Angular Acceleration Estimation in Hybrid INDI Using Extended Kalman Filtering

Antonio Minafra

TUDelft

Angular Acceleration Estimation in Hybrid INDI Using Extended Kalman Filtering

Thesis report

by

Antonio Minafra

to obtain the degree of Master of Science
at the Delft University of Technology
to be defended publicly on March 6, 2026 at 10:00

Thesis committee:

Chair:	Dr. Olaf Stroosma
Supervisors:	Dr. Erik-jan van Kampen
	Direnc Atmaca
External examiner:	Dr. Erwin Mooij
Place:	Faculty of Aerospace Engineering, Delft
Project Duration:	May, 2025 - February, 2026
Student number:	5027993

Faculty of Aerospace Engineering · Delft University of Technology

TU Delft

Preface

This report contains the results of eight months of thesis work and serves as the final requirement for obtaining the Master of Science in Aerospace Engineering at TU Delft, with a specialization in Control & Simulation.

This thesis also marks the end of a chapter that began six years ago, when I moved to the Netherlands to start my Bachelor's in Aerospace Engineering. These years have been filled with challenges, changes, and unforgettable experiences, but above all with tremendous personal and professional growth. I feel incredibly grateful to all the people who have been part of this journey.

First and foremost, I would like to thank my parents, who made this path possible from the very beginning and who have always believed in me getting to the end of the path. A special thank you goes to my girlfriend, Sofia, who has been by my side throughout this journey and supported me through both the good and the difficult moments. I am also deeply grateful to my friends in Delft, who became my second family. To Giacomo and Silvio, my former roommates, and to Marco, Giorgio, Alfonso, Gigi, and Sergio: thank you for the many dinners, parties, trips, and adventures we shared, and for all the ones still to come. Thanks as well to Emma, Archie, Ionut, Sara, Vittorio, Jonny, Dimitri, Marley, Serban, Flavia and many others. The list could go on, but if you know you have been part of it, you know – thank you for being there.

Special thanks also go to my friends from my hometown, Bari. Thank you to Alessandro, Luciano, Emanuele, and Gabriele, whom I have known for more than ten years, and with whom I have shared so much over the years and still do every time I return home.

I would also like to express my gratitude to my supervisors, Erik-Jan and Direnc, for their guidance, availability, and valuable feedback throughout this work. Thank you for your time, your support, and for giving me this opportunity.

With the hope of never losing the curiosity that drives me every day,

Antonio Minafra
Delft, February 2026

Contents

Nomenclature

List of Abbreviations

α	angle of attack
$\bar{c}$	mean aerodynamic chord
β	sideslip angle
$\boldsymbol{\nu}$	virtual control input vector
$\boldsymbol{\omega}$	angular rate vector
$\boldsymbol{\omega}_b$	body-axis angular rate vector
$\boldsymbol{f}(\boldsymbol{x})$	nonlinear system
δ_a	aileron deflection
δ_e	elevator deflection
δ_r	rudder deflection
$\mathbf{I}$	aircraft inertia tensor
$\mathbf{M}$	total moment vector in body axes
ω_c	complementary-filter crossover frequency
ω_d	bandwidth parameter of the filtered derivative
ϕ	roll angle
ϕ_m	measured roll angle
ψ	yaw angle / heading
ρ	air density
σ^2	variance
σ^2_{IMU}	IMU measurement noise variance
σ^2_{lat}	lateral-axis process noise variance used in EKF tuning
σ^2_{long}	longitudinal-axis process noise variance used in EKF tuning
τ	time lag (e.g., in autocorrelation)
τ_{IMU}	IMU sensor delay
θ	pitch angle
θ_m	measured pitch angle
$\tilde{y}_k$	innovation at time step k
ζ_d	damping ratio of the filtered derivative
b	wing span
c	chord
C_l	rolling moment coefficient
C_m	pitching moment coefficient
C_n	yawing moment coefficient
C_x	force coefficient in the x direction
C_y	force coefficient in the y direction
C_{l_a}	airframe contribution to rolling moment coefficient
$C_{l_{\delta_a}}$	aileron contribution to rolling moment coefficient
$C_{l_{\delta_r}}$	rudder contribution to rolling moment coefficient (roll coupling)
C_{m_a}	airframe contribution to pitching moment coefficient
$C_{m_{\delta_e}}$	elevator contribution to pitching moment coefficient
C_{n_a}	airframe contribution to yawing moment coefficient
$C_{n_{\delta_a}}$	aileron contribution to yawing moment coefficient (yaw coupling)
$C_{n_{\delta_r}}$	rudder contribution to yawing moment coefficient
dt	time step
e_p, e_q, e_r	angular-rate tracking errors in roll, pitch, and yaw
e_ϕ, e_θ, e_ψ	attitude tracking errors in roll, pitch, and yaw
$G(\boldsymbol{x})$	control effectiveness matrix
g_0	gravitational acceleration constant
H	linearized measurement Jacobian matrix

h height (altitude)

$h(\cdot)$ nonlinear measurement function

$H_f(s)$ filtered derivative transfer function

$H_s(s)$ synchronization filter transfer function for the actuator path

I_{xx}, I_{yy}, I_{zz} principal moments of inertia about body axes

I_{xz} product of inertia about the body x-z axes

J cost/objective function

K_k Kalman gain matrix at time step k

K_{aug} augmentation/blending gain matrix used in proportional Hybrid INDI fusion

$K_{D_\phi}, K_{D_\theta}, K_{D_\psi}$ outer-loop derivative gains for roll, pitch, and yaw attitude control

K_{I_p}, K_{I_q} inner-loop integral gains for roll and pitch rate control

$K_{P_p}, K_{P_q}, K_{P_r}$ inner-loop proportional gains for roll, pitch, and yaw rate control

K_{P_β}, K_{I_β} sideslip controller proportional and integral gains

$K_{P_\phi}, K_{P_\theta}, K_{P_\psi}$ outer-loop proportional gains for roll, pitch, and yaw attitude control

L rolling moment

L_T, M_T, N_T thrust-induced roll, pitch, and yaw moments

M pitching moment

N yawing moment

n_y lateral load factor

p, q, r angular rates (roll, pitch, yaw rates)

P_k state estimation error covariance matrix

PLA_1, PLA_2 Power Level Angle command of engine 1 and engine 2

Q process-noise covariance matrix

R measurement-noise covariance matrix

$r(\tau)$ autocorrelation function at lag τ

S wing reference area

s Laplace variable

$S(s)$ low-pass complementary filter transfer function

S_k innovation covariance matrix at time step k

T controller gain vector used in optimization

t time

$T(s)$ high-pass complementary filter transfer function

T_s sampling time

u control input vector

V airspeed

v_k measurement noise at time step k

V_{TAS} true airspeed

w_k process noise at time step k

x_k state vector at time step k

z_k measurement vector at time step k

List of Figures

1 Introduction

The design of flight control systems (FCS) is driven by the requirement to ensure adequate stability, flying qualities, and robustness across the aircraft operating envelope. For aircraft relying partially or fully on the FCS for control during flight, these requirements are essential to guarantee safe operation and acceptable handling qualities under nominal conditions as well as in the presence of disturbances and uncertainties. Traditionally, such control laws have been developed using linear control techniques combined with gain scheduling to address nonlinear aerodynamic effects. While effective in practice, this approach does not inherently guarantee stability or performance between scheduled operating points and becomes increasingly complex as the number of operating conditions increases.

To address these limitations, Nonlinear Dynamic Inversion (NDI) was introduced as an alternative to gain-scheduled linear control. NDI algebraically cancels the systems's nonlinearities resulting in globally linearized input–output behavior. This approach offers several advantages, most notably a uniform performance across the flight envelope. However, a key limitation of NDI lies in its strong dependence on an accurate representation of the full system dynamics, which remains challenging particularly under off-nominal conditions or for novel configurations. Model–plant mismatches and unmodeled dynamics can significantly degrade performance and compromise stability.

To mitigate the reliance on an accurate nonlinear plant model, Incremental Nonlinear Dynamic Inversion (INDI) has been developed. Rather than inverting the full nonlinear dynamics, INDI performs inversion on the incremental behavior of the system around the current operating point, relying solely on the control effectiveness matrix. This formulation makes INDI naturally more robust to model-plant mismatches and unmodeled dynamics compared to conventional NDI. The method has been successfully demonstrated on a wide range of platforms, including manned aircraft, quadrotor UAVs, and hybrid configurations. A defining characteristic of INDI is its reliance on measured state derivatives, such as angular accelerations, and accurate measurements of the current control inputs.

Despite its robustness advantages, sensor-based INDI introduces new practical challenges. Accurate and time-synchronized measurements of angular accelerations and control inputs are required to avoid oscillatory behavior. In most aircraft applications, angular acceleration is not directly measured and must instead be obtained by differentiating angular rate measurements from inertial measurement units (IMUs). The differentiation process amplifies sensor noise, which is widely recognized as one of the main limitations of INDI. Furthermore, delays and mismatches between sensor and actuator signal paths can significantly degrade stability margins. Synchronization of these signals is therefore essential and is commonly achieved through the use of dedicated synchronization filters. Nonetheless, measurement noise and delay remain among the primary challenges in practical INDI implementations.

Hybrid Incremental Nonlinear Dynamic Inversion (Hybrid INDI) was developed to address these shortcomings by combining the advantages of sensor-based and model-based approaches. Hybrid INDI blends angular acceleration information obtained from sensors with estimates generated by an on-board model (OBM) of the aircraft. By fusing these two information paths, Hybrid INDI aims to exploit the fast response and low noise characteristics of model-based predictions while retaining the accuracy of sensor measurements at low and medium frequencies. Several strategies have been proposed in the literature to perform this fusion. Proportional blending techniques augment the model-based estimate with a weighted contribution of the measured acceleration error, offering simplicity and improved robustness compared to purely sensor-based INDI, but at the cost of reduced stability margins in the presence of delays. Extended State Observers (ESO) treat model uncertainties and disturbances as additional states to be estimated, providing improved noise attenuation and reduced dependence on precise model knowledge. Predictive

filtering approaches use past measurements to forecast future accelerations, improving robustness to uncertainty, while Fixed-Lag Smoothing techniques use Kalman-based estimators to reduce noise and delay effects by incorporating delayed measurement information. Among these methods, the Complementary Filter (CF) has emerged as the most widely adopted solution. The CF blends sensor and model signals based on their frequency characteristics, applying a low-pass filter to the sensor path and a high-pass filter to the OBM estimate such that the two filters complement each other. This approach is simple to implement, computationally efficient, and has demonstrated good robustness to sensor noise, delays, and model–plant mismatches in comparison to sensor-based INDI.

An alternative approach to angular acceleration estimation within the Hybrid INDI framework is based on classic Kalman filtering. Kalman-filter-based estimators inherently combine model predictions with sensor measurements using a statistical least-squares weighting derived from the assumed noise properties of both information sources. In this work, a real-time Extended Kalman Filter (EKF) is developed and integrated into a Hybrid INDI flight control architecture for attitude control. The EKF-based estimator provides an alternative to the fixed-frequency blending performed by the Complementary Filter, adapting the relative weighting between model and sensor information based on the estimated state uncertainty. The main research question to be answered from this study is the following:

Main Research Question

In the context of flight control using Hybrid INDI, how can a Kalman Filter-based approach be used to provide an optimal blend of sensor measurements and on-board model predictions?

The above question has been divided into 4 sub-questions listed in Section 3.9. The proposed approach is evaluated using a nonlinear model of the Cessna Citation II small passenger aircraft. Two linear controllers operating in tandem are synthesized and tuned simultaneously based on predefined performance requirements. The parameters of both the CF and EKF estimators are determined using a robust tuning methodology that accounts for user-defined ranges of sensor noise, delay, and model–plant uncertainty, optimizing filter performance under worst-case conditions. The CF- and EKF-based Hybrid INDI architectures are then systematically compared under nominal and degraded operating scenarios to assess their relative performance and robustness.

This report is organized as follows. Part I presents the scientific article, which introduces the aircraft model, the methodology used to answer the research questions, and the main results. The results of the literature study that identified the research gap and research questions is described in Part II, together with the activities planned for the research. Chapter 4 contains additional results obtained during the research that contributed to answering the research questions. Chapter 5 presents the verification and validation activities performed for the model and the tools used. Finally, Chapter 6 and Chapter 7 present the conclusions of the study and recommendations for future research.

Part I

Article

Angular Acceleration Estimation in Hybrid INDI Using Extended Kalman Filtering

Antonio Minafra

Incremental Nonlinear Dynamic Inversion (INDI) is effective at controlling nonlinear aircraft dynamics by inversion of the control effectiveness matrix, but relies on accurate and time-synchronized measurements of the control input and the state derivative. Hybrid INDI address this limitation by blending sensor measurements with model-based estimates, most commonly through Complementary Filtering. In this work, an Extended Kalman Filter (EKF) is proposed as an alternative angular acceleration estimation method within a Hybrid INDI attitude control framework. A real-time EKF is developed and integrated into the control architecture of a nonlinear aircraft model, and its performance is compared against a baseline Complementary Filter approach. The parameters of both estimators are tuned using a robust multi-model optimization procedure that accounts for sensor noise, delay, and model-plant uncertainties. Simulation results under nominal and degraded conditions demonstrate that the EKF-based Hybrid INDI approach provides estimation performance comparable to that of the Complementary Filter, with improvements in scenarios of combined sensor noise and delay degradation.

Nomenclature

$\alpha,\ \beta$	=	angle of attack, sideslip angle
$\boldsymbol{\nu}$	=	virtual control input vector
$\boldsymbol{\omega},\ \boldsymbol{\omega}_b$	=	angular-rate vector, body-axis angular-rate vector
$\boldsymbol{f}(\boldsymbol{x})$	=	nonlinear state dynamics
$\mathbf{I},\ I_{xx},\ I_{yy},\ I_{zz},\ I_{xz}$	=	inertia tensor, principal moments of inertia, and product of inertia
$\mathbf{M}$	=	total moment vector in body axes
$\omega_c,\ \omega_d,\ \zeta_d$	=	CF crossover frequency, derivative-filter bandwidth and damping ratio
$\phi,\ \theta,\ \psi$	=	roll angle, pitch angle, yaw angle
$\rho,\ g_0$	=	air density, gravitational acceleration
$\sigma^2,\ \sigma^2_{\text{IMU}}$	=	variance, IMU measurement noise variance
$\sigma^2_{lat},\ \sigma^2_{long}$	=	lateral and longitudinal process-noise variances used in EKF tuning
$\tau,\ \tau_{\text{IMU}}$	=	time lag, IMU sensor delay
$b,\ \bar{c},\ S$	=	wing span, mean aerodynamic chord, wing reference area
$C_l,\ C_m,\ C_n$	=	rolling, pitching, and yawing moment coefficients
$C_x,\ C_y$	=	force coefficients in body x and y directions
$C_{l_a},\ C_{m_a},\ C_{n_a}$	=	airframe contributions to aerodynamic moment coefficients
$C_{l_{\delta a}},\ C_{m_{\delta e}},\ C_{n_{\delta r}}$	=	primary control-surface contributions to moment coefficients
$C_{l_{\delta r}},\ C_{n_{\delta a}}$	=	cross-coupling control-surface contributions to moment coefficients
F, H	=	linearized state and measurement Jacobian matrices
$G(\boldsymbol{x})$	=	control effectiveness matrix
$h,\ V,\ V_{\text{TAS}}$	=	altitude, airspeed, true airspeed
$H_f(s),\ H_s(s)$	=	filtered-derivative and synchronization filter transfer functions
$K_k,\ P_k$	=	Kalman gain and state estimation error covariance at time step k

$L,\ M,\ N$	=	roll, pitch, and yaw moments
$L_T,\ M_T,\ N_T$	=	thrust-induced roll, pitch, and yaw moments
n_y	=	lateral load factor
$p,\ q,\ r$	=	body-axis roll, pitch, and yaw rates
$Q,\ R$	=	process-noise and measurement-noise covariance matrices
$r(\tau)$	=	autocorrelation function at lag τ
$S(s),\ T(s)$	=	low-pass and high-pass complementary-filter transfer functions
$s,\ t,\ T_s,\ dt$	=	Laplace variable, time, sampling time, time step
$S_k,\ \tilde{y}_k$	=	innovation covariance and innovation (residual) at time step k
$u,\ \delta_a,\ \delta_e,\ \delta_r$	=	input vector, aileron, elevator, and rudder deflections
$v_k,\ w_k$	=	measurement noise and process noise at time step k
$x_k,\ z_k$	=	state vector and measurement vector at time step k

I. Introduction

The design of flight control systems (FCS) aims to ensure an adequate level of stability and robustness across the aircraft operating envelope. For aircraft relying partially or fully on automatic flight control, these requirements are essential to guarantee safe operation under nominal conditions as well as in the presence of disturbances and uncertainties. Conventional flight control laws are typically based on linear control techniques combined with gain scheduling to handle nonlinear aerodynamic effects. While effective in practice, this approach does not inherently guarantee stability or performance between scheduled operating points and leads to increasing design complexity.

Nonlinear Dynamic Inversion (NDI) was introduced as an alternative to gain-scheduled control by mathematically canceling the nonlinearities of the system, resulting in globally linearized input–output dynamics [1]. Although NDI offers uniform performance across the flight envelope, it relies on an accurate representation of the full system dynamics. In practice, model–plant mismatches and unmodeled dynamics can significantly degrade performance and compromise stability.

Incremental Nonlinear Dynamic Inversion (INDI) was developed to reduce dependence on the system plant by performing inversion on the *incremental* system dynamics using only the control effectiveness matrix. This makes INDI inherently more robust to parametric uncertainties and unmodeled dynamics, and its advantage over NDI control has been successfully demonstrated in a variety of practical applications [2–4]. INDI control is most effective when state derivatives, such as angular accelerations, can be accurately measured, together with the current control inputs applied to the actuators.

Despite its robustness to model mismatches and parametric uncertainty, INDI presents challenges related to synchronization and estimation. Angular acceleration is typically not measured directly and must be obtained by differentiating angular rate measurements from inertial sensors, a process that amplifies noise. In addition, delays and mismatches between sensor and actuator signal paths can significantly reduce stability margins. Measurement noise and delay therefore constitute the primary limitations in the implementation of INDI control laws.

Hybrid Incremental Nonlinear Dynamic Inversion (Hybrid INDI) addresses these limitations by blending differentiated sensor measurements with estimates generated by an On-Board Model (OBM). This approach is powerful because it exploits the low-noise characteristics of model-based predictions while retaining the accuracy of sensor measurements at low and medium frequencies. Several blending strategies have been used in the literature, including proportional blending [5], observer-based approaches [6], and Kalman-based approaches [7]. Among these, the Complementary Filter (CF) has emerged as the most widely adopted

solution due to its simplicity, computational efficiency, and robustness to sensor noise, delays, and model–plant mismatches. Its effectiveness has been demonstrated in various airborne systems [5, 8, 9].

An alternative estimation approach within the Hybrid INDI framework is based on Kalman filtering. Kalman filter-based estimators combine model predictions and sensor measurements using a statistical least-squares weighting derived from the assumed noise characteristics of the constituent signals. In this work, a real-time Extended Kalman Filter (EKF) is developed and integrated into a Hybrid INDI attitude and rate control architecture as a time-varying alternative to the fixed blending provided by the Complementary Filter (CF). The proposed approach is evaluated using a nonlinear model of the Cessna Citation II aircraft, and its performance is systematically compared against the baseline CF-based Hybrid INDI controller under both nominal and degraded conditions.

The remainder of this paper is organized as follows. Section II presents the nonlinear aircraft model used in this study, including the relevant states, inputs, equations of motion, and sensor characteristics. Section III presents the INDI framework and its hybrid formulation. The baseline Hybrid INDI controller based on complementary filtering, together with the controller tuning methodology, is described in Section IV.A. Section V details the development and integration of the EKF within the Hybrid INDI framework. The simulation results and a comparative performance assessment of the CF and EKF approaches under nominal and degraded conditions are presented in Section VI. Finally, Section VII summarizes the main conclusions and lists recommendations for future work.

II. Model Description

The aircraft model used for this research is the laboratory aircraft Cessna Citation II PH-LAB. The Citation II is a twin-jet fixed wing passenger aircraft equipped with two Pratt & Whitney JT15D-4 turbofan engines and able to perform a wide range of operations thanks to its large flight envelope.

A. States and Inputs

The dynamics of interest in this research are the rotational dynamics of the aircraft. These dynamics are represented by the attitude angles and rates, expressed in the body reference frame, described by the state vector in Equation 1.

$$x = \begin{bmatrix} p_b & q_b & r_b & \phi & \theta & \psi \end{bmatrix}^T \tag{1}$$

The aircraft rotation is controlled by using the three main aerodynamic surfaces: elevator, ailerons and rudder deflections. Additionally, the thrust can also be controlled by adjusting the Power Level Angle (PLA) of each engine. Thrust will be held at the a constant trimming value for every simulation. The complete input vector is shown in Equation 2.

$$u = \begin{bmatrix} \delta_a & \delta_e & \delta_r & PLA_1 & PLA_2 \end{bmatrix}^T \tag{2}$$

B. Equations of Motion

The nonlinear rotational dynamic equations of a rigid-body aircraft relate the angular acceleration to the applied moments and the mass properties of the aircraft. In this work, the rotational equations of motion are derived under the standard rigid-body aircraft assumptions. The aircraft is modeled as a rigid body with constant mass properties over the simulated maneuver, and the rotational dynamics are written about the center of gravity in the body reference frame. To derive the dynamic equations, the inertia matrix is first defined in Equation 3 for an aircraft assumed symmetric in the x-z plane, such that $I_{xy} = I_{yz} = 0$ while I_{xz} is

retained.

$$\mathbf{I} = \begin{bmatrix} I_{xx} & 0 & -I_{xz} \\ 0 & I_{yy} & 0 \\ -I_{xz} & 0 & I_{zz} \end{bmatrix} \tag{3}$$

The total moment vector around the axes of the body frame $\mathbf{M}$ is given by:

$$\mathbf{M} = \mathbf{I}\dot{\omega}_b + \omega_b \times \mathbf{I}\omega_b \tag{4}$$

where ω is the aircraft attitude rate expressed in body frame and defined as $\omega_b = \begin{bmatrix} p_b & q_b & r_b \end{bmatrix}$. The total moment vector can be split into moments due to aerodynamics, namely rolling (L), pitching (M), and yawing (N), and moments due to the engine thrust force:

$$\mathbf{M} = \begin{bmatrix} L \\ M \\ N \end{bmatrix} = \frac{1}{2}\rho V^2 S \begin{bmatrix} bC_l \\ \bar{c}C_m \\ bC_n \end{bmatrix} + \begin{bmatrix} L_T \\ M_T \\ N_T \end{bmatrix} \tag{5}$$

The aerodynamic moments are represented through look-up-table-based quasi-steady aerodynamic coefficients, which depend on the flight condition. Since both engine thrust settings are kept constant at the level specified by the trim condition, thrust-induced roll and yaw moments are assumed zero for the flight condition considered, while the thrust-induced pitching moment is treated as constant. Hence, the total moment equation can be rewritten as:

$$\mathbf{M} = \begin{bmatrix} L \\ M \\ N \end{bmatrix} = \frac{1}{2}\rho V^2 S \begin{bmatrix} bC_l \\ \bar{c}C_m \\ bC_n \end{bmatrix} + \begin{bmatrix} 0 \\ M_T \\ 0 \end{bmatrix} \tag{6}$$

where b is the wing span, $\bar{c}$ represents the mean aerodynamic cord, ρ is air density, V is airspeed, S is the wing surface area, and C_l, C_m, C_n are non-dimensional aerodynamic coefficients. To transform the equations of motion into a control-affine form suitable for dynamic inversion, the aerodynamic coefficients are rewritten into their two main contributions: the ones due to the aerodynamic surface deflections, and the contributions from the airframe. The moment equation above can therefore be rewritten as:

$$\begin{bmatrix} L \\ M \\ N \end{bmatrix} = \frac{1}{2}\rho V^2 S \begin{bmatrix} bC_{l\delta_a} & 0 & bC_{l\delta_r} \\ 0 & \bar{c}C_{m\delta_e} & 0 \\ bC_{n\delta_a} & 0 & bC_{n\delta_r} \end{bmatrix} \begin{bmatrix} \delta_a \\ \delta_e \\ \delta_r \end{bmatrix} + \frac{1}{2}\rho V^2 S \begin{bmatrix} bC_{l_a} \\ \bar{c}C_{m_a} \\ bC_{n_a} \end{bmatrix} + \begin{bmatrix} 0 \\ M_T \\ 0 \end{bmatrix} \tag{7}$$

Finally, substituting Equation 7 into Equation 4 and solving for the angular acceleration gives the rotational dynamic equation of the aircraft, in Equation 8.

$$\dot{\omega} = \mathbf{I}^{-1} \left\{ \frac{1}{2}\rho V^2 S \begin{bmatrix} bC_{l\alpha} \\ \bar{c}C_{m\alpha} \\ bC_{n\alpha} \end{bmatrix} - \omega \times \mathbf{I}\omega \right\} + \mathbf{I}^{-1} \frac{1}{2}\rho V^2 S \begin{bmatrix} bC_{l\delta_a} & 0 & bC_{l\delta_r} \\ 0 & \bar{c}C_{m\delta_e} & 0 \\ bC_{n\delta_a} & 0 & C_{n\delta_r} \end{bmatrix} \begin{bmatrix} \delta_a \\ \delta_e \\ \delta_r \end{bmatrix} + \begin{bmatrix} 0 \\ M_T \\ 0 \end{bmatrix} \tag{8}$$

The derivatives of the attitude angles follow directly from the kinematic relationship of the body reference frame rotating with respect to a local Earth-fixed navigation frame. In this work, the attitude kinematics are expressed using Euler angles (ϕ, θ, ψ).

$$\dot{\phi} = p_b + \tan\theta(q_b \sin\phi + r_b \cos\phi) \tag{9}$$

$$\dot{\theta} = q_b \cos\phi - r_b \sin\phi \tag{10}$$

$$\dot{\psi} = \frac{q_b \sin\phi + r_b \cos\phi}{\cos\theta} \tag{11}$$

C. Observations

The research aircraft is equipped with several sensors for measuring relevant variables needed for aircraft control. An overview of the observable variables can be found in Equation 12.

$$y = \begin{bmatrix} p_b & q_b & r_b & \phi & \theta & \psi & \alpha & \beta & V_{TAS} & h & n_y \end{bmatrix}^T \tag{12}$$

From the above equation it can be seen that the state vector is directly observable. Additionally, the angle of attack and sideslip α and β, the true airspeed V_{TAS}, the aircraft altitude h and the accelerometer output in the y-direction n_y are also measured.

D. Sensor Characteristics

The observation variables are obtained through on-board sensor measurements. To model the real sensors, real world effects are added to the *clean* observed signal. The effects considered in this research are noise, delay, sampling time and quantization. Noise is assumed to be white, zero-mean and following a Gaussian distribution, defined by the variance. Delay and sampling time are measured in seconds and have to be a multiple of the simulation time step, which is the minimum sampling time. Values for these quantities differ per sensor and match the value of the most recent sensors on board of the PH-LAB. An overview of these sensors can be found in Table 1 [10, 11].

Table 1. Sensor Data Summary

Sensor	Signal	Noise σ^2	Sampling time [ms]	Delay [ms]
AHRS	ϕ, θ	$1.0 \cdot 10^{-9}$	20	90
Hall Sensor	$\delta_a, \delta_e, \delta_r$	$5.5 \cdot 10^{-7}$	10	<1
Air data boom	α, β	$1.0 \cdot 10^{-9}$	10	100
Air data sensor	$V_{\text{TAS}}, h, \dot{h}$	$8.5 \cdot 10^{-4}$	60	300
IMU	$p, q, r, \phi, \theta, \psi, n_y$	$4.0 \cdot 10^{-7}$	1	15

III. INDI and Hybrid INDI

A. Incremental Nonlinear Dynamic Inersion (INDI)

Incremental Nonlinear Dynamic Inversion (INDI) is a development of Nonlinear Dynamic Inversion (NDI), both based on principles of feedback linearization. While NDI cancels nonlinearities through inversion using a model of the system dynamics, it requires accurate knowledge of the nonlinear function $\boldsymbol{f}(\boldsymbol{x})$, making it sensitive to modeling errors and parameter uncertainties. To mitigate this dependency, the Incremental Nonlinear Dynamic Inversion (INDI) approach has been developed. The central idea of INDI is to exploit the fact that although the nonlinear dynamics may be unknown, instantaneous changes in the system's behavior, the *increments*, can be obtained using measurable signals. INDI thus performs inversion not on the full nonlinear model, but on the local incremental behavior around the current operating point.

The INDI inversion law mostly follows the same steps as for NDI, and it starts from the definition of a general, input-affine nonlinear system as in Equation 13

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + G(\boldsymbol{x})\, u, \quad x \in \mathbb{R}^n,\ u \in \mathbb{R}^m \tag{13}$$

By applying a first-order Taylor expansion around the current operating point $(\boldsymbol{x}_0, \boldsymbol{u}_0)$, the dynamics can be

approximated locally:

$$\dot{\boldsymbol{x}} \;=\; \boldsymbol{f}(\boldsymbol{x}_0) + G(\boldsymbol{x}_0)\,\boldsymbol{u}_0 \;+\; \frac{\partial}{\partial \boldsymbol{x}}\left[\boldsymbol{f}(\boldsymbol{x}) + G(\boldsymbol{x})\,\boldsymbol{u}\right]_{(\boldsymbol{x}_0,\boldsymbol{u}_0)}(\boldsymbol{x}-\boldsymbol{x}_0) \;+\; \frac{\partial}{\partial \boldsymbol{u}}\left[\boldsymbol{f}(\boldsymbol{x}) + G(\boldsymbol{x})\,\boldsymbol{u}\right]_{(\boldsymbol{x}_0,\boldsymbol{u}_0)}(\boldsymbol{u}-\boldsymbol{u}_0) \quad (14)$$

where the first two terms represent the derivative of the current state:

$$\boldsymbol{f}(\boldsymbol{x}_0) + G(\boldsymbol{x}_0)\,\boldsymbol{u}_0 = \dot{\boldsymbol{x}}_0$$

and the second Jacobian term can be simplified into:

$$\frac{\partial}{\partial \boldsymbol{u}}\left[\boldsymbol{f} + G(\boldsymbol{x})\,\boldsymbol{u}_0\right](\boldsymbol{u}-\boldsymbol{u}_0) = G(\boldsymbol{x}_0)\;(\boldsymbol{u}-\boldsymbol{u}_0)$$

A key assumption in INDI is the time-scale separation principle, which assumes that with fast enough actuators, the change in state between two successive time steps is negligible compared to the change in control input, thus the two can be treated separately. The state increments in Equation 14 can be neglected, as described by Equation 15:

$$\boldsymbol{x} - \boldsymbol{x}_0 \approx 0 \quad (15)$$

Using Equation 15 into Equation 14 leads to Equation 16:

$$\dot{\boldsymbol{x}} \;\approx\; \dot{\boldsymbol{x}}_0 + G(\boldsymbol{x_0})\;(\boldsymbol{u}-\boldsymbol{u}_0) \quad (16)$$

A virtual control input can be defined as the input to the linearization loop $\boldsymbol{\nu} = \dot{\boldsymbol{x}}$. The control input can be obtained by inverting Equation 16 and solving for u:

$$\boldsymbol{u} = \boldsymbol{u}_0 + G^{-1}(\boldsymbol{x}_0)(\boldsymbol{\nu} - \dot{\boldsymbol{x}}_0) \quad (17)$$

Equation 17 removes the dependency on the plant dynamics $\boldsymbol{f}(\boldsymbol{x})$ but introduces the terms u_0 and $\boldsymbol{x}_0$.
Due to its reliance on measurements, this formulation is commonly referred to as *sensor-based INDI*. An alternative formulation known as *model-based INDI* also exists. In this variant, the current derivative of the state $\dot{\boldsymbol{x}}_0$ is not obtained through measurements, but instead approximated using an On-Board Model (OBM), as shown in Equation 18:

$$\boldsymbol{u} = G(\boldsymbol{x}_0)^{-1} \cdot \Big[\boldsymbol{\nu} - \underbrace{\left(\boldsymbol{f}_{\mathrm{OBM}}(\boldsymbol{x}_0) + G(\boldsymbol{x}_0)\cdot\boldsymbol{u}_0\right)}_{=\dot{\boldsymbol{y}}_{\mathrm{OBM}}}\Big] + \boldsymbol{u}_0 \quad (18)$$

In recent years, sensor-based INDI gained significant traction within the flight control community due to its robustness against model uncertainties and reduced reliance on the system's plant [12, 13]. Grondman et al. (2018) presented the first successful demonstration of INDI on the same aircraft of this research [2]. The main challenge in INDI implementation is its dependency on sensor measurements, where limited sampling frequencies, noise filtering, and other processing can lead to performance degradation and oscillatory responses. Furthermore, obtaining direct measurements of the state derivative, $\underline{\dot{x}}_0$ is often difficult, and differentiating measurements magnifies the noise present in the signal. Model-based INDI is mainly affected by limitations tied to parameter uncertainties and model mismatches between the system plant and the OBM, which has been shown to degrade the controller's performance [14].

B. Hybrid Incremental Nonlinear Dynamic Inversion (Hybrid INDI)

Hybrid INDI was introduced primarily to address the challenges present in both sensor and model-based INDI control systems. The main enhancement of Hybrid INDI is integrating both system model information and sensor measurements. This is generally obtained by augmenting the outputs of the OBM with additional data from sensors, and combining them into a single state estimate used in the inversion. By blending model estimates with sensor measurements, Hybrid INDI benefits from the fast response and high-frequency prediction of the OBM, while still being accurate at lower frequencies thanks to sensor measurements.
Hybrid INDI starts from the same general input-affine system:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + G(\boldsymbol{x})\boldsymbol{u} \tag{19}$$

Unlike standard INDI which obtains the state derivative using either sensor data or a model, Hybrid INDI utilizes both to obtain an estimate via a fusion method:

$$\hat{\dot{\boldsymbol{x}}} = \mathcal{F}\left(\dot{\boldsymbol{x}}_{\mathrm{s}}, \dot{\boldsymbol{x}}_{\mathrm{OBM}}\right) \tag{20}$$

The control law then follows the familiar incremental structure, using the estimated state derivative:

$$\boldsymbol{u} = \boldsymbol{u}_0 + G(\boldsymbol{x}_0)^{-1}\left(\boldsymbol{\nu} - \hat{\dot{\boldsymbol{x}}}_0\right) \tag{21}$$

Numerous fusion strategies have been investigated in the literature for Hybrid INDI to combine $\underline{\dot{x}}_{\mathrm{meas}}$ and $\underline{\dot{x}}_{\mathrm{OBM}}$. These include a Proportional Blend using an augmentation matrix K_{aug} [12, 15], Complementary Filters (CF) based on frequency-domain blending [8, 9], and Extended-State Observers (ESO) that treat unknown dynamics and disturbances as an extended state [6, 12]. In addition, other approaches such as a linear Predictive Filter (Predictive INDI, PINDI) [16] and Fixed-Lag Smoothing [7] have been proposed. A Kalman-filter-based estimation has also been proposed in literature as a possible fusion strategy [8], since it inherently exploits both measurement and model simultaneously to output a statistical estimate based on sensor and model noise characteristics.

In this paper, a new method for estimation of the state derivative used for Hybrid INDI flight control is investigated. The method considered is an Extended Kalman Filter (EKF), which is widely used in the literature for nonlinear state estimation by combining predictions made by a process model with corrections from sensor measurements via local linearization. This approach fits well within the Hybrid INDI control framework, where model outputs are augmented with sensor information to obtain an estimate of the state derivative. The control problem considered is attitude and rate control of the Citation II aircraft, described in the previous section.

A first part of the research relates to the integration of a real-time EKF within a Hybrid INDI control system. The EKF-based estimation within the Hybrid INDI controller is compared to a Baseline controller using the more standard Complementary Filter. The comparison is carried out by introducing realistic sensor measurements corrupted by noise and delay, together with uncertainties in the plant parameters. The goal of the research is to investigate whether EKF-based estimation could bring any advantage in tracking performance and response stability for Hybrid INDI flight control under both nominal and degraded conditions.

IV. Baseline Hybrid INDI Controller

A. Baseline Controller Description

Putting together the Hybrid INDI theory and the model considered for this research leads to the final Hybrid INDI equations. From Equation 19, which is the standard control-affine equation for a generic system, the

control input for inversion is given by Equation 21. Using the model in question for the aircraft rate, described by Equation 8, the control effectiveness matrix is inverted to obtain the control law in Equation 22.

$$\begin{bmatrix}\delta_a\\ \delta_e\\ \delta_r\end{bmatrix} = \begin{bmatrix}\delta_{a_0}\\ \delta_{e_0}\\ \delta_{r_0}\end{bmatrix} + \mathbf{I}\left(\frac{1}{2}\rho V^2 S \begin{bmatrix} bC_{l_{\delta_a}} & 0 & bC_{l_{\delta_r}} \\ 0 & \bar{c}C_{m_{\delta_e}} & 0 \\ bC_{n_{\delta_a}} & 0 & C_{n_{\delta_r}}\end{bmatrix}\right)^{-1}\left(\begin{bmatrix}\nu_{\dot{p}}\\ \nu_{\dot{q}}\\ \nu_{\dot{r}}\end{bmatrix} - \begin{bmatrix}\hat{\dot{p}}_b\\ \hat{\dot{q}}_b\\ \hat{\dot{r}}_b\end{bmatrix}\right) \tag{22}$$

Where the virtual input vector $\boldsymbol{\nu}_{\dot{\omega}} = \begin{bmatrix}\nu_{\dot{p}} & \nu_{\dot{q}} & \nu_{\dot{r}}\end{bmatrix}^T$ represents the desired angular acceleration. The angular acceleration at the current time, $\hat{\dot{\omega}}$, is obtained through a combination of sensor measurements and OBM predictions, as outlined by Equation 20. The information coming from both sources is fused using a Complementary Filter (CF). The Complementary Filter approach fuses model outputs with sensor measurements by blending signals based on their frequency characteristics.

Using this approach, a low-pass filter $S(s)$ is applied to sensor measurements, which contain high-frequency noise, while a high-pass filter $T(s)$ is applied to model outputs, which respond quickly to control inputs but contain model mismatches, which can be treated as low-frequency noise components. By setting an appropriate crossover frequency ω_c, these filters complement each other to produce an all-pass estimate of the state derivative. The complementary filter used for this study is a first-order filter characterized only by the crossover frequency which sets the dependency more on either model or sensor. This differs from other second-order CF used in Hybrid INDI literature, characterized by two parameters, and in which the low pass filter acts also as the derivative filter [5, 8, 9]. The high and low pass filter used for the CF in this research are described in Equation 23 Note that these filters complement each other so that $S(s) + T(s) = 1$

$$T(s) = \frac{s}{s+\omega_c} \qquad\qquad S(s) = \frac{\omega_c}{s+\omega_c} \tag{23}$$

As angular acceleration $\dot{\omega}$ is not directly measured, it needs to be differentiated from the measured angular rate. A second-order filtered derivative is used for simultaneous differentiation and noise attenuation, described by the transfer function in Equation 24. This filter is characterized by two parameters: the natural frequency $\omega_d = 25$, which sets the filter bandwidth and thus determines the trade-off between responsiveness and noise attenuation, and the damping ratio $\zeta_d = 0.7$, which governs the transient behavior of the filter, in particular the level of overshoot and oscillatory response[8].

$$H_f(s) = \frac{\omega_d^2\, s}{s^2 + 2\zeta_d\omega_d s + \omega_d^2} \tag{24}$$

The On-Board Model (OBM) predictions for the angular accelerations, $\dot{\omega}_m$, are obtained using the same rotational dynamics as in Equation 8. When sensor delay, noise and model mismatches are absent, the $\dot{\omega}$ calculated by the OBM is equal to the rotational acceleration of the plant. Plant-model mismatches will be then introduced to test the robustness of the estimation method presented in this research.

The current input given to the system u_0 in Equation 22 is measured using the actuator sensors which monitor the current position of the actuators. As highlighted by previous research the feedback path of the input and the feedback path of the state derivative need to be correctly synchronized [8, 9, 12]. This is achieved by introducing the same filter dynamics and known delays of the state feedback path in the actuator feedback path. The synchronization filter is then given by Equation 25 in which τ_{IMU} is the IMU sensor delay as listed in Table 1

$$H_s(s) = \left[H_f(s)\cdot\frac{S(s)}{s} + T(s)\right] e^{-s\tau_{\text{IMU}}} \tag{25}$$

The desired response for rotational rate control can be obtained by synthesizing an appropriate linear controller. The linear controller used for the inner loop rate control consist of a PI controller for which the proportional and integrator gains operate on the error between the desired and actual angular rate. The structure of the PI controller is the following:

$$\begin{aligned} \nu_{\dot{p}}(t) &= K_{P_p}\, e_p(t) + K_{I_p} \int_0^t e_p(\tau)\, \mathrm{d}\tau \\ \nu_{\dot{q}}(t) &= K_{P_q}\, e_q(t) + K_{I_q} \int_0^t e_q(\tau)\, \mathrm{d}\tau \\ \nu_{\dot{r}}(t) &= K_{P_r}\, e_r(t) \end{aligned} \tag{26}$$

Where e_ω is the error between the desired and current angular rate of the aircraft $\omega_d - \omega_s$. The proportional term is used to shape the rate error response and the integral term to eliminate steady-state error.
An outer control loop is also introduced to give reference commands directly on the aircraft attitude angles. Attitude control is achieved by using the kinematic relation between attitude angles and angular rates of Equation 11 and applying NDI inversion to obtain the desired attitude angle derivatives as in Equation 27.

$$\omega_d = \begin{bmatrix} p_d \\ q_d \\ r_d \end{bmatrix} = \begin{bmatrix} 1 & \tan\theta \sin\phi & \tan\theta \cos\phi \\ 0 & \cos\phi & -\sin\phi \\ 0 & \dfrac{\sin\phi}{\cos\theta} & \dfrac{\cos\phi}{\cos\theta} \end{bmatrix}^{-1} \begin{bmatrix} \nu_{\dot{\phi}} \\ \nu_{\dot{\theta}} \\ \nu_{\dot{\psi}} \end{bmatrix} \tag{27}$$

As for the inner controller, the virtual input vector $\boldsymbol{\nu}_{\dot{\theta}} = \begin{bmatrix} \nu_{\dot{\phi}} & \nu_{\dot{\theta}} & \nu_{\dot{\psi}} \end{bmatrix}^T$ can be shaped using linear control. For the outer loop, a PD controller with the structure of Equation 28 is used. Here the error e_θ is the difference between the commanded attitude and the measured one, $\theta_d - \theta_s$.

$$\begin{aligned} \nu_{\dot{\phi}}(t) &= K_{P_\phi}\, e_\phi(t) + K_{D_\phi} \frac{d}{dt} e_\phi(t) \\ \nu_{\dot{\theta}}(t) &= K_{P_\theta}\, e_\theta(t) + K_{D_\theta} \frac{d}{dt} e_\theta(t) \\ \nu_{\dot{\psi}}(t) &= K_{P_\psi}\, e_\psi(t) + K_{D_\psi} \frac{d}{dt} e_\psi(t) \end{aligned} \tag{28}$$

The outer controller also includes a sideslip controller similar to Grondman et al.[2] and using the assumptions of Miller [17]. The main objective of sideslip control is to manually control the sideslip angle in absence of a Fly-By-Wire system for the yaw channel in order to achieve coordinated turns ($\beta_d = 0$). The control command is formulated as a desired yaw rate which serves as the reference input for the inner-loop angular rate controller, as in Equation 26.

$$r_d = \frac{g_0}{V_{\text{tas}}} \left(n_y + \sin\phi_s \cos\theta_s\right) - \left(K_{I_\beta} \int_0^t (\beta_d - \beta_s) d\tau - K_{P_\beta} \beta_s\right) \tag{29}$$

The architecture of the baseline CF controller using INDI inversion and synchronization is shown in Figure 1.

B. Controller Tuning

The tuning approach for finding the optimal gains for the inner and outer loop controllers follows the method already proposed in literature by Grondman [2] which in terms come from the work of Looye et al. [18] [19] carried out at DLR, which was scaled down and adapted for this research. The method is based on multi-objective optimization of controller gains using a number of demands which have to be met by the controller. Since some model parameters, for instance the CF crossover frequency, external to the controller

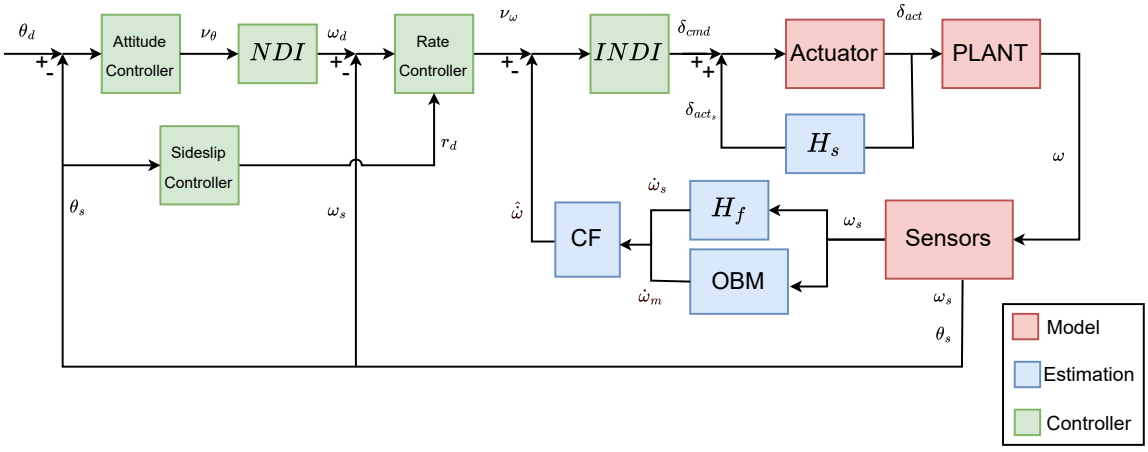


Figure 1. Baseline Controller diagram.

can influence its performance and the gains found are dependent on these parameters, some assumptions are made first, which allow to simplify the tuning problem and split the tuning into two different parts, one for the longitudinal and one for the lateral:

- If ideal signals are used, then the controller performance is independent of the hybrid estimation, therefore of ω_c.
- Even if the lateral and longitudinal axes are coupled ($I_{xz} \neq 0$), the influence from one to another can be assumed to be negligible, i.e. a large rolling moment induces a low pitching moment, which does not lead to instability in the pitch axis.
- It is assumed that control surfaces only influence the motion in their corresponding axes: elevators for longitudinal pitch and ailerons & rudder for lateral-directional roll and yaw.

The multi-objective optimization algorithm `fminimax` from MATLAB is used to find an appropriate vector of gains T for both the inner and outer controllers, that satisfies user-imposed objectives. The objectives, or criteria j used for tuning refer to the performance qualities for a response, given a simple attitude tracking task. A comprehensive list of these criteria for both axes is given in Table 2 and Table 3. For the lateral control case a step input of 5 deg/s is applied to the roll angle command ϕ for a duration of 1 s, while the sideslip angle is constrained to $\beta = 0$. For the longitudinal control case, a step input of 5 deg/s is applied to the pitch angle command θ for a duration of 1 s. The response is assumed to have reached steady-state level after a simulation time of $15s$

Criteria c_j are scaled by dividing them by the corresponding demand value , as in Equation 30 to obtain the scaled criteria $\hat{c}_j$. Values lower than 1 are considered unsatisfactory. The cost function $J(T)$ in Equation 31 evaluates all scaled criteria and picks the maximum value. The optimization algorithm tries to find the gains that satisfy all the objectives simultaneously.

$$\hat{c}_j(T) = \frac{c_j(T)}{d_j} \qquad (30) \qquad\qquad J(T) = \max_j \hat{c}_j(T) \qquad (31)$$

The gains for the inner and outer loops for both axes that have been found by the optimization algorithm with the above criteria and demands are Table 4. The sideslip controller gains are reported separately in Table 5 as they were manually tuned until a satisfactory and stable response was achieved.

Table 2. Lateral control performance requirements

Calculation	Demand
Overshoot ϕ [%]	15
Rise time ϕ [s], 10% to 90%	1.0
Settling time ϕ within 2% error [s]	6.0
Undershoot ϕ after first peak [%]	2
$\max\|\dot{\delta}_a\|$ [deg/s]	10
Cumulative absolute error β [deg]	0.5
Steady-state error after 15s ϕ [% of step value]	1.0

Table 3. Longitudinal control performance requirements

Calculation	Demand
Overshoot θ [%]	10
Rise time θ [s], 10% to 90%	1.5
Settling time θ within 2% error [s]	6.0
Undershoot θ after 15 s [%]	2
$\max\|\dot{\delta}_e\|$ [deg/s]	10
Steady-state error θ [% of final value]	1.0

Table 4. Final inner and outer controller gains

(a) Outer controller gains

Gain	Value
K_{P_ϕ}	2.37
K_{D_ϕ}	1.00
K_{P_θ}	2.60
K_{D_θ}	0.77

(b) Inner controller gains

Gain	Value
K_{P_p}	7.50
K_{P_q}	10.00
K_{P_r}	2.43
K_{I_p}	2.00
K_{I_q}	1.03

Table 5. Sideslip controller gains

Gain	Value
K_{P_β}	1.90
K_{I_β}	0.90

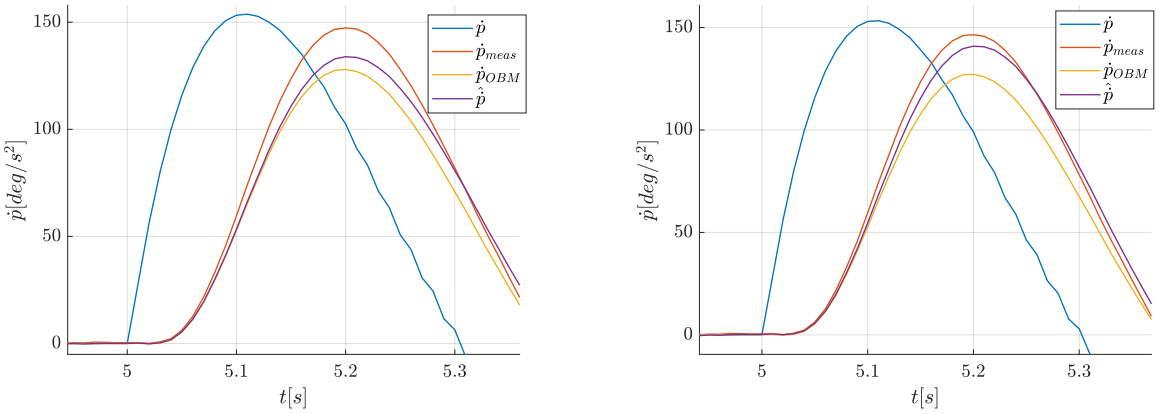
C. Optimal ω_c

The crossover frequency ω_c of the complementary filter defines the frequency threshold at which sensor measurements and OBM predictions are blended in the frequency domain. Consequently, there is no single globally optimal value for ω_c, as its optimal selection depends on the relative degradation levels of the measurements and the model-based estimates. For the *nominal* case, a crossover frequency of 35 rad/s is the optimal frequency in terms of lowest tracking error. The estimation of angular acceleration using a CF with this frequency can be seen in Figure 2b. The measured acceleration $\dot{p}_{meas}$ (orange) is similar in magnitude but lags behind the true acceleration $\dot{p}$ (blue). The OBM estimate $\dot{p}_{OBM}$ (yellow) is similar to the measurement, as the OBM inputs are the aircraft measurements. The CF hybrid $\hat{\dot{p}}$ (purple) between sensor and OBM sits in the middle of the two curves, leaning more towards the sensor. If ω_c is lowered to 5 rad/s then the CF estimate moves closer to $\dot{p}_{OBM}$.

D. Complementary Filter ω_c Tuning

A robust filter is developed which accounts for simultaneous degradations in both the sensor and OBM paths. To achieve this a set of uncertainty ranges for the most relevant model and sensor parameters was defined, as listed in Table 6 [2, 9]. An uncertainty of 30% is assumed for model parameters, together with an additional sensor delay of up to 30 ms and noise variance of $4 \cdot 10^{-7}$.

To determine a robust value for the crossover frequency under these uncertainty conditions, a multi-modal value tuning approach is adopted [2]. The algorithm pseudo-code is explained in the flowchart in Figure 3 The tuning process starts from a predefined vector of initial candidate values for ω_c. For each initial value, the system is evaluated at the extremes of the uncertain parameter range, and the three worst-performing cases based on the root-mean-square (RMS) of the tracking error are identified. The parameter is then tuned over a



(a) $\dot{p}$ estimation comparison for $\omega_c = 5\ rad/s$ **(b) $\dot{p}$ estimation comparison for $\omega_c = 35\ rad/s$**

Figure 2. Estimation of $\dot{p}$ from sensors and OBM.

Table 6. Parameter uncertainty ranges

Parameter	Uncertainty Range	Nominal
Sensor parameters		
$\Delta\tau_{\text{IMU}}$ [s]	$[0,\ 0.03]$	0.015
$\Delta\sigma^2_{\text{IMU}}$	$[0,\ 10^{-4}]$	$4 \cdot 10^{-7}$
Lateral parameters		
ΔI_{xx}	$[-0.3,\ 0.3]$	0
ΔI_{zz}	$[-0.3,\ 0.3]$	0
ΔC_{l_α}	$[-0.3,\ 0.3]$	0
$\Delta C_{l_{\delta_a}}$	$[-0.3,\ 0.3]$	0
$\Delta C_{l_{\delta_r}}$	$[-0.3,\ 0.3]$	0
Longitudinal parameters		
ΔI_{yy}	$[-0.3,\ 0.3]$	0
ΔC_{m_α}	$[-0.3,\ 0.3]$	0
$\Delta C_{m_{\delta_e}}$	$[-0.3,\ 0.3]$	0

one-dimensional grid centered around the current value, and the grid point yielding the lowest RMS error across the three worst cases is selected. A sensitivity check is then performed by comparing the RMS gradient between neighboring grid points; if the improvement is below a user-defined sensitivity threshold, the current value is considered optimal. Otherwise, the grid resolution is refined by one order of magnitude, and the tuning procedure is repeated. The optimal value obtained at the end of each iteration is used as the initial value for the next iteration, and the process continues until a predefined number of iterations is reached.

In Table 7, the optimal values of ω_c obtained after each tuning iteration are reported for the longitudinal control case. Regardless of the selected starting value, the tuning algorithm consistently converges to an optimal crossover frequency of approximately 136 rad/s already after the first iteration, which means that there is a well-defined global optimal value for ω_c within the uncertainty ranges listed above. As no further change in the optimal value is observed in the second iteration, the algorithm terminates after only two

iterations.
A similar convergence behavior is observed for the lateral control case, for which an optimal crossover frequency of approximately 109 rad/s is obtained. For simplicity, a single complementary filter is used on both axes. For this reason, the final crossover frequency is selected as the average of the optimal longitudinal and lateral values, resulting in $\omega_c = 122$ rad/s.
An additional observation concerns the magnitude of ω_c. For the uncertainty ranges considered, the resulting value favors sensor-based signals, although uncertainties are equally distributed between the OBM and the sensor parameters. This is because the measured states are also used as inputs to the OBM, causing sensor degradations to affect both estimation paths. As a result, the selected crossover frequency differs from previous choices in literature, based instead on nominal plant and sensor bandwidth [5, 8, 9].

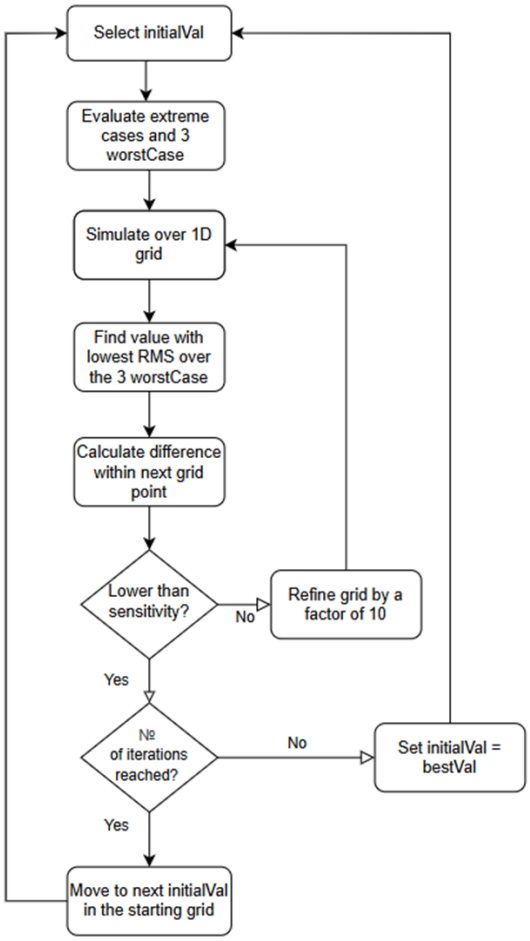


Figure 3. Pseudocode algorithm used for parameter tuning

V. EKF-Based Estimation

This work proposes a standard online Extended Kalman Filter (EKF) to blend sensor measurements and model predictions for angular velocity and acceleration estimation, to be used for INDI control. The standard EKF algorithm has been implemented, which is described extensively in literature [20, 21] and will not be reported here in its completeness for the sake of brevity. The EKF assumes a nonlinear process and measurement model both assumed to be corrupted by white Gaussian noise. Noise is modeled using covariance matrices,

Table 7. Tuning results for ω_c for the longitudinal case

Starting value ω_c [rad/s]	Optimal ω_c [rad/s]	
	Iteration 1	**Iteration 2**
15	135.20	136.86
25	135.20	136.86
45	135.20	136.86
60	136.86	136.86
90	136.86	136.86
120	136.86	136.86
150	136.86	136.86

Q and R for the model and measurements respectively. The confidence in the state estimation is expressed through the state covariance matrix $P_{k+1|k}$ which is updated recursively at each time-step k using the Kalman gain K which also dynamically weights the model-based state prediction against the corrections provided by the sensor measurements. In this sense, the EKF can be interpreted as a time-varying extension of the Complementary Filter used for the baseline controller.

A. EKF Algorithm

The proposed algorithm of the EKF is implemented online using interconnected Simulink function blocks and is integrated within the INDI controller loop. The EKF cannot be used directly for angular acceleration estimation since a state-transition model for $\dot{\omega}$ is not available. The filter estimates the angular velocity vector instead, using the rotational dynamics as the state transition model and direct measurements from the IMU. The process model used for the filter is:

$$x_k = \boldsymbol{\omega_k} \qquad z_k = \boldsymbol{\omega_{s_k}} \tag{32}$$

$$\omega_k = f(\omega_{k-1}, \delta_{k-1}) + w_{k-1} \qquad z_k = h(\omega_k) + v_k \tag{33}$$

where $f(\cdot)$ is Equation 8, δ represents the vector of surface deflections $\begin{bmatrix}\delta_a & \delta_e & \delta_r\end{bmatrix}$ and $h(\cdot)$ is the 3×1 identity vector since the states are directly measured by the IMU. Q and R are diagonal matrices initialized by choosing the appropriate value of the process and measurement noise variance σ^2. The process noise variance is set to two distinct values for the lateral and longitudinal axes, while the measurement noise variance is set to be equal to the sensor noise variance as in Table 1. The tuning of these parameters is discussed in the next subsection. As such the noise matrices are:

$$Q = \begin{bmatrix} \sigma^2_{lat} & 0 & 0 \\ 0 & \sigma^2_{long} & 0 \\ 0 & 0 & \sigma^2_{lat} \end{bmatrix} \qquad R = \begin{bmatrix} \sigma^2_{IMU} & 0 & 0 \\ 0 & \sigma^2_{IMU} & 0 \\ 0 & 0 & \sigma^2_{IMU} \end{bmatrix} \tag{34}$$

Furthermore, the state covariance matrix $P_{0|0}$ is initialized as the identity matrix. The Jacobian of the state transition matrix is Equation 35, derived by taking the partial derivative of Equation 8 with respect to the state where c_1 to c_9 are inertial constants. The measurement Jacobian function H_{k+1} is simply the identity matrix, as the states are measured directly by the IMU.

$$F_k = = \left.\frac{\partial f_d(x, u_{m,k})}{\partial x}\right|_{x=\hat{x}_{k|k}} = \begin{bmatrix} c_2 q & c_1 r + c_2 p & c_1 q \\ c_5 r - 2c_6 p & 0 & c_5 p + 2c_6 r \\ c_8 q & c_8 p - c_2 r & -c_2 q \end{bmatrix} \tag{35}$$

$$H_{k+1} = \left.\frac{\partial h(x)}{\partial x}\right|_{x=\hat{x}_{k+1|k}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{36}$$

The filter uses the covariance and Jacobian matrices to recursively update the estimate of the state, as in Equation 37, based on the weighting between process and measurements set initially by Q and R and the estimate at the previous time-step. The innovation $\tilde{y}_{k+1}$ is fundamental as it gives insights on the difference between the model prediction and the measurements and therefore on the validity of the Kalman filter assumptions of whiteness and uncorrelation.

$$\hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + K_{k+1}\left(z_{m,k+1} - H_{k+1}\,\hat{x}_{k+1|k}\right) \quad \text{with} \quad \tilde{y}_{k+1} = z_{m,k+1} - H_{k+1}\,\hat{x}_{k+1|k} \tag{37}$$

Since this EKF implementation estimates the angular rate and not the angular acceleration, which is also needed for INDI inversion, the output of the EKF is passed through the same derivative filter as the one used in the baseline controller, Equation 24. The filtered estimated angular rate is then passed to the inner rate controller while the angular acceleration is subtracted from the desired angular acceleration $\nu_{\dot{\omega}}$. An overview of how the EKF estimation fits within the Hybrid INDI control architecture is visualized in Figure 5.

B. EKF Synchronization

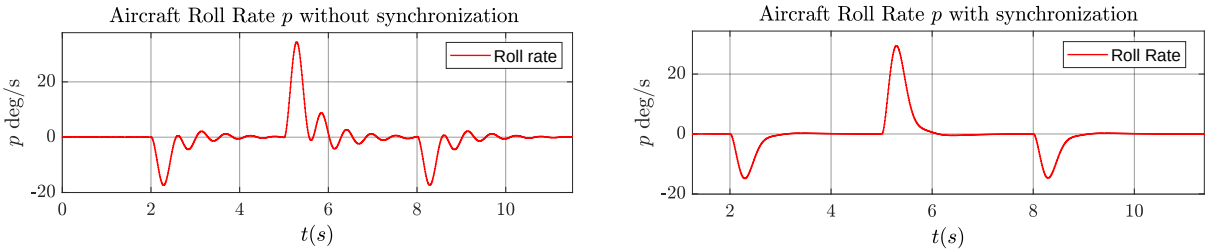
The EKF filter dynamics introduce lag, especially for small values of the Q/R ratio. To synchronize the actuator path, the same smoothing dynamics should be replicated on that path. However, this cannot be achieved with a single time-invariant filter, since the EKF is a time-varying filter. A practical way to tackle this issue is to pass the actuator measurement through a discrete-time smoothing filter parameterized by the instantaneous EKF Kalman gain K_k, as in Equation 38. The filter is effective because it reproduces, sample-by-sample, the same time-varying smoothing law that the EKF applies to the measured angular rates, which is the main source of lag in the EKF. The method is valid under the assumptions that the Kalman gain is diagonally dominant and that each state is controlled mainly by its corresponding control effector, which is valid for the system considered. The effect of the discrete-time smoother on the system dynamics can be observed by comparing the oscillatory aircraft Roll Rate estimation without the filter in Figure 4a against the more stable estimation with the filter in Figure 4b. Additionally, as with the CF, to compensate the lag generated by the derivative filter and sensor, an alternative synchronization filter is also introduced, described by the transfer function of Equation 39.

$$\delta_k^{\text{sync}} = \left(I - \text{diag}(K_k)\right)\delta_{k-1}^{\text{sync}} + \text{diag}(K_k)\,\delta_k \tag{38}$$

$$H_{s_{alt}}(s) = \left(H_f(s)\cdot\frac{1}{s}\right)e^{-s\,\tau_{\text{IMU}}} \tag{39}$$

C. EKF Parameter Tuning

The remaining EKF parameters to tune are the entries of the process-noise covariance matrix Q. This is widely recognized as a nontrivial task in the Kalman filtering literature [22, 23]. There is no unified procedure to translate parameter uncertainty and model–plant mismatch into a set of values for the process-noise covariance. For *nominal* conditions, the noise variance that produces the lowest RMS tracking error is $\sigma^2_{lat} = 2.15e^{-6}$ and $\sigma^2_{long} = 8.6e^{-6}$, values close to σ^2_{IMU}, suggesting a similar blending as CF. The process covariance matrix i also tuned for robustness, using the same approach employed to determine the optimal crossover frequency in Section IV.D. The assumed sensor-noise and model-parameter uncertainty ranges are identical to those listed



(a) Roll Rate response without smoothing filter **(b) Roll Rate response with smoothing filter**

Figure 4. Effect of smoothing filter for synchronization on Roll Rate response

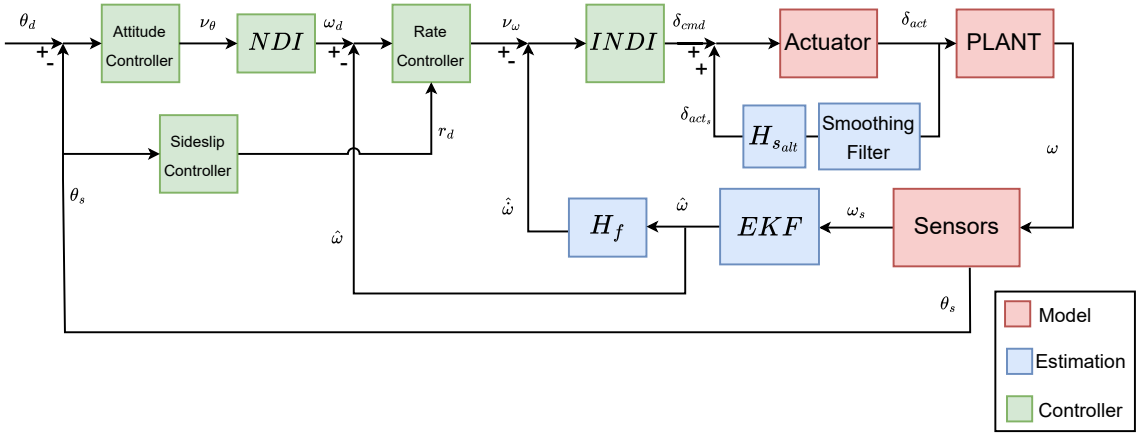
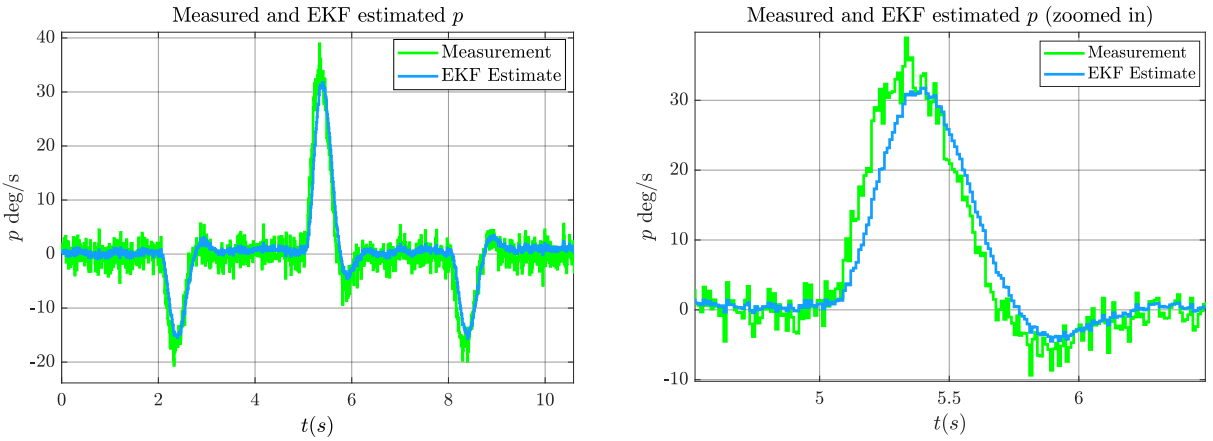


Figure 5. INDI with EKF diagram.

in Table 6. Keeping the R matrix fixed and tuning only the Q matrix allows to find the optimal Q/R ratio which is a measure on how the filter weights the model and measurements information in the Kalman gain. The absolute magnitudes of Q and R mainly affect the scale of the covariance matrix P_k, which quantify the filter's uncertainty of the estimation [24, 25]. The tuning results for the model noise variance of the two axes are presented in Table 8. More iterations than the CF are necessary before convergence to a final value. The EKF estimates of the aircraft angular rate show effective reduction of disturbances when compared to noisy measured rates in Figure 6a and Figure 6b, which proves that the EKF is correcting the measured signal with the model predictions.

Table 8. Tuning results for σ^2_{lat} and σ^2_{long}

Starting value σ^2	**Optimal σ^2_{lat}**		**Optimal σ^2_{long}**		
	Iteration 1	**Iteration 2**	**Iteration 1**	**Iteration 2**	**Iteration 3**
$1e^{-9}$	$2.5e^{-4}$	$1.17e^{-6}$	$1e^{-1}$	$2.1e^{-6}$	$4.6e^{-6}$
$1e^{-7}$	$1e^{-5}$	$1.17e^{-6}$	$1e^{-1}$	$2.1e^{-6}$	$4.6e^{-6}$
$1e^{-7}$	$1e^{-5}$	$1.17e^{-6}$	$1e^{-1}$	$2.1e^{-6}$	$4.6e^{-6}$
$1e^{-6}$	$1.17e^{-6}$	$1.17e^{-6}$	$6e^{-2}$	$2.1e^{-6}$	$4.6e^{-6}$
$1e^{-5}$	$1.17e^{-6}$	$1.17e^{-6}$	$9.1e^{-5}$	$2.1e^{-6}$	$4.6e^{-6}$
$1e^{-4}$	$8.2e^{-5}$	$1.17e^{-6}$	$2.2e^{-6}$	$4.6e^{-6}$	$4.6e^{-6}$



(a) Measured and estimated Roll Rate **(b) Measured and estimated Roll Rate (zoomed in)**

Figure 6. Comparison of sensor measurements and EKF estimates corrected with model predictions

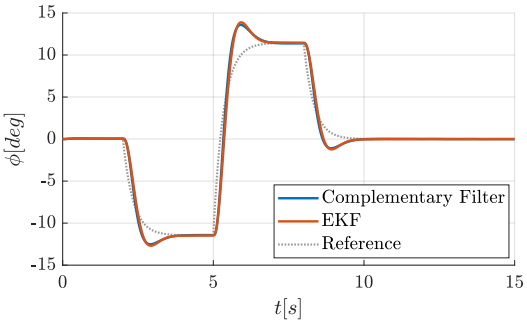
VI. Results & Discussion

Both the Complementary Filter (CF) and the Extended Kalman Filter (EKF) parameters were tuned using the same robust tuning methodology. Their performance in an attitude tracking task is evaluated under both nominal and degraded conditions. Performance is assessed primarily through a visual comparison of the aircraft attitude response relative to the reference command, as well as through RMS tracking error.

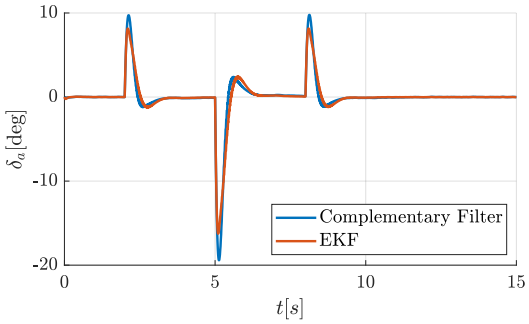
A doublet command of 0.2 rad and 0.1 rad for roll and pitch respectively is superimposed on the trimmed attitude as reference for the motion. Under nominal conditions, with no added noise delay or uncertainty, the performance of the controllers with the two different estimation methods can be seen in Figure 7a and Figure 8a. The filter parameters used for these simulations correspond to the values obtained for nominal conditions. The controller achieves accurate tracking of the reference trajectory, as expected given that it was tuned to satisfy predefined performance criteria. A moderate overshoot is observed in the roll angle ϕ, while the pitch angle θ exhibits no overshoot. No noticeable phase lag or lead relative to the reference command is present, and the steady-state error remains close to zero. The responses obtained using the CF and the EKF are virtually identical, indicating that the angular acceleration estimate $\hat{\dot{\omega}}$ produced by the EKF closely matches the blended sensor–model estimate provided by the CF. The control surface deflections shown in Figure 7b and Figure 8b are also very similar, with the EKF controller achieves the same response with less deflection than the CF.

Additional simulations are performed to investigate the sensitivity of the control system to sensor noise, transport delay, and model–plant mismatches. The filter parameters ω_c and Q employed in these simulations correspond to the values tuned for robustness, as described in subsection V.C. Sensitivity is assessed by varying the gains associated with model and sensor parameters and observing the resulting changes in system response, both in terms of stability and oscillatory behavior as well as RMS tracking error. Uncertainty levels beyond those considered during the tuning process are also examined to assess the limits before instability.

A mismatch in the form of a gain on the parameter ΔI_{xx} is introduced in the plant. Higher ΔI_{xx} makes the aircraft harder to roll, leading to a higher tracking error, as shown in Figure 9b. Conversely, a reduction in ΔI_{xx} facilitates the roll maneuver, but only up to a point, beyond which instabilities arise and the response becomes highly oscillatory, as illustrated in Figure 9a. The EKF controller shows less sensitivity to changes of this parameter compared to the CF.

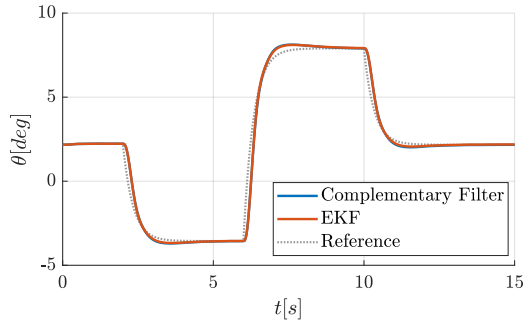


(a) Roll angle tracking performance under nominal conditions.

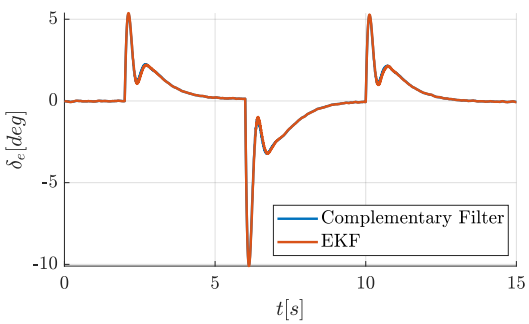


(b) Aileron deflection corresponding to the nominal roll maneuver.

Figure 7. Nominal roll-axis response and control effort for CF- and EKF-based estimation.



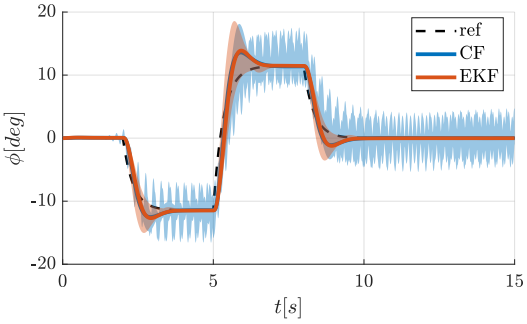
(a) Pitch angle tracking performance under nominal conditions.



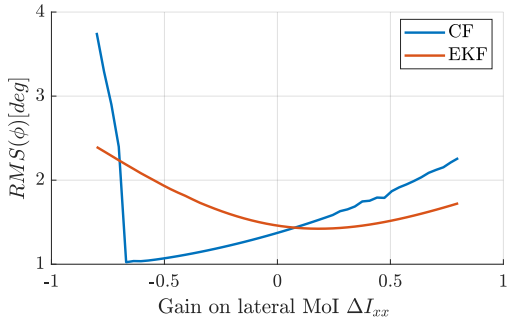
(b) Elevator deflection corresponding to the nominal pitch maneuver.

Figure 8. Nominal pitch-axis response and control effort for CF- and EKF-based estimation.

An increase in the aileron effectiveness parameter $\Delta C_{l_{\delta a}}$ produces the opposite effect, as shown in Figure 10b, making the aircraft easier to roll. At very low levels of control effectiveness, the response remains stable for the EKF but shows an unstable behavior for the CF. The EKF shows less sensitivity to this mismatch, indicating an improvement in robustness.

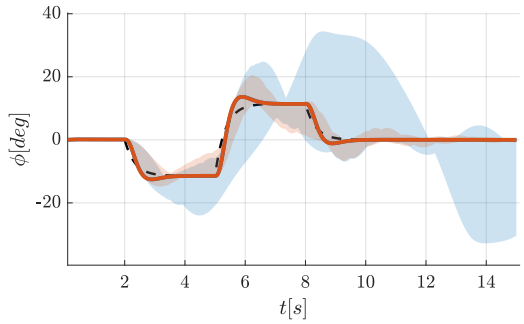


(a) Roll angle response under variations of the lateral moment of inertia ΔI_{xx}.

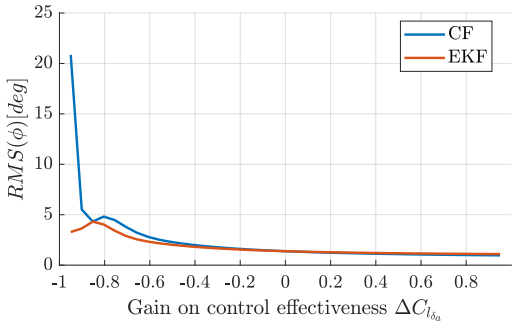


(b) RMS tracking error as a function of the lateral moment of inertia variation ΔI_{xx}.

Figure 9. Effect of lateral moment of inertia variations on roll response and tracking performance.



(a) Roll angle response under variations of the aileron effectiveness parameter $\Delta C_{l_{\delta a}}$.

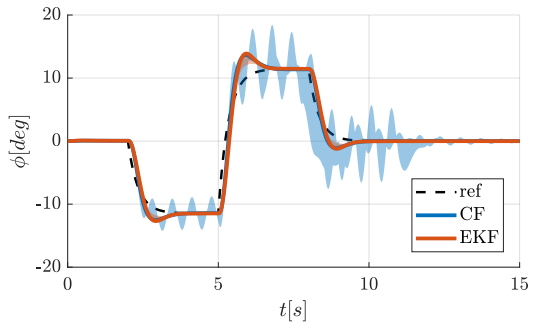


(b) RMS tracking error as a function of aileron effectiveness variation $\Delta C_{l_{\delta a}}$.

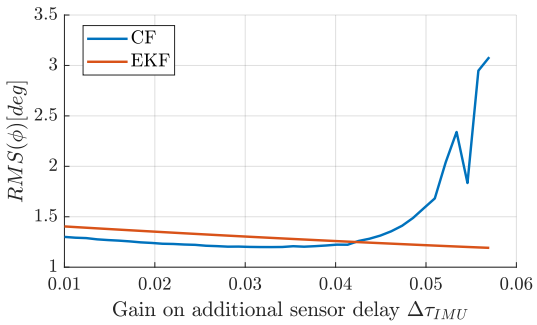
Figure 10. Effect of aileron effectiveness variations on roll response and tracking performance.

Next, sensitivity to sensor delay and noise is investigated. As shown in Figure 11b, increasing the transport delay in the IMU measurement path initially leads to a reduction in RMS tracking error, up to the delay value of 0.03 s used during tuning, as listed in Table 6. Beyond a delay of 0.04 s, oscillatory behavior begins to appear for both the CF controller, with oscillation amplitudes increasing as the delay is further increased, as illustrated in Figure 11a. The EKF shows clear improvements in robustness to additional delay, due to the time-varying nature of this filter and the synchronization filter based on the sample Klamn gain. Further simulations indicate that divergence for the EKF occurs at delays of approximately 0.07 s. Moreover, during delayed roll maneuvers, the pitch response becomes quickly unstable for the CF but not for the EKF, even if this axis is not directly excited.

Sensitivity to increased noise variance in the IMU measurement signal is evaluated. As shown in Figure 12a, the RMS tracking error increases monotonically with increasing noise magnitude, as expected. The CF and EKF exhibit nearly identical responses and error levels, and aside from increased oscillations, the closed-loop system remains stable even for noise variances as high as 10^{-1}.



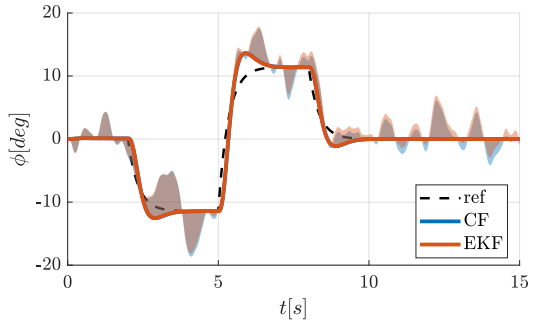
(a) Attitude response under increasing IMU transport delay.



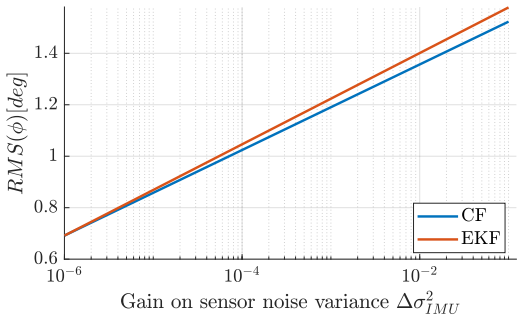
(b) RMS tracking error as a function of IMU transport delay.

Figure 11. Effect of IMU transport delay on roll response and tracking performance.

To assess the robustness of the estimation methods under combined model uncertainty and sensor degradation, 500 Monte Carlo simulations were performed. In each simulation, the parameters listed in Table 6 were varied using a uniform random distribution within their specified ranges. Table 9 summarizes the RMS attitude estimation error the simulations for both the CF and the EKF, evaluated separately for the lateral and longitudinal axes. For both axes, the two filters exhibit nearly identical mean RMS error, indicating



(a) RMS tracking error under increasing IMU measurement noise variance.



(b) Attitude response under increasing IMU measurement noise variance.

Figure 12. Effect of IMU measurement noise variance on roll response and tracking performance.

comparable performance under nominal conditions. However, the EKF demonstrates a significantly reduced variability. The resulting 2σ bound is notably smaller for the EKF which highlights improved robustness and reduced occurrence of high-error cases. These results are consistent with the sensitivity analysis discussed in the earlier paragraphs.

Table 9. Monte Carlo RMS attitude error statistics over 500 simulation runs

(a) Longitudinal axis

Filter	Mean	σ	Mean + 2σ
CF	0.48661	0.053	0.59
EKF	0.48517	0.032	0.54

(b) Lateral axis

Filter	Mean	σ	Mean + 2σ
CF	0.66237	0.125	0.91
EKF	0.65495	0.075	0.80

Overall, it can be concluded that, in the context of angular acceleration estimation for Hybrid INDI control laws, the EKF-based estimation approach constitutes a valid alternative to the CF. The performance of the two methods is comparable in most scenarios, with the EKF providing improvements over the CF only in cases of combined model and sensor parameter uncertainty. The EKF is also marginally less sensitive to unknown time delay and model-plant mismatches in the control effectiveness matrix. At the same time, the EKF requires a more complex implementation and additional computational steps, whereas the CF implementation is more straightforward.

VII. Conclusion

This work investigated the estimation of angular acceleration within the framework of aircraft attitude and rate control using Hybrid Incremental Nonlinear Dynamic Inversion (INDI). Past efforts have defined the Complementary Filter (CF) as the main method to combine sensor measurements, affected by delay and high-frequency noise, with On-Board-Model (OBM) predictions, which are subject to model–plant mismatches, in order to obtain an estimate of angular acceleration, a quantity that is typically not measured directly. In this study, the Extended Kalman Filter (EKF) was identified as a viable alternative estimation method, as its formulation naturally combines model-based predictions with sensor measurements through a statistically least-squares weighting based on assumed noise characteristics. An EKF-based estimation algorithm was developed for the attitude control of a small passenger aircraft and evaluated against a baseline CF-based approach. Two linear controllers were synthesized simultaneously to meet predefined performance

requirements, while the filter parameters were tuned using a robust multi-model approach accounting for uncertainties in both sensor and model–plant parameters. The robust tuning results indicate that, when combined model and sensor degradations are considered, sensor information is favored over OBM predictions, as the latter are also driven by the same sensor measurements. Simulation results obtained under identical reference commands demonstrate that the EKF-based architecture represents a feasible alternative for state estimation within the Hybrid INDI framework. The EKF shows performance improvements over the CF in scenarios of combined sensor delays, noise and model–plant mismatches. Furthermore, the parameters selected allow the INDI controller to keep the system stable in spite of signal degradation in the sensor and OBM path.

The validation of the EKF-based estimation approach should be strengthened by extending the simulation campaigns to include piloted flight-test data or hardware-in-the-loop experiments, enabling evaluation under realistic sensor and actuator conditions. In addition, the computational burden introduced by the Kalman Filter algorithm should be assessed, since it requires significantly more computational power than the Complementary Filter used in the baseline controller and could become a limitation for smaller systems.

Beyond Kalman filter-based methods, alternative nonlinear estimation techniques could also be considered. For instance, the H_∞ filter minimizes worst-case estimation error and may offer increased robustness to modeling uncertainties [26], while optimization-based approaches such as Moving Horizon Estimation and Maximum Likelihood Estimation can explicitly account for constraints and nonlinearities, at the cost of increased computational complexity.

Promising extensions of the EKF include adaptive variants that tune filter parameters online rather than relying on fixed values, potentially improving robustness to varying model parameters and system or sensor faults. Adaptive approaches applicable to Hybrid INDI already exist in the literature, including adaptive Complementary Filters based on optimization techniques such as least-squares estimation [27].

Finally, an alternative direction for improving angular acceleration estimation is to include it as a Kalman Filter state and make it directly measurable. This may be achieved using angular accelerometers in addition to rate gyros and fusing the measurements, as highlighted by [28]. When angular accelerometers are not feasible, multiple linear accelerometers can be used instead, as recent improvements in INDI control for UAV have shown, yielding higher accuracy than numerical differentiation of angular rate measurements and reduced sensitivity to sensor noise and mounting errors [29].

References

[1] Snell, S. A., Enns, D. F., and Garrard, W. L., "Nonlinear Inversion Flight Control for a Supermaneuverable Aircraft," *Journal of Guidance, Control, and Dynamics*, Vol. 15, No. 4, 1992, pp. 976–984. doi: 10.2514/3.20932.

[2] Grondman, F., Looye, G., Kuchar, R. O., Chu, Q. P., and Van Kampen, E.-J., "Design and Flight Testing of Incremental Nonlinear Dynamic Inversion-based Control Laws for a Passenger Aircraft," *2018 AIAA Guidance, Navigation, and Control Conference*, American Institute of Aeronautics and Astronautics, Kissimmee, FL, USA, 2018. doi: 10.2514/6.2018-0385.

[3] Smeur, E. J. J., Chu, Q., and De Croon, G. C. H. E., "Adaptive Incremental Nonlinear Dynamic Inversion for Attitude Control of Micro Air Vehicles," *Journal of Guidance, Control, and Dynamics*, Vol. 39, No. 3, 2016, pp. 450–461. doi: 10.2514/1.G001490.

[4] Smeur, E. J. J., De Croon, G. C. H. E., and Chu, Q., "Cascaded Incremental Nonlinear Dynamic Inversion for MAV Disturbance Rejection," *Control Engineering Practice*, Vol. 73, 2018, pp. 79–90. doi: 10.1016/j.conengprac.2018.01.003.

[5] Kim, C., "A hybrid INDI control for ensuring flying qualities in failures of Xcg measurement subsystem," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, Vol. 237, No. 4, 2023, p. 972–991. doi: 10.1177/09544100221113429.

[6] Lyu, H., Ye, Z., Chen, Y., Zhao, T., Gong, Z., Liu, X., Qin, B., and Chen, K., "Extended-state-observer-based angular acceleration estimation for supersonic aircraft lateral–directional control," *Appl. Sci. (Basel)*, Vol. 13, No. 11, 2023, p. 6598.

[7] Ludeña Cervantes, T. J., Choi, S. H., and Kim, B. S., "Flight Control Design using Incremental Nonlinear Dynamic Inversion with Fixed-lag Smoothing Estimation," Vol. 21, No. 4, 2020, pp. 1047–1058. doi: 10.1007/s42405-020-00273-8, URL https://link.springer.com/10.1007/s42405-020-00273-8.

[8] Kumtepe, Y., Pollack, T., and Van Kampen, E.-J., "Flight control law design using hybrid incremental nonlinear dynamic inversion," *AIAA SCITECH 2022 Forum*, American Institute of Aeronautics and Astronautics, Reston, Virginia, 2022.

[9] Traas, T., "Hybrid INDI with enhanced FEP," Master's thesis, October 2024. URL https://repository.tudelft.nl/record/uuid:1c3fb8d1-4e7a-4bda-8ce0-537a69fe64f8, accessed: 2025-07-12.

[10] Konatala, R. B., Van Kampen, E.-J., Looye, G., Milz, D., and Weiser, C., "Flight Testing Reinforcement Learning based Online Adaptive Flight Control Laws on CS-25 Class Aircraft," *AIAA SCITECH 2024 Forum*, American Institute of Aeronautics and Astronautics, Orlando, FL, 2024. doi: 10.2514/6.2024-2402, URL https://arc.aiaa.org/doi/10.2514/6.2024-2402.

[11] van 't Veld, R. C., "Incremental Nonlinear Dynamic Inversion Flight Control: Stability and Robustness Analysis and Improvements," Master's Thesis, Delft University of Technology, Delft, Netherlands, September 2016. URL https://repository.tudelft.nl/file/File_72437d3f-ba51-4718-be28-a368de7d02f3?preview=1.

[12] Steinert, A., Raab, S., Hafner, S., Holzapfel, F., and Hong, H., "From fundamentals to applications of incremental nonlinear dynamic inversion: A survey on INDI – Part I," *Chin. J. Aeronaut.*, 2025, p. 103553.

[13] Steinert, A., Raab, S., Hafner, S., Holzapfel, F., and Hong, H., "Advancements in incremental nonlinear dynamic inversion and its components: A survey on INDI – Part II," *Chin. J. Aeronaut.*, 2025, p. 103591.

[14] Harris, J. J., "F-35 flight control law design, development and verification," *2018 Aviation Technology, Integration, and Operations Conference*, American Institute of Aeronautics and Astronautics, Reston, Virginia, 2018.

[15] Kim, C.-S., Ji, C.-H., and Kim, B. S., "Development of flight control law for improvement of uncommanded lateral motion of the fighter aircraft," *Int. J. Aeronaut. Space Sci.*, Vol. 21, No. 4, 2020, pp. 1059–1077.

[16] Sieberling, S., Chu, Q. P., and Mulder, J. A., "Robust Flight Control Using Incremental Nonlinear Dynamic Inversion and Angular Acceleration Prediction," Vol. 33, No. 6, 2010, pp. 1732–1742. doi: 10.2514/1.49978, URL https://arc.aiaa.org/doi/10.2514/1.49978.

[17] Miller, C., "Nonlinear Dynamic Inversion Baseline Control Law: Architecture and Performance Predictions," *AIAA Guidance, Navigation, and Control Conference*, American Institute of Aeronautics and Astronautics, Portland, Oregon, 2011. doi: 10.2514/6.2011-6467, URL https://arc.aiaa.org/doi/10.2514/6.2011-6467.

[18] Looye, G. H. N., "An Integrated Approach to Aircraft Modelling and Flight Control Law Design," Ph.D. Thesis, Delft University of Technology, Delft, NL, 2008. URL https://repository.tudelft.nl, iSBN: 978-90-9022646-0.

[19] Looye, G., Willemsen, D., Bauschar, J., and Moennich, W., "Flight testing robust autoland control laws," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, 2001. doi: 10.2514/6.2001-4208.

[20] Muske, K. R., and Edgar, T. F., "Nonlinear State Estimation," *Nonlinear Process Control*, edited by M. A. Henson and D. E. Seborg, Prentice Hall PTR, 1997, Chap. 6, pp. 311–370.

[21] Ribeiro, M. I., *Kalman and Extended Kalman Filters: Concept, Derivation and Properties*, Institute for Systems and Robotics, Lisboa, PORTUGAL, 2004. URL https://www.researchgate.net/publication/2888846_Kalman_and_Extended_Kalman_Filters_Concept_Derivation_and_Properties.

[22] Tandeo, P., Ailliot, P., Bocquet, M., Carrassi, A., Miyoshi, T., Pulido, M., and Zhen, Y., "A Review of Innovation-Based Methods to Jointly Estimate Model and Observation Error Covariance Matrices in Ensemble Data Assimilation," *Monthly Weather Review*, Vol. 148, No. 10, 2020, p. 3973–3994. doi: 10.1175/MWR-D-19-0240.1.

[23] "Evaluation of Adaptive Extended Kalman Filter Algorithms for State Estimation in Presence of Model-Plant Mismatch," Vol. 46, 2013, p. 184–189. doi: 10.3182/20131218-3-IN-2045.00175.

[24] Reid, I., and Term, H., "Estimation II," , 2014. URL https://www.robots.ox.ac.uk, accessed: 2026-01-26.

[25] Akhlaghi, S., Zhou, N., and Huang, Z., "Adaptive adjustment of noise covariance in Kalman filter for dynamic state estimation," *2017 IEEE Power Energy Society General Meeting*, IEEE, Chicago, IL, 2017, p. 1–5. doi: 10.1109/PESGM.2017.8273755, URL http://ieeexplore.ieee.org/document/8273755/.

[26] Jiang, C., Zhang, S.-B., and Zhang, Q.-Z., "A New Adaptive H-Infinity Filtering Algorithm for the GPS/INS

Integrated Navigation," *Sensors*, Vol. 16, No. 12, 2016. doi: 10.3390/s16122127, URL https://www.mdpi.com/1424-8220/16/12/2127.

[27] "Least square estimation-based adaptive complimentary filter for attitude estimation," Vol. 41, 2019, p. 235–245. doi: 10.1177/0142331218755234.

[28] Cakiroglu, C., Van Kampen, E.-J., and Chu, Q. P., "Robust Incremental Nonlinear Dynamic Inversion Control Using Angular Accelerometer Feedback," *2018 AIAA Guidance, Navigation, and Control Conference*, American Institute of Aeronautics and Astronautics, Kissimmee, Florida, 2018. doi: 10.2514/6.2018-1128, URL https://arc.aiaa.org/doi/10.2514/6.2018-1128.

[29] Jeong, H., Jeong, J., Suk, J., and Kim, S., *Angular Acceleration Estimation with Off-CG Accelerometers for Incremental Nonlinear Dynamic Inversion Control*, 2024. doi: 10.2514/6.2024-2566, URL https://arc.aiaa.org/doi/abs/10.2514/6.2024-2566.

Part II

Literature Study & Research Proposal

3

Literature Review & Research Proposal

This chapter begins with an overview of classical and modern linear flight control methods in Section 3.1, which motivates the limitations of linear control and gain scheduling. This naturally leads to Section 3.2, where the main nonlinear flight control approaches are introduced and compared. Building on this background, Section 3.3 presents Nonlinear Dynamic Inversion (NDI), followed by Section 3.4, which introduces Incremental Nonlinear Dynamic Inversion (INDI) as a more robust alternative with reduced model dependence. The review then narrows to Hybrid INDI in Section 3.5, where the main blending strategies and their advantages and limitations are discussed, and to Kalman filtering in Section 3.6, which provides the foundation for the proposed approach. Based on these elements, the research proposal and research plan is presented in Section 3.7 and Section 3.10, where the motivation and scope of the study are formalized into a research goal, a main research question with sub-questions, and a research plan.

3.1. Classical and Modern Linear Flight Control

Early flight control relied on classical feedback loops and frequency-domain analysis. Fundamental work in the early 1900s established the PID controller and stability criteria that allowed engineers to design reliable autopilots and stability augmentation systems (SAS). Fundamental was the work of Nyquist, who developed a graphical frequency-response criterion to determine closed-loop stability of linear feedback systems [1]. Bode's classic book *Network Analysis and Feedback Amplifier Design* introduced Bode plots and gain/phase margin concepts for feedback design [2]. Bode's methods enabled engineers to shape open-loop frequency response for desired closed-loop behavior, which, together with the technological advancements of the mid-century, drove the analog flight control system design. The American Bureau of Aeronautics (BuAer) issued various reports in the first half of the 20th century, which became the standard reference for automatic flight control design, with emphasis on aircraft equations of motion, aerodynamic effects on modes of motion, and feedback control. These text later evolved into the extensive book by McRuer et al., providing design techniques for stability augmentation, autopilots, and handling-qualities analysis [3]. The text bridges the gap between theory and practical design for SISO control , using what are nowadays considered *classical* techniques such as PID, lead-lag compensators and gain scheduling.

PID control is effective for SISO systems, but it is complicated in the case of MIMO systems and does not provide an optimal control solution, which had become a central problem for controller design, together with the state-estimation problem, which involves finding the best estimate of a system's state using noisy measurements and a model of the system's behavior. Both these problems were addressed in the early 60s by the work of R. E. Kalman, with incredible contributions to the field of optimal control, by the development of Linear-Quadratic-Regulator (LQR) [4] and the Kalman Filter, leading the way to what would be later defined as *modern* control. This denomination encompasses also later contributions on Adaptive and Robust control and other less conventional and more recent approaches.

Even if the aforementioned techniques are able to obtain an optimal control solution for multiple states and uncertain measurements, they still rely on linear description of the system, which is usually obtained by linearization of the nonlinear plant around an operating point. Gain scheduling is a widely used approach to handle significant changes across a system's operating envelope, implemented by adjusting the controller parameters online according to one or more measurable scheduling variables. In practice, a family of controllers is designed for representative operating points, and a scheduling law, often simply an interpolation, is used to blend the appropriate controller gains during operation. While this offers a simple solution to altering dynamics, it relies on specific conditions to ensure global stability, robustness, and performance characteristics. These conditions are that the scheduling variable should be slow-varying and represent as much as possible the nonlinearities of the system [5, 6]. Fast changing variables,

scheduling on the wrong variables and strong nonlinearities due to driving the system away from its equilibrium, leads to complex transients and overall degraded performance [7].

3.2. Nonlinear Flight Control

The limitations of gain scheduling, which ultimately stem from relying on local linearization and a collection of locally valid linear controllers, have motivated researchers to develop control design approaches that can act on the nonlinear dynamics of the system directly. Jacobian linearization provides a linear model about an equilibrium point, but this approximation is only locally valid at the operating point, as already mentioned, and becomes increasingly inaccurate as the system moves across a wide operating envelope. Nonlinear control methods therefore explicitly exploit the nonlinear system structure to shape the closed-loop dynamics, providing a single control structure that remains effective over a wider operating region. In the next subsections three of the most effective nonlinear control methods employed for flight control are going to be discussed more in detail: Feedback Linearization, Backstepping and Sliding Mode Control.

3.2.1. Feedback Linearization Control

Feedback Linearization Control (FLC) is a type of nonlinear control method that, unlike linear techniques, can generate a linear model that precisely represents the original nonlinear model across a broad spectrum of operating conditions. FLC provides an exact, algebraic cancellation of the nonlinearities for systems that satisfy certain geometric conditions. [8]. Consider a general nonlinear SISO system of the form:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})u, \qquad y = h(x) \tag{3.1}$$

with x the state, u the input and y the output, $\boldsymbol{f}(\boldsymbol{x})$ and $\boldsymbol{g}(\boldsymbol{x})$ smooth vector fields, and $h(x)$ a smooth scalar function. The general procedure of FL involves different steps:

(a) differentiating the output y until a direct relationship between u and $y^{(i)}$ is found, as in Equation 3.2. The first integer r that makes the term $L_g L_f^{i-1} h(x)$ nonzero is by definition the *relative degree* of the system. Here the notation $L_f^i h(x)$ refers to the i^{th} Lie derivative of the scalar function $h(x)$ along the vector field f.

$$y^{(r)} = L_f^r h(x) \; + \; L_g L_f^{r-1} h(x)\, u \tag{3.2}$$

(b) Then a feedback law can be derived which will cancel the nonlinearities in the system :

$$u = \frac{1}{L_g L_f^{r-1} h(x)} \Big(\nu - L_f^r h(x) \Big), \tag{3.3}$$

(c) at this point the system can be written in the so-called normal form by applying a local change of coordinates, also called a local *diffeomorphism* $z = \Phi(x)$, to transform the system into a normal form that exposes a linear chain of integrators and, when $r < n$, a separate internal dynamics.

(d) Lastly, standard pole-placement techniques can be used to design the pseudo-control ν in order to obtain stable closed-loop dynamics and guarantee the desired transient performance.

The Feedback Linearization method can be constructed with two nested feedback loops, where the inner loop accomplishes the linearization and the outer loop ensures the stabilization of the closed-loop dynamics, as represented in Figure 3.1.

The relative degree r of a nonlinear system is analogous to the difference between poles and zeros for a linear system, and it is an indication of the 'hidden dynamics' of the system. The system possesses internal (zero) dynamics of order $n - r$ and their stability must be checked. The validity of FLC typically extends over a larger region of the state-space than Jacobian linearization but remains local rather than global. Feedback Linearization also has a number of important limitations [9]:

- It cannot be used for all nonlinear systems. Systems must have a well-defined relative degree and stable internal dynamics. For instance, non-minimum phase systems, such as flexible-joint robots or certain aircraft configurations cannot be fully feedback linearized due to their unstable zero dynamics.

- The full state needs to be measured. This is because the feedback law often depends explicitly on state derivatives up to the relative degree. Without full information about the states, the controller cannot correctly cancel the nonlinearities or apply the coordinate transformation needed for linearization.
- Robustness is not guaranteed in the presence of parameter uncertainty or unmodeled dynamics.

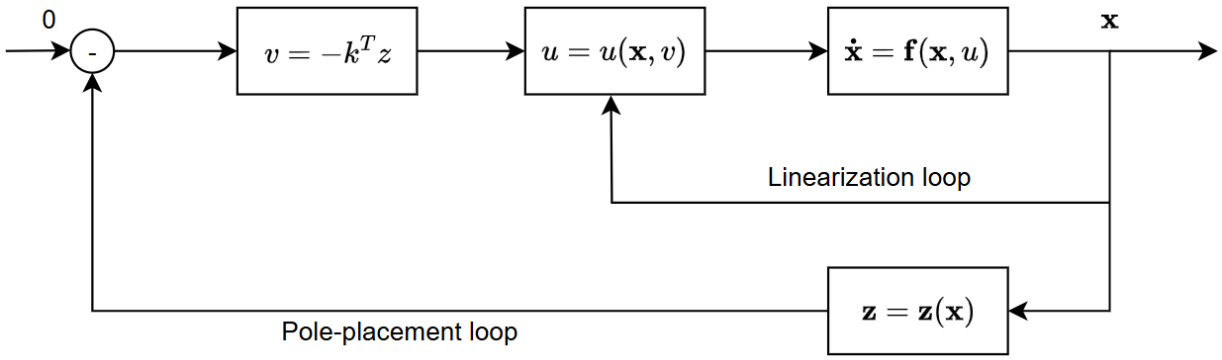


Figure 3.1: Feedback Linearization block diagram

3.2.2. Backstepping Control

Backstepping Control (BSC) is a recursive, Lyapunov-based design methodology tailored for a class of nonlinear systems known as *strict-feedback systems*. In such systems, the control input affects the dynamics of the outermost subsystem, and the state variables are hierarchically coupled. The general form of a strict-feedback system is given by [10]:

$$\begin{aligned} \dot{x}_1 &= f_1(x_1) + g_1(x_1)x_2 \\ \dot{x}_2 &= f_2(x_1, x_2) + g_2(x_1, x_2)x_3 \\ &\vdots \\ \dot{x}_{n-1} &= f_{n-1}(x_1, \ldots, x_{n-1}) + g_{n-1}(x_1, \ldots, x_{n-1})x_n \\ \dot{x}_n &= f_n(x_1, \ldots, x_n) + g_n(x_1, \ldots, x_n)u \end{aligned} \tag{3.4}$$

The backstepping procedure constructs the control law step-by-step, beginning with the innermost part of the system and working backwards toward the actual control input u. At each stage, a *virtual control input* is introduced for a simplified subsystem, and a stabilizing function is designed for that subsystem using a control-Lyapunov function.

This recursive process ensures that the entire system is stabilized through the construction of a global Lyapunov function. One of the primary advantages of Backstepping over Feedback Linearization is its flexibility, as it does not require the cancellation of all nonlinear terms. This allows beneficial nonlinear terms to remain in the closed-loop dynamics, potentially improving stability or transient performance.

The main disadvantage of Backstepping lies in the repeated differentiation of virtual control inputs, which leads to an ”explosion of terms” [11]. This significantly increases the complexity of designing a controller, particularly for higher-order systems. Another limitation is that the system must be expressed in strict-feedback form to apply the recursive Backstepping required. However, robust and adaptive variants of BSC have been introduced, which effectively improve the closed-loop tracking performance of nonlinear systems with uncertainties [12].

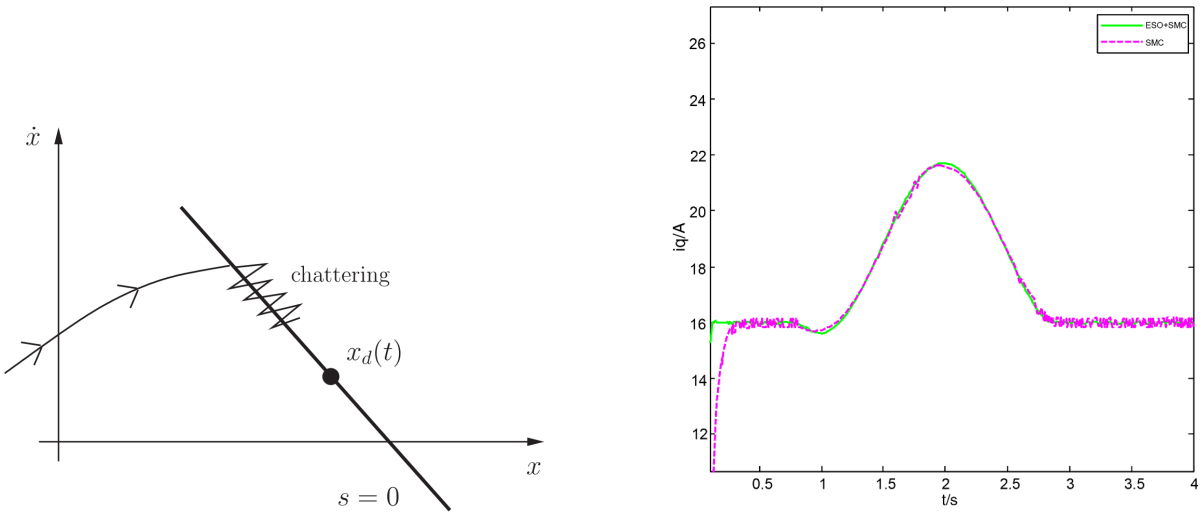
3.2.3. Sliding Mode Control

Sliding Mode Control (SMC) is a nonlinear control technique that drives the system state to a predefined sliding surface and constraints to remain there, therefore achieving the desired system's behavior. For a nonlinear system of n-states a sliding surface of lower dimension can be designed as a linear combination of the system's state variables. The controller is designed to drive the state trajectory towards the sliding surface and, once reached, it enforces motion along the surface, effectively reducing the order of the motion. This is achieved through the use of a sign or saturation function, which switch the control input at high frequency.

Sliding mode control is a nonlinear control law that is more robust in the presence of system uncertainties and external

disturbances. The gain of the controller can be adjusted to ensure the system stays along the desired sliding surface. The approach is also relatively straightforward to implement.

Nevertheless, there exist some drawbacks. The most important one is 'chattering', which refers to the rapid jittering of the system state between two sides of the sliding surface, as can be seen in Figure 3.2a, which shows rapid changes along the sliding surface as the system approaches this surface. Chattering occurs due to the finite time between switching from one side to another, and it can be exacerbated by internal delays in the system. There exists a trade-off between chattering and robustness, and mitigations to reduce its effect can be employed. For instance, Tang et al introduced of a non-linear extended state observer (ESO), which effectively reduces the chattering problem in control for wind turbines, by observing and compensating the total disturbance of the system in real time [13]. However, chattering remains a major practical issue in the application of SMC.



(a) Chattering behaviour near a sliding surface [14]

(b) Original vs ESO-mitigated chattering for an SMC controller [13]

Figure 3.2: Chattering phenomena in Sliding Mode Control.

3.2.4. Comparison of the approaches

The aforementioned nonlinear control approaches have been tested in the different practical applications. A number of studies has individuated how SMC is preferred for its superior robustness against uncertainties and external disturbances. It also tolerates uncertainties in the model and performs well under noisy conditions, making it applicable across diverse fields like robotics and UAVs. However, chattering can degrade performance, increase actuator usage producing more wear and energy consumption [11]. In contrast, FLC provides flexible design and, under ideal conditions without noise or uncertainty, can achieve satisfactory results with efficient, chatter-free inputs [15]. Nevertheless, FLC's main drawbacks are its high sensitivity to sensor noise and model uncertainty, the requirement for an exact mathematical model of the nonlinear system, and the involvement of high-order derivative terms in its design, which can lead to complex computations and instability in the presence of disturbances. Overall, SMC demonstrates superior robustness and lower tracking error compared to FLC, despite SMC's known chattering effect [16].

Regarding Backstepping, the method offers a particularly strong stability framework because of how the controller is constructed recursively using Lyapunov functions [17]. Compared to FLC, Backstepping is generally less reliant on exact cancellation of all nonlinear terms, and this often gives the designer more freedom to retain nonlinear dynamics that may be beneficial. Compared to SMC, Backstepping avoids the discontinuous switching action that causes chattering, which can be advantageous in applications where actuator smoothness is important. On the other hand, application of Backstepping comes with a higher analytical and implementation burden for more complex systems. This practical limitation has motivated variants such as Dynamic Surface Control specifically to reduce implementation complexity and the explosion of terms arising due to multiple differentiation [18]. In addition, its application requires the plant to be written in strict-feedback form, which may limit the applicability without first reformulating the system [12].

3.3. Nonlinear Dynamic Inversion (NDI)

Nonlinear Dynamic Inversion (NDI) is the name given in the aerospace community to a practical application of the Feedback Linearization (FL) technique. The two methods are theoretically equivalent, with FL representing a general mathematical framework, and NDI being its implementation for controller design of aircraft and spacecraft with nonlinear dynamics.

The general formulation of NDI starts by considering a generic, nonlinear MIMO system affine in the control input:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + G(\boldsymbol{x})\,\boldsymbol{u}, \qquad \boldsymbol{y} = h(\boldsymbol{x}) \tag{3.5}$$

where $\boldsymbol{f}(\boldsymbol{x})$ represents the nonlinear system dynamics, and $G(\boldsymbol{x})$ the control-effectiveness matrix. It is important to note how for the MIMO case, compared to the SISO case of Equation 3.1, the input u and output $\boldsymbol{y}$ have become m-dimensional vectors rather than scalars. $G(\boldsymbol{x})$ is an nxm matrix, with m the number of inputs, n the number of state equations. In order for NDI to work directly on the MIMO system, we need to have as many inputs as there are outputs.

Each output $h_i(\boldsymbol{x})$ is associated with an *individual* relative degree r_i. If the total relative degree $\sum_{i=1}^{m} r_i$ is strictly less than the number of states n, then the system possesses internal dynamics of order $n - \sum r_i$. These internal dynamics must be stable to ensure overall system stability.

As with Feedback Linearization, NDI requires the computation of Lie derivatives of the outputs with respect to the system dynamics. The pseudo-control vector $\boldsymbol{\nu}$ of linear coordinates is given by Equation 3.6 and the nonlinear matrices of Lie derivatives $\boldsymbol{b}(\boldsymbol{x})$ and $A(\boldsymbol{x})$ are given by Equation 3.7

$$\dot{\boldsymbol{y}} = \boldsymbol{\nu}(\boldsymbol{x}) = \boldsymbol{b}(\boldsymbol{x}) + A(\boldsymbol{x})\,\boldsymbol{u} \tag{3.6}$$

$$\begin{aligned}
\boldsymbol{\nu}(\boldsymbol{x}) &= \begin{bmatrix} \phi_{r_1}^{j^1} \\ \phi_{r_2}^{j^2} \\ \vdots \\ \phi_{r_m}^{j^m} \end{bmatrix}, \quad \boldsymbol{b}(\boldsymbol{x}) = \begin{bmatrix} L_{\boldsymbol{f}}^{r_1} h_1(\boldsymbol{x}) \\ L_{\boldsymbol{f}}^{r_2} h_2(\boldsymbol{x}) \\ \vdots \\ L_{\boldsymbol{f}}^{r_m} h_m(\boldsymbol{x}) \end{bmatrix} \\
A(\boldsymbol{x}) &= \begin{bmatrix} L_{g_1} L_{\boldsymbol{f}}^{r_1-1} h_1(\boldsymbol{x}) & \cdots & L_{g_m} L_{\boldsymbol{f}}^{r_1-1} h_1(\boldsymbol{x}) \\ L_{g_1} L_{\boldsymbol{f}}^{r_2-1} h_2(\boldsymbol{x}) & \cdots & L_{g_m} L_{\boldsymbol{f}}^{r_2-1} h_2(\boldsymbol{x}) \\ \vdots & \ddots & \vdots \\ L_{g_1} L_{\boldsymbol{f}}^{r_m-1} h_m(\boldsymbol{x}) & \cdots & L_{g_m} L_{\boldsymbol{f}}^{r_m-1} h_m(\boldsymbol{x}) \end{bmatrix}
\end{aligned} \tag{3.7}$$

The inversion control law that cancels the nonlinearities and produces linear input-output dynamics is:

$$\boldsymbol{u} = A^{-1}(\boldsymbol{x})\,[\boldsymbol{\nu}(\boldsymbol{x}) - \boldsymbol{b}(\boldsymbol{x})] \tag{3.8}$$

Here, $A(\boldsymbol{x})$ is referred to as the decoupling matrix and must be invertible for the control law to be defined. If the number of inputs does not match the number of outputs, a generalized (pseudo) inverse must be used instead.

The inversion process described above is not guaranteed to succeed for arbitrarily nonlinear processes, but it is in the case of controlling an aircraft's rotational dynamics. For the rotational motion of an aircraft, $\boldsymbol{f}(\boldsymbol{x})$ represents the nonlinear rotational dynamics, and the inputs u correspond to actuator deflections (ailerons, elevators, rudder), with their effects on the states captured in $G(\boldsymbol{x})$.

In both rate and attitude control applications, the outputs are chosen directly from the state vector. For rate control, the control variables are the body rates $[p, q, r]$, and for attitude control they are the Euler angles $[\phi, \theta, \psi]$. In both cases, the control effectors generate torques around the roll, pitch, and yaw axes. As a result, a direct matrix relation can obtained, as shown in Equation 3.9, where $A(\boldsymbol{x})$ has been replaced by the control-effectiveness matrix and $\boldsymbol{b}(\boldsymbol{x})$ by the nonlinear plant dynamics. For both cases, the total relative degree equals the number of states ($\sum r_i = n$), meaning that the system has no internal (zero) dynamics.

$$\boldsymbol{u} = G^{-1}(\boldsymbol{x})\,[\boldsymbol{\nu}(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{x})] \tag{3.9}$$

3.3.1. Application of NDI in Flight Control

NDI began to be employed in aircraft flight control in the early 1990s, and was found to be suited for aircraft with fast and unstable dynamics. One of the first published implementations was by Snell et al. (1992), who applied NDI to a super-maneuverable research aircraft. The authors differentiated between fast (angular rates) and slow (angle of attack, sideslip, bank angle) variables, performed inversion on the fast loop using control effectors and thrust vectoring, then applied inversion on the slow loop using commands for the fast states as inputs. The NDI system outperformed a comparable gain-scheduled controller in providing accurate control of sideslip and lateral acceleration and in reducing control deflections [19].

Colgren et al. (1997) applied multivariable dynamic inversion to the F-117A stealth fighter, using a cascaded inversion design [20]. Full nonlinear simulations demonstrated that the NDI controller met performance and stability requirements across the flight envelope, achieving desired control objectives without the need for gain scheduling.

Smith developed a simplified NDI approach based on fundamental flight mechanics principles [21]. This method had the advantage of requiring minimal a priori knowledge of vehicle aerodynamics and demonstrated excellent robustness to uncertainties in the mathematical model.

Adams and Banda developed an integrated approach by combining NDI with μ-synthesis, utilizing an inner/outer loop control structure where NDI in the inner loop equalized plant dynamics across the flight envelope, eliminating the need for tedious gain scheduling in the outer loop's $\mu-$synthesis compensator [22]. Concurrently, Bacon and Ostroff (2000) presented a modified NDI specifically for re-configurable flight control systems, which was proven to operate effectively under deteriorated conditions such as actuator failures, and missing or damaged control surfaces [23].

As with Feedback Linearization, the success of NDI relies heavily on the accuracy of the system model, in particular the control effectiveness matrix. Mismatches between the real aircraft and the onboard model can lead to poor tracking or instabilities. Furthermore, the basic NDI law is sensitive to external disturbances, and must be combined with other control methodologies or augmented with adaptive elements to improve its robustness.

3.4. Incremental Nonlinear Dynamic Inversion (INDI)

The NDI approach described in the previous section has been proved to be an attractive alternative to the practice of gain scheduling. However, it requires accurate knowledge of the system dynamics, particularly the nonlinear function $\boldsymbol{f}(\boldsymbol{x})$. This dependence makes NDI sensitive to modeling errors and parameter uncertainties, which is especially problematic in application to novel or experimental aircraft configurations, because knowledge of the plant dynamics may be incomplete or inaccurate.

To mitigate this dependency, the Incremental Nonlinear Dynamic Inversion (INDI) approach has been developed. The central idea of INDI is to exploit the fact that although the nonlinear dynamics may be unknown, instantaneous changes in the system's behavior, the *increments*, can be obtained using measurable signals. INDI thus performs inversion not on the full nonlinear model, but on the local incremental behavior around the current operating point. The nonlinearities are canceled based on measured changes in state and input, rather than requiring an explicit model of $\boldsymbol{f}(\boldsymbol{x})$.

The INDI inversion law mostly follows the same steps as for NDI, and it is here derived from a general input-affine nonlinear system, as in Equation 3.10. It should be noted, however, that INDI can also be derived from an input non-affine nonlinear system.

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + G(\boldsymbol{x})\, u, \quad x \in \mathbb{R}^n,\ u \in \mathbb{R}^m \tag{3.10}$$

By applying a first-order Taylor expansion around the current operating point $(\boldsymbol{x}_0, \boldsymbol{u}_0)$, the dynamics can be approximated locally:

$$\dot{\boldsymbol{x}} \;=\; \boldsymbol{f}(\boldsymbol{x}_0) + G(\boldsymbol{x}_0)\,\boldsymbol{u}_0 \;+\; \frac{\partial}{\partial \boldsymbol{x}}\left[\boldsymbol{f}(\boldsymbol{x}) + G(\boldsymbol{x})\,\boldsymbol{u}\right]_{(\boldsymbol{x}_0,\boldsymbol{u}_0)}(\boldsymbol{x}-\boldsymbol{x}_0) \;+\; \frac{\partial}{\partial \boldsymbol{u}}\left[\boldsymbol{f}(\boldsymbol{x}) + G(\boldsymbol{x})\,\boldsymbol{u}\right]_{(\boldsymbol{x}_0,\boldsymbol{u}_0)}(\boldsymbol{u}-\boldsymbol{u}_0) \tag{3.11}$$

where the first two terms represent the derivative of the current state:

$$\boldsymbol{f}(\boldsymbol{x}_0) + G(\boldsymbol{x}_0)\,\boldsymbol{u}_0 = \dot{\boldsymbol{x}}_0$$

and the second Jacobian term can be simplified into:

$$\frac{\partial}{\partial \boldsymbol{u}}\left[\boldsymbol{f} + G(\boldsymbol{x})\,\boldsymbol{u}_0\right](\boldsymbol{u}-\boldsymbol{u}_0) = G(\boldsymbol{x}_0)\ (\boldsymbol{u}-\boldsymbol{u}_0)$$

A key assumption in INDI is the time-scale separation principle, which assumes that, with fast enough actuators, the change in state between two successive time steps is negligible compared to the change in control input, thus the two can be treated separately. The state increments in Equation 3.11 can be neglected, as described by Equation 3.12.

$$\boldsymbol{x} - \boldsymbol{x}_0 \approx 0 \tag{3.12}$$

Using Equation 3.12 into Equation 3.11 leads to Equation 3.13:

$$\dot{\boldsymbol{x}} \;\approx\; \dot{\boldsymbol{x}}_0 + G(\boldsymbol{x_0})\;(\boldsymbol{u} - \boldsymbol{u}_0) \tag{3.13}$$

If the outputs are assumed to be the same as the state variables, as it is the case often for flight control applications, then $\dot{\boldsymbol{y}} = \dot{\boldsymbol{x}}$. As in the NDI derivation, a virtual control input can be defined as the input to the linearization loop $\boldsymbol{\nu} = \dot{\boldsymbol{x}}$, which can be designed by linear controllers to achieve the desired dynamics. The control input can be obtained by inverting Equation 3.13 and solving for u:

$$\boldsymbol{u} = \boldsymbol{u}_0 + G^{-1}(\boldsymbol{x}_0)(\boldsymbol{\nu} - \dot{\boldsymbol{x}}_0) \tag{3.14}$$

$$\boldsymbol{u} = \boldsymbol{u}_{0,s} + G^{-1}(\boldsymbol{x}_{0,s})(\boldsymbol{\nu} - \dot{\boldsymbol{x}}_{0,s}) \tag{3.15}$$

Equation 3.15 removes the dependency on the plant dynamics $\boldsymbol{f}(\boldsymbol{x})$ which was the main limitation of NDI, but introduces the terms u_0 and $\boldsymbol{x}_0$, the input and state at the current time step, respectively. The availability of these two terms poses a new challenge, as the current state derivative $\dot{\boldsymbol{x}}_0$ needs to be obtained from sensor measurements, which are degraded by delay, bias, noise and sampling errors in practical applications. In some instances, this quantity is not even directly available and must be estimated from other measurements. Similarly, accurate knowledge of u_0 assumes availability of the precise position of actuators, which may not always be reliable. Ensuring that these signals are synchronized is fundamental for practical implementations of the INDI approach.

Due to its reliance on measurements, this formulation is commonly referred to as *sensor-based INDI*. An alternative formulation known as *model-based INDI* also exists. In this variant, the current derivative of the state $\dot{\boldsymbol{x}}_0$ is not obtained through measurements, but instead approximated using an on-board model (OBM). In other words, $\dot{\boldsymbol{x}}_0$ is replaced with its model-based equivalent $\dot{\boldsymbol{y}}_{\text{OBM}}$, as shown in Equation 3.16. This equation is almost identical to the NDI inversion law of Equation 3.9, except for the presence of the current control input $\boldsymbol{u}_0$. The advantages of using model-based INDI rather than NDI are model-dependent.

$$\boldsymbol{u} = G(\boldsymbol{x}_0)^{-1} \cdot \Big[\boldsymbol{\nu} - \underbrace{\left(\boldsymbol{f}_{\text{OBM}}(\boldsymbol{x}_0) + G(\boldsymbol{x}_0) \cdot \boldsymbol{u}_0\right)}_{=\dot{\boldsymbol{y}}_{\text{OBM}}}\Big] + \boldsymbol{u}_0 \tag{3.16}$$

3.4.1. Application of INDI in Flight Control

In recent years, INDI gained significant traction within the flight control community due to its robustness against model uncertainties and reduced reliance on the system's plant. Steinert et al. (2025) provide a comprehensive overview of the method's development, its variants, and practical applications in aerospace systems [24, 25].

Early applications of INDI in flight control, particularly for advanced fighter aircraft, highlighted its potential for reconfigurability and robustness against model uncertainties.

The modified NDI method implemented by Bacon and Ostroff (2020) is notable for not requiring a model of the baseline vehicle but relying mainly on feedback of accelerations and effector positions. The approach was demonstrated on an advanced tailless aircraft with Innovative Control Effectors (ICE), inherently unstable in its lateral-directional axes at high speeds [23]. The reconfigured control system was shown to perform almost as well as the original system. A key insight from this work was that changes in the aerodynamic forces and moments could be directly measured by accelerometers, allowing the control system to regain control of the damaged vehicle.

Bordignon and Bessolo (2002) applied Dynamic Inversion control to the X-35B STOVL aircraft, addressing the challenging control allocation problem posed by strict low-speed performance demands and the combined use of aerodynamic and propulsive effectors [26]. The authors used a cascaded, generalized pseudo-inverse algorithm coupled with a high-fidelity onboard model to successfully manage effector failures scenarios, avoid structural coupling, and maximize control authority across the flight envelope.

Following these foundational efforts, INDI's application expanded to other fields of flight control. Dickman (2005) developed a fast and cost-effective method for prototyping control systems for Micro Air Vehicles (MAVs) [27]. The

flight control solution, based on a sensor-rich Incremental NDI law, was demonstrated on a 30mm wingspan vehicle. Grondman et al. (2018) presented the first successful demonstration of INDI on a full-size CS-25 certified aircraft, the PH-LAB experimental aircraft [28]. The flight tests showed that the INDI controller clearly outperformed classical NDI, achieving smaller overshoots in pitch tracking and compensating for asymmetric thrust during simulated engine failures.

One of the main challenges in INDI implementation is its sensitivity to measurement delays, particularly between the output derivative feedback (e.g., angular acceleration) and the actuator command path. Research has consistently shown that the largest stability regions and lowest tracking errors are achieved when the delays between these signals are minimized or made equal [25]. The conventional method is that a synchronization filter is applied to the measured actuator feedback, equivalent to the low-pass filter used for differentiating the pseudo-control signal, so to introduce the same delay. Steffensen et al. (2023) propose using cascaded complementary filters to obtain an un-delayed estimate of the output derivative, showing that this approach maintains closed-loop stability where conventional synchronization on actuator feedback fails in scenarios with strongly coupled input effectiveness. [29].

Building on these advancements, researchers have explored INDI's potential in fault-tolerant control architectures. Lu et al. (2016) proposed a trajectory controller that combines INDI, applied to flight path and angular rate loops, with standard NDI for position and attitude control [30]. The controller demonstrates accurate trajectory tracking in the presence of actuator faults, such as control surface jamming, by reallocating the control authority to unaffected actuators.

To further improve robustness and adaptability, recent research incorporated adaptive mechanisms into INDI. Smeur et al. (2016) developed an adaptive INDI scheme that estimates the control effectiveness of a MAV online, achieved using a Least Mean Square (LMS) algorithm to continuously update the $G(x_0)$ matrix [31]. Pineau et al. (2023) proposed an $\mathcal{L}_1$ adaptive augmentation scheme for an INDI autopilot on dual-spin guided projectiles under uncertainty in the control effectiveness parameters [32]. The $\mathcal{L}_1$ adaptive part comprises a state predictor, an adaptation law to estimate uncertainties, and a low-pass filter that balances performance with delay margins.

3.5. Hybrid Incremental Nonlinear Dynamic Inversion

Hybrid Incremental Nonlinear Dynamic Inversion (Hybrid INDI) was introduced primarily to address significant challenges present in both sensor and model-based INDI control systems.

Sensor-based INDI is highly dependent on sensor measurements. Factors like limited sampling frequencies, noise filtering, and other processing can cause measurements to not reflect accurately the true system state at the correct instant, leading to performance degradation and oscillatory responses. Furthermore, obtaining direct measurements of angular accelerations is often difficult, as aircraft typically use gyroscopes which measure angular rates. Differentiating rates to obtain angular accelerations magnifies the noise of the measured signal. Model-based INDI instead suffers from similar limitations as traditional NDI: in flight regimes characterized by strong nonlinearities, such as at high angles of attack, model inaccuracies become significant, and the performance of this approach has been shown to degrade [33].

Hybrid INDI specifically aims to alleviate these problems by integrating both system model information and sensor measurements. This is generally obtained by augmenting the outputs of the On-Board Model (OBM) with additional data from sensors, and combining them into a single state estimate used in the inversion to compute the required incremental control input. A general block diagram of Hybrid INDI from Harris (2018) is shown in Figure 3.3

By blending model estimates with sensor measurements, Hybrid INDI benefits from the fast response and high-frequency prediction of the OBM, which responds quickly to inputs and generally has lower noise levels for its estimates, while still being accurate at lower frequencies thanks to sensor measurements.

3.5.1. Theory of Hybrid INDI

Hybrid INDI maintains the control-affine structure of conventional INDI:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + G(\boldsymbol{x})\boldsymbol{u} \tag{3.17}$$

However, unlike standard INDI which obtains the state deriavtive $\dot{\boldsymbol{x}}$ using either sensor data or an onboard model (OBM), Hybrid INDI utilizes both to obtain an estimate $\dot{\boldsymbol{x}}$ via a fusion method:

$$\hat{\dot{\boldsymbol{x}}} = \mathcal{F}\left(\dot{\boldsymbol{x}}_{\mathrm{s}}, \dot{\boldsymbol{x}}_{\mathrm{OBM}}\right) \tag{3.18}$$

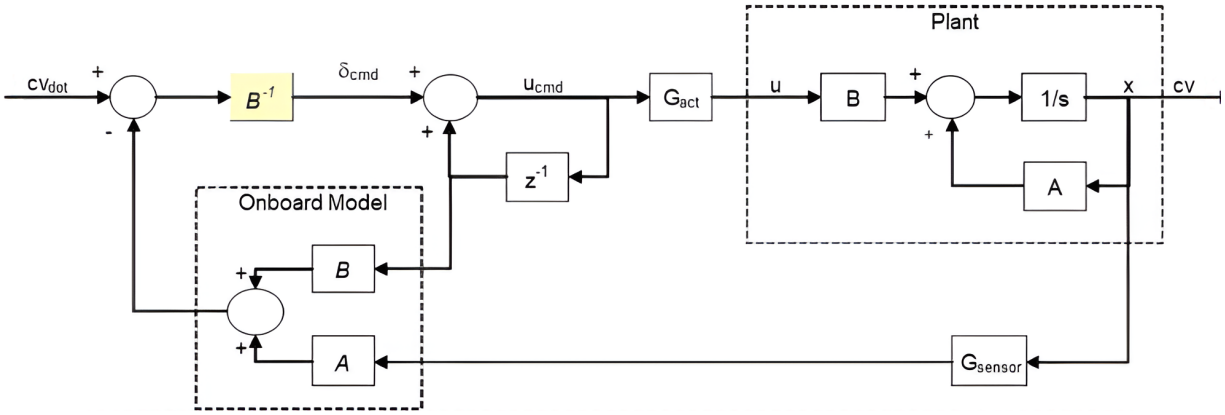


Figure 3.3: Hybrid INDI general structure [33]

where $\dot{x}_s$ is is the state derivative obtained from sensor measurements, $\dot{x}_{\text{OBM}}$ is the predicted state derivative from the on-board model and $\mathcal{F}(\cdot)$ is a fusion operator. This operator takes different forms depending on the chosen implementation. The different fusion approaches include Complementary Filters (CF) which blend signals based on their frequency characteristics, Extended State Observers (ESO) which offer superior noise attenuation, Augmentation Matrices or proportional blending techniques, and Predictive Filters that use past data to forecast future accelerations.

The control law then follows the familiar incremental structure, using the estimated state derivative:

$$\boldsymbol{u} = \boldsymbol{u}_0 + G(\boldsymbol{x}_0)^{-1}\left(\nu - \hat{\dot{\boldsymbol{x}}}_0\right) \tag{3.19}$$

3.5.2. Proportional Blend

One of the simplest approaches to the blending of sensors and model is by augmenting the model-based INDI with a proportional feedback of the error between the measured state derivative and the model-based estimate, as in Equation 3.20. A linear proportional blend is then obtained, where the weighting is determined by the augmentation matrix $\boldsymbol{K}_{aug}$, an n-dimensional diagonal matrix with elements typically between 0 and 1. While Kim et al. found a value of 0.8 for all diagonal elements of $\boldsymbol{K}_{aug}$ to be effective for a fighter aircraft [34], this blending can also be scheduled across the flight envelope to adapt to varying model uncertainties [24].

$$\hat{\dot{\boldsymbol{x}}} = \boldsymbol{K}_{aug} \cdot \dot{\boldsymbol{x}}_s + \left(1 - \boldsymbol{K}_{aug}\right) \cdot \dot{\boldsymbol{x}}_{\text{OBM}} \tag{3.20}$$

The primary advantages of this method include its simplicity in implementation and its ability to offer better robustness compared to sensor-based INDI, particularly when significant time delays are present in sensors and actuators. However, a significant disadvantage is that it can reduce the overall stability margin of the control system, with this reduction becoming more pronounced as time delays in the angular acceleration signal increase [34]. This necessitates careful synchronization design of the control surface feedback path to compensate for the potential stability loss [35].

3.5.3. Complementary Filter (CF)

The Complementary Filter (CF) approach fuses model outputs with sensor measurements by blending signals based on their frequency characteristics, as shown in Equation 3.21

Typically, a low-pass filter $S(s)$ is applied to the sensor measurements, which contain high-frequency noise, while a high-pass filter $T(s)$ is applied to the model outputs, which respond quickly to control inputs but has mismatches, which can be treated as low-frequency noise components. The complementary filter combines these two signals: it maintains the low-frequency information from the sensor, combining it with the model's accurate high-frequency prediction. By setting appropriate cut-off frequencies, these filters complement each other to produce an all-pass estimate of the state derivative. The filter indirectly performs differentiation of the measurements while mitigating noise [36].

$$\hat{\dot{\boldsymbol{x}}}(s) = T(s) \cdot \dot{\boldsymbol{x}}_{\text{OBM}}(s) + S(s) \cdot \dot{\boldsymbol{x}}_s(s) \tag{3.21}$$

where $T(s)$ and $S(s)$ are high-pass and low-pass transfer functions, respectively, such that $T(s) + \frac{1}{s}S(s) = 1$.

A standard second-order filter formulation from [36] is shown in Equation 3.22 where $K_P = 2\zeta\omega_n$ and $K_I = \omega_n^2$ set the cut-off frequency and damping of the filters.

$$T(s) = \frac{s^2}{s^2 + K_P s + K_I} \qquad S(s) = \frac{K_P s + K_I}{s^2 + K_P s + K_I} \tag{3.22}$$

A key advantage of CF is that it maintains good performance despite model mismatches and external disturbances, when compared to standard INDI, with the on-board model not requiring extreme precision. Kumptepe et al (2022) show that a CF can perform sufficiently well even without a dedicated synchronization filter due to the transient state derivative feedback provided by the on-board model, although synchronization can further improve performance [36].

A significant disadvantage lies in the trade-off between noise alleviation, response time, and model mismatch cancellation [37]. The cut-off frequency ω_n of the filter determines this balance. For instance, smaller ω_n attenuates more sensor noise, but the response to disturbances and discrepancies in the model becomes slower. Conversely, a higher value of ω_n introduces more information from the measurements but also amplifies its noise. This necessitates careful selection of the cutoff frequency value.

3.5.4. Extended-State Observer (ESO)

The Extended State Observer (ESO) is another method utilized within Hybrid INDI to estimate state derivatives, showcased by Lyu et al. (2023) [38]. ESO operates on the principle of treating unknown dynamics and external disturbances as an extended state which is estimated. The estimated state derivative is derived as a linear combination of the model output and the measured output, with specific observer gains, determined via pole placement.

The main advantage of ESO is its ability to achieve superior noise reduction and attenuation when compared to CF, without introducing additional computational complexity. This enhanced noise performance is attributed to a higher roll-off in its low-pass filter characteristics [24]. Additionally, ESO reduces the onboard model requirements without substantially compromising estimation accuracy [38].

Despite these benefits, a noted disadvantage is that ESO may exhibit worse tracking performance compared to Complementary Filters [37].

3.5.5. Other Blending Methods

Beyond the approaches already discussed, several other methods have been explored for state derivative estimation in Hybrid INDI. Sieberling et al. (2010) [39] used a linear Predictive Filter to forecast future angular accelerations, using present measurements of angular rates. This approach was referred in the study as Predictive INDI (PINDI), and was shown to improve the robustness of the controller under model and parametric uncertainties, compared to traditional NDI.

Ludeña Cervantes et al. (2020) tried to mitigate the effect of noise and delay on the performance of the aircraft responses, proposing a Kalman-based estimator [40]. The study employs the Fixed-Lag Smoothing technique, derived from the Kalman estimation equations, by fusing information from multiple sensors to achieve minimal estimation error variance. The method proved to effectively eliminate sensor noise and improve control command execution.

3.6. Kalman Filter

The Kalman Filter (KF) is one of the most widely used state estimation algorithms. Developed by Rudolf E. Kalman in the 1960s and notoriously employed for the Apollo program, it is nowadays the basis for many applications in autonomous vehicle control and navigation. The Kalman Filter is a recursive estimator which uses a model for future predictions and measurements for corrections, under the assumption that both are corrupted by white Gaussian noise. For a linear model, this filter is the optimal state estimator, since it minimizes the estimation error in the least-squares sense [41].

Over the years, several variations of the KF have been developed. The Extended Kalman Filter (EKF) extends KF theory to nonlinear processes by applying a first-order Taylor expansion at each time step. The Unscented Kalman Filter (UKF), instead of linearizing the model, uses a deterministic sampling technique to propagate the probability distribution through the nonlinear system. It is generally more accurate than the EKF for highly nonlinear systems, but at an increased computational cost [42]. Numerous adaptive variants of the KF have been developed that change

the values of the Gaussian noise statistics at each time-step [43, 44], typically by using innovation-based metrics to update the covariance matrices online in order to maintain consistency under time-varying noise levels and modeling uncertainties. Joint and Ensemble variants aim instead at estimating additional parameters together with the original states, such as unmodeled dynamics, slowly-varying disturbances, or sensor biases, by embedding these quantities into the estimation problem as augmented states or parameters [45, 46].

The estimate of a KF is a statistically-weighted blend of model predictions and sensor measurements. In this sense the KF can be seed as a time-varying variant of the Complementary Filter. Kalman Filters and Complementary filters are both widely used in sensor fusion applications for attitude determination. Because the Kalman Filter is stochastic and the Complementary Filter is deterministic, researchers frequently compare them in terms of accuracy and processing power. The two filters have been proved to be mathematically equivalent if the noise processes are stationary and the filter gains are constant [47]. Comparative studies for attitude estimation have highlighted how the KF is more resilient to bias and noise, while the CF is preferred for embedded system applications for its simplicity and small computational cost [48].

3.7. Research Proposal

In the previous sections, a survey was conducted on the scientific literature concerning nonlinear control techniques, with particular attention to their advantages and limitations in the presence of parametric uncertainties, measurement noise, and actuator/sensor faults. The study progressively narrowed its focus to Incremental Nonlinear Dynamic Inversion (INDI), and more specifically to its Hybrid variant. Hybrid INDI has been shown to preserve most of the strengths of the standard INDI approach, but showing greater adaptability by combining sensor feedback with On Board Model predictions, thereby compensating for the limitations of either source when used in isolation.

Upon further exploration of the Hybrid INDI literature, it becomes clear that active research continues on various aspects of this control strategy. Among the open questions are: how best to blend model predictions with sensor measurements, how to dynamically adapt the blending depending on flight conditions or degradation and how to handle faults in the plant or in the sensors.

In this context, a promising approach that has been recommended in some recent studies is the integration of Kalman Filter (KF) theory into the Hybrid INDI framework. This approach is motivated by the fact that a key challenge in Hybrid INDI lies in accurately estimating the aircraft's angular accelerations. A Kalman Filter can provide a statistically optimal estimate of this and other states by dynamically weighting model predictions and sensor measurements based on their uncertainty. This approach mirrors the principle of Complementary Filters, already used in some Hybrid INDI architectures [36, 37], but extends it into a time-varying alternative.

The above hypothesis sets the basis for the goal of the present research. It is further formalized into one main Research Question and four Sub-questions, which will be addressed over the course of the study.

3.8. Research Goal

From the above discussion, the goal of the research has been formalized into the following statement:

"The goal of this research is to develop Kalman Filter–based blending strategies for Hybrid INDI, in order to improve state estimation and control performance under nominal and degraded flight conditions"

3.9. Research Questions

The research goal can be described by the main research question below, which has been further divided into four sub-questions.

Main Research Question

In the context of flight control using Hybrid INDI, how can a Kalman Filter-based approach be used to provide an optimal blend of sensor measurements and on-board model predictions?

Sub-question 1

What are the possible architectures for integrating a Kalman Filter for state estimation within a Hybrid INDI framework?

Sub-question 2a

How does the Hybrid INDI with a Kalman Filter–based blend perform under nominal flight conditions?

Sub-question 2b

How does the Hybrid INDI with a Kalman Filter–based blend perform under degraded flight conditions, such as model-parameter uncertainties, sensor delay and noise?

Sub-question 3

How does the Hybrid INDI Kalman Filter–based blend compare to the Complementary Filter approach in terms of performance, under both nominal and degraded flight conditions?

3.10. Research Plan

The research has a total planned duration of 32 weeks and it is divided in 4 main phases. The *Literature Study* and *Research Plan*, is presented in Section 3.1 to Section 3.5. This phase will have a planned duration of 7 weeks. The successive phase is the *Implementation* phase, where the proposed Hybrid INDI with KF will be implemented in a MATLAB/Simulink simulation environment. This phase will have a planned duration of 12 weeks. After that, the *Analysis* phase will follow, with a duration of 9 weeks, which will be dedicated to extracting valuable insights into the performance of the developed architectures. Finally, the *Conclusion* phase will tell the outcome of the research and the results obtained. This final phase is estimated to last 4 weeks.

3.10.1. Work Packages Definition

To answer the research questions, a series of structured activities will be conducted. The different research phases have been divided into Work Packages (WP). Each WP is a collection of tasks related chronologically to each other, which shall be completed in a limited time. Below, a brief explanation of each WP is given, along with a breakdown of the tasks for each WP in Figure 3.4

WP-2.1 Model Verification: The nonlinear model of the Cessna Citation aircraft will be verified. This process could involve comparing the outputs of simulations with reference flight data or past simulations to ensure the fidelity of the model.

WP-2.2 Control Requirement Definition: The control goals will be defined,. These could be a combination of the following:

- Attitude tracking of Euler angles (Roll ϕ, Pitch θ, Yaw ψ) or angular rates (p, q, r).
- Flight Path control: altitude h, flight path angle γ, course angle χ.
- Robustness against uncertainties.
- Improvement of Handling Qualities.
- Fault tolerance and Failure Handling.

WP-2.3 Standard INDI Implementation: An inner-loop based on standard INDI theory shall be implemented to gain familiarity with the control strategy. This shall be coupled the design and tuning of the outer-loop linear controller.

WP-2.4 Hybrid INDI Baseline Implementation: An On-Board Model (OBM), simpler than then full non-linear model, shall be integrated into the control architecture, yielding a Hybrid INDI configuration which will likely be combined with a Complementary Filter (CF), and will serve as the *baseline* controller. The Hybrid INDI + CF controller will be tested to ensure it performs as intended under nominal conditions.

WP-2.5 KF Architecture Selection: A brainstorming session will identify all feasible KF architecture options. These may include configurations with one or more Kalman Filters and combinations with Complementary Filters or Extended State Observers (ESO). A subset will be selected based on theoretical performance and engineering judgment. The selected designs will be implemented in a MATLAB/Simulink environment. **Sub-question 1** shall be answered at this stage.

After the *Implementation* phase, the *Analysis* phase will follow. The core objective of the *Analysis* phase is to assess whether and how well each tested configuration fulfills the previously defined control requirements under various conditions. This phase shall include the evaluation of the performance of the selected architectures, comparison with the baseline controller and also the discussion of the results obtained from the simulations.

WP-3.1 Performance Evaluation under Nominal Conditions: The performance of the selected KF-based controllers will be evaluated relative to the baseline CF controller. **Sub-question 2a** shall also be answered at this point. The planned evaluation will involve both time-domain and frequency-domain performance metrics:

- **Time-domain metrics:** overshoot, settling time, and RMS of the tracking error.
- **Frequency-domain metrics:** Since the aircraft plant if non-linear, traditional frequency-domain methods cannot be applied. Limit cases for gain and phase margin can be used to evaluate how much delay and gain the system can tolerate. Alternatively, spectral analysis can be used for estimating the system's frequency response.

WP-3.2 Performance Evaluation under Degraded Conditions: Performance will be analyzed using the metrics of WP-3.1, now under degraded flight conditions, including:

- Introducing noise, bias, and delay into sensor measurements. Emphasis will be placed on identifying limit cases which will determine how much delay the system can tolerate before instability occurs.
- Inducing parametric uncertainties between the nonlinear plant and the OBM to simulate model mismatches.

WP-3.3 Monte Carlo Analysis: For robustness assessment, Monte Carlo simulations will be conducted with randomized combinations and severities of the degradation factors. These runs will be analyzed with statistical methods. **Sub-question 2b** and **Sub-question 3** shall be answered at this stage.

WP-3.4 EKF Behavior and Innovation Analysis: The innovation signals and state estimate convergence of each EKF configuration will be analyzed. The aim is to understand their influence on controller performance.

WP-4.1 Conclusion and Reflection: Draw the conclusions of the study, summarize which architecture performed best under which conditions, reflect on the broader implications of the results, recommend actions for future work on Hybrid INDI. Write about the work performed during the research in the thesis document and submit a scientific paper for possible publication. The **Main Research Question** shall be answered by the end of the study.

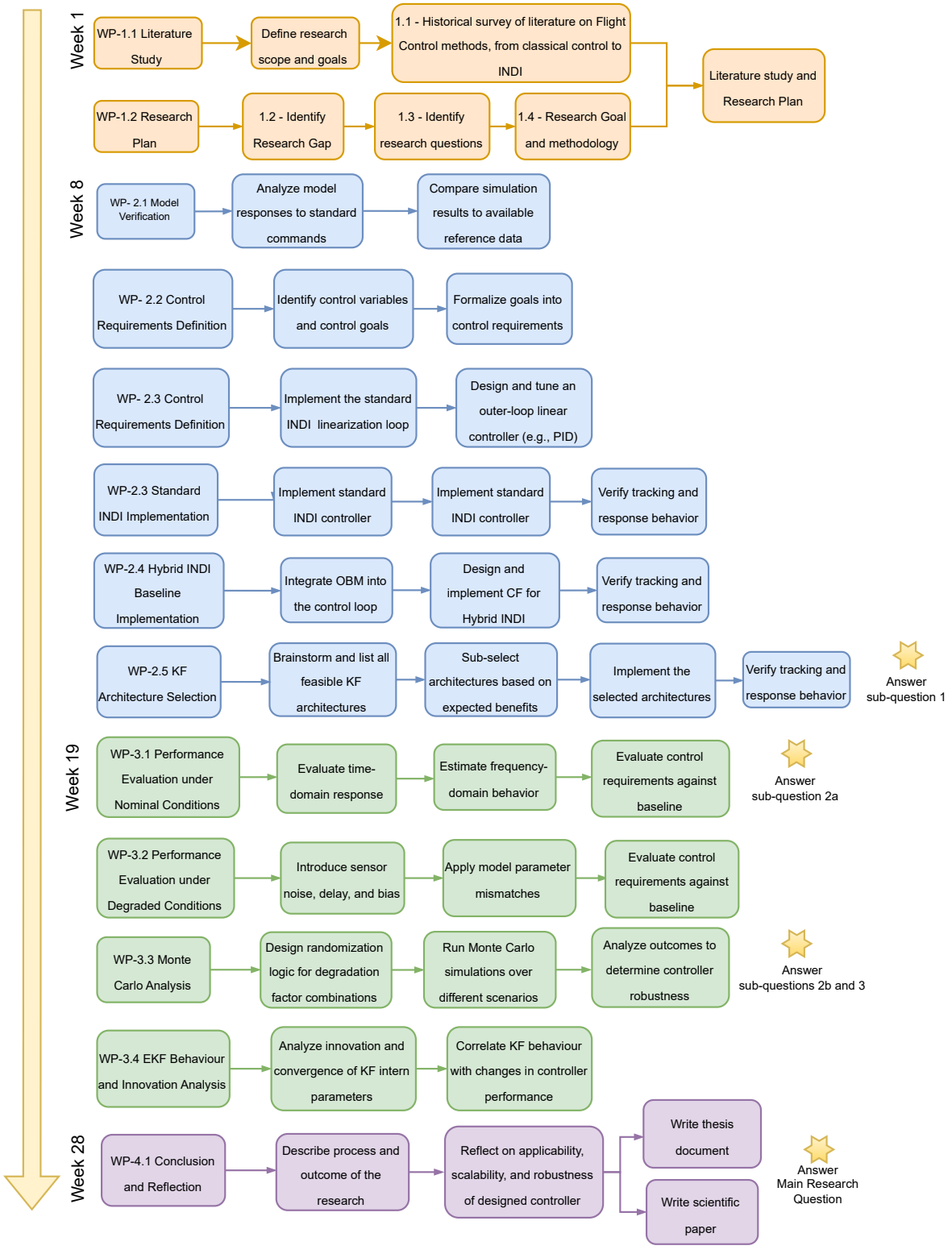


Figure 3.4: Research Plan Breakdown Structure

Part III

Additional Results

4

Additional Results

The following chapter presents additional analyses and results that complement the main findings of the scientific article and further support the discussion of estimation and filtering choices within the Hybrid INDI framework. Section 4.1 revisits the structure of the Complementary Filter (CF), motivating the two-stage filter architecture adopted in this work. Building on this, Section 4.2 investigates the selection of the CF crossover frequency ω_c and discusses how its optimal value varies with the relative degradation of the sensor and model paths. Section 4.3 then provides the Extended Kalman Filter (EKF) formulation and implementation details used in this research, including the underlying assumptions and the recursive estimation algorithm. Next, Section 4.4 explores adaptive online estimation of the EKF covariance matrices, with particular emphasis on innovation-based adaptation of the measurement-noise covariance. Section 4.5 presents a Joint EKF formulation for simultaneous state and parameter estimation and discusses the practical limitations encountered using this approach. Finally, Section 4.6 lists alternative EKF architectures for obtaining angular acceleration estimates more directly and highlights their potential for future research.

4.1. Complementary Filter Structure

Following the definition given by Mahony et al. [49] two filters, $F_1(s)$ and $F_2(s)$ can be termed complementary if

$$F_1(s) + F_2(s) = 1 \tag{4.1}$$

such that the low-frequency content is predominantly shaped by the low-pass filter $F_1(s)$ while the high-frequency content by the high-pass $F_2(s)$. Equation 4.1 ensures that the cut-off frequency of the low pass filter corresponds to the cut-off frequency of the high pass filter, and the two filters jointly represent the full frequency range without gain discontinuities in the ideal case. Complementary Filters can be of arbitrary order, with higher-order filters providing steeper roll-off, hence stronger attenuation of disturbances outside the pass band, but it also increases the number of tuning parameters and the risk of unintended peaking or phase lag if the parameters are not selected carefully.In practice, the observed performance differences between first- and second-order complementary filters can be modest for some applications, and the results often depend more strongly on parameter selection than on the order itself [50, 51]. Consequently, several works have tried adaptable approaches by solving optimization algorithm such as Least-Mean-Squares (LMS) [52] or Particle Swarm Optimization (PSO) [53] as well as fuzzy adaptive alternatives [51] or a cascaded filter structure which does not require any tuning [54]. Although adaptability has an edge over manual tuning for off-nominal cases, these techniques are highly dependent on prior knowledge of the range of gain parameters to be used.

For angular acceleration estimation, as in Hybrid INDI applications, a commonly used second-order CF structure for fusing a measured body rate ω_s with a model-based angular acceleration prediction $\dot{\omega}_m$ is

$$\dot{\omega}_{\text{est}} \;=\; \omega_s \, S_\omega(s) \;+\; \dot{\omega}_m \, T(s), \tag{4.2}$$

with

$$S_\omega(s) = \frac{K_P s^2 + K_I s}{s^2 + K_P s + K_I}, \qquad T(s) = \frac{s^2}{s^2 + K_P s + K_I}, \tag{4.3}$$

where the crossover frequency ω_c and damping ratio ζ are typically parameterized as $K_P = 2\zeta\omega_c$ and $K_I = \omega_c^2$. At first sight, $S_\omega(s) + T(s) \neq 1$; however, the complementary property becomes explicit when both contributions are expressed in the same signal domain (angular acceleration). Using $\dot{\omega}_s = s\,\omega_s$ yields

$$\omega_s \, S_\omega(s) = \dot{\omega}_s \left(\frac{S_\omega(s)}{s} \right), \qquad \text{with} \qquad F_1(s) \triangleq \frac{S_\omega(s)}{s} = \frac{K_P s + K_I}{s^2 + K_P s + K_I}. \tag{4.4}$$

Hence, (4.2) can be rewritten as

$$\dot{\omega}_{\text{est}} = \dot{\omega}_s\, F_1(s) + \dot{\omega}_m\, F_2(s), \qquad F_2(s) \triangleq T(s) = \frac{s^2}{s^2 + K_P s + K_I}, \tag{4.5}$$

and the complementary condition holds exactly:

$$F_1(s) + F_2(s) = \frac{K_P s + K_I}{s^2 + K_P s + K_I} + \frac{s^2}{s^2 + K_P s + K_I} = 1. \tag{4.6}$$

While (4.6) guarantees a mathematically consistent split in the *angular-acceleration* domain, the practical issue in (4.2) is that the sensor enters through ω_s (which is typically noise- and delay-corrupted), and the corresponding transfer from ω_s to $\dot{\omega}_{\text{est}}$ exhibits a constant high-frequency gain. Specifically,

$$\lim_{\omega\to\infty} S_\omega(j\omega) = \lim_{\omega\to\infty} \frac{K_P(j\omega)^2 + K_I(j\omega)}{(j\omega)^2 + K_P(j\omega) + K_I} = K_P = 2\zeta\omega_c. \tag{4.7}$$

Therefore, increasing ω_c not only the blending point of the sensor and model moves to higher frequencies, but also increases the asymptotic high-frequency gain proportionally to K_P. In other words, in this second-order structure the crossover tuning and the high-frequency leakage of sensor content are intrinsically coupled through $K_P = 2\zeta\omega_c$. This coupling can limit how large ω_c can be chosen in practice, especially when ω_s contains substantial high-frequency noise and/or delay.

To decouple the differentiation of angular rate from the sensor–model blending, this work adopts a two-stage structure. First, the measured rate ω_s is differentiated using a band-limited differentiator

$$\dot{\omega}_s = H_f(s)\, \omega_s, \tag{4.8}$$

with

$$H_f(s) = \frac{\omega_d^2\, s}{s^2 + 2\zeta_d\omega_d s + \omega_d^2}, \tag{4.9}$$

where ω_d and ζ_d are selected to achieve the desired trade-off between ideal derivative action at low frequency and roll-off at high frequency. The differentiator parameters were manually chosen for this application, $\omega_d = 25$ rad/s and $\zeta_d = 0.7$, and are consistent with previous INDI implementations [36].

In the second step the differentiated measurement $\dot{\omega}_s$ is then fused with the model predicted acceleration $\dot{\omega}_m$ using a first-order complementary filter:

$$\dot{\omega}_{\text{est}} = \dot{\omega}_s\, \frac{\omega_c}{s + \omega_c} + \dot{\omega}_m\, \frac{s}{s + \omega_c}. \tag{4.10}$$

The filters in (4.10) are complementary by construction, since

$$\frac{\omega_c}{s + \omega_c} + \frac{s}{s + \omega_c} = 1, \tag{4.11}$$

and the crossover is governed by a single parameter ω_c. Importantly, in the separated design of (4.9)–(4.10), the choice of ω_c does not change the differentiator dynamics set by ω_d and ζ_d. Increasing ω_c shifts the fusion toward trusting the measured acceleration over a wider band, but the high-frequency noise behavior remains strongly shaped by $H_f(s)$.

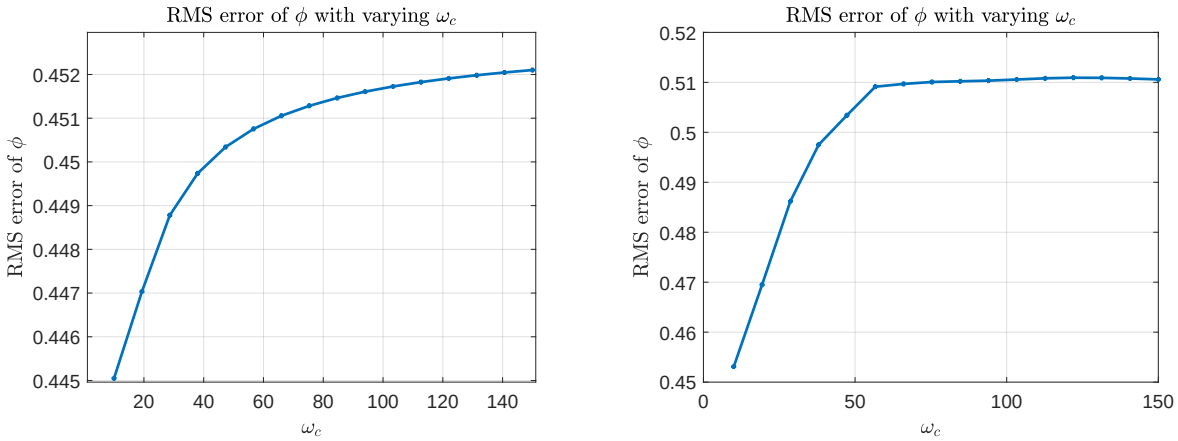
In the selected architecture, the sensor path is affected by high-frequency noise and delay, whereas the model path is primarily affected by low-frequency mismatch. In second-order structure of (4.2)–(4.7) the crossover frequency is tied to the high-frequency gain from ω_s to $\dot{\omega}_{\text{est}}$ via $K_P = 2\zeta\omega_c$, while the proposed two-filter design of (4.9) and (4.10) provides two independent tuning knobs: ω_d and ζ_d determine derivative fidelity and noise attenuation, while ω_c determines the sensor–model blending. This separation enables selecting the optimal ω_c which minimizes RMS estimation error without simultaneously increasing the high-frequency gain from the differentiated measurement. As a result, the crossover frequency identified in this work can be higher than values commonly reported for combined differentiator–CF structures, which indicates that a larger portion of the usable bandwidth is dominated by the sensor path for the considered implementation. The next sections explain the choice of ω_c to achieve the lowest tracking error for different situations.

4.2. Optimal Crossover Frequency

The crossover frequency of the complementary filter, ω_c determines the blending of the measurements and the model predictions in the frequency domain. For this application we are interested in the value of ω_c that provides the lowest tracking error. This value is not trivial to find since it changes based on the level of degradation in of the paths the filter is fusing. To illustrate this, the RMS tracking error can be plotted for different values of the crossover frequency to show how the ω_c corresponding to the minimum RMS value changes. In general, a higher value of ω_c translates into the filter output being dominated by sensor information while a lower value favors OBM predictions. Figure 4.1a shows RMS values for the *ideal* condition, when both sensor and model path are ideal, unperturbed and un-delayed signals. The RMS error value is basically constant, as both input paths share the same information. When delay and white noise are introduced, as in Figure 4.1b in the measurement path, having a lower ω_c value is beneficial, as OBM predictions are more reliable for INDI control. The opposite happens when the parameters of the OBM Figure 4.2a do not exactly match the aircraft plant, Figure 4.2a. Finally if degradation and uncertainties exist in both paths, the ω_c value that yields the lowest error is found at a midpoint, such as in Figure 4.2b where the optimal crossover frequency lies at about 80 rad/s.

The analysis above highlights how the optimal value for the CF crossover frequency changes based on the quality of the input signal, therefore by manually tuning the parameters is hard to create a robust filter. The tuning technique used in this work, described in detail in Section 2.4 aims at creating a robust filter by looking for the parameters that obtain the average lowest RMS error among a number of worst-case scenarios using different starting values and multiple iterations.

A further note concerns the relationship between the OBM and the measurements. The input to the OBM is the current measurement of the state, provided by the sensors, which then serve as the base for the angular acceleration prediction of the OBM. If these measurements are degraded by noise and delay, this will inevitably affect the OBM prediction as well. This puts the OBM prediction in a disadvantage position compared to the measured information. It has been observed that, with small degradation only in the sensor path, the measurements still provide a better estimate than the OBM. Then, for lager values of noise and delay, the OBM predictions are favored. To limit this from happening the OBM input signal should be the 'true' state at all times, free from any sensor degradation, which is impossible to achieve in real applications. An alternative is to pre-process the measured state before feeding it to the OBM for instance with a low-pass filter.



(a) Optimal ω_c for the ideal scenario

(b) Optimal ω_c with sensor noise and delay

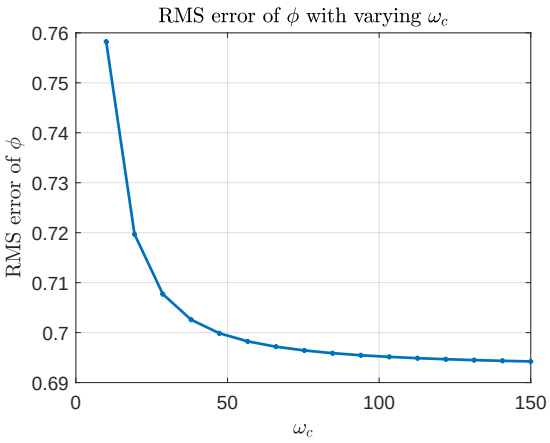
Figure 4.1: Optimal ω_c values for different scenarios

4.3. Extended Kalman Filter

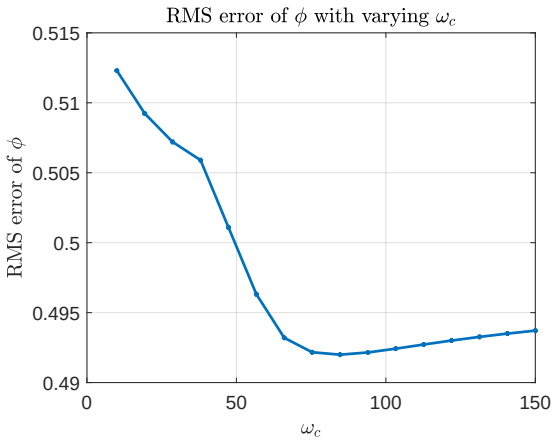
The Extended Kalman Filter (EKF) extends the classical Kalman Filter to nonlinear state estimation problems through local linearization. In discrete time, the EKF assumes a nonlinear process model and a nonlinear measurement model of the form:

$$x_k = f(x_{k-1}, u_{k-1}) + w_{k-1} \qquad z_k = h(x_k) + v_k \tag{4.12}$$

where x_k denotes the system state vector at time step k, u_{k-1} is the known input vector, and z_k is the measurement



(a) Optimal ω_c with model-plant mismatch



(b) Optimal ω_c with both degradation in the sensor and model path

Figure 4.2: Optimal ω_c for different scenarios

vector. The vectors w_{k-1} and v_k represent the process noise and measurement noise, respectively, and are assumed to be zero-mean, white, and Gaussian. These noise terms capture unmodeled dynamics, parametric uncertainties, sensor noise and delays affecting the model and measurement paths. The statistical properties of the noise processes are characterized by the covariance matrices:

$$Q = \mathbb{E}\left[w_{k-1}w_{k-1}^T\right], \qquad R = \mathbb{E}\left[v_k v_k^T\right],$$

which are specified a priori together with the initial state covariance matrix $P_{0|0}$. The EKF aims to minimize the mean squared error (MSE) of the state estimate, under these Gaussian noise assumptions. This minimization is realized via the Kalman gain K, which dynamically weights the model-based state prediction against the corrections provided by the sensor measurements. In this sense, the EKF can be interpreted as a time-varying extension of the Complementary Filter used in the baseline controller.

The assumptions for effective state estimation using the Extended Kalman Filter are the following:

- The system is Markovian: the state x_k depends only on x_{k-1} (and known inputs u_k), not on earlier states.
- Process noise w_k and measurement noise v_k are zero-mean, white (uncorrelated in time), and mutually uncorrelated, with known finite covariances Q and R.
- The initial state estimate $\hat{x}_0$ and its covariance P_0 are given, finite, and reasonably close to the true state.
- The nonlinear state and measurement models $x_k = f(x_{k-1}, u_{k-1}) + w_{k-1}$; $z_k = h(x_k) + v_k$ are sufficiently smooth; $f(\cdot)$ and $h(\cdot)$ are at least once differentiable in the region of interest.
- The Jacobian matrices $F_k = \frac{\partial f}{\partial x}\big|\hat{x}_{k-1}$ and $H_k = \frac{\partial h}{\partial x}\big|\hat{x}_{k|k-1}$ exist and are finite.
- The state (and measurement) distributions can be reasonably approximated as Gaussian, so that propagating only the mean and covariance is adequate (no strong multimodality or heavy tails).
- The chosen process and measurement noise models (Q, R) are consistent with the real system so that the filter does not become systematically overconfident or diverge.

4.3.1. EKF Algorithm

The generalized algorithm for the real-time implementation of the Extended Kalman Filter (EKF) [55] is described below and follows a discrete-time formulation consistent with the implementation in the Simulink environment. The EKF operates recursively by alternating between a time update (prediction) and a measurement update (correction) at each simulation time step.

Step 1: Initialization: At the start of the simulation, the state estimate and its uncertainty are initialized as

$$\hat{x}_{0|0} = x_0, \qquad P_{0|0} = P_0,$$

where x_0 denotes the first available measurement of the state and P_0 is initialized as an identity matrix of dimensions equal to the number of states. This initialization represents the best available prior knowledge of the system state and its associated uncertainty.

Step 2: State prediction: Given the measured input sample $u_{m,k}$ at time step k, the discrete-time nonlinear process model is used to propagate the state estimate forward in time,

$$\hat{x}_{k+1|k} = f_d\big(\hat{x}_{k|k}, u_{m,k}\big), \tag{4.13}$$

where $f_d(\cdot)$ denotes the discretized state transition function of the On Board Model (OBM) already described in Section 2.2.

Step 3: Linearization of the process model: Because the process model is nonlinear, it needs to be linearized about the predicted state to obtain the state-transition Jacobian:

$$F_k = \left.\frac{\partial f_d(x, u_{m,k})}{\partial x}\right|_{x=\hat{x}_{k|k}}. \tag{4.14}$$

Step 4: Covariance prediction: The predicted (a priori) covariance matrix is computed using the discrete-time covariance propagation

$$P_{k+1|k} = F_k\, P_{k|k}\, F_k^T + Q, \tag{4.15}$$

where Q is the discrete-time process noise covariance matrix, which by the assumptions listed before, directly affects the system states.

Step 5: Measurement linearization: At time $k+1$, the predicted state estimate is corrected using the measurement vector $z_{m,k+1}$. The nonlinear measurement model is linearized once around the predicted state,

$$\hat{z}_{k+1} = h\big(\hat{x}_{k+1|k}\big), \qquad H_{k+1} = \left.\frac{\partial h(x)}{\partial x}\right|_{x=\hat{x}_{k+1|k}}. \tag{4.16}$$

Step 6: Measurement update: Using the linearized measurement model, the innovation covariance and the Kalman gain are computed as in Equation 4.17 and Equation 4.18 respectively, where R denotes the measurement noise covariance matrix.

$$S_{k+1} = H_{k+1}\, P_{k+1|k}\, H_{k+1}^T + R, \tag{4.17}$$

$$K_{k+1} = P_{k+1|k}\, H_{k+1}^T\, S_{k+1}^{-1}, \tag{4.18}$$

The innovation, also called measurement residual, is defined as the difference between the current and the predicted (a priori) measurement:

$$\tilde{y}_{k+1} = z_{m,k+1} - h(x_{k+1})\, \hat{x}_{k+1|k}, \tag{4.19}$$

and the prediction of the state estimate is then updated with the innovation by applying a correction based on the Kalman gain. This yields the posterior state estimate:

$$\hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + K_{k+1}\, \tilde{y}_{k+1}. \tag{4.20}$$

Step 7: Covariance update: Finally, the posterior covariance matrix is updated using the Joseph stabilized form, which preserves symmetry and improves numerical robustness [56].

$$P_{k+1|k+1} = (I - K_{k+1}H_{k+1})\, P_{k+1|k}\, (I - K_{k+1}H_{k+1})^T + K_{k+1} R K_{k+1}^T, \tag{4.21}$$

After completion of the measurement update, the posterior state estimate $\hat{x}_{k+1|k+1}$ and covariance $P_{k+1|k+1}$ are stored and used as initial conditions for the next time step. This recursive procedure is repeated for the full duration of the simulation.

4.4. Adaptive Estimation of Q and R matrices

A fundamental challenge when applying the EKF is the appropriate initialization and tuning of the process- and measurement-noise covariance matrices, Q and R. This challenge becomes more relevant when the uncertainty characteristics are not constant, for instance due to additional noise, delay or model-parameter mismatches. This is the case for this research, where one objective is to assess EKF performance under such off-nominal conditions. To improve the estimation capabilities of the EKF under varying uncertainty conditions, it is therefore desirable to allow the noise covariance matrices to vary online as well.

A literature review was conducted on methods for adaptive covariance estimation. Most approaches rely on the innovation sequence, defined as differences between the current and the predicted measurements. A recent survey [57] categorizes adaptive methods into two main statistical families: *method-of-moments*, which enforces agreement between the theoretical and empirical moments of the innovations, and *likelihood-based* methods that employ the likelihood of the observations. In this work, the likelihood-based approaches were not explored further, since the estimation of Q and R is generally ill-conditioned when relying only on instantaneous measurements and requires data from multiple simulations [58]. Within the category of methods-of-moments, covariance matching was selected as a practical candidate. The central idea is to make the sample covariance of the innovation sequence consistent with its theoretical value. The actual covariance matrix of the innovation vector is calculated approximately by its sample covariance and matched with its theoretical form. Early work by Mehra proposed a moving-average estimator for R [59], and Wang later modified this procedure to ensure positive definiteness, which is required for a valid covariance matrix [60].

A more recent paper replaced the moving window with a forgetting factor [43] and extended the approach to estimate both R and Q simultaneously, resulting in an Adaptive Extended Kalman Filter. The approach introduces the recursive updates in Equation 4.22 and Equation 4.23. The forgetting factor α controls the adaptation rate, where smaller values lead to faster adaptation but increased estimation variance. In particular, R_k is updated using the *residual* ε_k rather than the innovation y_k, where the residual is defined as the difference between the current measurement and the *a posteriori* predicted measurement.

$$R_k = \alpha R_{k-1} + (1-\alpha)\left(\varepsilon_k \varepsilon_k^T + H_k P_{k+1|k} H_k^T\right) \qquad \text{with } \varepsilon_k = z_{m,k+1} - H_{k+1}\,\hat{x}_{k+1|k+1} \tag{4.22}$$

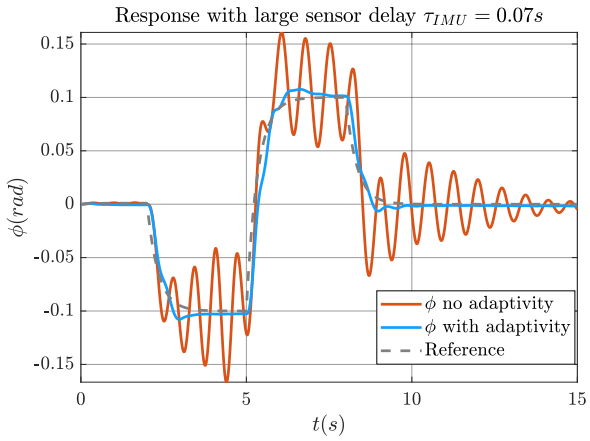
$$Q_k = \alpha Q_{k-1} + (1-\alpha)\left(K_k y_k y_k^T K_k^T\right) \tag{4.23}$$

The proposed adaptive EKF was implemented and tested using multiple values for the forgetting factor and under different uncertainty scenarios. The method proved to be effective only for estimating R_k, whereas the update in Equation 4.23 frequently drove Q_k towards zero. For this reason only Equation 4.22 was used. The adaptive estimation of R was particularly effective in scenarios with added delay and increased noise in the measurement path. Figure 4.3a shows the difference between the EKF with a fixed R matrix and the adaptive EKF for an added delay of $0.07s$ in the sensor path. Here $\alpha = 0.8$ was used. The adaptive algorithm improves the response by increasing the diagonal entries of R during the aircraft rotations as in Figure 4.3b, making the estimation more weighted towards model predictions. Additional simulations indicate that this adaptation improves robustness to delays up to $\tau_{IMU} = 0.5s$.

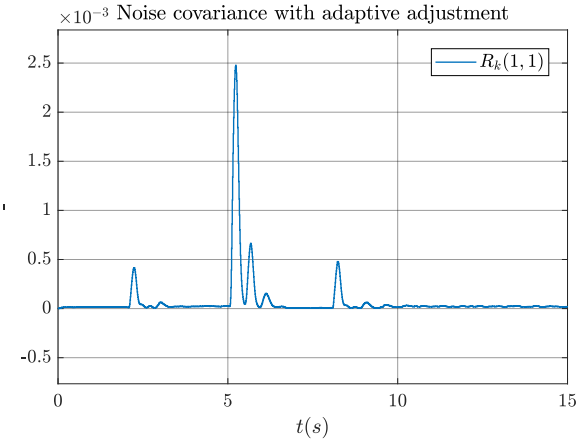
Despite these benefits, the adaptive EKF did not provide improvements over the regular EKF for model-parameter mismatch, as the algorithm did not successfully adapt the process-noise covariance in a beneficial way, contrary to the behavior reported in the original paper[43]. For these scenarios, the fixed values of Q and R obtained using the tuning procedure described in Section 2.5 yielded a better tracking performance. While this does not invalidate adaptive covariance estimation as a concept, it highlights that covariance adaptation is dependent on the method used and its application. Other adaptive approaches from literature may yield better results, but they were not implemented due to time constraints. Nevertheless, the results demonstrate that EKF performance can be significantly improved by adaptive features under specific off-nominal conditions, and that further research into online covariance adaptation remains a promising direction.

4.5. Joint EKF for Online Parameter Estimation

Kalman filtering frameworks can be extended by augmenting the state vector with additional variables, enabling simultaneous state estimation and online identification of parameters. This approach is commonly referred to as Joint state and parameter estimation or Joint EKF (J-EKF) and is one of the most popular approaches used to compensate for model-plant mismatches. In a typical formulation, the parameter variations are modeled as a random-walk process, and the parameter estimates are updated jointly with the EKF states within the same Bayesian recursive estimation [61]. The algorithm of the J-EFK is identical to the classic EKF algorithm, but with an augmented state vector that includes the parameter(s) θ to be identified.



(a) Improvements using adaptive estimation of R with added sensor delay



(b) Changing noise covariance value for ϕ with adaptive algorithm

Figure 4.3: Adaptive estimation of measurement noise covariance matrix R

In this work, the parameter dynamics are assumed to follow a random walk,

$$\theta_{k+1} = \theta_k + w_{\theta,k},$$

where $w_{\theta,k}$ represents a zero-mean process noise term that accounts for slow parameter drift and unmodeled effects. Using this assumption, the prediction step becomes

$$\hat{x}_{k+1|k} = f_d\big(\hat{x}_{k|k}, \theta_k, u_{m,k}\big), \qquad \hat{\theta}_{k|k-1} = \hat{\theta}_{k-1|k-1}, \tag{4.24}$$

Defining the following augmented vectors and covariance matrices,

$$\hat{x}^a = \begin{bmatrix} \hat{x} \\ \hat{\theta} \end{bmatrix}, \qquad Q^a = \begin{bmatrix} Q & \mathbf{0} \\ \mathbf{0} & Q_\theta \end{bmatrix} \qquad R^a = \begin{bmatrix} R & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$$

where $\hat{x}^a$ is the augmented state vector and Q_θ is the process-noise covariance associated with the random-walk parameter model. The state-transition Jacobian is augmented to include the partial differential with respect to the parameter.

$$F_k = \begin{bmatrix} \dfrac{\partial f_d}{\partial x} & \dfrac{\partial f_d}{\partial \theta} \\ 0 & I \end{bmatrix} \tag{4.25}$$

The remainder of the EKF recursion follows the standard EKF algorithm described earlier, applied now to the augmented state. Importantly, the filter must be initialized with an appropriate value of Q_θ, which determines how rapidly the parameter is allowed to vary. Since θ has no deterministic state-transition dynamics, Q_θ effectively sets the wandering magnitude of the parameter estimate and becomes an additional quantity to be tuned.

This Joint EKF was implemented for online identification of the mismatched model parameters, such as the inertial constants and the control effectiveness. In the conducted simulations, the identified parameters did not converge to stable and accurate estimates of the true mismatch. A primary reason is that the augmented parameter is are often only poorly observable, and can only be identified through cross-excitation with other states. In practice, this means that the parameter correction relies heavily on cross-correlation with the physical states and on the local Jacobian $\partial f_d / \partial \theta$. Additionally, because the parameter evolution is driven by the random-walk model, the balance between adaptation speed and estimate stability is dominated by the choice of Q_θ. The high sensitivity to tuning and the resulting variability in identification rendered the approach insufficiently robust for the model-plant mismatch scenarios considered. For these reasons, the Joint EKF was not included in the final comparison with the other controllers. The results indicate that additional work is required, for example, the introduction of separate excitation maneuvers or alternative estimation frameworks, before the method can be reliably used for online parameter identification.

4.6. Alternative EKF Architectures

This section considers alternative EKF architectures aimed at directly obtaining an estimate for the angular acceleration $\dot{\omega}$ for INDI feedback. This can be done by including $\dot{\omega}$ as a filter state and using additional measurements to strengthen observability.

4.6.1. EKF with Angular Acceleration State

The EKF can be augmented to include the angular acceleration $\dot{\omega}$ as an additional state, rather than estimating only the angular rate ω and subsequently differentiating it using a separate filter. In this way, $\dot{\omega}$ is available directly from the EKF output, which can reduce reliance on external derivative filtering. There are multiple ways to implement this augmentation. First, a process model is required to describe the state transition of the angular acceleration. A straightforward approach is to assume a constant angular acceleration model that evolves only through a random walk, analogous to the Joint EKF described earlier. This yields the following process model:

$$\begin{bmatrix} \omega_k \\ \dot{\omega}_k \end{bmatrix} = \begin{bmatrix} f(\omega_{k-1}, \delta_{k-1}) \\ \dot{\omega}_{k-1} \end{bmatrix} + \begin{bmatrix} w_{\omega_k} \\ w_{\dot{\omega}_k} \end{bmatrix} \tag{4.26}$$

Alternatively, instead of a constant process for angular acceleration, an *angular jerk* model can be used instead. The idea is to explicitly model the time-variation of angular acceleration by augmenting the state with the angular jerk $j \triangleq \ddot{\omega}$ and modeling jerk as the driving process noise acting on $\dot{\omega}$. This is analogue to the widely used constant-acceleration kinematic models in Kalman filtering, where the highest derivative is assumed constant over one sampling interval and its changes are driven by white noise [62].

The issue with this implementation lies in the observability of the state derivative. If only angular rate measurements are used, then the angular acceleration is not directly observed, and there is no correction with the measurement in the update step. As a result, $\dot{\omega}$ is weakly observable and can become dominated by the process model and noise tuning rather than the data.

4.6.2. EKF with Angular Acceleration Measurement

To overcome the issue of observability, an angular acceleration measurement can be incorporated within the EKF framework. This can be achieved in two main ways.

The first strategy is to differentiate the angular rate prior to the EKF, i.e., placing the filtered derivative after the rate measurement and before the EKF. This approach provides an artificial angular acceleration measurement, but it violates one of the key Kalman Filter assumptions, namely that the measurement noise components are uncorrelated. This means that the measurement noise covariance matrix R can no longer be assumed diagonal, since the differentiated signal is constructed from the same gyro samples used to measure angular rate, making the resulting measurement noises correlated. If this correlation is neglected in the EKF formulation, the filter may become inconsistent or overconfident, which can lead to noisy acceleration estimates and potentially degrade the INDI feedback path. A consistent implementation therefore requires explicitly accounting for the correlated measurement noise (e.g., through a full R matrix or an equivalent augmentation of the measurement/noise model).

A second approach is to directly measure angular acceleration using on-board sensors. Angular accelerometers are widely used in aviation, particularly in high-performance aircraft and unmanned aerial systems (UAS). Most angular accelerometers function either by employing piezoelectric materials or by utilizing a fluid-filled ring chamber shaped like a pendulum. Research has shown that such sensors can be modeled by high-order transfer functions [63], and these models are also employed for INDI control. Cakiroglu et al. used an angular acceleration sensor in conjunction with angular rate feedback for INDI control of the same aircraft considered in this research. The control performance did not improve substantially due to mismatches in time delay in the multiple feedback paths [64]. He recommended that these two sensors should be combined using a sensor fusion approach (i.e., Kalman Filtering) to ‘‘obtain fast, accurate and reliable angular rates’’.

Following the discussion in the paragraph above, a simple linear KF structure was developed for sensor fusion of IMU and angular accelerometer measurements $\dot{\omega}_s$. In this case the KF state vector is augmented with $\dot{p}, \dot{q}, \dot{r}$ as in Equation 4.27 and Equation 4.28. A constant acceleration model is assumed, while the angular rates are now obtained via integration of angular accelerations, without using the rotational dynamics model of Equation 4.13. The process and measurement models are then described by Equation 4.29 and Equation 4.30 respectively. Assuming angular accelerometer lag and noise parameters to be equal to the IMU, the KF structure can be substituted to the EKF. The

response can be seen in Figure 4.4a and Figure 4.4b for roll and pitch angle commands. This alternative structure was not considered further and was not compared with the EKF controller since the use of angular accelerometers lies outside the scope of this research, but it constitutes a valid direction for future research.

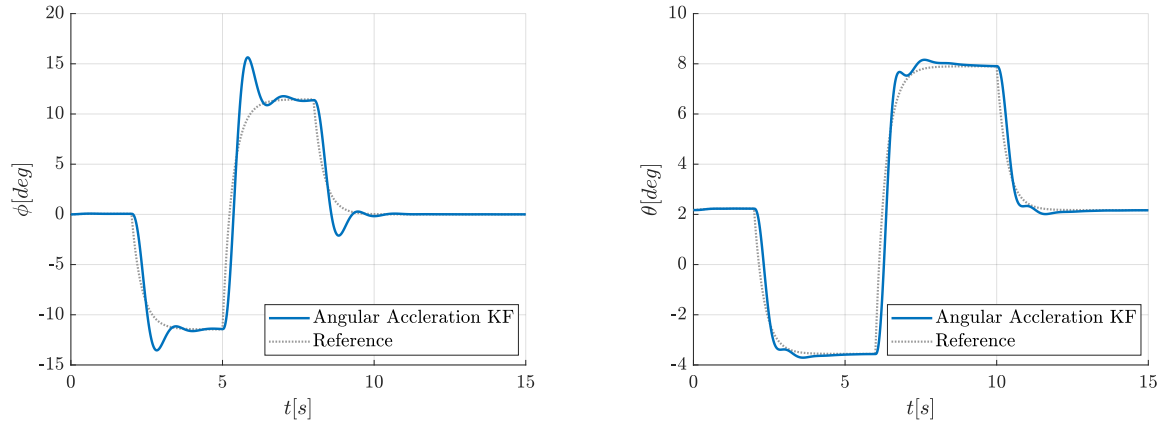
$$x_k = \begin{bmatrix} p_k & q_k & r_k & \dot{p}_k & \dot{q}_k & \dot{r}_k \end{bmatrix}^T = \begin{bmatrix} \omega_k & \dot{\omega}_k \end{bmatrix}^T \tag{4.27}$$

$$z_k = \begin{bmatrix} p_{s,k} & q_{s,k} & r_{s,k} & \dot{p}_{s,k} & \dot{q}_{s,k} & \dot{r}_{s,k} \end{bmatrix}^T = \begin{bmatrix} \omega_{s,k} & \dot{\omega}_{s,k} \end{bmatrix}^T \tag{4.28}$$

$$x_{k+1} = Ax_k + w_k, \qquad A = \begin{bmatrix} \mathbf{I}_3 & \Delta t\, \mathbf{I}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix} \tag{4.29}$$

$$z_k = Hx_k + v_k, \qquad H = \mathbf{I}_6 \tag{4.30}$$

Angular acceleration measurements can also be obtained without a dedicated sensor by using linear accelerometers.



(a) Roll angle tracking with KF and angular acceleration measurements **(b)** Pitch angle tracking with KF and angular acceleration measurements

Figure 4.4: Kalman Filter with angular accelerometer

Padgaonkar et al. utilized nine single-axis acceleration readings from four distinct non-coplanar locations to calculate angular velocity and acceleration components [65]. Costello et al. calculated the angular velocity and angular acceleration of projectiles by utilizing triaxial acceleration data, which involved using an array of accelerometers to measure 3D linear acceleration at multiple non-collinear points on the body, allowing for the calculation of rotational kinematics without gyroscopes [66]. Past research has proved that these solutions be applied to INDI control, and are a valid alternative for systems where the use of an angular accelerometer is not feasible due to weight or space constraints, for instance small UAVs [67].

5

Verification and Validation

This chapter presents the verification and validation activities carried out on the aircraft model and on the EKF estimation framework used in this research. Section 5.1 first addresses model verification through unit-step response tests on the available control inputs, with the objective of checking the general aircraft dynamics and the adopted sign conventions. Section 5.2 continues the analysis by exciting the main aircraft eigenmodes and comparing the responses with reference data from flight dynamics reported in the literature. Finally, Section 5.3 presents statistical consistency checks for the Extended Kalman Filter, based on the innovation sequence and innovation covariance, to assess whether the main Kalman Filter assumptions remain sufficiently valid in the absence of ground-truth data.

5.1. Model Verification

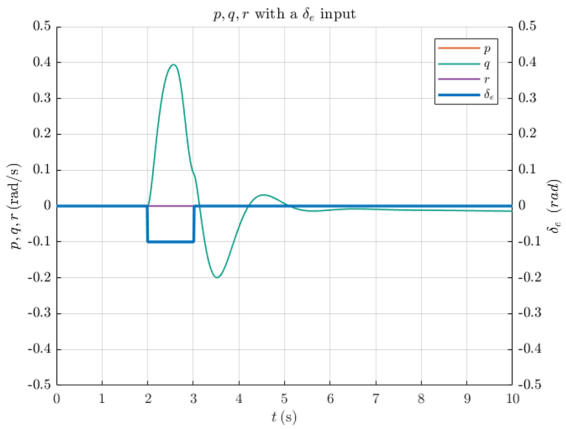
The MATLAB/Simulink model used for this research, described in Section 2.2, needs to be verified and validated. Verification is carried out using two different approaches. First, the general aircraft dynamics and sign conventions are verified by applying a short step input of -0.1 rad for 1 s to each of the available aerodynamic control surfaces. The responses of the model to these individual inputs are then compared with the standard expected responses for a stable twin-engine aircraft. The results of these tests are as follows:

- -0.1 rad $\boldsymbol{\delta_e}$: As observed in Figure 5.1a, the pitch rate increases positively as expected, which verifies the axis and sign conventions described in the previous sections. After the elevator returns to its initial position, the pitching motion damps quickly and settles just below the initial value. This is expected, since the aircraft exhibits positive static longitudinal stability, producing a restoring pitching moment that offsets the initial elevator deflection and results in a new equilibrium slightly below trim.
- -0.1 rad $\boldsymbol{\delta_a}$: As observed in Figure 5.1b, the roll rate increases positively until the ailerons return to zero. After this, the roll rate continues oscillating and damps slowly. From the plot, the adverse yaw effect can also be noticed, as well as a slight increase in pitch. The latter is due to two combined effects. The first is inertia coupling, which develops when an aircraft rolls about an axis different from its principal inertial axis, which is the case for the Citation II since its aerodynamic axis is located below the inertial axis. This is further confirmed by the term $\dot{q}_b = \frac{I_{zz}-I_{xx}}{I_{yy}} p_b r_b$ when the rotational dynamics are fully expanded.
- -0.1 rad $\boldsymbol{\delta_r}$: Deflecting the rudder causes the aircraft to yaw in the direction of the deflection, as seen in Figure 5.2a. Rudder deflection also induces an opposite roll rate and, consequently, a pitch rate due to inertia coupling. The yawing motion is slowly damped, as expected.
- 0.2 $\boldsymbol{\delta_t}$: If the thrust is increased asymmetrically on the left engine only, the aircraft starts yawing slowly in the positive direction, as expected (see Figure 5.2b). As the nose yaws to the right, the starboard wing speeds up, generating more lift and causing a roll to the right as well. The aircraft also pitches up as more engine power is provided.

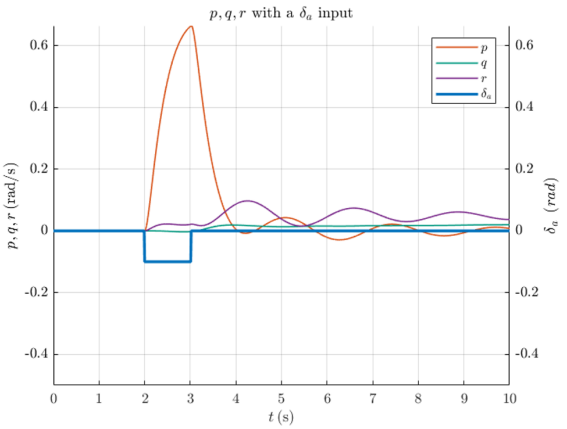
5.2. Mode Excitation Maneuvers

The validation approach consists of exciting the aircraft's eigenmodes, analyzing their time evolution, and comparing them with available reference data from flight dynamics literature for a general transport aircraft.

A doublet pulse command on the elevator is superimposed on the trimmed elevator deflection, thereby exciting the short-period mode. During this motion, the pitch rate quickly follows the elevator command and returns to zero moderately quickly, as shown in Figure 5.3a. θ and α vary together, thus causing very little change in the flight path

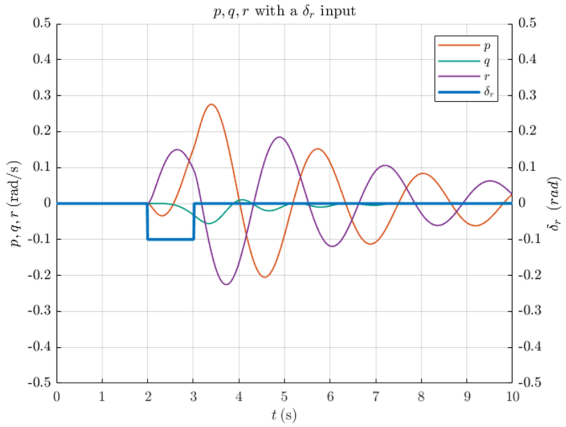


(a) System response to a negative $1s$ elevator step input

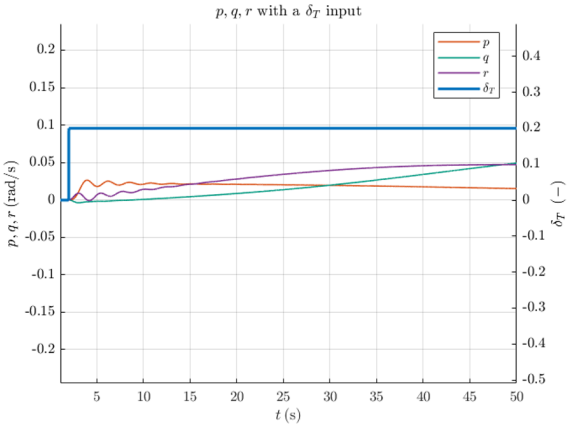


(b) System response to a negative $1s$ aileron step input

Figure 5.1: Longitudinal/lateral control surface unit-step responses.



(a) System response to a negative $1s$ rudder step input



(b) System response to an increase in thrust on the left engine

Figure 5.2: Yaw control unit-step responses.

angle, as shown in Figure 5.3b. The dynamics of the motion correspond to the reference plot from the literature, shown in Figure 5.3c.

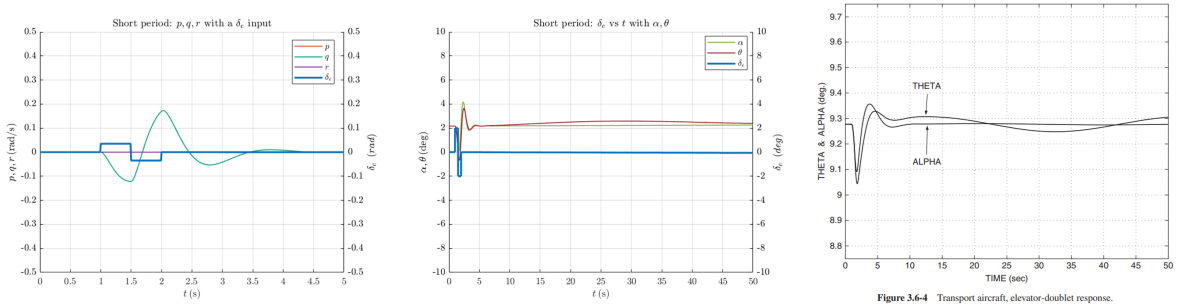
The phugoid motion can be excited by superimposing a 3 s doublet pulse on the steady-state throttle setting. The result is a long-period pitch-rate oscillation, which is slowly damped, as shown in Figure 5.4a. The angle of attack remains mostly unaffected (see Figure 5.4b). The response is consistent with the plot from the literature in Figure 5.4c. Note that the initial pitch attitude is opposite to that in the literature plot, since the Citation II has its engines mounted above the center of gravity rather than below it, as is the case for larger aircraft, causing a negative pitching moment.

A doublet pulse on the aileron initiates rolling, which damps rapidly with minimal oscillation, as can be seen in Figure 5.5a. This corresponds to excitation of the aperiodic roll lateral mode. A large doublet pulse on the rudder excites the well-known lateral oscillatory mode called Dutch roll, which couples yaw and roll motions. As can be seen in Figure 5.5b, r leads p by approximately 90° in phase, with lightly damped oscillations.

5.3. EKF Consistency Checks

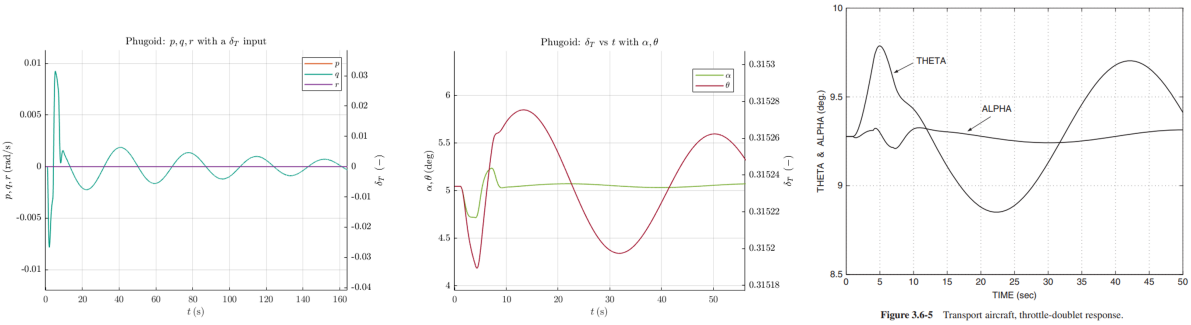
In typical cases, the performance of the Kalman Filter estimation is evaluated by comparison with ground-truth data. When such data are not available, as in this research, statistical methods can be used to assess whether the Kalman Filter assumptions have been violated [68].

A fundamental quantity for the Kalman Filter is the innovation sequence. According to the underlying assumptions,



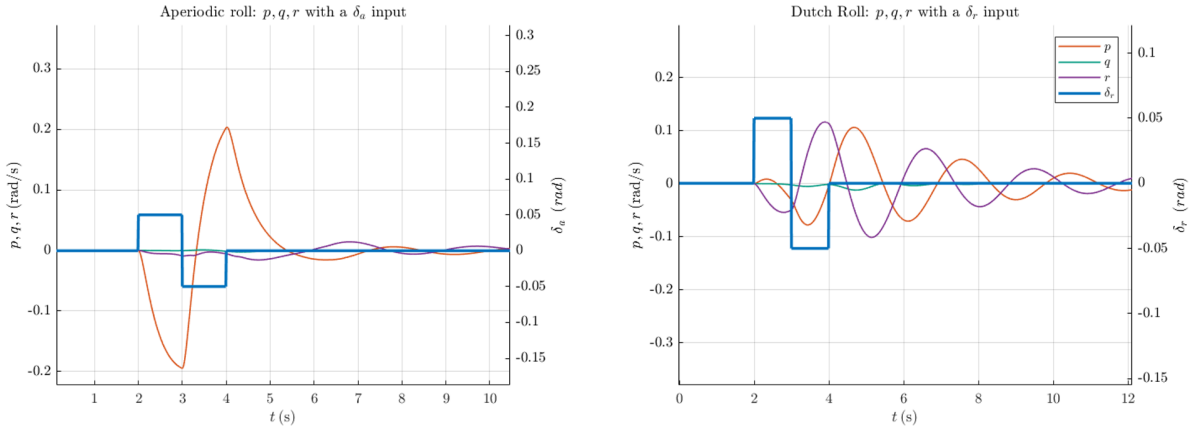
(a) Short period excitation, pitch rate q **(b)** α and θ during short period mode **(c)** α and θ from literature

Figure 5.3: Short Period maneuver validated against literature.



(a) Phugoid excitation, pitch rate q **(b)** α and θ during Phugoid mode **(c)** α and θ from literature

Figure 5.4: Phugoid maneuver validated against literature.



(a) Aperiodic roll excitation **(b)** Dutch Roll excitation

Figure 5.5: Lateral modes.

the innovation is defined as the difference between the current measurement z_k and the predicted measurement, given by $h(x_{k+1})\,\hat{x}_{k+1|k}$. As such, the innovation cannot exhibit a predictable bias, implying that it should be a zero-mean sequence. At the same time, the innovation should also be uncorrelated with itself. Mathematically, this is expressed as:

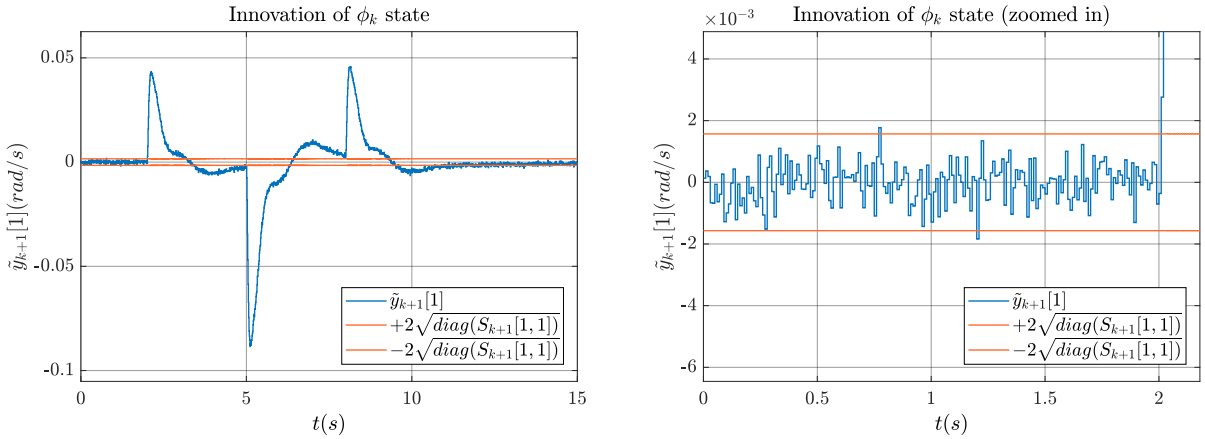
$$\mathbb{E}\left[\tilde{y}_k\right] = 0 \tag{5.1}$$

$$\mathbb{E}\left[\tilde{y}_k \tilde{y}_{k-\tau}^{\top}\right] = 0, \qquad \tau \neq 0 \tag{5.2}$$

These properties can be assessed by inspecting the innovation sequence over time for a controlled roll motion, as plotted in Figure 5.6a, and by evaluating the autocorrelation function $r(\tau)$, shown in Figure 5.8a. From both plots, it can be concluded that, for the simulation sample considered, the innovation is neither zero-mean nor white. Ideally, the innovation should be randomly scattered around zero, and its autocorrelation should resemble a delta function, as illustrated by the literature examples in Figure 5.7 and Figure 5.8b, respectively. In the present case, the innovation is correlated with the aircraft motion, which is attributed to the inherent sensor delay: the current and predicted measurements do not correspond to the same sample, and the resulting mismatch scales with the amplitude of the motion. This effect can be partially corrected by introducing an additional delay after the prediction step; however, doing so makes synchronization with the rest of the INDI loop more difficult.

The nonzero mean of the innovation can be explained in the same way. When steady state is achieved, the innovation becomes approximately zero-mean, as can be seen in the first 2 seconds of the simulation in Figure 5.6b. For this interval, the mean of the innovation is calculated as $6 \cdot 10^{-6}$.

To verify whether the Q and R matrices have been tuned correctly in a statistical sense, a consistency test can be performed using the innovation covariance S_k. From the Kalman Filter derivation, $\tilde{y}_k$ should be bounded by S_k. This can be checked by plotting the 2σ bound set by the covariance matrix, i.e., $\pm 2\sqrt{(S_{k_{1,1}})}$, and verifying whether 95% of the values of $\tilde{y}_k$ lie within these bounds. Figure 5.6b suggests that, for steady-state values, this is indeed the case, although slightly more than 95% of the values lie inside the bounds. This indicates that the process covariance values in Q could be slightly lowered; however, since they have been tuned for best tracking response, they are left unchanged.



(a) Innovation of ϕ over time

(b) Innovation of ϕ over time (close-up)

Figure 5.6: Innovation over time

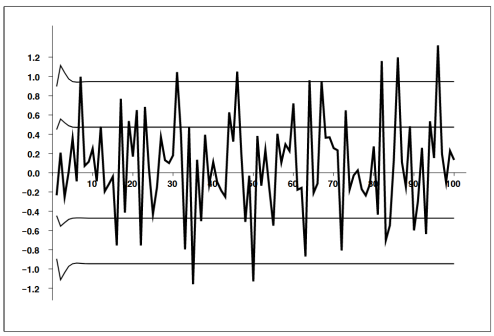
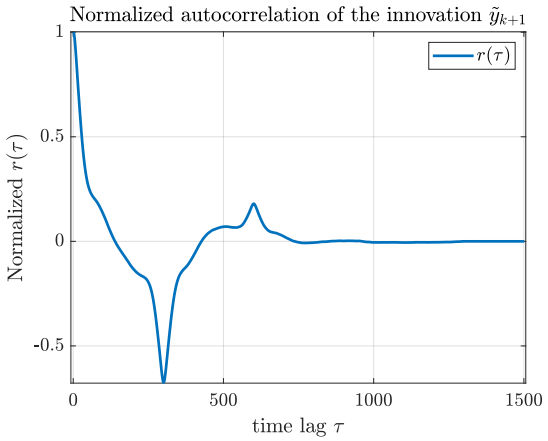
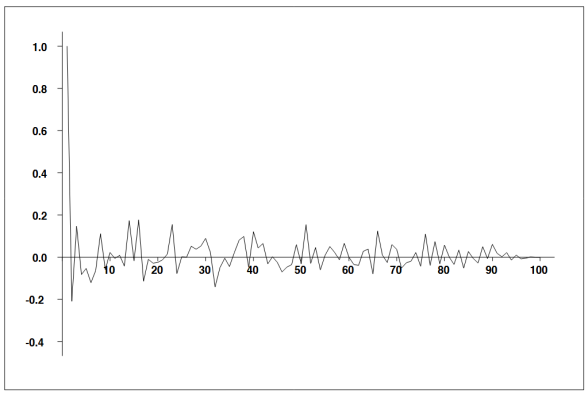


Figure 5.7: Uncorrelated and zero-mean innovation [68]



(a) Autocorrelation of the innovation sequence normalized by the first sample $\tilde{y}_0$



(b) Autocorrelation of an uncorrelated innovation sequence[68]

Figure 5.8: Autocorrelation of innovation sequence

Part IV

Closure

6

Conclusion

In this research, an alternative approach to angular acceleration estimation within the Hybrid INDI framework is proposed, based on an Extended Kalman Filter (EKF). The research starts with a literature review in Chapter 3, beginning with nonlinear control approaches, which identifies INDI and its Hybrid version as simple yet powerful nonlinear control methods. The various approaches for Hybrid INDI are compared, and the motivation to investigate KF-based approaches is introduced through five research questions. The proposed system for implementation is a high-fidelity nonlinear model of the Cessna Citation II, a small passenger aircraft. A baseline Hybrid INDI controller is first constructed using Complementary Filter (CF) blending, as described in Section 2.4. An EKF algorithm for estimation is then proposed in Section 2.5 and integrated within the Hybrid INDI framework. The parameters of both the CF and EKF are tuned using a multi-model robust algorithm based on [28]. The EKF- and CF-based controllers are subsequently evaluated under nominal and degraded conditions in Section 2.6. Furthermore, additional research regarding optimal filter parameters and alternative Kalman filter-based methods is presented and discussed in Chapter 4. Finally, verification and validation of the model and of the methods used are presented in Chapter 5.

The five research questions laid out at the end of the literature study are repeated below and answered with the findings of this research.

Main Research Question

In the context of flight control using Hybrid INDI, how can a Kalman Filter-based approach be used to provide an optimal blend of sensor measurements and on-board model predictions?

Simulations have shown that a Kalman Filter-based approach represents a feasible alternative for a time-varying blend of model predictions and sensor measurements for the estimation of angular rate and angular acceleration, and can be applied within the Hybrid INDI control framework. An optimal blend of the two information paths can be obtained by adjusting the filter noise covariance matrices Q and R. A tuning approach that considers a pre-defined range of uncertainties for model and sensor parameters is employed to identify the filter parameters that yield the lowest tracking error among worst-case scenarios. These parameters provide robustness to variations within, and in some cases beyond, the parameter space considered.

Sub-question 1

What are the possible architectures for integrating a Kalman Filter for state estimation within a Hybrid INDI framework?

Kalman Filters (KF) can be integrated in numerous ways. Extended and Unscented KFs are the two main baseline implementations of this type of filter, due to the nonlinear nature of INDI control. In this research, an Extended Kalman Filter (EKF) is implemented for angular rate estimation, and the resulting estimate is subsequently differentiated to obtain angular acceleration. To compensate for the EKF lag and maintain synchronization between feedback paths, a smoothing filter based on the Kalman gain is introduced in the actuator feedback path. Angular acceleration can also be included as a filter state, but it requires additional sensors for observability. Adaptive approaches in the form of recursive functions can also be employed to vary the value of R based on the measurement residual. Joint and ensemble KF variants could allow the simultaneous estimation of the state and additional uncertain parameters.

Sub-question 2a

How does the Hybrid INDI with a Kalman Filter–based blend perform under nominal flight conditions?

When using an EKF for blending sensor measurements and model predictions and under nominal conditions the Hybrid INDI controller follows the reference command accurately and steadily, with minimal tracking error. This is primarily due to the inner and outer linear controllers, which are synthesized using specific performance metrics on the output response, allowing for high bandwidth, overshoot below 15%, and low actuator usage. The EKF blending effectively compensates for the noise and delay in the measurements using model predictions, and synchronization between the different feedback paths is achieved, resulting in a successful application of the INDI control law.

Sub-question 2b

How does the Hybrid INDI with a Kalman Filter–based blend perform under degraded flight conditions, such as model-parameter uncertainties, sensor delay and noise?

The KF-based blend maintains good performance even under off-nominal conditions, due to the blending between measurements and model predictions. For model-plant mismatch in parameters such as the lateral inertia or the control effectiveness coefficient, the tracking error varies slowly and the response remains stable even with variations of ±80%. The tracking error increases approximately linearly with increasing noise in the rate measurement. Finally, the KF-based blend maintains stable attitude control even in the presence of additional delay, up to 70 ms.

Sub-question 3

How does the Hybrid INDI Kalman Filter–based blend compare to the Complementary Filter approach in terms of performance, under both nominal and degraded flight conditions?

Under nominal conditions, the EKF estimation shows an identical response when compared with the more standard Complementary Filter (CF) estimation used in the baseline controller. The KF-based blend provides improvements in robustness to model-plant mismatches and sensor delay. For increasing mismatches between the model and plant parameters, the EKF estimation delivers lower tracking error than the CF, especially for very high mismatches (±80%). No clear improvement over the CF is observed in scenarios with high measurement noise. The EKF estimation provides improvements over the CF particularly in the presence of additional sensor delay, where the CF is far more sensitive to an increase in delay than the EKF-based controller. Additionally, if adaptive laws are used to vary the measurement noise covariance at each time step, robustness to delay can be improved further. Finally, when all the above effects are combined, the EKF shows improved robustness, as the worst-case errors are lower than those of the CF for the same scenario.

The next section, Chapter 7, provides recommendations and advice for future research in the field of Hybrid INDI, especially using Kalman Filter-based approaches.

7

Recommendations

The research described in this report shows that the EKF is a valid alternative for angular acceleration estimation within the Hybrid INDI framework. However, the results and discussion also highlight that several aspects could benefit from further investigation, and that additional directions for future research have been opened.

The validation of the EKF-based estimation approach should be strengthened by extending the simulation campaigns to include piloted flight-test data or hardware-in-the-loop experiments. Such scenarios would enable the evaluation of controller performance under realistic sensor and actuator conditions. Another important aspect to consider is the computational burden introduced by the Kalman Filter algorithm, which requires significantly more computational power than the Complementary Filter used in the baseline controller. This can be assessed by timing the code execution or by analyzing algorithmic complexity using Big O notation. This aspect was not discussed in the present research because the focus was on application in aircraft flight control, which usually takes place on machines with sufficient computational power, but it could become an issue for smaller systems.

Beyond Kalman filter-based methods, other nonlinear estimation techniques could also be considered. For instance, the H_∞ filter seeks to minimize the worst-case estimation error rather than the expected least-squares error, potentially offering increased robustness to modeling uncertainties [69]. In addition, optimization-based approaches such as Moving Horizon Estimation and Maximum Likelihood Estimation provide alternative formulations that explicitly account for constraints and nonlinearities, but at the cost of increased computational complexity.

As discussed in Section 4.4, promising alternatives to the EKF approach include adaptive variants for online tuning of the filter parameters, rather than relying on fixed values. Adaptivity is a well-researched aspect in flight control, since it brings improvements in cases of varying model parameters or system and sensor faults, which cannot be effectively handled by static systems. The Hybrid INDI framework could benefit from these improvements, and several adaptive approaches that could be applied to Hybrid INDI already exist in the literature, including adaptive Complementary Filters based on optimization techniques such as least-squares estimation [70].

Finally, an alternative direction for improving angular acceleration estimation lies in including it as a Kalman Filter state and making this quantity directly measurable. This may be achieved through the use of angular accelerometers in addition to rate gyros and by using sensor fusion techniques, as already highlighted by past research [64]. When the use of angular accelerometers is not feasible, multiple linear accelerometers can be used instead, as recent improvements in INDI control for UAV have shown, yielding higher accuracy than numerical differentiation of angular rate measurements, with a corresponding reduction in sensitivity to sensor noise and mounting errors [67].

References

[1] H. Nyquist. ‘‘Regeneration Theory’’. In: *Bell System Technical Journal* 11 (1 1932), pp. 126–147. DOI: 10.1002/j.1538-7305.1932.tb02344.x.

[2] Hendrik W. Bode. *Network Analysis and Feedback Amplifier Design*. Bell Telephone Laboratories Series. First published September 1945. New York, NY: D. Van Nostrand Company, Inc., 1945.

[3] D. T. McRuer et al. *Aircraft Dynamics and Automatic Control*. Princeton: Princeton University Press, 1990. DOI: 10.1515/9781400855988.

[4] Rudolf Emil Kalman et al. ‘‘Contributions to the theory of optimal control’’. In: *Boletín De La Sociedad Matemática Mexicana* 5.2 (1960), pp. 102–119.

[5] Jeff Shamma et al. ‘‘Analysis of Gain Scheduled Control for Nonlinear Plants’’. English (US). In: *IEEE Transactions on Automatic Control* 35.8 (Aug. 1990), pp. 898–907. DOI: 10.1109/9.58498.

[6] Jeff Shamma et al. ‘‘Gain Scheduling: Potential Hazards and Possible Remedies’’. In: *Control Systems, IEEE* 12 (July 1992), pp. 101–107. DOI: 10.1109/37.165527.

[7] Duc H. Nguyen et al. ‘‘Transient Dynamics Assessment of a Gain-Scheduled Aircraft Controller Using Nonlinear Frequency Approach’’. en. In: *Journal of Guidance, Control, and Dynamics* 44.9 (2021), pp. 1692–1699. DOI: 10.2514/1.G005866.

[8] Michael A. Henson et al. *Nonlinear Process Control*. Upper Saddle River, NJ: Prentice Hall PTR, 1996. 570 pp.

[9] Michael A. Henson et al. ‘‘Feedback Linearizing Control’’. In: *Nonlinear Process Control*. Ed. by Dale E. Seborg et al. Vol. 4. Advances in Chemical Engineering. Originally appeared as Ph.D. thesis Chapter 4 (University of California, Santa Barbara, 1992). Upper Saddle River, NJ, USA: Prentice Hall, 1997, pp. 149–231.

[10] Miroslav Krstic et al. *Nonlinear and adaptive control design*. Adaptive and learning systems for signal processing, communications, and control. New York: Wiley, 1995. 563 pp.

[11] Fadi Alyoussef et al. ‘‘A Review on Nonlinear Control Approaches: Sliding Mode Control, Back-Stepping Control and Feedback Linearization Control’’. In: *Proceedings of the International Engineering and Natural Sciences Conference (IENSC)*. Diyarbakır, Turkey, Nov. 2019, pp. 608–619.

[12] Lijia Cao et al. ‘‘Robust Adaptive Backstepping Control for a Class of Uncertain Nonlinear System’’. In: *Proceedings of the 2016 Chinese Control and Decision Conference (CCDC)*. Yinchuan, China: IEEE, May 2016.

[13] Yongwei Tang et al. ‘‘Non-linear extended state observer-based sliding mode control for a direct-driven wind energy conversion system with permanent magnet synchronous generator’’. In: *The Journal of Engineering* 2019.15 (2019), pp. 613–617. DOI: https://doi.org/10.1049/joe.2018.9392. URL: https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/joe.2018.9392.

[14] Riccardo Furlan et al. ‘‘Friction compensation in the interstand looper of hot strip mills: A sliding-mode control approach’’. In: *Control Engineering Practice - CONTROL ENG PRACTICE* 16 (Feb. 2008), pp. 214–224. DOI: 10.1016/j.conengprac.2006.12.002.

[15] Daewon Lee et al. ‘‘Feedback linearization vs. adaptive sliding mode control for a quadrotor helicopter’’. In: *International Journal of Control, Automation and Systems* 7.3 (June 2009), pp. 419–428. DOI: 10.1007/s12555-009-0311-8. URL: http://link.springer.com/10.1007/s12555-009-0311-8 (visited on 06/09/2025).

[16] Gaya Prasad et al. ‘‘A Comparison Between Sliding Mode Control and Feedback Linearization’’. In: *2018 5th IEEE Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*. 2018 5th IEEE Uttar Pradesh Section International Conference on Electrical, Electronics and

Computer Engineering (UPCON). Gorakhpur: IEEE, Nov. 2018, pp. 1–5. DOI: 10.1109/UPCON.2018.8597038. URL: https://ieeexplore.ieee.org/document/8597038/ (visited on 06/09/2025).

[17] Ola Harkegard. *Backstepping Designs for Aircraft Control – What is there to gain?* Tech. rep. LiTH-ISY-R-2339. Presented at CCSSE 2001, Norrkoping, Sweden. Division of Automatic Control, Department of Electrical Engineering, Linkopings universitet, Mar. 2001.

[18] D. Swaroop et al. ''Dynamic surface control for a class of nonlinear systems''. In: *IEEE Transactions on Automatic Control* 45.10 (2000), pp. 1893–1899. DOI: 10.1109/TAC.2000.880994.

[19] S. A. Snell et al. ''Nonlinear Inversion Flight Control for a Supermaneuverable Aircraft''. In: *Journal of Guidance, Control, and Dynamics* 15.4 (July 1992), pp. 976–984. DOI: 10.2514/3.20932.

[20] R. Colgren et al. ''Dynamic Inversion Applied to the F-117A''. In: *Modeling and Simulation Technologies Conference*. New Orleans, LA, USA: American Institute of Aeronautics and Astronautics, Aug. 1997. DOI: 10.2514/6.1997-3786.

[21] P. Smith. ''A Simplified Approach to Nonlinear Dynamic Inversion Based Flight Control''. In: *23rd Atmospheric Flight Mechanics Conference*. Boston, MA, USA: American Institute of Aeronautics and Astronautics, Aug. 1998. DOI: 10.2514/6.1998-4461.

[22] R. J. Adams et al. ''An Integrated Approach to Flight Control Design Using Dynamic Inversion and μ-Synthesis''. In: *1993 American Control Conference*. San Francisco, CA, USA: IEEE, June 1993, pp. 1385–1389. DOI: 10.23919/ACC.1993.4793098.

[23] B. Bacon et al. ''Reconfigurable Flight Control Using Nonlinear Dynamic Inversion with a Special Accelerometer Implementation''. In: *AIAA Guidance, Navigation, and Control Conference and Exhibit*. Denver, CO, USA: American Institute of Aeronautics and Astronautics, Aug. 2000. DOI: 10.2514/6.2000-4565.

[24] Agnes Steinert et al. ''From fundamentals to applications of incremental nonlinear dynamic inversion: A survey on INDI – Part I''. en. In: *Chin. J. Aeronaut.* (Apr. 2025), p. 103553.

[25] Agnes Steinert et al. ''Advancements in incremental nonlinear dynamic inversion and its components: A survey on INDI – Part II''. en. In: *Chin. J. Aeronaut.* (May 2025), p. 103591.

[26] K. Bordignon et al. ''Control Allocation for the X-35B''. In: *2002 Biennial International Powered Lift Conference and Exhibit*. Williamsburg, VA, USA: American Institute of Aeronautics and Astronautics, Nov. 2002. DOI: 10.2514/6.2002-6020.

[27] G. Dickman. ''Rapid Prototyping of Micro Air Vehicle Control Systems''. In: *Infotech@Aerospace*. Arlington, VA, USA: American Institute of Aeronautics and Astronautics, Sept. 2005. DOI: 10.2514/6.2005-7068.

[28] F. Grondman et al. ''Design and Flight Testing of Incremental Nonlinear Dynamic Inversion-based Control Laws for a Passenger Aircraft''. In: *2018 AIAA Guidance, Navigation, and Control Conference*. Kissimmee, FL, USA: American Institute of Aeronautics and Astronautics, Jan. 2018. DOI: 10.2514/6.2018-0385.

[29] R. Steffensen et al. ''Filter and Sensor Delay Synchronization in Incremental Flight Control Laws''. In: *Aeronautical Science (AS)* 6.2 (June 2023), pp. 285–304. DOI: 10.1007/s42401-022-00186-2.

[30] P. Lu et al. ''Aircraft Fault-Tolerant Trajectory Control Using Incremental Nonlinear Dynamic Inversion''. In: *Control Engineering Practice* 57 (Dec. 2016), pp. 126–141. DOI: 10.1016/j.conengprac.2016.09.010.

[31] E. J. J. Smeur et al. ''Adaptive Incremental Nonlinear Dynamic Inversion for Attitude Control of Micro Air Vehicles''. In: *Journal of Guidance, Control, and Dynamics* 39.3 (Mar. 2016), pp. 450–461. DOI: 10.2514/1.G001490.

[32] S. Pineau et al. ''L1 Adaptive Augmentation of an Incremental Nonlinear Dynamic Inversion Autopilot for Dual-Spin Guided Projectiles''. In: *AIAA SCITECH 2023 Forum*. National Harbor, MD & Online: American Institute of Aeronautics and Astronautics, Jan. 2023. DOI: 10.2514/6.2023-1998.

[33] Jeffrey J Harris. ''F-35 flight control law design, development and verification''. In: *2018 Aviation Technology, Integration, and Operations Conference*. Atlanta, Georgia: American Institute of Aeronautics and Astronautics, June 2018.

[34] Chong-Sup Kim et al. ‘‘Development of flight control law for improvement of uncommanded lateral motion of the fighter aircraft’’. en. In: *Int. J. Aeronaut. Space Sci.* 21.4 (Dec. 2020), pp. 1059–1077.

[35] Chong-Sup Kim et al. ‘‘Stability margin and structural coupling analysis of a hybrid INDI control for the fighter aircraft’’. en. In: *Int. J. Aeronaut. Space Sci.* 22.5 (Oct. 2021), pp. 1154–1169.

[36] Yagiz Kumtepe et al. ‘‘Flight control law design using hybrid incremental nonlinear dynamic inversion’’. In: *AIAA SCITECH 2022 Forum*. San Diego, CA & Virtual: American Institute of Aeronautics and Astronautics, Jan. 2022.

[37] T.J.J. Traas. ‘‘Hybrid INDI with enhanced FEP’’. Accessed: 2025-07-12. Master’s thesis. Delft University of Technology, Oct. 2024. URL: https://repository.tudelft.nl/record/uuid:1c3fb8d1-4e7a-4bda-8ce0-537a69fe64f8.

[38] Huitao Lyu et al. ‘‘Extended-state-observer-based angular acceleration estimation for supersonic aircraft lateral–directional control’’. en. In: *Appl. Sci. (Basel)* 13.11 (May 2023), p. 6598.

[39] S. Sieberling et al. ‘‘Robust Flight Control Using Incremental Nonlinear Dynamic Inversion and Angular Acceleration Prediction’’. In: *Journal of Guidance, Control, and Dynamics* 33.6 (Nov. 2010), pp. 1732–1742. DOI: 10.2514/1.49978. URL: https://arc.aiaa.org/doi/10.2514/1.49978 (visited on 06/17/2025).

[40] Tito J. Ludeña Cervantes et al. ‘‘Flight Control Design using Incremental Nonlinear Dynamic Inversion with Fixed-lag Smoothing Estimation’’. In: *International Journal of Aeronautical and Space Sciences* 21.4 (Dec. 2020), pp. 1047–1058. DOI: 10.1007/s42405-020-00273-8. URL: https://link.springer.com/10.1007/s42405-020-00273-8 (visited on 06/17/2025).

[41] Maria Isabel Ribeiro. *Kalman and Extended Kalman Filters: Concept, Derivation and Properties*. en. Lisboa, PORTUGAL: Institute for Systems and Robotics, Apr. 2004. URL: https://www.researchgate.net/publication/2888846_Kalman_and_Extended_Kalman_Filters_Concept_Derivation_and_Properties.

[42] M. St-Pierre et al. ‘‘Comparison between the unscented kalman filter and the extended kalman filter for the position estimation module of an integrated navigation information system’’. en. In: *IEEE Intelligent Vehicles Symposium, 2004*. Parma, Italy: IEEE, 2004, pp. 831–835. DOI: 10.1109/IVS.2004.1336492. URL: http://ieeexplore.ieee.org/document/1336492/.

[43] Shahrokh Akhlaghi et al. ‘‘Adaptive adjustment of noise covariance in Kalman filter for dynamic state estimation’’. en. In: *2017 IEEE Power & Energy Society General Meeting*. Chicago, IL: IEEE, 2017, pp. 1–5. DOI: 10.1109/PESGM.2017.8273755. URL: http://ieeexplore.ieee.org/document/8273755/.

[44] A. H. Mohamed et al. ‘‘Adaptive Kalman Filtering for INS/GPS’’. en. In: *Journal of Geodesy* 73.4 (May 1999), pp. 193–203. DOI: 10.1007/s001900050236.

[45] Di Ding et al. ‘‘A Joint Online Estimation Method for Aircraft Aerodynamic Parameters and Thrust Deviation’’. In: *International Journal of Aerospace Engineering* 2025.1 (2025). DOI: 10.1155/ijae/1890214.

[46] Zeming Liang et al. ‘‘An Enhanced Adaptive Ensemble Kalman Filter for Autonomous Underwater Vehicle Integrated Navigation’’. In: *Drones* 8.12 (2024), p. 711. DOI: 10.3390/drones8120711.

[47] W. T. Higgins. ‘‘A Comparison of Complementary and Kalman Filtering’’. In: *Ieee Transactions on Aerospace and Electronic Systems* AES-11 (3 1975), pp. 321–325. DOI: 10.1109/taes.1975.308081.

[48] Gonzalo Perez Paina et al. ‘‘Experimental comparison of Kalman and complementary filter for attitude estimation’’. In: Aug. 2011.

[49] R. Mahony et al. ‘‘Complementary filter design on the special orthogonal group SO(3)’’. en. In: *Proceedings of the 44th IEEE Conference on Decision and Control*. Seville, Spain: IEEE, 2005, pp. 1477–1484. DOI: 10.1109/CDC.2005.1582367. URL: http://ieeexplore.ieee.org/document/1582367/.

[50] Bin Liu et al. ‘‘Application of Multi-order Complementary Filtering Technique in Dynamic Attitude Measurement of Forestry Special Robots’’. en. In: *Proceedings of the 2017 International Conference on Electronic Industry and Automation (EIA 2017)*. Suzhou, China: Atlantis Press, 2017. DOI: 10.2991/eia-17.2017.14. URL: http://www.atlantis-press.com/php/paper-details.php?id=25881957.

[51] Spyros Kontelis et al. ‘‘Comparison of Complementary Filters Implementations for Unmanned Aerial Vehicles’’. en. In: *2022 Panhellenic Conference on Electronics Telecommunications (PACET)*. Tripolis, Greece: IEEE, Dec. 2022, pp. 1–4. DOI: 10.1109/PACET56979.2022.9976362. URL: https://ieeexplore.ieee.org/document/9976362/.

[52] Parag Narkhede et al. ‘‘Least square estimation-based adaptive complimentary filter for attitude estimation’’. en. In: *Transactions of the Institute of Measurement and Control* 41.1 (Jan. 2019), pp. 235–245. DOI: 10.1177/0142331218755234.

[53] Shashi Poddar et al. ‘‘PSO Aided Adaptive Complementary Filter for Attitude Estimation’’. en. In: *Journal of Intelligent Robotic Systems* 87.3–4 (2017), pp. 531–543. DOI: 10.1007/s10846-017-0507-8.

[54] Parag Narkhede et al. ‘‘Cascaded Complementary Filter Architecture for Sensor Fusion in Attitude Estimation’’. en. In: *Sensors* 21.6 (Mar. 2021), p. 1937. DOI: 10.3390/s21061937.

[55] Kenneth R. Muske et al. ‘‘Nonlinear State Estimation’’. In: *Nonlinear Process Control*. Ed. by Michael A. Henson et al. Prentice Hall PTR, 1997. Chap. 6, pp. 311–370.

[56] Richard S. Bucy et al. *Filtering for Stochastic Processes with Applications to Guidance*. Vol. 23. Interscience Tracts in Pure and Applied Mathematics. New York: Interscience Publishers, 1968.

[57] Pierre Tandeo et al. ‘‘A Review of Innovation-Based Methods to Jointly Estimate Model and Observation Error Covariance Matrices in Ensemble Data Assimilation’’. en. In: *Monthly Weather Review* 148.10 (Oct. 2020), pp. 3973–3994. DOI: 10.1175/MWR-D-19-0240.1.

[58] Ricardo Todling. ‘‘A lag-1 smoother approach to system-error estimation: sequential method’’. In: *Quarterly Journal of the Royal Meteorological Society* 141.690 (2015), pp. 1502–1513. DOI: https://doi.org/10.1002/qj.2460. eprint: https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.2460. URL: https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.2460.

[59] R. Mehra. ‘‘Approaches to adaptive filtering’’. In: *IEEE Transactions on Automatic Control* 17.5 (1972), pp. 693–698. DOI: 10.1109/TAC.1972.1100100.

[60] Jinling Wang. ‘‘Stochastic Modeling for Real-Time Kinematic GPS/GLONASS Positioning’’. en. In: *Navigation* 46.4 (Dec. 1999), pp. 297–305. DOI: 10.1002/j.2161-4296.1999.tb02416.x.

[61] ‘‘Evaluation of Adaptive Extended Kalman Filter Algorithms for State Estimation in Presence of Model-Plant Mismatch’’. en. In: 46 (Dec. 2013), pp. 184–189. DOI: 10.3182/20131218-3-IN-2045.00175.

[62] Ethan Eade. *Process Noise for Gaussian Random Walks*. en. Mar. 2017. URL: https://ethaneade.com/process_noise_random_walk.pdf.

[63] Dyah Jatiningrum et al. ‘‘Modelling an Angular Accelerometer using Frequency-Response Measurements’’. en. In: *AIAA Guidance, Navigation, and Control Conference*. San Diego, California, USA: American Institute of Aeronautics and Astronautics, Jan. 2016. DOI: 10.2514/6.2016-1139. URL: https://arc.aiaa.org/doi/10.2514/6.2016-1139.

[64] Can Cakiroglu et al. ‘‘Robust Incremental Nonlinear Dynamic Inversion Control Using Angular Accelerometer Feedback’’. en. In: *2018 AIAA Guidance, Navigation, and Control Conference*. Kissimmee, Florida: American Institute of Aeronautics and Astronautics, Jan. 2018. DOI: 10.2514/6.2018-1128. URL: https://arc.aiaa.org/doi/10.2514/6.2018-1128.

[65] A. J. Padgaonkar et al. ‘‘Measurement of Angular Acceleration of a Rigid Body Using Linear Accelerometers’’. In: *Journal of Applied Mechanics* 42.3 (1975), pp. 552–556. DOI: 10.1115/1.3423640.

[66] Mark Costello et al. ‘‘Determining Angular Velocity and Angular Acceleration of Projectiles Using Triaxial Acceleration Measurements’’. In: *Journal of Spacecraft and Rockets - J SPACECRAFT ROCKET* 39 (Jan. 2002), pp. 73–80. DOI: 10.2514/2.3784.

[67] Hoijo Jeong et al. ‘‘Angular Acceleration Estimation with Off-CG Accelerometers for Incremental Nonlinear Dynamic Inversion Control’’. In: *AIAA SCITECH 2024 Forum*. 2024. DOI: 10.2514/6.2024-2566. eprint: https://arc.aiaa.org/doi/pdf/10.2514/6.2024-2566. URL: https://arc.aiaa.org/doi/abs/10.2514/6.2024-2566.

[68] Ian Reid. *Estimation II: Discrete-time Kalman filter*. Lecture Notes, University of Oxford. Hilary Term 2001. URL: https://www.robots.ox.ac.uk/~ian/Teaching/Estimation/LectureNotes2.pdf.

[69] Chen Jiang et al. ‘‘A New Adaptive H-Infinity Filtering Algorithm for the GPS/INS Integrated Navigation’’. In: *Sensors* 16.12 (2016). DOI: 10.3390/s16122127. URL: https://www.mdpi.com/1424-8220/16/12/2127.

[70] ‘‘Least square estimation-based adaptive complimentary filter for attitude estimation’’. en. In: 41 (Jan. 2019), pp. 235–245. DOI: 10.1177/0142331218755234.