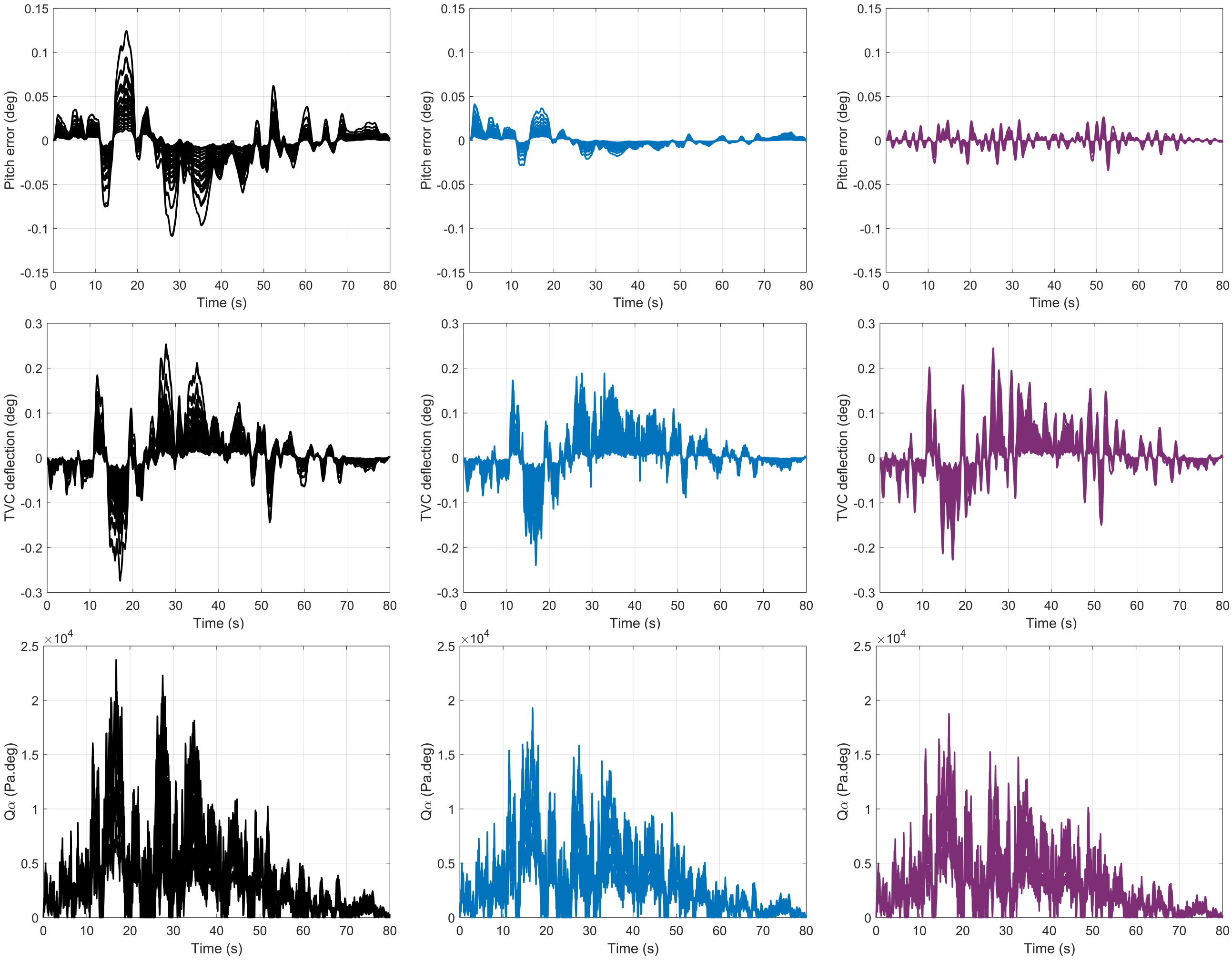
This section analyses and compares the nonlinear time-domain performance the controllers developed in Sec. 4. Figure 5 shows dispersed responses of the $2^8 = 256$ corner-cases within the uncertainty level of Table 1 when subjected to the same wind turbulence input v_{w}, modelled as described in Sec. 3. From the top to the bottom rows, the figure depicts the obtained pitch error θ_{err}, TVC deflection β and aerodynamic load indicator $Q\alpha$ along the trajectory. From left to right, the figure depicts results using the scheduled PD controller (Fig. 5a, in black), scheduled PD controller with $\dot{q}$ feedback (Fig. 5b, in blue) and INDI controller with low-pass filter (Fig. 5c, in purple). The pure INDI controller (wihout low-pass filter) is not shown as it leads to unacceptably high TVC rates.

From Fig. 5a to 5b, a reduction in the dispersion of all the indicators can be observed. This joint reduction clearly demonstrates the benefit of including $\dot{q}$ feedback in the control design. The pitch error (and partially the $Q\alpha$) is further reduced when using the INDI controller with low-pass filter, as depicted in Fig. 5c, at the expensive of higher TVC deflections (although still comparable to the pure PD controller). Note that $Q\alpha$ minimisation was not a specific control design objective in this case, but comes as a direct consequence of smaller pitch and drift errors, as indicated in Eq. (17).

In order to more clearly visualise these trends, Fig. 6a shows the wind response results using the same RMS θ_{err} vs. $\dot{\beta}$ plot of Fig. 4. Each point in Fig. 6a corresponds to a single simulation from Fig. 5. As anticipated, the pure PD controller (in black) provides the largest errors but the smallest TVC rates while, on the other hand, the pure INDI controller (in red) provides the smallest errors but the largest TVC rates. The PD controller with $\dot{q}$ feedback (in blue) and the INDI controller with low-pass filter (in purple) lie in-between the two extremes, with the latter controller performing better than the former (i.e. with slightly smaller errors and TVC rates) but only marginally.

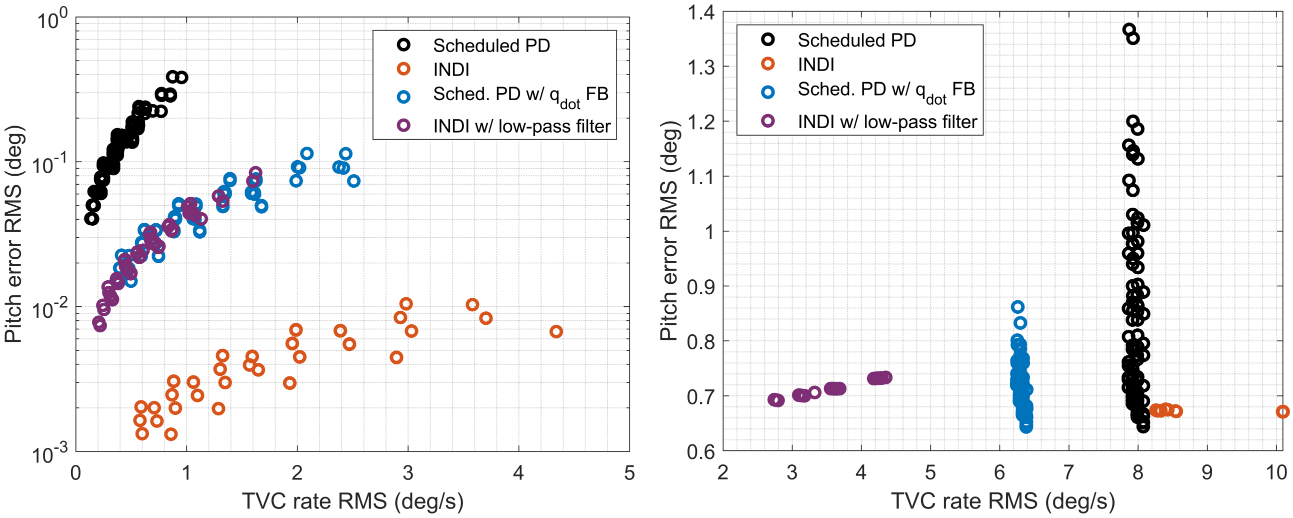
In order to complement the analysis, Fig. 6b shows the same type of results for a step command in θ_{cmd}. As before, the pure PD controller (in black) leads by far to the largest errors and the pure INDI controller (in red) to the largest TVC rates. Performance in terms of error and TVC rate improves using either the PD controller with $\dot{q}$ feedback or the INDI controller with low-pass filter, and the difference between these two controllers is now more evident than for the wind responses.

In nominal conditions, it was known from Fig. 4 that, for the same error, the INDI controller with low-pass filter (in purple) provides a smaller TVC rate than the PD controller with $\dot{q}$ feedback (in blue). Nonetheless, Fig. 6a shows that the former controller performs better also in terms of error, having a range of dispersion that is approximately four times smaller. The smaller error dispersion of the INDI controller with low-pass filter comes at the expense of a larger TVC rate dispersion, but its maximum value remains significantly lower than that of the PD controller with $\dot{q}$ feedback.



(a) Scheduled PD controller (b) Scheduled PD controller w/ $\dot{q}$ FB (c) INDI controller w/ low-pass filter

Figure 5: Comparison of Monte-Carlo wind responses using different controllers



(a) Wind responses (b) Step responses

Figure 6: Overview of Monte-Carlo results using different controllers

INDI-based controllers, by relying on angular acceleration and control input measurements/estimates, are known to be more sensitive to sensor noise and actuator delays than classical linear controllers. In order to assess this sensitivity, Fig. 7 extends Fig. 6a using the INDI controller with low-pass filter, showing wind simulation results with different combinations of:

- Gaussian noise on the angular rate signal, with $3\sigma = \{0, 0.05, 0.1\}$ deg/s, which affects the estimates of both q and $\dot{q}$ through Eq. (36);
- Time delay of $\{0, 40, 80\}$ ms on the signal commanded to the TVC actuator, corresponding to a delay of $\{0, 1, 2\}$ control samples.

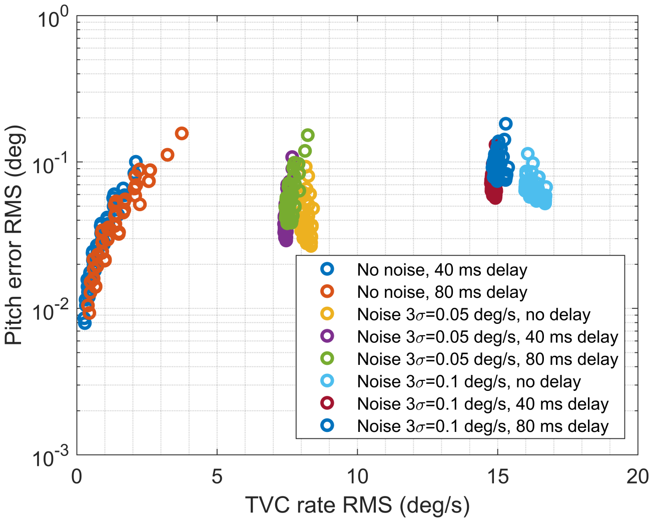


Figure 7: Impact of sensor noise and actuator delays on INDI controller (w/ low-pass filter)

From Fig. 7, it can be observed that, for the ranges considered, delays on the TVC signal have very little impact on the controller's performance. Noise on the angular rate signal, on the other hand, leads to a more noticeable degradation, with the resulting TVC rates increasing approximately linearly with the noise variance. This type of understanding is therefore critical when designing and sizing INDI-based GNC software and hardware. The impact of angular rate noise would likely be minimised by using a higher-order derivative filter $H_{\dot{q}}(\mathrm{s})$ or by including an angular acceleration sensor in the GNC system.

6 FREQUENCY-DOMAIN ROBUST STABILITY ANALYSIS

Because of the nonlinear nature of INDI, attaining an analytical proof of stability of INDI-based controllers [30] is much less trivial than for classical linear controllers. In order to mitigate this shortcoming, this section introduces a simple yet insightful frequency-domain approach to quantify stability degradation related to an imperfect feedback linearisation and to deviations from the control tuning conditions. This section is therefore focused on the controller developed in Sec. 4.4, not on a full comparison of controllers.

The proposed approach is based on linearised models of the nonlinear launcher simulator with the INDI control law in the loop at different flight conditions and on the fact that, for a perfect feedback linearisation, the channel $\nu(\mathrm{s}) \rightarrow \theta(\mathrm{s})$ behaves as a double integrator (recall Eq. (34)). The INDI controller design was carried out under this assumption.

Linearised models of $\nu(\mathrm{s}) \rightarrow \theta(\mathrm{s})$ can be obtained thanks to MATLAB® routine:

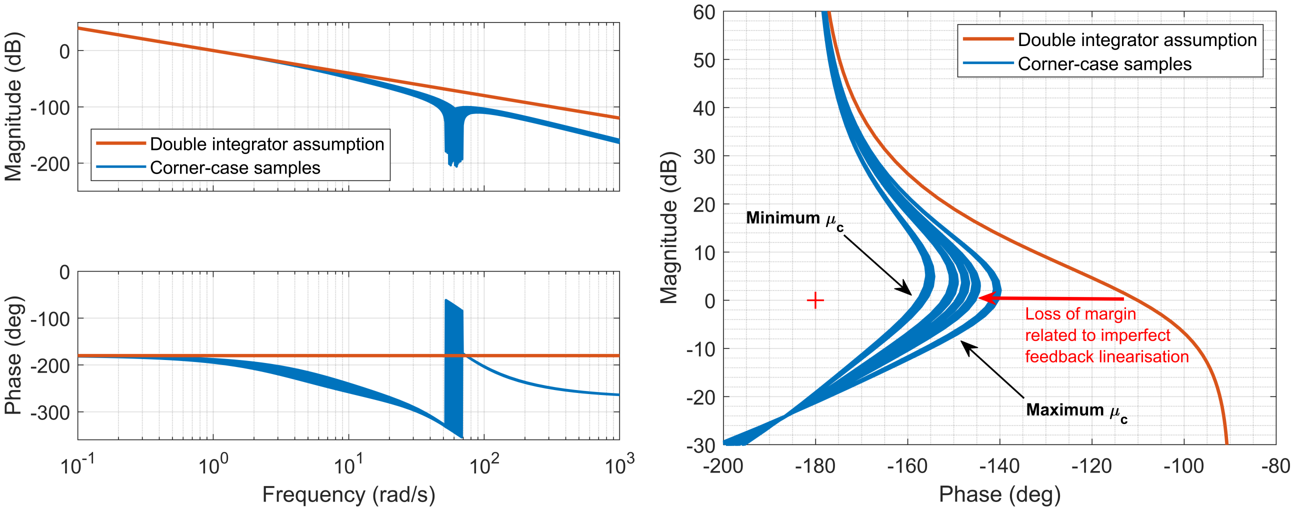
```
linearize(mdl, findop(mdl, t), ...)
```

where mdl is the Simulink® file instantiated with a certain configuration and t is the flight time instant. The analysis in this section considers the $2^8 = 256$ corner-cases (within the uncertainty level of Table 1) and 33 instants (spaced every 2.5 seconds along the trajectory).

Figure 8a shows the frequency response of the aforementioned linearised models (in blue), together with the "perfect" double integrator assumption (in red). This figure shows two important features:

- A mismatch between the linearised models and the double integrator assumption, which grows with the frequency and arises from the fact that drift motion, TWD effects, actuator dynamics and $H_{\hat{q}}(\mathrm{s})$ filter were neglected in the feedback linearisation;
- A dispersion of the linearised models, which is caused by deviations from the control tuning conditions due to the uncertain and time-varying nature of the model's parameters.

These models can be employed to assess the system's stability margins when the loop is closed using Eq. (37). To do so, it is convenient to plot the responses in a Nichols chart, which is depicted in Fig. 8b. For a detailed explanation of the application of Nichols charts to launcher stability assessment, the reader is referred to [33].



(a) Bode plot $\nu(\mathrm{s}) \rightarrow \theta(\mathrm{s})$

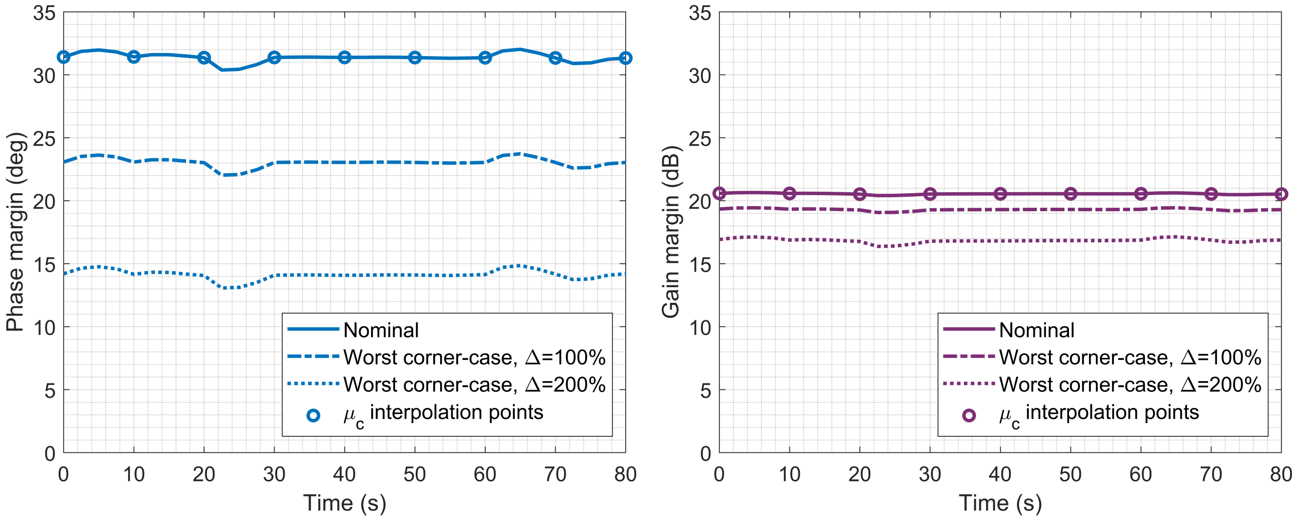
(b) Nichols chart $\theta_{\mathrm{err}}(\mathrm{s}) \rightarrow \theta(\mathrm{s})$

Figure 8: Frequency responses of linearised INDI-controlled plants for stability analysis

The impact of the imperfect feedback linearisation on the system's stability becomes evident from Fig. 8b: the phase margin is reduced approximately by half and the system can be gain-destabilised, which is not the case under the double integrator assumption. Nonetheless, all phase and gain margins remain substantial. When this is not the case, the linearised models of $\nu(\mathrm{s}) \rightarrow \theta(\mathrm{s})$ can be employed instead of Eq. (34) to re-tune the INDI outer control law. The stability margins are naturally driven by the value of μ_{c}, which is the main dependency of the INDI controller (recall Table 2). Accordingly, the margins become smaller for smaller values of μ_{c} as the system's control effectiveness decreases, and vice-versa.

In order to assess the degradation caused by uncertainties and time variations, the phase and gain margins are plotted as a function of time in Fig. 9a and b, respectively. These figures show the nominal margins (in continuous line), the worst (minimum) corner-case margins with the uncertainty level of Table 1 ($\Delta = 100\%$, in dash-dotted line) and the worst corner-case margins with twice the uncertainty level ($\Delta = 200\%$, in dotted line). The $N = 9$ control tuning points, i.e. the interpolation nodes of μ_{c}, are indicated in the figures using circular marks. The main results are then summarised in Table 3.

From Fig. 9, it can be seen that, at the control tuning points, nominal phase and gain margins are constant throughout the flight. This is expected because the closed-loop of Eq. (37) is time-invariant.



(a) Phase margin vs. time

(b) Gain margin vs. time

Figure 9: Nominal and worst-case margins of INDI controller (w/ low-pass filter)

Table 3: Stability margin budget ($\Delta = 100\%$)

Case	Phase margin	Gain margin
Double integrator assumption	69.84 deg	∞
Nominal case	31.37 deg	20.55 dB
Nominal case w/ deviation from μ_c interpolation points	30.38 deg	20.41 dB
Worst corner-case	23.04 deg	19.30 dB
Worst corner-case w/ deviation from μ_c interpolation points	22.03 deg	19.07 dB

Between tuning points there is naturally a variation in margins due to mismatches between actual and interpolated values of μ_c. Nonetheless, this variation is extremely limited and leads to a degradation of only 1 deg and 0.14 dB.

Stability degradation due to uncertainties is about one order of magnitude higher, leading to margin losses of 8.3 deg and 1.3 dB. In practice, the resulting stability margins must provide enough room to accommodate the impact of dynamical effects that were not considered in this study, such as flexible modes and non-collocated sensing. Nonetheless, the worst-case margins are plentiful, which suggests the feasibility of INDI-based launcher attitude control. In fact, the worst-case values remain acceptable even when the assumed level of uncertainty is doubled ($\Delta = 200\%$, shown only in Fig. 9, not in Table 3 for the sake of conciseness).

7 CONCLUSIONS

In conclusion, this paper presented a feasibility study of Incremental Nonlinear Dynamic Inversion (INDI) applied to a launcher ascent flight control scenario and highlighted its potential benefits over the traditional *ad hoc* linear control approach widely studied and implemented in practice. The paper introduced the INDI technique which mainly cancels the nonlinearities of a (nonlinear) system by means of state/output feedback and transforms it into a *linear form*, making it suitable to be controlled by a single linear control law without the need for gain-scheduling or other nonlinear approach (sliding mode, etc.). The paper also discussed the challenges associated with INDI-based control, such as sensitivity to sensor noise and actuator delay, and the difficulty of obtaining an analytical proof of stability. However, the potential benefits of INDI-based control outweighed these challenges as it was shown in a comprehensive nonlinear simulation campaign which considered wind disturbances and parameter uncertainties.

Finally, the paper proposed a simple, yet insightful linearisation-based approach to evaluate stability degradation and deviations from the (nominal) control tuning conditions. The results obtained in this study suggest that the INDI-based control approach could bring relevant improvements to launcher GNC, which may facilitate the transition to data-driven methods in the future. Outlook of this work will be furthering the analysis in terms of limits of performance (worst-case analysis) as well as addressing the impact of flexible modes and non-collocated sensing.

ACKNOWLEDGEMENTS

The authors would like to thank Mr. Massimo Casasco, head of ESA's GNC section, for making this feasibility study possible.

REFERENCES

[1] A. Marcos, D. Navarro-Tapia, P. Simplício, and S. Bennani, "Robust Control for Launchers: VEGA study case," *Journal of SICE*, vol. 3, no. 59, 2020.

[2] J. J. Slotine and W. Li, *Applied Nonlinear Control*. Prentice Hall Inc, 1990.

[3] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Prentice Hall, 2002.

[4] J. Reiner, G. J. Balas, and W. L. Garrard, "Flight Control Design Using Robust Dynamic Inversion and Time-scale Separation," *Automatica*, vol. 32, no. 11, pp. 1493–1504, 1996.

[5] P. R. Smith, "A Simplified Approach to Nonlinear Dynamic Inversion Based Flight Control," in *proceedings of the AIAA Atmospheric Flight Mechanics Conference*, 1998.

[6] G. Looye, "Design of Robust Autopilot Control Laws with Nonlinear Dynamic Inversion," *at–Automatisierungstechnik*, vol. 49, no. 12, pp. 523–531, 2001.

[7] T. J. Lombaerts, H. O. Huisman, Q. P. Chu, J. A. Mulder, and D. A. Joosten, "Flight Control Reconfiguration based on Online Physical Model Identification and Nonlinear Dynamic Inversion," in *proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit*, 2008.

[8] S. Bennani, and G. Looye, "Flight Control Law Design for a Civil Aircraft using Robust Dynamic Inversion," in *proceedings of the IEEE/SMC CESA'98*, 1998.

[9] P. R. Smith and A. Berry, "Flight Test Experience of a Nonlinear Dynamic Inversion Control Law on the VAAC Harrier," in *proceedings of the AIAA Atmospheric Flight Mechanics Conference*, 2000.

[10] B. J. Bacon and A. J. Ostroff, "Reconfigurable Flight Control using Nonlinear Dynamic Inversion with a Special Accelerometer Implementation," in *proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit*, 2000.

[11] B. J. Bacon, A. J. Ostroff, and S. M. Joshi, "Nonlinear Dynamic Inversion Reconfigurable Controller utilizing a Fault-tolerant Accelerometer Approach," NASA Langley Research Center, Tech. Rep., 2000.

[12] B. J. Bacon, A. J. Ostroff, and S. M. Joshi, "Reconfigurable NDI Controller using Inertial Sensor Failure Detection & Isolation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 37, no. 4, pp. 1373–1383, 2001.

[13] H. B. Chen and S. G. Zhang, “Robust Dynamic Inversion Flight Control Law Design,” in *proceedings of the ISSCAA 2008, 2nd International Symposium on Systems and Control in Aerospace and Astronautics*, 2008.

[14] S. Sieberling, Q. P. Chu, and J. A. Mulder, “Robust Flight Control Using Incremental Nonlinear Dynamic Inversion and Angular Acceleration Prediction,” *Journal of Guidance, Control and Dynamics*, vol. 33, no. 6, pp. 1732–1742, 2010.

[15] 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*, vol. 21, no. 8, pp. 1065–1077, 2013.

[16] P. Lu, E.-J. van Kampen, and Q. P. Chu, “Robustness and Tuning of Incremental Backstepping Approach,” in *proceedings of the AIAA Guidance, Navigation and Control Conference*, 2015.

[17] P. Lu, E.-J. van Kampen, “Active Fault-Tolerant Control System using Incremental Backstepping Approach,” in *proceedings of the AIAA Guidance, Navigation and Control Conference*, 2015.

[18] E. J. Smeur, Q. P. Chu, and G. C. de Croon, “Adaptive Incremental Nonlinear Dynamic Inversion for Attitude Control of Micro Air Vehicles,” *Journal of Guidance, Control and Dynamics*, vol. 39, no. 3, pp. 450–461, 2016.

[19] E. J. Smeur, G. C. de Croon, and Q. P. Chu, “Gust Disturbance Alleviation with Incremental Nonlinear Dynamic Inversion,” in *proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016.

[20] C. Vlaar, *Incremental Nonlinear Dynamic Inversion Flight Control.* MSc thesis, Delft University of Technology, Faculty of Aerospace Engineering, 2014.

[21] F. Grondman, G. Looye, R. O. Kuchar, Q. P. Chu, and E. van Kampen, “Design and Flight Testing of Incremental Nonlinear Dynamic Inversion-based Control Laws for a Passenger Aircraft,” in *proceedings of the AIAA Guidance, Navigation and Control Conference*, 2018.

[22] T. Keijzer, G. Looye, Q. P. Chu, and E.-J. van Kampen, “Design and Flight Testing of Incremental Backstepping based Control Laws with Angular Accelerometer Feedback,” in *proceedings of the AIAA SciTech Forum*, 2019.

[23] P. Acquatella B., W. Falkena, E.-J. van Kampen, and Q. P. Chu, “Robust Nonlinear Spacecraft Attitude Control Using Incremental Nonlinear Dynamic Inversion,” in *proceedings of the AIAA Guidance, Navigation, and Control Conference*, 2012.

[24] P. Acquatella B. and Q. P. Chu, “Agile Spacecraft Attitude Control: an Incremental Nonlinear Dynamic Inversion Approach,” *IFAC-PapersOnLine*, vol. 53, no. 2, 2020.

[25] P. Acquatella B., E. van Kampen, and Q. P. Chu, “A Sampled-data Form of Incremental Nonlinear Dynamic Inversion for Spacecraft Attitude Control,” in *proceedings of the AIAA SciTech Forum*, 2022.

[26] E. Mooij, “Robust Control of a Conventional Aeroelastic Launch Vehicle,” in *proceedings of the AIAA SciTech Forum*, 2020.

[27] E. Mooij, and X. Wang, “Incremental Sliding Mode Control for Aeroelastic Launch Vehicles with Propellant Slosh,” in *proceedings of the AIAA SciTech Forum*, 2021.

[28] E. Mooij, “Dynamic Inversion Heat-Flux Tracking for Hypersonic Entry,” in *proceedings of the AIAA SciTech Forum*, 2023.

[29] P. Rickmers *et al.*, “The Reusability Flight Experiment – ReFEx: From Design to Flight – Hardware,” in *proceedings of the International Astronautical Congress (IAC)*, 2021.

[30] Wang, X., van Kampen, E., Chu, Q.P., and Lu, P., “Stability Analysis for Incremental Nonlinear Dynamic Inversion Control,” *Journal of Guidance, Control, and Dynamics*, vol. 42, no. 5, pp. 1116–1129, 2019.

[31] P. Simplício, A. Marcos, and S. Bennani, “Reusable Launchers: Development of a Coupled Flight Mechanics, Guidance and Control Benchmark,” *Journal of Spacecraft and Rockets*, vol. 57, no. 1, pp. 74–89, 2020.

[32] P. Simplício, A. Marcos, and S. Bennani, “Launcher Flight Control Design using Robust Wind Disturbance Observation,” *Acta Astronautica*, vol. 186, pp. 303–318, 2021.

[33] 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*, vol. 39, no. 6, pp. 1342–1355, 2016.

[34] D. Navarro-Tapia, A. Marcos, S. Bennani, and C. Roux, “Structured H-infinity and Linear Parameter Varying control design for the VEGA Launch Vehicle,” in *The 7th European Conference for Aeronautics and Space Sciences*, Milan, Italy, Jul 3–6 2017.