Robust $\mathcal{H}_\infty$ Controller Design For INDI-Controlled Quadcopters Using Online Parameter Identification

Tom Aantjes

TU Delft

Robust $\mathcal{H}_\infty$ Controller Design For INDI-Controlled Quadcopters Using Online Parameter Identification

Thesis Report

by

Tom Aantjes

to obtain the degree of Master of Science
at the Delft University of Technology
to be defended publicly on November 5, 2025 at 9:00

Thesis committee:

Chair:	Dr. Salua Hamaza
Supervisors:	Dr. Ewoud Smeur
	Dr. Spilios Theodoulis
	Ir. Till Blaha
External examiner:	Dr. Alessandro Bombelli
Place:	Faculty of Aerospace Engineering, Delft
Project Duration:	December, 2024 - November, 2025
Student number:	5065917

An electronic version of this thesis is available at https://repository.tudelft.nl/.

Faculty of Aerospace Engineering · Delft University of Technology


TU Delft
Delft
University of
Technology

Acronyms

AHA Artificial Hummingbird Algorithm
ARE Algebraic Ricatti Equation

CA Control Allocation

DE Differential Evolution

FDD Fault Detection and Diagnosis
FTC Fault Tolerant Control

GA Genetic Algorithm
GPI Generalized Policy Iteration
GPU Graphical Processing Unit
GWO Grey Wolf Optimization

HJB Hamilton-Jacobi-Bellman
HJI Hamilton-Jacobi-Isaacs

IAE Integral Absolute Error
IMU Inertial Measurement Unit
INDI Incremental Nonlinear Dynamic Inversion
IQC Integral Quadratic Constraint
IRL Integral Reinforcement Learning
ISE Integral of the Squared Error
ITAE Integral Time Multiple Absolute Error

LMI Linear Matrix Inequality
LMS Least Mean Squares
LPV Linear Parameter-Varying
LQ Linear Quadratic
LQR Linear Quadratic Regulator
LQT Linear Quadratic Tracking
LS Least Squares
LTI Linear Time Invariant

MS Mixed Sensitivity

NCF Normalized Coprime Factor
NCP Normalized Coprime Factorization
NDI Nonlinear Dynamic Inversion
NN Neural Network

OPFB Output-Feedback

PD Proportional Derivative
PI Policy Iteration
PID Proportional-Integral-Derivative
PSO Partical Swarm Optimization

RL Reinforcement Learning
RLS Recursive Least Squares
RMS Root-Mean-Square

SGA Standard Genetic Algorithm
SPUA Simultaneous Policy Update Algorithm

UAV Unmanned Aerial Vehicle

VFA Value Function Approximation

ZOH Zero Order Hold

Contents

1 Introduction **1**

2 Literature Review **2**
2.1 Incremental Non-Linear Dynamic Inversion . . . 2
2.2 Optimal Control . . . 4
2.3 Robust Control . . . 5
2.4 Weighting Filter Optimization . . . 9
2.5 Optimal and Robust Control with Reinforcement Learning . . . 9
2.6 Gain-Scheduling . . . 11
2.7 Auto-Tuning . . . 13
2.8 Conclusion and Discussion . . . 14

3 Research Questions **17**

4 Robust $\mathcal{H}_\infty$ Controller Design for INDI-Controlled Quadcopters Using Online Parameter Identification **18**
4.1 Introduction . . . 18
4.2 System Model . . . 19
4.3 Attitude Control Design Setup . . . 25
4.4 Control Design and Analysis for a Symmetric Quadcopter . . . 28
4.5 Symmetric Quadcopter Results . . . 33
4.6 Gain-Scheduled Controller Design and Analysis . . . 37
4.7 Gain-Scheduled Controller Results . . . 40
4.8 Conclusion and Future Work . . . 44
4.9 References . . . 44

5 Conclusions & Recommendations **47**
5.1 Conclusions . . . 47
5.2 Recommendations . . . 48

References **48**

1

Introduction

Unmanned Aerial Vehicles (UAVs) have become increasingly popular due to their versatility, agility, and wide range of applications, from aerial photography and surveillance to search and rescue operations [1], [2]. However, the performance and stability of these systems heavily depend on the precise tuning of their control parameters, which can be a challenging task. In particular, ensuring robustness against uncertainties or disturbances makes controller synthesis a time-consuming endeavor requiring experienced engineers.

Recent work by Blaha [3] has shown that it is possible for a quadrotor to identify its actuator model with onboard compute resources in less than a second. This allows for the recovery of a quadrotor from a throw without any preprogrammed gains or system parameters by inverting the identified model using an Incremental Nonlinear Dynamic Inversion (INDI) controller. This has been further extended to include arbitrary orientation of the Inertial Measurement Unit (IMU), creating a flexible system for a large set of drone configurations [4]. However, the tuning of inner and outer loop gains is performed based on an approximate pole-placement procedure with no explicit robustness or performance guarantees.

If this procedure could be extended to include automatic, robust determination of an outer-loop controller, it could provide a versatile and complete control solution. This would allow for good, robust performance for arbitrary multicopter configurations without requiring prior knowledge of the system. To this end, this research focuses on designing a gain-scheduled controller using signal-based $\mathcal{H}_\infty$ closed-loop shaping techniques that can achieve robust performance for quadcopters with a wide range of actuator properties.

This report starts with a literature review on robust control methods, INDI control and other relevant concepts for controller synthesis in Chapter 2. From the research gaps identified in the literature review, the research questions are formulated in Chapter 3. The scientific article describing the modelling, controller synthesis and results based on nonlinear simulations and experiments can be found in Chapter 4. Finally, the conclusions and recommendations are presented in Chapter 5.

2

Literature Review

This chapter presents a literature review focused on INDI control for multirotors and robust control methods, from which a research gap and subsequent research objective and questions can be identified. Section 2.1 explores the application of INDI to quadrotors, highlighting key developments in the field. In Section 2.2, advancements in optimal control and their practical applications are examined. Section 2.3 delves into the $\mathcal{H}_\infty$ control problem and discusses various solution strategies. The focus then shifts to optimization techniques for the automatic tuning of weighting filters in optimal and robust control, presented in Section 2.4. The application of gain-scheduling in robust control is addressed in Section 2.6. Finally, Section 2.7 reviews auto-tuning procedures for Proportional-Integral-Derivative (PID) control, followed by a discussion of the literature and the identification of research gaps in Section 2.8.

2.1. Incremental Non-Linear Dynamic Inversion

A challenge in quadrotor control is the nonlinear nature of the system dynamics, which can cause certain control strategies to perform poorly when these nonlinearities become significant or when external disturbances such as wind gusts are encountered [5], [6]. One approach to deal with this is to employ model-based methods, such as Nonlinear Dynamic Inversion (NDI). These techniques generate a linearizing control law by inverting a model of the quadrotor dynamics [7], allowing the required control inputs to achieve a desired system state to be computed directly. However, the effectiveness of this approach is sensitive to model inaccuracies, and practical limitations such as sensor constraints or cost considerations can make obtaining an accurate model challenging [7].

An incremental form of NDI, known as INDI, has been developed to reduce model dependence [8], [9]. Instead of modeling the angular acceleration based on the system state, this approach measures the current angular acceleration and computes an increment of the control input to achieve the desired change. A key advantage is that the onboard model only requires the plant control effectiveness. Sieberling et al. [8] demonstrate that this method allows errors arising from unmodeled dynamics or modeling inaccuracies to be directly measured and compensated. Pollack and van Kampen [10] further show that INDI is more robust against regular perturbations in the output dynamics compared to NDI.

A challenge of INDI is that angular acceleration cannot be directly measured, so angular rate measurements from the gyroscope must be differentiated. This significantly increases noise effects and necessitates the use of a filter on the measured angular rates [9], [11]. As a result, the measured output state and actuator state are not perfectly synchronized, which can degrade closed-loop performance [11]. This is also referred to as the synchronization effect. Further analysis of the open-loop responses by Pollack and van Kampen [10] for both NDI- and INDI-based controllers reveals that INDI exhibits higher open-loop gain at low frequencies and higher crossover frequencies. This indicates reduced robustness at high frequencies due to insufficient open-loop roll-off [10], making INDI sensitive to measurement noise, unmodeled high-frequency dynamics, and time delays [8]. Additionally, the influence of actuator and filter dynamics [12], as well as inaccuracies in control effectiveness matching [13], on the time delay margin has been reported. Taken together, these characteristics highlight both the strengths and limitations of INDI, showing that while it im-

proves robustness against regular output perturbations compared to NDI, this comes with reduced robustness elsewhere.

Smeur et al. [9] solved the synchronization effect issue by applying the synchronization filter $H(s)$ used on the differentiated gyroscope outputs to the incremented inputs. As a consequence, the inner angular acceleration control loop reduces to the actuator dynamics $A(z)$, as shown in Figure 2.1. This controller also includes gains for the attitude η and angular rate Ω. An important observation is that this result applies to the ideal inversion case, where the control parameters are perfect and no disturbances are present. In non-ideal cases, the performance and robustness will depend on the filter characteristics.

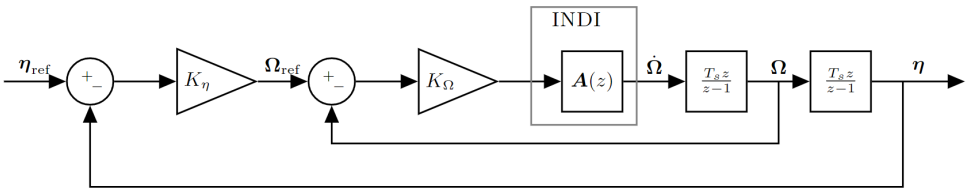


Figure 2.1: The design of the attitude controller based on the closed-loop response of the INDI controller, extracted from [9].

Although INDI reduces model dependence compared to NDI, nominal coefficients for the control effectiveness parameters are still required to compute the incremental control inputs. A challenge is that the effectiveness parameters are not constant during flight, due to external factors such as varying voltage or aerodynamic conditions. To address this, Smeur et al. [9] propose an adaptive approach that estimates the control effectiveness parameters online using the Least Mean Squares (LMS) algorithm [14]. Their results demonstrated that this method enables the controller to maintain consistent performance even when the moment of inertia changes. Furthermore, the study showed that this adaptive method offers a general advantage, as it can account for and identify performance differences between the actuators themselves.

Later work by Smeur et al. [15] presents an adaptive, cascaded control system that includes both the attitude inner loop described previously and an outer position loop based on INDI. In this work, the control effectiveness matrix from [9] was extended with an additional parameter that predicts changes in thrust based on actuator inputs. It was shown that, using the LMS algorithm, the estimated parameters converge to the control effectiveness values determined offline. Furthermore, the study demonstrated that the system exhibited a faster response to wind disturbances compared to a PID-controlled quadrotor.

Blaha [3] developed a method that enables a quadrotor, under the assumptions that the IMU orientation is known and the four rotors point upwards, to recover from a throw without any prior knowledge of the control effectiveness or motor parameters. To achieve this, the control effectiveness parameters needed to be identified significantly faster than possible with the LMS method presented in [9], [15]. This was accomplished by employing the Recursive Least Squares (RLS) algorithm in combination with an excitation sequence that ensures the gyroscope measurements remain within their sensing limits. Simulations showed that the Root-Mean-Square (RMS) errors between the true and estimated parameters are typically below 10%.

The inner and outer loop gains were tuned by leveraging the collapse of the inner loop to the actuator dynamics, as shown in Figure 2.1 [9]. This approach allowed for the successive application of a pole-placement technique to achieve the desired damping ratios for each loop. The study reports that these gains provide adequate performance for stabilizing the quadrotor and controlling its position; however, no evaluation of trajectory or attitude tracking performance is provided.

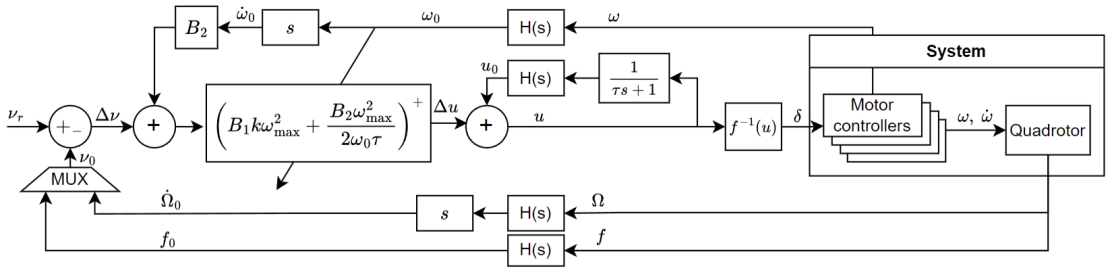


Figure 2.2: INDI inner loop similar to [9] with linearized motor acceleration dynamics, extracted from [3].

Blaha extended his work to include the online estimation of the IMU position relative to the center of gravity, its orientation, and the thrust direction of all motors [4]. It was shown that performing two throw-and-catch experiments allows for accurate estimation of the IMU location and orientation, after which the method described in [3] can be used to recover the quadrotor. Additionally, RLS was employed to determine the optimal thrust directions that minimize rotations. This enables position control under the assumption that the IMU alignment is restricted to yaw-free rotations. Furthermore, the approach allows the most efficient hover direction to be determined for overactuated systems such as hexarotors. Overall, this generalizes the ability of a UAV capable of static hover to determine its control effectiveness.

These studies highlight several important properties of INDI in the context of quadrotors. In particular, under ideal conditions, INDI exhibits linearizing properties that allow the inner angular acceleration loop to be effectively reduced to the actuator dynamics. This means the nominal system could be characterized purely by its actuator properties. Moreover, adaptive implementations of INDI enable online identification of all relevant system parameters, which could be used to determine a suitable robust controller.

2.2. Optimal Control

Optimal control methods aim to minimize a performance cost function. For nonlinear systems or complex cost functions, this problem is generally difficult to solve analytically. However, for linear systems with quadratic performance indices, well-established solution methods exist [16]. This section discusses these Linear Quadratic (LQ) methods and their applications to quadrotors.

One such method is the state-feedback Linear Quadratic Regulator (LQR), which yields a controller that minimizes a quadratic cost function and typically consists of terms that penalize deviations from desired states and control effort [17]. When the state-space model of the plant is known, the optimal control law can be obtained by solving a single Algebraic Ricatti Equation (ARE). Srivastava et al. [18] applied the LQR method with full-state feedback for a quadrotor based on a linearized hover model. By augmenting the A and B matrices of the model, they also presented an integrator LQR approach that addresses steady-state errors. The selection of the Q and R matrices directly affects the resulting performance. It was found that the yaw-axis control performance was particularly sensitive to the choice of Q and R compared to the other axes.

Foehn and Scaramuzza [19] applied the LQR method to the control of agile quadrotors by formulating a state-dependent LQR. This approach unifies the translational and rotational motor dynamics of the quadrotor, which are often neglected but are crucial for agile control. The full system is linearized at each time step, after which the state-feedback ARE is solved. These computations were performed at approximately $10\,\text{Hz}$ on a small quadcopter platform, enabling optimal control during trajectory tracking.

In many cases, full state feedback might not be desirable or even feasible, which is why research has been conducted on Output-Feedback (OPFB) control. Solving this problem is significantly more difficult than the state-feedback case, as it requires solving three coupled, nonlinear matrix

equations involving three unknowns [17]. Gadewadikar et al. [20] provided a procedure for obtaining an OPFB controller that requires solving only two coupled matrix equations. This is combined with an iterative algorithm that can efficiently obtain a solution to the OPFB control problem for LQR without the need for initial stabilizing gains. This method is also applicable to $\mathcal{H}_\infty$ controllers which will be discussed in later sections.

An extension of LQR is Linear Quadratic Tracking (LQT), which considers tracking a trajectory rather than merely stabilizing a system at an equilibrium point [21]. This approach incorporates a reference signal into the cost function, enabling improved tracking performance in dynamic environments. The resulting control law consists of a static state- or output-feedback component and a feedforward component that depends on the output of the closed-loop plant when driven by the reference. Koksal et al. [22] applied LQT to a quadrotor in combination with an adaptive algorithm to compensate for inertial uncertainties.

2.3. Robust Control

After the rise of optimal controllers designed through LQR and LQT, deficiencies arose because these methods did not account for model uncertainties. Consequently, such optimal controllers could exhibit poor performance or instability during operation [23]. In response, Zames [24] formulated the $\mathcal{H}_\infty$ control problem, which considers both plant uncertainty and imposes a limit on the gain from disturbances to the plant output. The main objective is illustrated in Figure 2.3 [16].

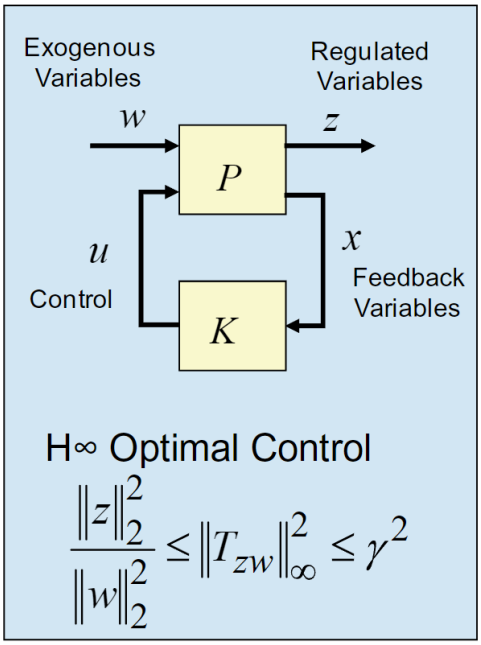


Figure 2.3: $\mathcal{H}_\infty$ optimal control block diagram, extracted from [16].

In this model, the goal is to minimize the $\mathcal{H}_\infty$ norm of the transfer functions from a disturbance w to performance output z ($T_{w\to z}(P,K)$) through a stabilizing K.

Doyle et al. [25] solved the $\mathcal{H}_\infty$ problem by showing that it requires the solution of two AREs. In the state-feedback case, this reduces further to a single ARE. Later, Gahinet and Apkarian [26] revisited the problem using Linear Matrix Inequalities (LMIs). A major drawback of these solutions is that the resulting controllers are full-order, which makes implementation difficult for many systems [27]. Implementing lower-order controllers proved challenging because the optimization problem becomes non-convex [27]. Eventually, Apkarian and Noll [28] solved the $\mathcal{H}_\infty$ problem for structured controllers through a non-smooth optimization technique. The work of Schoon [29] further demonstrated that Lyapunov-based optimization methods are effective for solving the optimization problem for static output feedback controller design.

2.3.1. $\mathcal{H}_\infty$ Closed-Loop Shaping

The $\mathcal{H}_\infty$ closed-loop shaping control design method focuses on shaping the closed-loop transfer functions between the exogenous inputs w and performance outputs z of the control system [16]. One common approach for closed-loop shaping is the mixed-sensitivity method, the main principle of which is illustrated in Figure 2.4 [16]. In this example, the transfer functions for the disturbance sensitivity S, complementary sensitivity T, and control sensitivity KS are shaped using weighting filters. By performing $\mathcal{H}_\infty$ optimization, the controller is tuned such that each transfer function attains its desired shape, enabling specific performance characteristics in defined frequency ranges depending on the application. In general, selecting appropriate weighting filters is not trivial and often requires iterative design and tuning [16], [30]. Variations of this approach also exist, such as the method initially proposed by Kwakernaak [31], which only shapes the S and KS transfer functions.

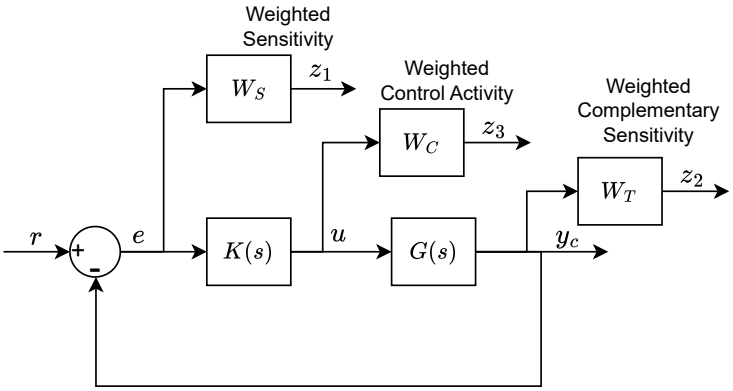


Figure 2.4: Block diagram showing weighted sensitivity, control activity, and complementary sensitivity, based on [16].

An issue highlighted by Sefton and Glover [32] is that pole–zero cancellations can occur in the transfer functions from input disturbances to error signals in the S/KS formulation. This is problematic because system stability may then depend on such cancellations, which will not occur in practice due to plant uncertainties. Tsai et al. [33] proposed a method for selecting weighting filters that prevent these cancellations and even allow for (partial) pole placement. Kwakernaak [34] also addressed this issue by introducing an additional weighting filter, $V = D_0^{-1}M$, which depends on the plant through D_0 but also enables partial pole placement via modifications to M. Later, Cao et al. [35] demonstrated that this method can yield results equivalent to the two-block Normalized Coprime Factorization (NCP) formulation developed by McFarlane and Glover [36], but with polynomial matrix formulations instead of coprime factorizations.

Huang et al. [30] applied the S/KS approach to the actuator system of a horizontal stabilizer and observed that it does not shape all important transfer functions for robust control. In particular, the transfer function SG, which represents the closed-loop behavior of the plant under disturbances, was found to amplify the effect of input disturbances on the output when the plant was poorly damped. To address this issue, a four-block formulation has been proposed, shaping the functions S, SG, KS, and KSG, thereby resolving the limitation. The method proposed by Kwakernaak [34] was also shown to be effective. However, when comparing tuning procedures, the four-block was preferred since the partially fixed structure of V complicates the construction of the weighting filters and restricts tuning flexibility.

The four-block method has also been applied to attitude control design for quadrotors with tilting rotors [37], where it effectively handled system uncertainties and achieved improved performance compared to classical controller designs. Furthermore, Biertümpfel et al. [38] applied the four-block method to a space launcher with varying control objectives. In this case, the weighting filters were defined as time-varying, incorporating a dynamic scaling factor, which enabled the computation of a time-varying control law throughout the ascent phase as the control objectives evolved.

Pineau et al. [39] applied the Mixed Sensitivity (MS) approach to the autopilot design of dual-spin projectiles using an INDI controller. For the outer loop, an S/KS MS formulation was employed, while for the inner loop an additional model-following constraint, M, was included, resulting in an $S/KS/M$ structure. This allowed a desired dynamic response to be specified, which the optimization aimed to match. In this framework, the INDI structure used for gain tuning was approximated by the actuator dynamics combined with an integrator. To represent the discrete control system more accurately, approximated Zero Order Hold (ZOH) and computational delay transfer functions were also included.

Methods that shape the complementary sensitivity function T, related to the attenuation of measurement noise and robustness against multiplicative uncertainty, also exist. An S/T approach was applied by Thomas et al. [40] to the control design of a quadcopter, demonstrating good noise rejection and tracking performance.

Hachem et al. [41] presented a signal-based closed-loop shaping method to design a structured outer-loop $\mathcal{H}_\infty$ controller for a quadcopter using INDI control. Simulation and experimental results showed that this approach significantly improved disturbance rejection in both attitude and position tracking compared to a conventional PD controller.

2.3.2. $\mathcal{H}_\infty$ Open-Loop Shaping

In addition to closed-loop shaping, an $\mathcal{H}_\infty$ open-loop shaping method has also been proposed by McFarlane and Glover [42]. This approach employs a precompensator W_1 and a postcompensator W_2 to shape the singular values of the plant in open-loop, G, as illustrated in Figure 2.5.

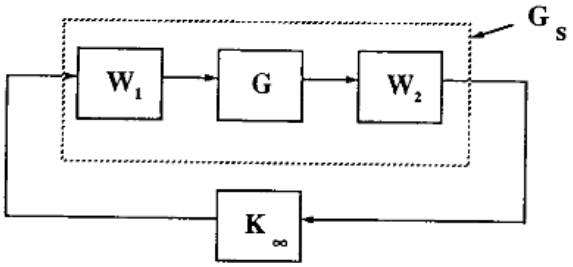


Figure 2.5: Shaped plant in the $\mathcal{H}_\infty$ loop-shaping design process, extracted from [43].

The shaped plant, G_S, is then robustly stabilized using $\mathcal{H}_\infty$ optimization with respect to coprime factor uncertainty. This methodology provides robust stabilization not only for the nominal plant but also for a family of perturbed plants. The lowest achievable γ, and thus the maximal achievable robustness margin, can be determined using the state-space model as described by McFarlane and Glover [36]. The controller can be computed from the solution of two AREs [44]. The choice of W_1 and W_2 enables a trade-off between performance and robustness. By finally combining W_1 and W_2 with the synthesized controller, the desired closed-loop controller can be implemented.

Chen et al. [45] implemented $\mathcal{H}_\infty$ loop shaping for quadrotor control in both position and attitude, and validated the approach using nonlinear simulations. They demonstrated that this procedure achieves good robustness, disturbance rejection, and tracking performance. Similar results were obtained by Turkoglu and Jafarov [46], who also compared the loop shaping approach to classical PID control. Their study showed that the loop shaping method yields significantly reduced overshoot and improved settling times for both disturbance rejection and tracking performance.

Li et al. [47] applied this technique to a quadcopter with tilting rotors, enabling full control of six degrees of freedom. In this case, a linear controller obtained via $\mathcal{H}_\infty$ loop shaping was combined with a control allocation scheme to achieve robust six-degree-of-freedom control. Compared to the PID controller, the loop-shaped controller exhibited much better robustness against external disturbances, sensor noise, actuator delays, and model parameter variations. However, tracking

performance was somewhat degraded, which was attributed to the loop shaping controller being systematically tuned for robustness, as opposed to the manual tuning of the PID controller.

Cardenas [48] applied $\mathcal{H}_\infty$ open-loop shaping to the inner attitude loop of a quadrotor using an INDI control system. In this case, an idealized inversion, as described in Figure 2.1, was assumed, meaning that the robustification was performed against uncertainty in the actuator dynamics and the idealized dynamics. The results showed that the loop-shaped controller achieved larger stability margins and improved disturbance rejection compared to a manually tuned Proportional Derivative (PD)-INDI structure.

2.3.3. Comparison of $\mathcal{H}_\infty$ Open and Closed-Loop Shaping

Bates [23] discusses some disadvantages of MS compared to the previously discussed loop-shaping. One is that the optimal γ has to be computed iteratively as opposed to being computable directly, since the robustness margins are determined a priori when using loop-shaping. Additionally, while MS can only account for additive and multiplicative uncertainty, open-loop-shaping can obtain robustness with respect to Normalized Coprime Factor (NCF) uncertainty. This also provides balanced robustness and performance properties at the plant input and output. As opposed to the S/KS MS method, no pole-zero cancellations are present in loop shaping. This can be avoided in MS but would require additional weighting functions which complicate the design process as more parameters must be chosen.

In the work of Huang et al. [30], a comparison was made between controllers obtained using open- and closed-loop shaping techniques. Four different methods were applied for the controller design of a horizontal stabilizer: the four-block method, loop shaping, the S/KS method, and a modified adaptation of S/KS. Overall, the performance achieved with each methodology was comparable, with no single approach proving conclusively superior, suggesting that both open- and closed-loop methods can yield satisfactory results. Ultimately, the four-block MS approach was preferred, as it enables direct influence on closed-loop performance through adjustments of the weighting filters.

In contrast, open-loop shaping cannot directly affect KS, SG, or KSG; instead, it relates closed-loop objectives to the open-loop singular values of the compensated system [43]. Consequently, closed-loop performance requirements are expressed in terms of the low- and high-frequency characteristics of the open-loop system. Closed-loop shaping, on the other hand, explicitly considers the entire frequency domain [43]. This approach also provides additional opportunities, such as directly constraining disk-based stability margins, which are related to the sensitivity and co-sensitivity functions [49]. For example, Theodoulis et al. [50] demonstrate the use of such constraints to optimize disk margins separately at the plant input, output, and combined inputs and outputs.

2.3.4. μ-synthesis

The structured singular value μ, introduced by Doyle [51], allows for describing and analyzing structured uncertainty. However, no method exists to directly synthesize a μ-optimal controller [44]. A combination of $\mathcal{H}_\infty$ synthesis and μ-analysis, known as DK-iteration, is available and enables modeling of complex disturbances. Hamza et al. [52] demonstrated that a μ controller combined with a disturbance observer improved performance under large parametric uncertainties compared to a full-state-feedback disturbance observer controller and a standalone μ controller. Another trajectory tracking controller for quadrotors based on μ-synthesis was compared to an LQR solution, which demonstrated superior performance under uncertainty [53]. Pollack [54] introduced a multi-objective structured $\mathcal{H}_\infty$ synthesis routine based on hybrid DK-iteration. This was applied on a NDI, INDI and hybrid-INDI control structure.

Masuda and Uchiyama [55] used μ-synthesis for the controller design of a quad tilt-wing UAV. Using this approach, the controller was synthesized to achieve robust stability under structured parameter uncertainties. A primary advantage over $\mathcal{H}_\infty$ synthesis is that robust stability and performance can be explicitly accounted for. Panza et al. [56] described the synthesis of a controller using μ-synthesis for a similar platform, considering unstructured uncertainties. This demonstrated improved robust stability and performance compared to a baseline controller.

2.4. Weighting Filter Optimization

As discussed in the previous sections, weighting filters play a crucial role in achieving robust and optimal control performance. Therefore, their selection requires careful consideration. As the goal is to obtain a robust controller for a wide range of multicopter configurations, choosing weighting filters applicable to all drones might not be possible. As a consequence, methods for automatically optimizing these weighting filters would be valuable for any approach. This section presents several methods proposed for the automatic tuning of weighting filters.

Ata and Gencal [57] compared several optimization approaches for the LQR design of a quadrotor, including Standard Genetic Algorithm (SGA), Differential Evolution (DE), Partical Swarm Optimization (PSO), and Grey Wolf Optimization (GWO). The Integral of Absolute Error (Integral Absolute Error (IAE)) was used as the performance indicator to directly penalize tracking errors. Their results show that the SGA approach did not produce satisfactory performance, whereas the other algorithms achieved acceptable results. However, no discussion was provided regarding the computational efficiency of these methods. Given that stochastic optimization algorithms typically require numerous simulation iterations, it is likely that they are unsuitable for real-time implementation. This inference is supported by the work of Feyel et al. [58], who optimized the weighting parameters for an $\mathcal{H}_\infty$ loop-shaping procedure using PSO. While their results were promising, the optimization required approximately 20 hours of computation, which the authors noted to be faster than manual tuning by a control engineer.

Kanokmedhakul et al. [59] conducted a comparative study of metaheuristic approaches for tuning the weighting filters in a UAV control system. Among the eleven methods evaluated, the Artificial Hummingbird Algorithm (AHA) and GWO algorithms demonstrated the best performance. However, the study did not report the computational times of the compared approaches.

Thomas et al. [40] presented an automated framework for designing weighting filters for MS $\mathcal{H}_\infty$ control applied to a quadrotor. Rather than relying on manual tuning, the performance weighting filters were automatically optimized through a constrained nonlinear minimization problem implemented using the MATLAB OPTI Toolbox. The optimization process adjusts the sensitivity weights by minimizing a cost function that balances the IAEs of the output and control effort, while ensuring that robust stability constraints are satisfied. Compared to traditional trial-and-error methods, this approach streamlines the controller design process and yields controllers with robust performance, characterized by $\mathcal{H}_\infty$ norms well below unity. It also significantly reduces the required tuning time and effort.

Perez et al. [60] introduced a practical framework for the co-design of weighting filters and controller gains. In this approach, the S/KS MS filters are optimized by defining them as tunable gains. This means the optimizer considers them part of the controller and will optimize them based on the set $\mathcal{H}_\infty$ constraints. Compared to manually tuned weighting filters based on pole placement, the co-design approach demonstrates improved performance across multiple performance metrics. The resulting $\mathcal{H}_\infty$ autopilots exhibited shorter settling times, reduced control effort, and enhanced robustness to plant uncertainties. In contrast, the pole-placement method, although providing strong stability margins, led to longer disturbance rejection times, higher control amplitudes, and actuator rates approaching saturation. Furthermore, the co-design approach minimized oscillations under uncertain conditions and maintained a more stable response in turbulent environments. Overall, integrating weighting filter optimization into the controller design process offers a more systematic and efficient tuning methodology. This co-design strategy has also been extended to gain-scheduling problems [61], as discussed in Section 2.6.

2.5. Optimal and Robust Control with Reinforcement Learning

Even though methods for solving the $\mathcal{H}_\infty$ problem have been well established, they generally require the availability of a system model. In cases where the system model is unavailable or may change during operation, adaptive optimal controllers can be of interest [62]. One approach to this problem is the use of Reinforcement Learning (RL) to address design challenges associated with the previously described optimal and robust control problems. This section will discuss the relevant developments and methodologies employed in this context.

One RL technique derived from dynamic programming is Policy Iteration (PI) [63], which employs a two-step iteration consisting of policy evaluation and policy improvement. Kleinman [64] demonstrated that PI can be used to iteratively solve the ARE by solving a Lyapunov equation and updating the control gain. It should be noted that this approach requires an admissible input policy that is stabilizing and full knowledge of the system dynamics. Lian [62] extended this method to the LQT problem by formulating the LQT Lyapunov equation, allowing the PI iteration to solve the LQT problem. When the full system dynamics are known, this enables the iterative algorithm to converge to the optimal solution.

Vrabie et al. [65] developed a method to solve the LQR problem without knowledge of the system dynamics, using an adaptive critic structure. In this approach, the critic evaluates the infinite-horizon cost for a given controller, and the actor parameters are subsequently updated to reduce this cost. The iterative procedure uses the current state, x, and eliminates the need for the plant matrix A in the computation, while convergence to the solution of the ARE is still guaranteed. To determine the value of the critic, $V_i(x_i)$, a Least Squares (LS) problem is solved once a sufficient number of data points have been collected under appropriate excitation. Alternatively, a recursive estimation algorithm, such as gradient descent or RLS, can be used. The latter is preferred, as it does not scale computationally with the state-space dimension. This method was applied to finding the optimal controller for load-frequency control of a power system, achieving convergence in approximately 7 s. Accurate measurement of the full state is required, and an initial stabilizing controller must be present for the algorithm to function correctly.

This method was extended by Modares and Lewis [66] to the LQT problem. The main development is the formulation of a Bellman equation with augmented states that account for the reference trajectory. This formulation enables the setup of a LQT ARE, which can be solved offline using an algorithm similar to Kleinman's method [64], but incorporating a discount factor. Using Integral Reinforcement Learning (IRL), the LQT problem can then be solved online in a manner analogous to the previously described approach. An initial admissible policy is still required for convergence. The method assumes that the reference trajectory is generated by a linear command generator, $\dot{y}_d = Fy_d$, which limits the class of reference trajectories. Nevertheless, unit step inputs or sinusoidal reference signals are possible. It has been shown that the matrix F does not need to be known for the algorithm to converge.

Later, Modares et al. [67] further extended this method to the OPFB case. In addition, the LQR and LQT problems are unified through the use of a discounted ARE, with a specified limit on the discount factor beyond which stability is not guaranteed. This approach is implemented via reconstruction of the state and Value Function Approximation (VFA) using only measured output data. Methods are provided for performing this reconstruction both with and without knowledge of the system dynamics matrices A, B, and C.

For the nonlinear case, Mclain [68] provides an algorithm for solving the Hamilton-Jacobi-Bellman (HJB) equation for the nonlinear optimal regulation problem using a successive approximation technique. Essentially, this approach transforms the HJB equation into a sequence of linear partial differential equations, for which the author provides an approximate solution using Galerkin's global approximation method. The method is also extended to the Hamilton-Jacobi-Isaacs (HJI) equation, where the disturbance and control policies are updated iteratively. Convergence to the true solution is proven; however, it should be noted that this is done successively. For a given control strategy, the disturbance is iteratively updated until the worst-case scenario is found, after which the control is updated to improve performance for that disturbance. This procedure can lead to redundant iterations, resulting in poor computational efficiency [69]. Abu-Khalaf et al. [70] describe a method for solving the $\mathcal{H}_\infty$ regulation problem for nonlinear systems with input constraints. This is achieved by formulating a HJI equation that incorporates input constraints and applying a similar iterative scheme to that of [68]. As with the previous approach, nested loops are still required.

Vamvoudakis and Lewis [71] developed a synchronous policy iteration scheme that updates the disturbance and control policies simultaneously. This is achieved by approximating the value function with a Neural Network (NN), in combination with separate NNs representing the control action and the disturbance. By adapting the Generalized Policy Iteration (GPI) approach introduced

by Sutton and Barto [72], all three NNs can be tuned simultaneously. Tuning algorithms for the NNs are provided, and convergence to the solution of the HJI equation is proven. The synchronous updating scheme allows this algorithm to be executed in real-time. A limitation, however, is that a model of the system dynamics is required beforehand.

Wu and Luo [69] developed a PI algorithm for solving the $\mathcal{H}_\infty$ regulation problem using a Simultaneous Policy Update Algorithm (SPUA). Similar to the method of Vamvoudakis and Lewis [71], both the control and disturbance policies are updated simultaneously. However, a reformulation removes the dependency on the internal system dynamics, $f(x)$, by replacing it with online measurements of the states. It is proven that the solution still converges to the optimal control solution, although accurate measurement of all system states is required. Simulations show that the NNs converge in approximately 3 s, after which the $\mathcal{H}_\infty$ controller can be computed.

2.6. Gain-Scheduling

The goal in this work is to obtain a robust controller that can be utilized across a wide range of platforms. One potential solution is the use of gain-scheduled controllers, which will be described in this section. In general, any gain-scheduling technique follows the same four steps [73]. First, a linear parameter-varying representation of the plant must be computed, which can be achieved, for example, through Jacobian linearization of the nonlinear plant at a set of relevant operating points [73]. Next, linear design methods are applied to these models, resulting in a set of controllers corresponding to each design point. These controllers are then implemented as a function of certain scheduling variables to cover the entire operating domain.

Two main approaches can be distinguished. Classical gain-scheduling techniques, which are widely used, synthesize controller gains over a range of operating points and implement them through either interpolation or a defined gain surface [74]. The alternative approach is to define a scheduled controller directly as a function of the scheduling variables, referred to as Linear Parameter-Varying (LPV) control. In this section, both methodologies and their application to drone control will be discussed.

In classical gain scheduling, a finite set of Linear Time Invariant (LTI) controllers is interpolated based on the current operating point of the plant. Such a gain-scheduling approach has been applied to Fault Tolerant Control (FTC) of a quadrotor by Milhim et al. [75]. In this approach, sets of gains for specific fault conditions were manually tuned. These fault conditions were defined using a parameter that describes the loss of actuator effectiveness. Interpolation of this gain schedule was shown to maintain performance even when the actuators lose effectiveness. Sadeghzadeh et al. [76] combined this gain-scheduling approach with a Fault Detection and Diagnosis (FDD) scheme to automatically detect faults online, demonstrating good performance under fault conditions. Qiao et al. [77] further applied this approach to scenarios where a payload is dropped. In these methods, the fault scenarios had to be predefined, and the controller manually tuned. Moreover, the time between fault detection and gain adaptation needed to be sufficiently short, as excessive delay during the transition could cause the quadrotor to become unstable or crash. The reliance on manual tuning for specific operating and fault conditions also imposes significant design effort, particularly if the optimal gain schedule is highly nonlinear.

Building on these concepts, Nguyen et al. [78] incorporated $\mathcal{H}_\infty$ methods into gain-scheduled controllers to ensure robust performance under actuator failure conditions. Rather than relying on a predefined lookup table, the controller gains were modeled as bilinear functions of an effectiveness loss factor for each motor. MATLAB's `systune` function was then used to determine the $\mathcal{H}_\infty$-optimal parameters of the gain surface. This approach was subsequently extended to hexacopters [79] through a self-scheduling Control Allocation (CA) scheme, also based on actuator effectiveness loss. The methodology was further generalized to arbitrary multicopter configurations [80] by introducing the concept of virtual control effectiveness, enabling consistent tuning across different vehicle types. This formulation effectively replaces individual motor performance with virtual control effectiveness parameters.

Another perspective on gain-scheduling focuses on optimizing control structures under uncertainty. Lhachemi et al. [81] developed an $\mathcal{H}_\infty$ self-scheduled longitudinal flight control system

that accounts for uncertainties in mass and center-of-gravity location. In this case, point-by-point gain-scheduling was performed, after which it was found that a quadratic polynomial scheduling function is a reasonable shape. Then the variables in this scheduling function were optimized for a set of linearized plants representing the flight envelope. It was observed that this interpolation increased the global $\mathcal{H}_\infty$ norm of the controller from 1.48 to 3.86. By using the same quadratic structure but globally optimizing its parameters with `hinfstruct`, the global $\mathcal{H}_\infty$ gain was reduced to 1.56. However, improving performance at one operating point resulted in degraded performance at another. To address this trade-off, a normalized gain-scheduling technique was introduced, in which each design point was weighted by its initial $\mathcal{H}_\infty$ value. This approach improved performance at previously degraded operating points but slightly reduced it at the other end of the flight envelope, suggesting that further tuning is required.

Gain-scheduling techniques have also been applied to address nonlinear flight dynamics. Theodoulis et al. [82] employed MATLAB's `systune` toolbox to design a four-loop autopilot for a highly agile missile. In this case, gain-scheduling was used to manage the nonlinearities in the flight dynamics by adjusting the controller parameters based on the vehicle's position within the flight envelope. The design successfully satisfied both hard and soft performance constraints; however, the resulting gain surface was non-smooth, which the authors identified as a topic for future work. Simulation results further demonstrated that the controller maintained consistent performance under aerodynamic parameter uncertainties. In a subsequent study, Theodoulis and Proff [50] proposed a systematic procedure for gain-scheduling of missile autopilots, again leveraging the non-smooth optimization capabilities of `systune`. The approach allows either pointwise (a posteriori) tuning of individual operating points or global (a priori) optimization of the entire gain surface. Simulation results showed that the resulting controllers were robust to aerodynamic uncertainties, with both methodologies achieving comparable performance.

Finally, Rhenman and Theodoulis [61] developed a structured parametric $\mathcal{H}_\infty$ gain-scheduled controller for the ADMIRE aircraft, employing a two-stage feedback and feed-forward design to enhance pitch rate control. The design process incorporates co-design of the weighting filter parameters by applying the co-tuning method proposed in [60]. In this approach, a soft requirement is imposed on the weighting filters, allowing their parameters to be included directly in the optimization problem. Three controllers were synthesized, with a multi-model approach used to account for parameter variations in two of them. The first controller considers only the nominal model, the second applies the multi-model approach in the first stage, and the third uses it in both stages. The results indicate that employing the multi-model approach reduces nominal performance but improves robustness.

The classical gain scheduling approach does have some limitations as no guarantee on global stability or performance can be provided [73], [83]. In principle, the stability and flight performance characteristics are only guaranteed around the trim points. This can be overcome using LPV control, in which the controller is explicitly parameterized as a function of measurable scheduling variables [73]. In this case, the controller is still linear but the dynamics of the plant and its controller depend on a time-varying parameter.

The LPV control approach has been successfully applied to quadrotor tracking by Estrada et al. [84], demonstrating a robust controller capable of detecting sensor faults. Additional applications include optimal control using LPV methods [85], and the work of Rotondo et al. [86], which applied LPV-based reference tracking and showed strong performance in nonlinear simulations. This approach was further extended to account for actuator faults in the LPV framework [87]. In these studies, the controller was synthesized using LMI-based $\mathcal{H}_\infty$ optimization, ensuring regional pole placement and robustness against bounded uncertainties. More recently, Campos et al. [88] designed an adaptive $\mathcal{H}_\infty$ LPV controller for a convertible UAV, capable of handling transitions between hover, cruise, and forward flight.

Simões and Cavalcanti [83] addressed the problem of gain-scheduling for a missile autopilot that remains robust in the presence of uncertain operators, including those that are nonparametric or nonlinear dynamic uncertainties. Their method allows for specifying both the controller structure and the parameterization of the gain-scheduling function. Robustness to uncertain operators is achieved using Integral Quadratic Constraint (IQC) theory [89], which can capture behaviors such

as rate-bounded time-varying parameters and slope-restricted nonlinearities. A key advantage of this structured approach is that it enables robust performance while preserving flexibility in the controller order. Consequently, this method can achieve robust performance even under computational hardware constraints.

Vinco et al. [90] explored a strategy to reduce the computational complexity of LPV controller synthesis by reformulating nonlinear projectile dynamics as a polytopic system. In this approach, the number of scheduling parameters typically used in LPV design is reduced by first rewriting the standard LPV model as a quasi-LPV system, in which certain nonlinear parameter-varying terms are hidden. The quasi-LPV system is then transformed into a polytopic representation by expressing it as affine in the scheduling variables. This procedure requires an analysis of the system dynamics to determine suitable new scheduling functions. The resulting reduction in scheduling variables significantly lowers the computational complexity of controller synthesis. It should be noted, however, that this approach introduces conservativeness in the optimization, which can be mitigated through careful selection of the ranges for the scheduling functions. Chaabani and Azouz [91] applied a similar methodology for the control of a quadrotor with pivoting rotors. Here, a polytopic modeling technique was used to capture the highly nonlinear dynamics during take-off, and the effectiveness of the controller was demonstrated.

2.7. Auto-Tuning

Given the PD-like structure of the INDI controller under consideration, it can be of interest to examine auto-tuning methods developed for traditional control systems. One of the earliest and most well-known tuning methods for PID controllers is the Ziegler–Nichols method [92]. This heuristic approach determines the PID parameters based on the system's response to sustained oscillations, providing a straightforward means of achieving reasonable performance in many applications. However, it does not guarantee optimal robustness or performance in the presence of model uncertainties and external disturbances.

An alternative approach to PID auto-tuning is based on loop-shaping principles, as demonstrated by Gaikwad et al. [93]. This method is analogous to the $\mathcal{H}_\infty$ loop-shaping procedure, where the objective is to shape the closed-loop transfer function to achieve desired frequency-domain characteristics. The system is excited with an external signal in the frequency region around the target bandwidth, and a RLS algorithm is applied to online data to fit the PID gains to the desired loop shape. This approach enables adaptive tuning in real time, providing improved performance compared to static tuning methods. However, it requires sufficient system excitation and does not explicitly provide robustness guarantees in the presence of model uncertainties.

Hasseni and Abdou [94] proposed an auto-tuning method for decentralized PID controllers using a Genetic Algorithm (GA). The GA optimizes a fitness function based on the Integral of the Squared Error (ISE), which quantifies the tracking error over time. While this approach achieves good tracking performance, it does not explicitly ensure robustness against disturbances or parameter variations. Khuwaja et al. [95] extended this methodology by incorporating additional performance metrics, including the IAE and Integral Time Multiple Absolute Error (ITAE), which led to improved transient and steady-state behavior. Despite these advancements, neither method provides robustness guarantees, and their reliance on offline optimization limits applicability in dynamic or time-varying systems.

Additionally, Noordin et al. [96] investigated the use of PSO for tuning PID controllers. Although PSO-based tuning can effectively identify near-optimal control parameters, the method operates offline and does not explicitly account for robustness, limiting its suitability for systems subject to varying disturbances and uncertainties.

Moreno et al. [97] proposed a tuning procedure for a nonlinear PID-type controller. Their approach involved injecting a disturbance on one actuator and comparing performance with other control schemes such as sliding-mode control. The results indicated superior tracking performance, highlighting the potential benefits of nonlinear PID tuning in systems with significant nonlinearities. However, the method lacks generalization across different system configurations and disturbances.

More recent advancements leverage machine learning to improve controller tuning. Li et al. [98] combined Deep RL with an L_1 adaptive control algorithm to optimize controller parameters. The deep learning framework maximizes a reward function that penalizes large deviations from the reference signal, excessive overshoot, and violations of set boundary conditions. The learning process occurs offline and requires approximately 30 minutes of training on a consumer-grade Graphical Processing Unit (GPU). While this method significantly improves control performance, it does not inherently ensure robustness. To address this limitation, the authors incorporated a separate L_1 adaptive control module based on the work of Lewis [99]. The L_1 controller provides robustness guarantees by compensating for model uncertainties and disturbances, resulting in a hybrid approach that combines adaptive learning-based optimization with robust control principles.

Fuzzy logic control has also been explored as a method for automatic gain tuning. Abdelmaksoud et al. [100] implemented a self-regulating fuzzy PID controller to enhance disturbance rejection in a quadcopter. Essentially, this approach functions as a gain-scheduling scheme, adjusting the controller gains based on the current state of the quadcopter. However, it requires initial selection of the fuzzy rule scaling factors, which may not generalize well to significantly different systems, and it does not provide explicit robustness guarantees. Tran et al. [101] combined fuzzy tuning with a Strictly Negative-Imaginary controller for quadrotor trajectory tracking, which inherently improves system robustness.

2.8. Conclusion and Discussion

This section will discuss the important information from the preceding literature review to identify a research gap and motivate the research questions.

It was found that INDI is a promising controller in the context of quadrotor control, especially for adaptive or FTC applications [9], [15]. Methods that perform complete online parameter identification have also been developed [3], [4], allowing flight of a quadrotor without prior parameter knowledge. While such methods enable the initial stabilization of multicopter configurations without prior system information, they do not address the subsequent tuning of the outer-loop attitude controller required for robust flight performance. Consequently, obtaining a robust controller using system parameters identified online would provide a complete control solution and is therefore considered the main goal of this research. This leads to two main questions that must be addressed: what method for robust control should be used, and how can it be made applicable across a wide range of drone configurations?

Based on the literature, two promising directions can be considered to make the controller applicable to a wide range of drone configurations. The first is to synthesize a controller online based on the current motor parameters, as demonstrated for an LQR controller in [19]. For instance, the algorithm proposed by Gadewadikar et al. [20] for structured OPFB $\mathcal{H}_\infty$ controller synthesis could be applied on an onboard model of the system dynamics.

Previous work has also shown that the game-theoretic formulation of the $\mathcal{H}_\infty$ problem for both regulation and tracking, in linear as well as nonlinear cases, can be solved through RL and applied online. The main focus of these RL-based methods, however, lies in controller synthesis for systems with unknown dynamics. In the considered problem, the drone parameters are identified online, meaning that knowledge about the plant already exists. Consequently, extending $\mathcal{H}_\infty$ controller synthesis with RL would introduce unnecessary complications.

The second direction for designing a controller that can be applied across a wide range of drones is the development of a gain-scheduled controller that accounts for relevant motor properties. Combined with robust controller synthesis this could provide a widely applicable and robust control solution.

An advantage of gain-scheduling is that the controller can be synthesized offline. This allows for using computationally intensive methods, such as the non-smooth optimization techniques used for structured $\mathcal{H}_\infty$ controller synthesis. However, gain-scheduling over a wide range of platforms might require a large gain schedule to ensure broad applicability, whereas an online solution inherently adapts to system parameters. Particularly, if lookup tables with interpolation are used this could require significant storage. Nevertheless, a convenient property of INDI is that it reduces the

system dynamics to the actuator dynamics and two integrators. Consequently, only scheduling variables related to the actuator dynamics may be required. Gain-scheduled controllers employing smooth functions have also been demonstrated [50], [78], [83] which can alleviate this issue. Additionally, the use of LPV can also deal with this as it is directly parameterized against relevant system parameters. In general, the application of a gain-scheduled controller for a multicopter using an INDI controller has also not been found, indicating a research gap.

Both gain-scheduling and online synthesis of a $\mathcal{H}_\infty$ controller appear to be promising approaches for addressing the defined problem. Ideally, a comparative study between the two methods would be conducted. However, given the limited scope of this research, the focus is placed on the gain-scheduled controller. Based on the preceding analysis, this approach provides a suitable solution to the problem and is particularly attractive due to the potential of applying advanced controller synthesis techniques.

For gain-scheduling, two main approaches were identified: classical gain-scheduling and LPV control. The classical method is widely applied and has shown success in FTC and nonlinear control for both drones and other flight control systems. It typically relies on linearized models at selected operating points, with controllers designed for each point and gains interpolated based on measurable scheduling variables. Classical gain-scheduling can also be combined with structured $\mathcal{H}_\infty$ controller synthesis techniques to achieve robust control design. In particular, it has been extensively combined with signal-based $\mathcal{H}_\infty$ closed-loop shaping to produce robust and fault-tolerant gain-scheduled controllers. Nevertheless, a disadvantage of classical gain scheduling is that it has no global performance and stability guarantees.

The LPV methods address the limitations of classical control by explicitly incorporating the scheduling variables into both the plant and the controller. This formulation allows for global guarantees on stability and performance. Additionally, LPV methods can be combined with $\mathcal{H}_\infty$ synthesis techniques to achieve robust controller design. Finding the controller often involves solving a set of LMIs, which can be computationally intensive. The use of a polytopic approach can alleviate this issue by requiring LMIs to be solved only at a finite set of vertices while still guaranteeing stability and performance between them. However, this approach can introduce conservativeness, which must be mitigated by carefully selecting the polytope's dimensions.

The main purpose of a gain-scheduled controller in this context is to provide a solution that can be applied immediately after online identification of the system parameters. Unlike most gain-scheduling applications, the goal here is not to address nonlinear behavior or FTC. In fact, under ideal INDI control, the plant behaves linearly, as it reduces to the actuator dynamics and two integrators. This should be considered when deciding on a gain-scheduling method. The primary advantage of LPV control lies in its global stability and performance guarantees, which can be important for nonlinear or safety-critical systems. Although global performance guarantees are not available in the classical approach, strong performance and robustness have been demonstrated in drone control [45], [79], [102] as well as in other applications [50], [61], [82]. Given the linear nature of the INDI-controlled system, it is reasonable to expect similar results in this case. Consequently, the additional complexity introduced by LPV control appears to be unnecessary, and the classical technique will be pursued in this research.

For synthesizing $\mathcal{H}_\infty$ controllers, several methods have been described. Both open-loop shaping [45], [46], [47], [48] and closed-loop shaping [37], [40], [41] have been successfully applied to drones. Both approaches have also demonstrated effectiveness in achieving robust performance in the outer loop of an INDI controller [41], [48]. In general, comparable results can be achieved, assuming proper selection of the weighting filters. Given that comparable performance can be attained with either approach, it is not possible to establish a clear preference between them based solely on achievable performance.

However, an advantage of closed-loop shaping is its more intuitive design process, as specific closed-loop transfer functions can be directly shaped. Furthermore, signal-based closed-loop shaping allows the formulation of more advanced constraints that, for instance, bound disk margins at specific loop openings. The signal-based closed-loop shaping method has also been shown to be compatible with weighting-filter optimization techniques, such as co-design. Although other weighting-filter optimization algorithms exist, co-design is particularly attractive because it can

be directly integrated with the non-smooth optimization used for structured controller synthesis [60]. With this approach, weighting-filter parameters such as bandwidth can be optimized across the entire design space for improved performance, as demonstrated by Rhenman [61]. This is especially relevant here, as the achievable performance between different multirotor platforms will inherently change based on their actuator properties. Considering these factors, the decision was made to implement signal-based $\mathcal{H}_\infty$ closed-loop shaping in combination with gain scheduling.

Based on the above discussion, a research gap has been identified in the development of a gain-scheduled controller for an INDI-controlled quadrotor. The potential of $\mathcal{H}_\infty$ closed-loop shaping in obtaining such a controller is also highlighted. This leads to the research objective and research questions defined in Chapter 3.

3

Research Questions

Based on the discussion of the literature review in Section 2.8, the following research objective has been defined.

Research Objective

Devise a method for obtaining a robust attitude controller for a quadcopter utilizing an INDI control system for the angular acceleration loop with parameters that are identified online.

Based on the insights gained from the literature review, a promising approach to achieving the research objective is to use signal-based $\mathcal{H}_\infty$ closed-loop shaping to synthesize a gain-scheduled controller, scheduled against the identified motor parameters. This leads to the main research question outlined below.

Research Question

How can gain-scheduling be used for robust attitude control of a quadcopter utilizing an INDI control system for the angular acceleration loop, with parameters that are identified online?
1.1 How can uncertainty be characterized for a quadcopter using INDI combined with online system identification?
1.2 What is an effective controller structure that can meet desired specifications while enabling gain-scheduling?
1.3 What parameters should the controller be scheduled against?
1.4 How can a gain-scheduled controller be synthesized to achieve optimal performance and robustness across a variety of drone configurations?

Robust $\mathcal{H}_\infty$ Controller Design For INDI-Controlled Quadcopters Using Online Parameter Identification

Tom Aantjes
Faculty of Aerospace Engineering, Delft University of Technology, Delft, 2629HS, The Netherlands

This work presents the design of a robust gain-scheduled controller for attitude control of quadcopters, using an Incremental Nonlinear Dynamic Inversion (INDI)-based inner loop with online identification of its system parameters. A linearized model including uncertainty in the identified parameters and unmodeled dynamics is presented for robustness analysis, followed by a set of design requirements. With this, a cascaded feedback attitude controller with a feedforward filter was synthesized for a symmetric quadcopter using signal-based $\mathcal{H}_\infty$ closed-loop shaping. Subsequent linear analysis demonstrated good robustness margins and performance characteristics, which were further validated through nonlinear simulations and experimental flights, showing good performance under uncertainty. This methodology was then extended using co-design to develop a gain-schedule for varying actuator time constants. The approach achieved the requirements over the entire range. The resulting gain-scheduled controller exhibited good stability margins, with nonlinear simulations confirming effective tracking performance under uncertainty. Experimental evaluation of the gain-scheduled controller was conducted through flight tests with full online parameter identification. Even though the identified parameters during these tests were far outside the defined uncertainty range, acceptable flight performance comparable to simulation results was maintained for actuator time constants below 40 ms. For slower actuators, performance was degraded but this may have been due to the extreme uncertainties rather than the controller itself.

I. Introduction

Unmanned Aerial Vehicles (UAVs) have become increasingly popular due to their versatility, agility, and wide range of applications, from aerial photography and surveillance to search and rescue operations [1, 2]. The performance and stability of these systems depend on the precise tuning of their control parameters, which can be a challenging task, especially when considering robustness against uncertainties or disturbances. In such cases, controller synthesis becomes a time-consuming endeavor that requires experienced engineers. Recent advances have demonstrated that quadrotors can rapidly identify their actuator models onboard, enabling recovery from throws and stabilization without preprogrammed gains or system parameters by combining Incremental Nonlinear Dynamic Inversion (INDI) with a rapid online system identification procedure [3, 4]. While this method allows for initial stabilization of multicopter configurations without prior system knowledge, it does not address the subsequent tuning of the outer-loop attitude controller required for robust flight performance. The purpose of this study is to provide a solution for synthesizing an attitude controller based on the parameters identified online that achieves good performance and robustness immediately after stabilization.

INDI has emerged as a powerful approach for flight control, featuring an inner loop for dynamic inversion and an outer loop for generating virtual control inputs via linear controllers [5, 6]. Good robustness and disturbance rejection properties through the use of INDI have been successfully shown on quadrotors [7]. Efforts have been made to implement an $\mathcal{H}_\infty$ controller for the outer attitude control loop of an INDI system to

enhance robustness and disturbance rejection. These include the use of signal-based $\mathcal{H}_\infty$ controller synthesis methods [8] and open-loop shaping techniques [9], both of which have demonstrated strong disturbance rejection and robustness characteristics in the outer loop controller. These methods leverage the key property that, under perfect INDI control, the inner loop reduces to the actuator dynamics, making the system linear and solely dependent on the actuator properties [5]. Synthesis of a robust gain-scheduled controller scheduled against actuator properties, combined with the rapid online system identification from [3], could thus provide a versatile control solution applicable to a wide range of multirotor platforms using INDI.

Fixed-structure and multi-objective, multi-model design methods [10–12] have been extensively used to obtain gain-scheduling controllers for a set of linearized models, as demonstrated in [13–15]. These methods have also been extended to the design of highly agile aircraft, incorporating co-design to simplify the selection of weighting filters and achieve optimal closed-loop characteristics across an entire range of plant models [16]. However, attempts to synthesize a gain-scheduled controller for a wide range of INDI-controlled multirotor configurations have not been found.

This work presents a robust gain-scheduled attitude controller that can be used by quadcopters with a wide range of actuator properties. When combined with online system identification, the proposed approach can achieve high robustness without requiring prior knowledge of the platform. The controller is designed and assessed using a model that includes an explicit uncertainty representation tailored to multirotor UAVs employing INDI controllers with online system identification. This model is used to synthesize a robust attitude controller for a symmetric quadrotor using signal-based $\mathcal{H}_\infty$ closed-loop shaping. Its robust characteristics are demonstrated through analysis of the closed-loop system, simulation, and experimental flights. The approach is then extended with co-design to synthesize a gain-scheduled controller applicable over a range of actuator time constants. Its effectiveness is validated through closed-loop analysis, nonlinear simulation, and experimental flights with full onboard system identification.

A description of the equations of motion, the INDI controller, the modeling of uncertainty, and their linearization is provided in section II. The robust control design problem, including performance specifications and the controller structure, is then presented in section III. The tuning procedure and results for a symmetric quadcopter, along with an analysis of the closed-loop system, are discussed in section IV. Validation through nonlinear simulations and experimental tests is presented in section V, while the extension to gain-scheduled control is described in section VI. Finally, section VII demonstrates the effectiveness of the gain-scheduled controller through simulations and flight experiments.

II. System Model

This section provides an overview of the attitude equations of motion, the INDI controller, and their integration into a nominal linearized plant with uncertainty for control design. Additionally, the experimental platform used for validation and a noise model used for analysis are described.

A. Equations of motion

Nominally, a rigid quadcopter symmetric in its body axes as depicted in Figure 1 is considered. The figure illustrates the geometric parameters l and b, the rotor rotation directions and their numbering, and the body reference frame.

The attitude with respect to hover is described by the Euler angles $\boldsymbol{\eta} = [\phi, \theta, \psi]^\top$. The angular rates in the body frame are defined as $\boldsymbol{\Omega} = [p, q, r]^\top$, and the angular accelerations as $\dot{\boldsymbol{\Omega}} = [\dot{p}, \dot{q}, \dot{r}]^\top$. The rotational dynamics are described by Euler's rotational equation of motion, as given in Equation 1 [8]. Here, $\boldsymbol{\tau_B}$ denotes the torque generated by the propellers about each body axis, and $\boldsymbol{\tau_d}$ represents the disturbance torques. The quadrotor's moments of inertia are represented by $I_v = \mathrm{diag}(I_{xx}, I_{yy}, I_{zz})$ because of symmetry.

$$\dot{\boldsymbol{\Omega}} = I_v^{-1} \left(\boldsymbol{\tau_B} - \boldsymbol{\Omega} \times I_v \boldsymbol{\Omega} + \boldsymbol{\tau_d} \right) \tag{1}$$

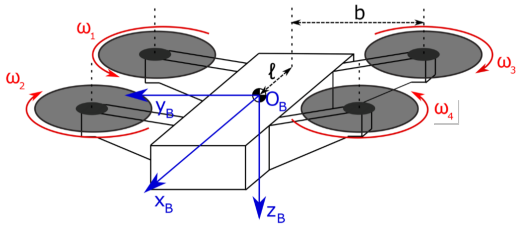


Figure 1. Quadrotor body frame, definition of geometric parameters and rotor numbering. Adapted from [17]

The torque $\boldsymbol{\tau_B}$ can be written as a function of the control effectiveness matrices B_1 and B_2 which arise from geometric and actuator parameters of the vehicle, as shown in Equation 2 [5]. In this equation, $\boldsymbol{T}$ represents the thrust for each motor and $\dot{\boldsymbol{\omega}}$, the rate of change of the motor velocity. Thrust can be related to the motor velocity ω according to $T = k\omega^2$, where k is the motor thrust constant [5].

$$\boldsymbol{\tau_B} = I_v \left(\frac{1}{2} B_1 \boldsymbol{T} + B_2 \dot{\boldsymbol{\omega}} \right) \tag{2}$$

B. INDI Controller

The angular acceleration of the quadrotor is controlled using an INDI controller, which is described in this section. Central to the INDI control law is an incremental model that relates changes in so-called pseudo-controls $\boldsymbol{\nu}$ to changes in the actuator states. Inversion of this model combined with sensor measurements can yield a control law that linearizes the system dynamics with respect to these pseudo-controls [5]. In this implementation, $\boldsymbol{\nu}$ corresponds to the angular acceleration $\dot{\boldsymbol{\Omega}}$, with the incremental model defined as in Equation 3 [3]. The increment in $\boldsymbol{\nu}$ is related to the increment in thrust T and the rate of change of motor velocity $\dot{\boldsymbol{\omega}}$, where incremented quantities are indicated by Δ, and the subscript 0 refers to the most recent measurement.

$$\Delta \boldsymbol{\nu} = \boldsymbol{\nu} - \boldsymbol{\nu}_0 = B_1 \Delta T + B_2 \Delta \dot{\boldsymbol{\omega}} \tag{3}$$

In the work of Blaha et al. [3] it was demonstrated that employing a thrust-normalized control variable $m_c \triangleq T_{\text{ss}}/T_{\text{max}}$ for motor commands enables a formulation of this incremental model that facilitates system identification and motor control. Here, T_{ss} denotes the steady-state thrust after the actuator dynamics have settled. Rewriting the incremental model in Equation 3 as a function of normalized thrust yields the expression shown in Equation 4. In this equation, ω_{max} is the maximum angular rate for each motor, and τ denotes the actuator time constant obtained from a first-order approximation of the actuator dynamics $A(s)$ shown in Equation 5. These parameters and the additional term on the left-hand side arise from an approximate relationship between $\dot{\boldsymbol{\omega}}$ and T, which enables a linearized formulation. For a full derivation, refer to [3].

$$\boldsymbol{\nu} - \boldsymbol{\nu}_f + B_2 \dot{\boldsymbol{\omega}}_f \approx \left(B_1 T_{\text{max}} + B_2 \frac{\boldsymbol{\omega}_{\text{max}}^2}{2 \boldsymbol{\omega}_f \tau} \right) \Delta \boldsymbol{m}_c \tag{4}$$

$$A(s) = \frac{1}{\tau s + 1} \tag{5}$$

The subscript f indicates that the most recent measurements are filtered through a synchronization filter $H(s)$. This filtering mitigates noise introduced by differentiating $\boldsymbol{\Omega}$ to obtain $\dot{\boldsymbol{\Omega}}$, and must also be applied to the current motor states and other measurements to maintain the validity of the INDI control law [5]. A second-order Butterworth filter with a cutoff frequency of 15 Hz is used for $H(s)$, as in [3]. The effectiveness of this choice is later verified through closed-loop analysis.

Solving Equation 4 for a reference $\boldsymbol{\nu}_r$ can be achieved through multiple methods such as a pseudo-inverse or active-set approaches [18]. In practice, an active-set method is employed, as it accommodates the bounds of $\boldsymbol{m_c}$ in the control allocation. Nonetheless, the pseudo-inverse can be effective and will be used for linear modelling. Since the Electronic Speed Control (ESC) does not directly control thrust, $\boldsymbol{m_c}$ must be converted to an ESC setpoint δ through inversion of a nonlinear function shown in Equation 6 [4]. Here, κ is a coefficient that can be identified online.

$$m_c(\delta) = \left[\kappa\delta + (1-\kappa)\sqrt{\delta}\right]^2 \tag{6}$$

As the control law computes only increments of the actuator state, knowledge of the current actuator state is necessary to determine the motor setpoints. When the motor speed $\boldsymbol{\omega}$ is directly measured, the current thrust can be computed from the thrust coefficient k. However, not all platforms can directly measure $\boldsymbol{\omega}$. On such platforms, the estimated first-order actuator dynamics can be applied to the motor setpoints $\boldsymbol{m_c}$ to estimate the current motor state. Although this approximation neglects nonlinear effects and additional actuator states, it has proven effective in practice [3] and will be considered here to make the controller more widely applicable.

C. Linearized Model

For controller design, the system is linearized about a level-hover operating point. When linearizing Equation 1 around $\boldsymbol{\Omega} = 0$, the gyroscopic term can be neglected, assuming that the effects of angular acceleration due to changes in $\boldsymbol{\Omega}$ and $\dot{\boldsymbol{\Omega}}$ are much smaller than those caused by changes in $\boldsymbol{\omega}$ and $\dot{\boldsymbol{\omega}}$ [5, 19]. This holds if the actuators are sufficiently fast and their influence outweighs the effects of aerodynamic forces and precession moments caused by changes in angular rates and body velocities [5].

To avoid dependence of the pseudo-inverse in the INDI controller on the current motor velocity, the B_2 matrix is assumed to be zero. Consequently, the yaw response in the linear model is less accurate as gyroscopic effects from the increasing spin rate of the propellers are neglected. Only yaw is affected, because the motors are aligned with z_B, which causes the gyroscopic effects of the propellers to influence rotations only about this axis. Given that roll and pitch dynamics are the most critical to track, this approximation is deemed acceptable for controller synthesis and linear analysis. In the implemented controller used for simulation and flight experiments, B_2 is still included to capture the full dynamics. However, the influence of uncertainty in this matrix on the closed-loop system is thus not explicitly analyzed in this work. For similar reasons, the nonlinear mapping between m_c and δ is assumed to be perfect as it mainly affects the system dynamics during large changes in motor setpoints. Consequently, the direct effect of uncertainties in this mapping on the closed-loop system is neglected in this study, but its implementation is still included in the actual controller.

Linearization of Equation 2 makes the transition from commanded thrust to motor angular rate linear. This means the actuator dynamics $A(s)$ can be considered to act directly on the commanded normalized thrust setpoints $\boldsymbol{m_c}$. Additionally, since B_2 is set to 0, the motor effects on angular acceleration can be fully characterized by the nominal control effectiveness matrix $E = B_1 T_{\max}$, shown in Equation 7. Here, the control effectiveness parameters of the motors about each body axis, $C_{\dot{p}}$, $C_{\dot{q}}$, and $C_{\dot{r}}$, are defined and are assumed to be identical for all motors. These parameters describe how the normalized thrust of each motor influences the angular acceleration about each body axis, combining geometric, inertial, and motor properties.

$$E = \begin{bmatrix} -C_{\dot{p}} & -C_{\dot{p}} & +C_{\dot{p}} & +C_{\dot{p}} \\ -C_{\dot{q}} & +C_{\dot{q}} & -C_{\dot{q}} & +C_{\dot{q}} \\ -C_{\dot{r}} & +C_{\dot{r}} & +C_{\dot{r}} & -C_{\dot{r}} \end{bmatrix} \tag{7}$$

The full linear plant with INDI controller is shown in Figure 2. Outside of the INDI controller, the

perturbed actuator dynamics $A_p(s)$ and perturbed control effectiveness matrix E_p are defined. These represent perturbed versions of $A(s)$ and E that include uncertainty, and their definitions are specified in subsection II.D. The output of the perturbed actuator dynamics $\boldsymbol{m}$ represents the current normalized thrust of the motors, while $\boldsymbol{\Omega}_m$ denotes the measured angular rates. The Moore–Penrose pseudoinverse, indicated by †, is applied to the nominal control effectiveness matrix within the INDI controller to solve Equation 4 for the reference signal $\boldsymbol{\nu}_r$. Finally, the onboard filtered estimate of the current motor state $\boldsymbol{m_f}$ is combined with $\Delta\boldsymbol{m_c}$ to obtain the motor setpoint $\boldsymbol{m_c}$.

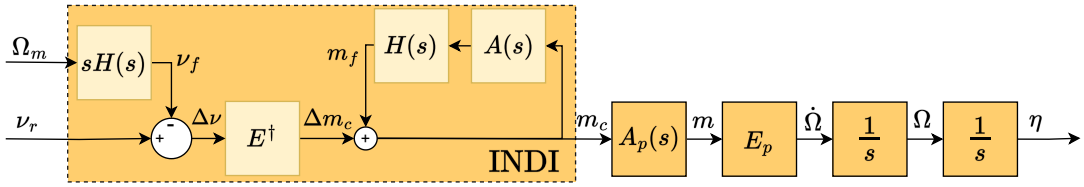


Figure 2. Linearized model of the quadrotor plant with INDI controller.

Under nominal conditions, where $A(s) = A_p(s)$, $E = E_p$, and the angular rate measurements $\boldsymbol{\Omega}$ are perfect, the INDI controller inverts the system dynamics [5]. As a consequence, the nominal system reduces to only the actuator dynamics $A(s)$ and two integrators. The nominal plant can therefore be fully characterized by the actuator time constant τ of $A(s)$.

D. Uncertainty Modelling

The nominal model introduced in the previous section does not perfectly represent the physical system, as the plant parameters cannot be known with perfect accuracy and some system dynamics are not captured by the model. To account for these discrepancies, this section introduces an uncertainty model for the control effectiveness matrix and actuator dynamics.

Control effectiveness: All coefficients in the E matrix are estimated online, which inherently introduces uncertainty into the control effectiveness model. Additional variability may arise from factors such as fluctuations in battery voltage or aerodynamic effects. To represent these sources of uncertainty, each control effectiveness coefficient is assumed to vary by up to 20 % from its nominal value. This deviation represents a conservative increase over the typical root-mean-square (RMS) error of 10 % observed in simulations of the online identification routine in [3], to account for additional uncertainty expected in practical operation. The uncertain control effectiveness for each axis and motor i is modeled as shown in Equation 8, where $r_C = 0.2$ defines the maximum relative deviation, δ_{C_x} is a scalar uncertainty parameter within this range, and the subscript p denotes the perturbed quantity.

$$\begin{aligned} C_{\dot{p}_{i_p}} &= C_{\dot{p}}\left(1 + r_C\delta_{C_{\dot{p}_i}}\right) \quad \text{with } |\delta_{C_{\dot{p}_i}}| < 1 \\ C_{\dot{q}_{i_p}} &= C_{\dot{q}}\left(1 + r_C\delta_{C_{\dot{q}_i}}\right) \quad \text{with } |\delta_{C_{\dot{q}_i}}| < 1 \\ C_{\dot{r}_{i_p}} &= C_{\dot{r}}\left(1 + r_C\delta_{C_{\dot{r}_i}}\right) \quad \text{with } |\delta_{C_{\dot{r}_i}}| < 1 \end{aligned} \tag{8}$$

Substituting these expressions into E yields the perturbed control effectiveness matrix E_p:

$$E_p = \begin{bmatrix} -C_{\dot{p}_{1_p}} & -C_{\dot{p}_{2_p}} & +C_{\dot{p}_{3_p}} & +C_{\dot{p}_{4_p}} \\ -C_{\dot{q}_{1_p}} & +C_{\dot{q}_{2_p}} & -C_{\dot{q}_{3_p}} & +C_{\dot{q}_{4_p}} \\ -C_{\dot{r}_{1_p}} & +C_{\dot{r}_{2_p}} & +C_{\dot{r}_{3_p}} & -C_{\dot{r}_{4_p}} \end{bmatrix} \tag{9}$$

Incorporating these uncertainties into the linear model enables direct analysis of their impact on closed-loop stability. To achieve this, a Linear Fractional Representation (LFR) of E_p is constructed as an upper

linear fractional transformation (LFT), combining an uncertainty-free system $M_E(s)$ with a perturbation matrix $\Delta_E = \text{diag}(\delta_{C_{\dot{p}_1}}, \ldots, \delta_{C_{\dot{p}_4}}, \delta_{C_{\dot{q}_1}}, \ldots, \delta_{C_{\dot{q}_4}}, \delta_{C_{\dot{r}_1}}, \ldots, \delta_{C_{\dot{r}_4}})$, as shown in Equation 10 [20]. This is illustrated schematically in Figure 3.

$$E_p = \mathcal{F}_\text{u}\left[M_E(s), \Delta_E\right] \tag{10}$$

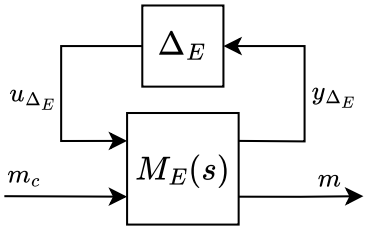


Figure 3. Upper LFT of the perturbed control effectiveness matrix E_p.

The state-space representation of the LFR is shown in Equation 11, where the uncertainty block is interfaced through fictitious inputs $\boldsymbol{u}_{\boldsymbol{\Delta_E}}$ and outputs $\boldsymbol{y}_{\boldsymbol{\Delta_E}}$, each containing one entry for every diagonal element in $\Delta_E(s)$. The block $M_{E_{22}}(s)$ corresponds to the nominal control effectiveness matrix E, while the remaining blocks of $M_E(s)$ describe how the actual system inputs and outputs interact with the fictitious uncertainty channels, thereby capturing the influence of uncertainty on the system dynamics.

$$\begin{bmatrix} \boldsymbol{y}_{\boldsymbol{\Delta_E}}(s) \\ \dot{\boldsymbol{\Omega}}(s) \end{bmatrix} = \begin{bmatrix} M_{E_{11}}(s) & M_{E_{12}}(s) \\ M_{E_{21}}(s) & M_{E_{22}}(s) \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_{\boldsymbol{\Delta_E}}(s) \\ \boldsymbol{m}(s) \end{bmatrix} \tag{11}$$

Actuator dynamics: The actuator dynamics are represented by a first-order system whose time constant is estimated online. This modeling approach introduces uncertainty arising from both identification errors and neglected higher-order effects. To characterize this uncertainty and assess how accurately the first-order approximation reproduces the true actuator behavior, experimental motor response measurements were conducted and compared with the corresponding online model estimates.

To avoid biasing the uncertainty model to a single motor type, two types of quadrotor motors were examined. The first was taken from a quadcopter equipped with 3-inch propellers, while the second originated from a smaller 75 mm frame platform, commonly referred to as a TinyWhoop. Each motor–propeller combination was excited with sinusoidal input signals spanning 36 logarithmically spaced frequencies between 6 Hz and 120 Hz, with each excitation lasting 1.5 s. The resulting motor responses were recorded to obtain experimental frequency responses, which were subsequently compared with both the first-order fits and the online model estimates, as illustrated in Figure 4 and Figure 5. Across most of the frequency range, the first-order model accurately captured the actuator dynamics for both motors. However, for the TinyWhoop motor, the approximation deteriorates beyond 150 rad/s, indicating unmodeled high-frequency dynamics.

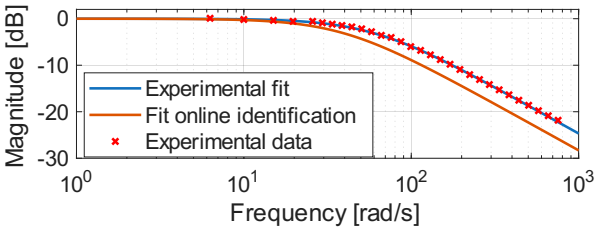


Figure 4. Identified actuator frequency response for the 3-inch drone.

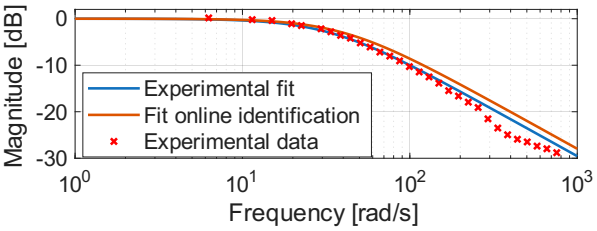


Figure 5. Identified actuator frequency response for the TinyWhoop.

To assess the accuracy of the online identification of the actuator time constant, the experimentally determined value τ_{exp} was compared with the online estimate τ_{id}. For the 3-inch drone, $\tau_{\text{id}} = 26.3\,\text{ms}$ compared to $\tau_{\text{exp}} = 17.0\,\text{ms}$, while for the TinyWhoop, $\tau_{\text{id}} = 25.0\,\text{ms}$ and $\tau_{\text{exp}} = 32.7\,\text{ms}$. These results indicate that the online identification both overestimates and underestimates the true actuator time constant, with deviations of up to 35 % relative to τ_{exp}.

Based on the above discussion, the actuator uncertainty model must capture both misidentification of the actuator time constant and unmodeled high-frequency dynamics. Accordingly, the perturbed actuator model $A_{p_i}(s)$ for each motor i incorporates both parametric and unstructured uncertainty, as shown in Equation 12. The parametric uncertainty is expressed as $\tau_{p_i} = \tau(1 + r_\tau \delta_{\tau_i})$, where $r_\tau = 0.4$ represents a conservative margin above the maximum observed deviation of 35 %. The unstructured component is characterized by the weighting function $w_m(s)$ defined in Equation 13 [20]. Here, $r_0 = 0.04$ and $r_\infty = 1.0$ specify the relative uncertainty at low and high frequencies, respectively. The crossover frequency τ_w, at which the relative uncertainty reaches 100 %, is set to $\tau/5$, making it dependent on the nominal actuator time constant. These values were determined to ensure that applying this uncertainty model to the experimentally identified first-order model bounds the experimental data. The uncertainty block $\Delta_{\tau_i}(s)$ is any stable transfer function satisfying $\|\Delta_{\tau_i}(s)\|_\infty < 1$.

$$A_{p_i}(s) = \frac{1}{\tau_{p_i} s + 1}\left(1 + w_m(s)\Delta_{\tau_i}(s)\right) \tag{12}$$

$$w_m(s) = \frac{\tau_w s + r_0}{\frac{\tau_w}{r_\infty} s + 1} \tag{13}$$

The resulting bounds and a set of uncertain realizations with respect to the experimentally identified first-order model of the TinyWhoop motor are shown in Figure 6, confirming that the experimental data is enclosed within the modeled uncertainty.

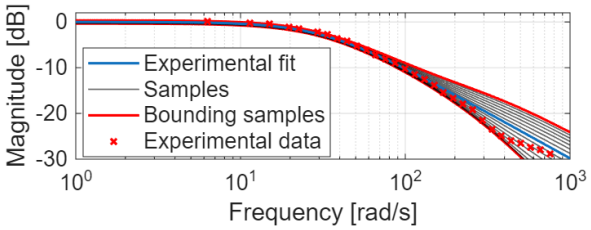


Figure 6. Unstructured uncertainty bounds for the actuator dynamics of the TinyWhoop.

The LFR representation of the uncertain actuator dynamics used in the linear model is shown in Equation 14. Here $\Delta_A(s) = \text{diag}\left(\delta_{\tau_1}, \ldots, \delta_{\tau_4}, \Delta_{\tau_1}(s), \ldots, \Delta_{\tau_4}(s)\right)$, and $M_A(s)$ is defined as in Equation 15. The nominal actuator dynamics are represented by the diagonal transfer matrix $M_{A_{22}}(s) = A(s)I_{4\times4}$.

$$A_p(s) = \mathcal{F}_{\text{u}}\left[M_A(s), \Delta_A(s)\right] \tag{14}$$

$$\begin{bmatrix} \boldsymbol{y}_{\boldsymbol{\Delta_A}}(s) \\ \boldsymbol{m}(s) \end{bmatrix} = \begin{bmatrix} M_{A_{11}}(s) & M_{A_{12}}(s) \\ M_{A_{21}}(s) & M_{A_{22}}(s) \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_{\boldsymbol{\Delta_A}}(s) \\ \boldsymbol{m_c}(s) \end{bmatrix} \tag{15}$$

E. Noise Model

The synchronization filter $H(s)$ is used to mitigate noise amplification that arises from differentiating the gyroscope measurements [5]. To evaluate its effectiveness for noise attenuation within the closed-loop system, a linear noise-shaping filter $N(s)$ representing the gyroscope noise spectrum was identified experimentally. Because the noise characteristics depend strongly on the specific platform, influenced by factors such as

the propellers and airframe, the tuning of the synchronization filter is inherently platform-dependent and does not generalize directly. Nevertheless, to provide insight into typical noise behavior and its effect on the closed-loop system, this work considers a noise model derived from both the quadrotor with 3-inch propellers and the TinyWhoop platform.

The noise characteristics of the gyroscope output were identified through spectral analysis of gyroscope data recorded during hover. Welch's power spectral density estimate was used with a Hamming window (50% overlap) and a segment length of 2048 [21], yielding a frequency resolution of 0.8 Hz while maintaining acceptable variance. The resulting amplitude spectral densities (ASDs) are shown in Figure 7. The spectra reveal that fo both platforms, the noise levels decrease with frequency until reaching a peak corresponding to the motor spin rate. For the noise model, the TinyWhoop data noise profile was overbounded as both spectra have a similar shape but the TinyWhoop exhibits higher noise across most frequencies except for the peak from the 3-inch drone. To this end, the shaping filter $N(s)$ was modeled as a combination of low-pass, lag, and band-pass elements. Although the band-pass approximation reduces accuracy around the peak, emphasis was placed on accurately capturing the peak itself as it dominates the response at high-frequencies.

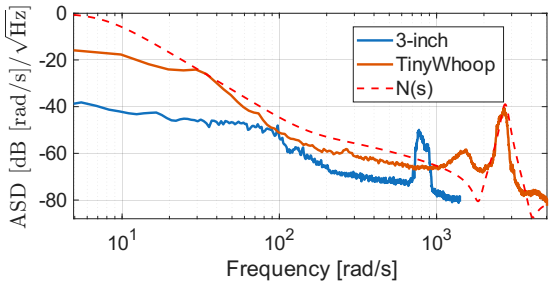


Figure 7. Amplitude spectral density of the gyroscope output during hover for a 3-inch drone and TinyWhoop.

F. Experimental Platform Description

In this work, the quadrotor equipped with 3-inch propellers is used both for experimental validation and as the basis for an initial controller design. It features four identical motors with control effectiveness parameters $C_{\dot{p}} = 400$ rad/s^2, $C_{\dot{q}} = 260$ rad/s^2, and $C_{\dot{r}} = 51$ rad/s^2, as well as an actuator time constant of $\tau = 17$ ms, determined from bench tests.

To experimentally evaluate controller performance under uncertainty in the actuator time constant, the actuators were intentionally slowed to a desired value τ_{des}. This was accomplished by applying the lag filter defined in Equation 16, with $\tau \leq \tau_{\text{des}}$. The filter was applied to the commanded motor setpoints before transmission to the ESCs, thereby creating a slower effective response. Only a lag filter is considered, as using a lead filter to increase actuator speed could push the actuators beyond their capabilities. This can introduce additional dynamics that would not otherwise be present. Because the motors can only be slowed down, experimentally covering a ±40% uncertainty range is not feasible with the actual nominal actuator time constant of 17 ms. Therefore, a larger nominal actuator time constant of 28 ms is adopted for controller design.

$$A_{ll}(s) = \frac{\tau s + 1}{\tau_{\text{des}} s + 1} \tag{16}$$

III. Attitude Control Design Setup

This section outlines the input-output structure used for control design, design requirements, and the controller configuration.

A. Design Layout

For control design, the design problem is formulated as a standard robust control configuration, consisting of the plant $P(s)$, controller $K(s)$, and the perturbation matrix $\Delta(s) = \text{diag}(\Delta_E, \Delta_A(s))$, as illustrated in Figure 8 [22]. In this formulation, a set of inputs and outputs used for controller synthesis and closed-loop analysis is defined.

Two output disturbances are considered, one acting on the angular acceleration, $\boldsymbol{d}_{\dot{\boldsymbol{\Omega}}} = [d_{\dot{p}}, d_{\dot{q}}, d_{\dot{r}}]^\top$, and one on the attitude output, $\boldsymbol{d}_{\boldsymbol{\eta}} = [d_\phi, d_\theta, d_\psi]^\top$. The disturbance on $\dot{\boldsymbol{\Omega}}$ is more physically accurate, as external forces such as wind directly affect angular acceleration. Nevertheless, an output disturbance on $\boldsymbol{\eta}$ is still included for controller synthesis, as it enables shaping of the closed-loop transfer function from $\boldsymbol{d}_{\boldsymbol{\eta}}$ to $\boldsymbol{\eta}$. This function affects both disturbance rejection performance and stability margins, and shaping it during controller synthesis allows achieving desired characteristics for both [22].

Other inputs include the attitude reference $\boldsymbol{r}_{\boldsymbol{\eta}} = [r_\phi, r_\theta, r_\psi]^\top$ for attitude control, input disturbances on the motor commands $\boldsymbol{d}_{\boldsymbol{i}} = [d_{m_1}, d_{m_2}, d_{m_3}, d_{m_4}]^\top$, and sensor noise $\boldsymbol{n}_{\boldsymbol{\Omega}} = [n_p, n_q, n_r]^\top$ in the gyroscope measurements. The sensor noise is shaped through the shaping filter $N(s)$, and to emulate noise in the measured attitude $\boldsymbol{\eta}_{\boldsymbol{m}}$, the signal is passed through an integrator. Although this does not fully capture the fusion of gyroscope and IMU measurements used in practice, it provides a reasonable approximation for frequency-domain analysis.

The output signals include $\boldsymbol{m}_{\boldsymbol{c}}$, $\dot{\boldsymbol{\Omega}}$, $\boldsymbol{\Omega}$, $\boldsymbol{\eta}$, and the model-following error $\boldsymbol{e}_{\mathbf{ref}_{\boldsymbol{\eta}}} = \boldsymbol{r}_{\boldsymbol{\eta}} - \boldsymbol{\eta}$ relative to a reference model $T_{\text{ref}}(s)$. The main purpose of $T_{\text{ref}}(s)$ is to allow the synthesis of a controller that makes the tracking response match desired dynamics. The structure of $T_{\text{ref}}(s)$ and the requirements on which it is based will be discussed in the next sections.

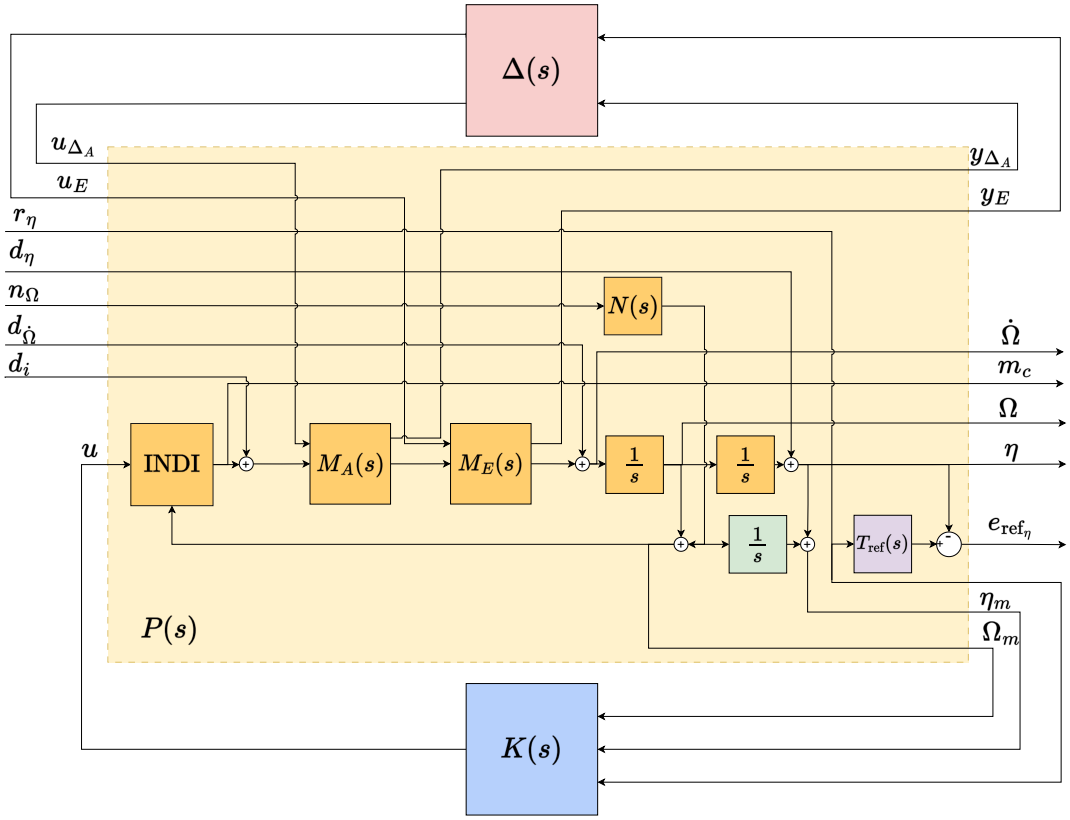


Figure 8. Input / output structure of the plant $P(s)$, controller $K(s)$ and perturbation matrix $\Delta(s)$ used for control design.

B. Design Requirements

The controller design is guided by a set of performance and robustness requirements. For robustness, a minimum nominal classical gain margin (GM) of 4 dB and a phase margin (PM) of 35° are specified at the actuator inputs $\boldsymbol{m_c}$ and the plant outputs $\dot{\boldsymbol{\Omega}}$, $\boldsymbol{\Omega}$, and $\boldsymbol{\eta}$. These values represent a relaxation of the AS94900 specification [23], which prescribes a GM of 6 dB and a PM of 45°, following the performance-oriented UAV design approach proposed by Berrios et al. [24]. To ensure robustness under uncertainty, a maximum reduction of 50 % from the nominal stability margins is permitted, consistent with the AS94900 requirements.

While classical gain and phase margins provide useful measures of robustness, they do not account for simultaneous variations in gain and phase or for perturbations in multiple channels simultaneously. A more comprehensive assessment can be obtained using disk margins, which quantify robustness against concurrent gain and phase perturbations [25]. These are represented as disks $D(\alpha_{\max}, \sigma)$, where $\alpha_{\max}$ defines the disk size and σ its eccentricity. Of specific interest is the special case $\sigma = 0$ which indicates symmetric gain and phase margins and will be considered here. To improve robustness against simultaneous perturbations in multiple loops, a minimum GM of 3 dB and PM of 19.5° is imposed for the symmetric multi-loop disk margins defined at the actuator input $\boldsymbol{m_c}$ and the plant output $\dot{\boldsymbol{\Omega}}$. Additional perturbations in $\boldsymbol{\Omega}$ and $\boldsymbol{\eta}$ could be considered but since these are integrated variables of each other, treating them as independent would be overly conservative.

The controller is further required to reject disturbances in both angular acceleration and actuator inputs, and to attenuate sensor noise at the plant input and output, which are standard control design objectives.

To prevent actuator saturation, the motor angular rate vector $\boldsymbol{\omega}$ must remain between 10 % to 100 % of the maximum angular rate $\omega_{\max}$ during a roll or pitch maneuver from +45° to −45°. This represents a generic requirement corresponding to flight with a saturation limit of ±45° in roll and pitch with an idle motor state at 10 %. Yaw is excluded from this requirement due to its low control effectiveness, which would dominate the constraint despite being the least critical axis for control. In practice, yaw is deprioritized in the control allocation.

Finally, to optimize rise time and the 5 % settling time, the nominal closed-loop attitude tracking response is designed with an overshoot between 4.5 % and 5 % [26]. These requirements are summarized and numbered below:

(R1) Minimum classical GM and PM of 4 dB and 35°, respectively, at the actuator input $\boldsymbol{m_c}$ and the plant outputs $\boldsymbol{\eta}$, $\boldsymbol{\Omega}$, and $\dot{\boldsymbol{\Omega}}$.

(R2) Minimum classical GM and PM under modeled uncertainty of 2 dB and 17.5°, respectively, for the actuator input $\boldsymbol{m_c}$ and the plant outputs $\boldsymbol{\eta}$, $\boldsymbol{\Omega}$, and $\dot{\boldsymbol{\Omega}}$.

(R3) Minimum multi-loop disk GM of 3 dB and PM of 19.5° for the actuator input $\boldsymbol{m_c}$ and the plant output $\dot{\boldsymbol{\Omega}}$.

(R4) Disturbance rejection for both actuator input and angular acceleration disturbances.

(R5) Attenuation of sensor noise at the actuator input and plant output.

(R6) Motor angular rates $\boldsymbol{\omega}$ within 10 %–100 % of $\omega_{\max}$ during roll and pitch maneuvers between −45° and +45°.

(R7) Nominal reference tracking performance with an overshoot between 4.5 % and 5.0 %.

C. Controller Structure

The controller structure used to achieve the requirements is shown in Figure 9 and consists of a feedback and feedforward part. The feedback controller contains the static gains $\boldsymbol{K_\eta} = K_\eta I_{3\times3}$ and $\boldsymbol{K_\Omega} = K_\Omega I_{3\times3}$, providing robust stability and disturbance rejection [22]. Although each gain could be tuned individually, they are taken as equal in this design. This is justified by the nominal decoupling and identical dynamics of each axis under ideal INDI control [5]. One could observe a lack of integral action in this controller structure, which may result in steady-state error and insufficient input disturbance rejection [22]. However, integrator-like behavior

is inherent to the incremental control structure combined with angular acceleration feedback in the INDI controller, and is therefore not strictly required in the outer loop [5]. The controller output $\boldsymbol{u}$ is the reference angular acceleration input to the INDI controller.

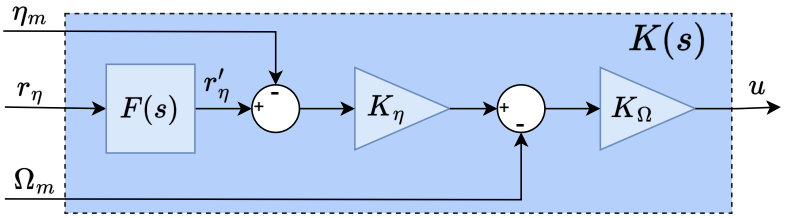


Figure 9. Attitude controller structure.

A feedforward filter $\boldsymbol{F}(s) = F(s)I_{3\times3}$ is applied to the reference to enhance nominal tracking performance without affecting the robustness of the system. The selected feedforward controller $F(s)$ is a first-order transfer function with a zero a_{ff} and a pole b_{ff}, as shown in Equation 17. This structure provides adequate performance while maintaining limited complexity with only two parameters.

$$F(s) = \frac{s/a_{\text{ff}} + 1}{s/b_{\text{ff}} + 1} \tag{17}$$

To enable synthesis of the feedforward component aimed at improving transient performance, a reference model describing the desired relationship between the attitude reference and the attitude output is required. While second-order reference models are commonly used to impose specific handling characteristics, analysis of the closed-loop system revealed that the dynamics between the feedforward filter output $\boldsymbol{r}'_{\boldsymbol{\eta}}$ and the attitude $\boldsymbol{\eta}$ are third-order. Imposing purely second-order behavior would therefore require a higher-order feedforward filter to compensate for the additional pole, increasing controller complexity. Since the main purpose of the reference model is to specify the desired transient response in terms of rise time and overshoot, and the third-order dynamics are not inherently undesirable, the model shown in Equation 18 is adopted. It is formulated as a second-order system with reference natural frequency ω_{ref} and damping ratio ζ_{ref}, augmented by an additional pole b_{ref}. The numerator includes the same pole to ensure a unity steady-state gain.

$$T_{\text{ref}}(s) = \frac{\eta_{\text{ref}}(s)}{r_\eta(s)} = \frac{\omega_{\text{ref}}^2}{s^2 + 2\zeta_{\text{ref}}\omega_{\text{ref}}s + \omega_{\text{ref}}^2} \frac{b_{\text{ref}}}{s + b_{\text{ref}}} \tag{18}$$

Faster actuators were found to facilitate better performance due to the increased plant bandwidth. To maintain optimal performance across different actuator time constants, a separate reference model is defined for each design point. To obtain this model systematically, the feedback component of the controller is synthesized first, and the resulting dynamics are used to define the reference model. Specifically, the third pole of the transfer function between $\boldsymbol{r}'_{\boldsymbol{\eta}}$ and $\boldsymbol{\eta}$ is assigned as b_{ref}, while the natural frequency ω_{ref} is determined from the dominant pole pair. The damping ratio ζ_{ref} is then tuned to yield an overshoot of approximately 5 %. By subsequently synthesizing the feedforward filter such that the closed-loop tracking dynamics match this reference model as closely as possible, the transient performance is improved consistently across all design points.

IV. Control Design and Analysis for a Symmetric Quadcopter

To gain insight into robust controller synthesis of an outer loop attitude controller for an INDI system, a controller is synthesized for the quadcopter described in subsection II.F. The uncertainty model is first simplified for a quadcopter with equal motor properties, followed by constraint selection, tuning, and system analysis.

A. Simplifications to Uncertainty Structure

Since a quadcopter with identical motor properties is considered, the uncertainty structure is simplified such that the perturbed control effectiveness values and the actuator time constant are equal for all motors. Consequently, the perturbed parameters in the control effectiveness matrix E_p and actuator dynamics $A_p(s)$ are identical for each motor i, as shown in Equation 19.

$$\begin{aligned} C_{\dot{p}_{i_p}} = C_{\dot{p}_p} &= C_{\dot{p}} \left(1 + r_C \delta_{C_{\dot{p}}}\right) \quad \text{with } |\delta_{C_{\dot{p}}}| < 1 \\ C_{\dot{q}_{i_p}} = C_{\dot{q}_p} &= C_{\dot{q}} \left(1 + r_C \delta_{C_{\dot{q}}}\right) \quad \text{with } |\delta_{C_{\dot{q}}}| < 1 \\ C_{\dot{r}_{i_p}} = C_{\dot{r}_p} &= C_{\dot{r}} \left(1 + r_C \delta_{C_{\dot{r}}}\right) \quad \text{with } |\delta_{C_{\dot{r}}}| < 1 \\ \tau_{i_p} = \tau_p &= \tau \left(1 + r_\tau \delta_\tau\right) \quad \text{with } |\delta_\tau| < 1 \end{aligned} \tag{19}$$

As a result, the perturbation matrices simplify to $\Delta_A(s) = \text{diag}(\delta_\tau, \Delta_{\tau_1}(s), \cdots, \Delta_{\tau_4}(s))$ and $\Delta_E = \text{diag}(\delta_{C_{\dot{p}}}, \delta_{C_{\dot{q}}}, \delta_{C_{\dot{r}}})$, reducing the number of uncertain parameters from twenty to eight.

B. Constraints and Weighting Filter Selection

The desired closed-loop characteristics that $K(s)$ must achieve are defined by a set of hard constraints that ensure effective disturbance rejection, sufficient stability margins, and nominal performance. These constraints will be detailed in this section.

Disturbance rejection: For output disturbance rejection, a constraint is imposed on the output sensitivity function $S_{o,\eta}$, which represents the transfer from the output disturbance $\boldsymbol{d_\eta}$ to the attitude output $\boldsymbol{\eta}$. This constraint has the form inf Equation 20, where $W_{S_{o,\eta}}(s)$ is a weighting filter the inverse of which represents the desired shape of $S_{o,\eta}$.

$$\|W_{S_{o,\eta}}(s) S_{o,\eta}(s)\|_\infty \leq 1 \tag{20}$$

Effective disturbance rejection requires the maximum singular value of $S_{o,\eta}$ to remain low at low frequencies (LF) [22]. To achieve this, a low-pass weighting filter $W_{S_{o,\eta}}(s)$ is employed, as defined in Equation 21. Because the frequency response is shaped by the inverse of the weighting filter, this corresponds to high-pass behavior in the sensitivity function. In this filter, $M_{S_{o,\eta}}$ specifies the desired gain at LF, $A_{S_{o,\eta}}$ the desired gain at high frequency (HF), and $\omega_{S_{o,\eta}}$ determines the crossover frequency.

$$W_{S_{o,\eta}}(s) = \frac{s/M_{S_{o,\eta}} + \omega_{S_{o,\eta}}}{s + \omega_{S_{o,\eta}} A_{S_{o,\eta}}} \tag{21}$$

The upper bound on $S_{o,\eta}$ constrains the skewed ($\sigma = 1$) disk margins, while its bandwidth determines disturbance rejection performance [25]. An upper limit of 6 dB is adopted, corresponding to a guaranteed gain margin of 6 dB and phase margin of 29°. Although this value is below the specified requirements, the transition region between the weighting filter slope and its HF gain effectively constrains the sensitivity peak. The actual constraint on the sensitivity function's peak is thus more severe.

A LF attenuation level of −50 dB is specified to ensure effective disturbance rejection, while a −3 dB bandwidth at 4.8 rad/s is imposed to define the desired crossover frequency[1].

To enhance rejection of input disturbances, a constraint is applied to the sensitivity times plant function S_oG, which defines the transfer from input disturbances $\boldsymbol{d_i}$ to the attitude output $\boldsymbol{\eta}$. The desired frequency response again follows a high-pass shape with a lower bound of −50 dB at LF to ensure sufficient attenuation. At HF, the singular values of S_oG closely follow those of the open-loop plant and therefore do not need to be

[1]To obtain the exact weighting filter coefficients, MATLAB's `makeweight` function is utilized [27].

constrained [22]. Consequently, a large upper bound of 30 dB is specified. A reference gain of 0 dB is set at 3.12 rad/s. This results in the second $\mathcal{H}_\infty$ constraint found in Equation 22.

$$\|W_{S_{oG}}(s)S_o(s)G(s)\|_\infty \leq 1 \tag{22}$$

Stability margins: To improve the multi-loop disk margins at the plant input $\boldsymbol{m_c}$ and output $\dot{\boldsymbol{\Omega}}$, a direct constraint on the disk margins is considered. Given an open-loop function L and its associated sensitivity function $S = I/(I+L)$, an $\mathcal{H}_\infty$ constraint on the minimum guaranteed disk stability margins can be formulated as Equation 23 [25]. Here, $\alpha_{\max}$ depends on the desired stability margins, and σ indicates the disk eccentricity.

$$\left\| \alpha_{\max} \left[S + \frac{\sigma - 1}{2} \right] \right\|_\infty \leq 1 \tag{23}$$

The guaranteed minimum and maximum gain margins ($\mathrm{GM_{min}}$, $\mathrm{GM_{max}}$) and phase margins ($\mathrm{PM_{min}}$, $\mathrm{PM_{max}}$) can be related to $\alpha_{\max}$ using Equation 24 and Equation 25 [25].

$$[\gamma_{\min}, \gamma_{\max}] = \left[\frac{2 - \alpha_{\max}(1-\sigma)}{2 + \alpha_{\max}(1+\sigma)}, \frac{2 + \alpha_{\max}(1-\sigma)}{2 - \alpha_{\max}(1+\sigma)} \right] \tag{24}$$

$$[\mathrm{PM_{min}}, \mathrm{PM_{max}}] = \left[-\arccos\left(\frac{1 + \gamma_{\min}\gamma_{\max}}{\gamma_{\min} + \gamma_{\max}} \right), \arccos\left(\frac{1 + \gamma_{\min}\gamma_{\max}}{\gamma_{\min} + \gamma_{\max}} \right) \right] \tag{25}$$

As mentioned in the requirements, symmetric margins are desired, which is why an eccentricity of $\sigma = 0$ is considered. These are also referred to as (S–T)-based disk margins, as they are related to the sensitivity S and complementary sensitivity T functions [25]. Although the constraint could be imposed directly on the multi-loop disk margins, it was instead applied to the margins at $\dot{\boldsymbol{\Omega}}$, since those were smaller than the margins at $\boldsymbol{m_c}$. Therefore, constraining the margins at $\dot{\boldsymbol{\Omega}}$ achieved a better balance of stability margins between the actuator input and plant output. The design requirements were met by enforcing a target gain margin of 6.75 dB and a phase margin of 40.6°, corresponding to $\alpha_{\max} = 0.740$.

Model following: For feedforward control design, a constraint is imposed on the model-following function M between the reference input $\boldsymbol{r_\eta}$ and the reference model following error $\boldsymbol{e_{\mathrm{ref}_\eta}}$. By minimizing the singular values of this function, the actual system dynamics are made to correspond more closely to the reference model. The reference model parameters, obtained after tuning the feedback controller, were $b_{\mathrm{ref}} = 12.93$, $\omega_{\mathrm{ref}} = 17.26$ rad/s, and $\zeta_{\mathrm{ref}} = 0.475$, following the procedure outlined in subsection III.C. This reference model has an improved 5 % settling time of 0.23 s and a rise time of 0.15 s, compared to 0.28 s and 0.19 s, respectively, obtained with the feedback controller alone.

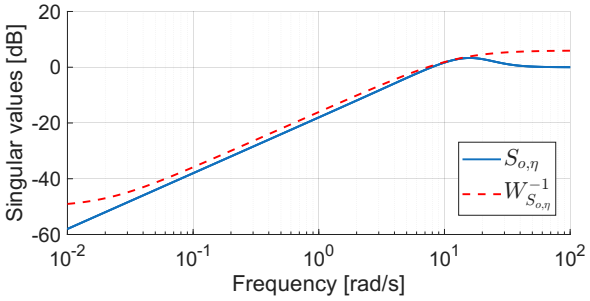
Since model matching cannot be achieved over infinite bandwidth, a high-pass reference shape is used. For this, a gain of −50 dB at LF and 0 dB at HF is considered. The bandwidth of the weighting filter was maximized, resulting in a singular value of −24.18 dB at 10 rad/s. This results in the final hard constraint shown in Equation 26.

$$\|W_M(s)M(s)\|_\infty \leq 1 \tag{26}$$

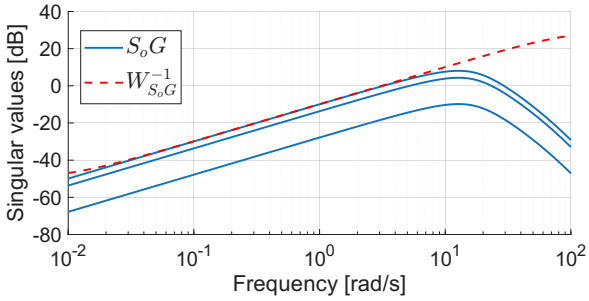
C. Tuning and Analysis

To synthesize the gains and feedforward filter while satisfying the previously defined constraints, a non-smooth optimization method is employed, enabling structured, multi-objective $\mathcal{H}_\infty$ controller synthesis [10]. Practically, this is performed using the `systune` function from the MATLAB Control Design Toolbox, where all specified constraints are treated as hard requirements that the optimization must satisfy [27]. The feedback controller is first synthesized based on the disturbance rejection and stability margin constraints, after which its gains are fixed and a feedforward filter is synthesized to satisfy the model-following constraint.

The singular values of the frequency responses and the corresponding weighting filters for the first two hard constraints on $S_{o,\eta}$ and S_oG are shown in Figure 10a and Figure 10b. Multiple singular value lines appear for S_oG due to differing control effectiveness in each axis. The integrator-like behavior of the INDI controller is also evident in the slope of S_oG at LF, as S_oG approximates the inverse of the controller at LF [22]. The resulting disk margins at $\dot{\boldsymbol{\Omega}}$ yield a GM of 6.75 dB and a PM of 40.63° which satisfies the stability margin constraint.



(a) $S_{o,\eta}$ **and** $W_{S_{o,\eta}}^{-1}$**.**



(b) S_oG **and** $W_{S_oG}^{-1}$**.**

Figure 10. Overview of disturbance rejection hard constraints after tuning.

The singular values of M and its weighting filter, shown in Figure 11, confirm that the constraint is satisfied. Furthermore, Figure 12 demonstrates that the desired 5 % overshoot and reference model following are achieved, particularly during the initial rise.

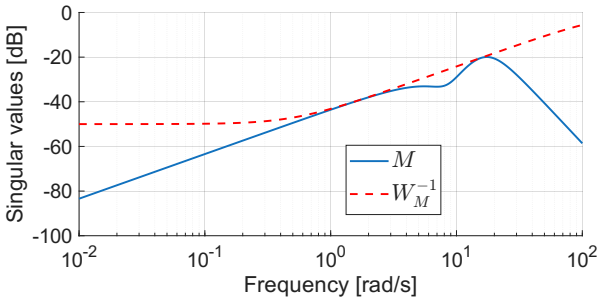


Figure 11. Model reference following function M **and its weighting filter** W_M^{-1}**.**

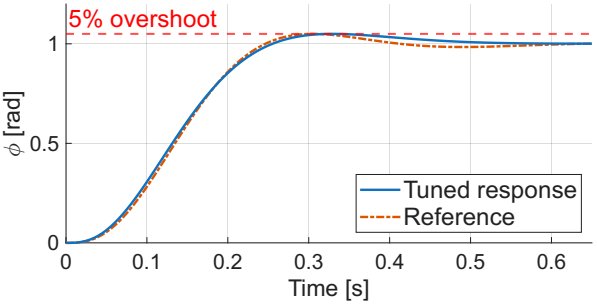


Figure 12. Nominal step response of the synthesized controller compared to the reference model.

The tuned feedback gains K_η and K_Ω, as well as the feedforward pole b_{ff} and zero a_{ff} are given in Equation 27.

$$K_\eta = 6.49, \quad K_\Omega = 16.60, \quad a_{\text{ff}} = 8.22, \quad b_{\text{ff}} = 10.73 \tag{27}$$

D. Frequency Domain Analysis

The disturbance rejection of angular acceleration disturbances and the noise attenuation at the plant input and output were not directly constrained during synthesis. To evaluate the resulting disturbance rejection capability of the closed-loop system, Figure 13 presents the frequency response from the angular acceleration disturbance $\boldsymbol{d}_{\dot{\boldsymbol{\Omega}}}$ to the attitude output $\boldsymbol{\eta}$. The small gain at LF indicates that disturbances in angular acceleration are effectively attenuated at the attitude output. Together with the characteristics of S_oG, this confirms that requirement R4 is satisfied.

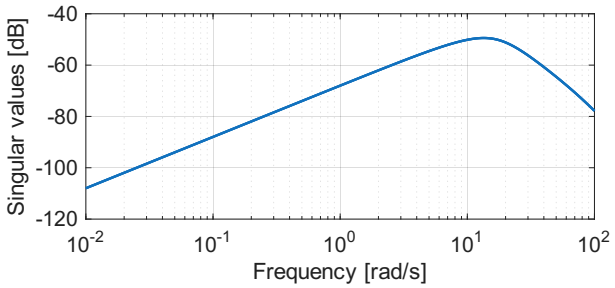


Figure 13. Frequency response from the disturbance in angular acceleration $d_{\dot{\Omega}}$ to the attitude η.

Noise attenuation at the plant output is analyzed through the frequency response between the gyroscope noise n_{Ω} and the angular rate output Ω, shown in Figure 14. The HF roll-off starts well before the actuator crossover frequency of 35.7 rad/s, demonstrating that the applied synchronization filter provides sufficient noise rejection. To illustrate the benefit of including the noise-shaping filter $N(s)$ in the analysis, the response assuming white noise is shown for comparison. Without this shaping filter, the response would suggest the need for a lower synchronization filter crossover frequency to achieve comparable attenuation, which would degrade performance [5]. The frequency response from gyroscope noise to the actuator input is depicted in Figure 15. A similar roll-off below the actuator crossover frequency confirms that noise attenuation is achieved at the actuator input as well. This confirms that, for this noise model and the used synchronization filter, the closed-loop system achieves noise attenuation at both the actuator input and plant output, thereby satisfying requirement R5.

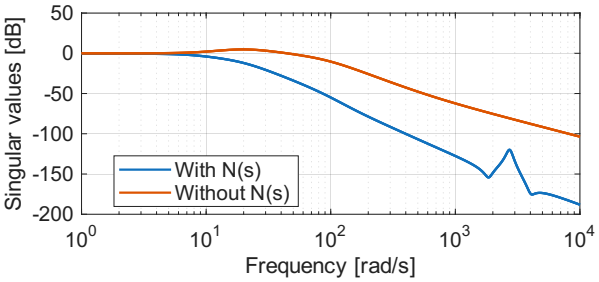


Figure 14. Frequency response from the gyroscopic noise n_{Ω} to the angular rate output Ω.

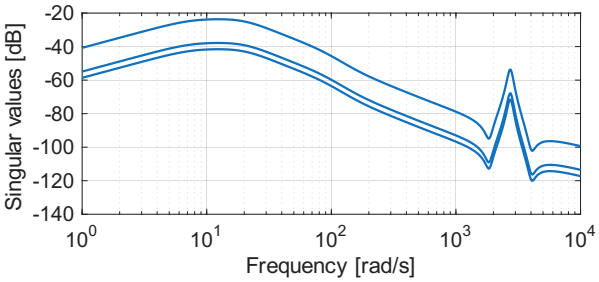


Figure 15. Frequency response from the gyroscopic noise n_{Ω} to the motor setpoints m_c.

E. Stability Margins

Stability margins are a key indicator of robustness, as emphasized in the design requirements, and are analyzed in this section. The nominal and worst-case classical stability margins and disk margins at the actuator inputs and plant outputs are summarized in Table 1. The worst-case values correspond to the minimum margins that may occur under the modeled uncertainty and were computed using MATLAB's `wcdiskmargin` function [27]. This function employs μ-analysis to determine lower and upper bounds on the worst-case disk margins. The absolute lower bound of the disk margins is obtained by solving a convex optimization problem [20], whereas the upper bound is found heuristically based on an actual uncertain realization. Convergence of the two bounds was achieved, indicating that the true worst-case margins were successfully identified.

The required nominal GM of 4 dB and PM of 35° are satisfied for all open-loop channels except at the angular rate and angular acceleration outputs. Although the GM values at these outputs even suggest instability, this apparent issue arises from the shape of the open-loop frequency response combined with the

classical GM definition. The more conservative disk margins confirm that the system remains stable and satisfies the requirements. Under the modeled uncertainty, the minimum required GM of 2 dB and PM of 17.5° are also met. The multi-loop disk margins evaluated at the actuator input and angular acceleration output yield a GM of 3.06 dB and a PM of 19.77°, satisfying requirement R3.

Table 1. Nominal and worst-case (S–T) based disk margins and classical stability margins.

Broken Loop	Nominal				Worst-case			
	DGM [dB]	DPM [°]	GM [dB]	PM [°]	DGM [dB]	DPM [°]	GM [dB]	PM [°]
ϕ / θ / ψ output	± 10.09	± 55.24	14.81	67.35	± 7.98	± 46.51	8.35	66.60
p / q/ r output	± 6.75	± 40.63	-630.5	42.32	± 3.33	± 21.43	-8.27	22.89
$\dot{p}$ / $\dot{q}$/ $\dot{r}$ output	± 6.75	± 40.63	-12.17	42.32	± 3.33	± 21.43	-484.3	22.89
m_{c_i} input	± 9.98	± 54.84	∞	57.12	± 3.48	± 22.34	5.54	28.48

F. Robust Stability Analysis

Although the stability margins provide a general indication of robustness, analyzing robustness with respect to the explicit uncertainty model offers deeper insight into how the modeled uncertainties affect closed-loop stability. To this end, a structured singular value (μ) analysis is performed on the uncertain system [20]. The value of μ is computed for each frequency, where its inverse represents the scaling factor of the modeled uncertainties that would lead to instability. The corresponding upper and lower bounds are shown in Figure 16. The convergence of the peak values indicates that the true worst-case μ has been obtained. The resulting largest value of $\mu = 0.47$ implies that the modeled uncertainties could be scaled by a factor of 2.13 before instability occurs. This represents a substantial robustness margin before the critical value of 1 is reached indicating good robustness against the modeled uncertainties.

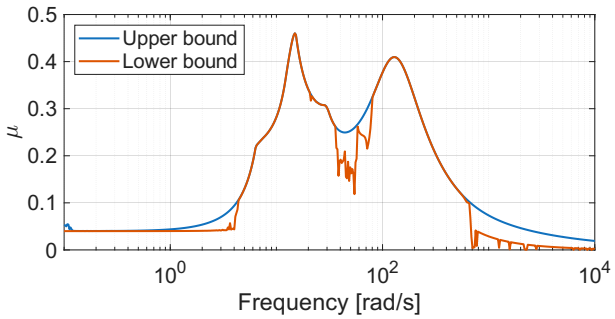


Figure 16. Structured singular value (μ) analysis for robust stability assessment.

V. Symmetric Quadcopter Results

This section describes the evaluation of the synthesized controller based on non-linear simulation and experimental flights.

A. Simulation Results

To assess the nominal and robust performance in the time domain, a pitch tracking task was simulated with nominal and uncertain plant parameters. Pitch was selected because the platform exhibits lower control

effectiveness around this axis than in roll, making it more critical for assessing actuator saturation risk. The uncertain plant parameters were generated from a grid of five equally spaced points between their respective minimum and maximum values for $C_{\dot{p}_p}$, $C_{\dot{q}_p}$, $C_{\dot{r}_p}$, and τ_p. In addition, a worst-case realization was obtained corresponding to the combination of uncertain parameters producing the largest $\mathcal{H}_\infty$ norm of $S_{o,\eta}$, given its direct relationship to tracking performance [22]. This realization was obtained using MATLAB's `wcgain` function [27].

The simulated pitch tracking performance under uncertainty is shown in Figure 17. In the nominal case, the system achieves a rise time of 0.15 s and a 5 % settling time of 0.23 s. Under uncertainty, additional oscillations and a worst-case overshoot of 11 % occur, after which the responses converge toward the nominal trajectory. Overall, no significant degradation in performance is observed. The normalized angular velocity of the first motor during the maneuver is shown in Figure 18 which shows that saturation is avoided by a wide margin, even under uncertainty. Due to the symmetry of the quadcopter and the motors being equal, similar results occur for the other motors and are omitted. Notably, the worst-case control effort peaks at twice the nominal value at approximately 0.8 s, reflecting the reduced control effectiveness and nonlinear relationship between motor speed and thrust. While not problematic in this case, it highlights that uncertainty in control effectiveness may become problematic for configurations with low control effectiveness.

One can observe that only a subset of the simulated realizations produces visibly distinct responses in both the pitch angle and motor speed, despite the total of 625 possible parameter combinations. This arises from the simplified uncertainty model, in which all motors retain identical control effectiveness about each body axis, even under uncertainty. This symmetry in thrust contributions prevents coupling effects between the rotational axes. Consequently, in this purely pitch-tracking maneuver, only variations in $C_{\dot{q}}$ and τ noticeably influence the system response.

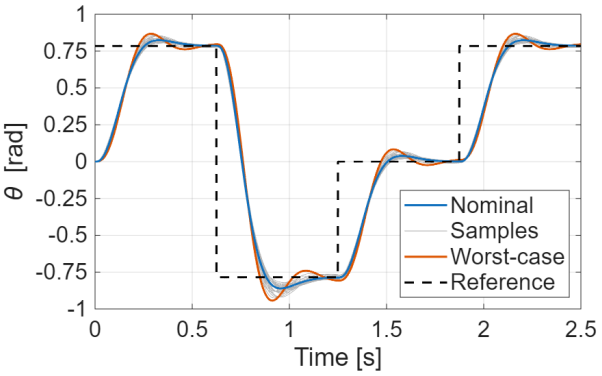


Figure 17. Simulated pitch angle tracking performance under uncertainty.

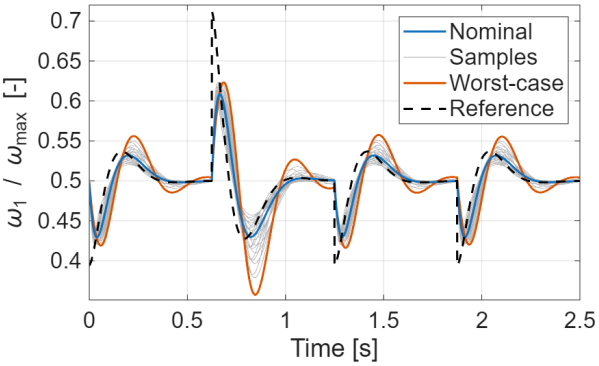


Figure 18. Simulated normalized motor angular velocity during pitch tracking under uncertainty.

B. Experimental Results

To validate the tuning model and assess robust performance experimentally, a set of roll doublet maneuvers was conducted using the platform described in subsection II.F. The experiment was performed under varying actuator time constants and control effectiveness parameters to evaluate controller performance under uncertainty. These variations were implemented by intentionally setting incorrect control effectiveness coefficients in the INDI controller and by slowing the actuator response to the desired time constant τ_{des} using a lag filter on the motor setpoints. Although modifying the estimated control effectiveness coefficients differs from the uncertainty model, in which the true control effectiveness itself varies, it produces a comparable mismatch between the INDI controller and the actual system.

Actuator time constants of $\tau = [17, 22, 28, 34, 40]$ ms were tested, spanning a ±40 % range around the nominal value of 28 ms. The control effectiveness coefficients were varied as $C_{\dot{p}_{\text{est}}} = [320, 400, 480]$, $C_{\dot{q}_{\text{est}}} = [218, 260, 312]$, and $C_{\dot{r}_{\text{est}}} = [40.8, 51, 61.2]$ to span the ±20 % uncertainty range for each coefficient. For each τ, the coefficients were set to their minimum, nominal, or maximum values, resulting in 15 configurations. Combinations of mixed minimum, nominal, and maximum values between the coefficients were omitted, as simulations indicated that tracking performance in pitch is primarily influenced by $C_{\dot{q}}$. Because the doublet is performed in roll, the focus was on covering the corresponding variations in $C_{\dot{p}}$ for each τ. To evaluate the influence of the feedforward filter, each configuration was tested both with and without feedforward, yielding a total of 30 flights.

Figure 19 shows the roll response to the doublet reference input for all flights. Only minor degradation in reference tracking performance is observed under uncertainty, indicating robust controller characteristics. The feedforward filter also improves the nominal rise time from 0.236 s to 0.178 s. In both cases, an unexpected steady-state error appears, that was not present in simulation and could not be conclusively explained. Possibly, this is due to increasing aerodynamic disturbances as the drone accelerates during the step up and down. Notably, no steady-state error is observed when the drone levels after the doublet, indicating that the controller can achieve a steady-state error of zero.

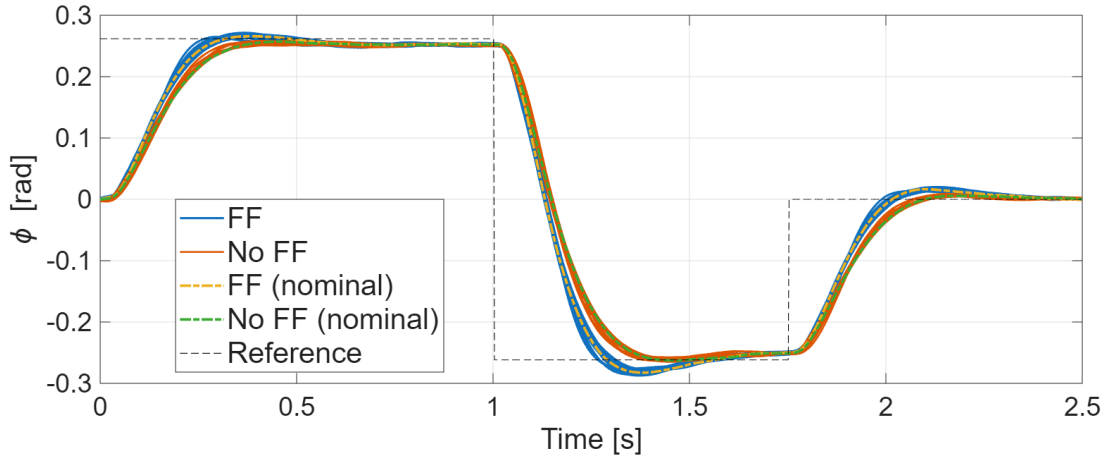


Figure 19. Roll response to a doublet input for a quadrotor under varying control effectiveness and actuator time constants, with and without feedforward.

To validate the accuracy of the tuning model, the experimental responses are compared with simulation results. Figure 20 presents simulated and experimental roll responses for varying actuator time constants τ, with $C_{\dot{p}_{\text{est}}} = 480$ rad/s^2. The model captures the initial rise accurately but diverges beyond approximately 0.15 rad, where the experiments exhibit a less pronounced overshoot. This discrepancy likely arises from unmodeled aerodynamic effects, such as increased damping due to rising velocity during the step maneuver. Nevertheless, the influence of motor dynamics is clearly captured, and the overall experimental trends remain consistent with the simulations. Similar agreement is observed in Figure 32 in the appendix, which compares simulated and experimental responses for different values of $C_{\dot{p}_{\text{est}}}$ with a fixed $\tau = 28$ ms.

Figure 21 shows the comparison for the return-to-level portion of the doublet. The fit is notably better for a larger portion of the initial rise which suggests that the velocity, which is significantly reduced at this point due to the doublet, is a contributing factor. However, confirming this requires further investigation in future work.

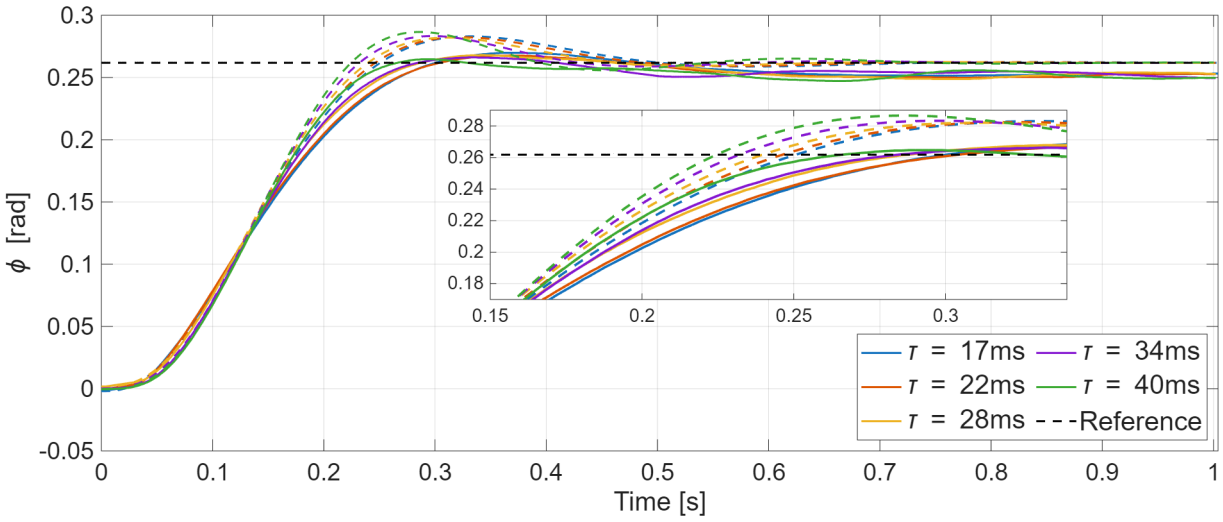


Figure 20. Experimental (solid) and simulated (dashed) roll response with uncertain parameters for $C_{\dot{p}\,\mathbf{est}} = 480$ rad/s^2 and different τ.

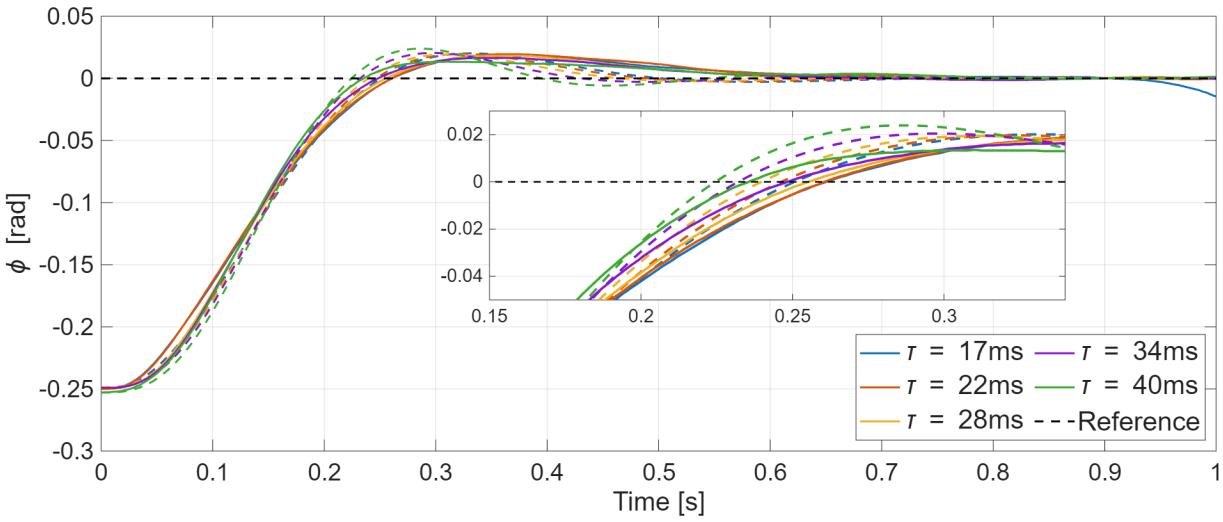


Figure 21. Experimental (solid) and simulated (dashed) roll response with uncertain parameters for $C_{\dot{p}\,\mathbf{est}} = 480$ rad/s^2 and different τ.

VI. Gain-Scheduled Controller Design and Analysis

This section extends the previous controller design to a gain-scheduled formulation suitable for platforms that identify their actuator properties online. The underlying assumptions of the gain-scheduled controller are first discussed, followed by the tuning procedure and the corresponding controller analysis.

A. Design Setup and Assumptions

In this work, the gain-scheduled controller uses the actuator time constant τ as the scheduling variable, as it primarily characterizes the nominal system dynamics under ideal INDI control. This choice neglects variations in control effectiveness, which mainly influence the required actuator effort. Lower control effectiveness values demand larger control inputs to achieve the same response, making platforms with low control effectiveness more susceptible to actuator saturation. Although incorporating control effectiveness into the scheduling scheme could reduce this risk, this aspect is not addressed here.

For the controller synthesis, a nominal control effectiveness of 300 $\mathrm{rad/s^2}$ per motor and axis is assumed as a baseline, comparable to the control effectiveness of the experimental quadcopter platform.

B. Constraints and Weighting Filter Selection

The $\mathcal{H}_\infty$ constraints imposed on the singular values of selected input–output frequency responses for the gain-scheduled controller synthesis are described below.

Disturbance rejection: As in the previous design, a constraint on $S_{o,\eta}$ is imposed to ensure disturbance rejection. Selecting an appropriate weighting filter for each design point is challenging because the actuator time constant directly influences the plant bandwidth and smaller bandwidths reduce robustness [28]. Using a single filter across all design points would therefore require a conservative bandwidth to accommodate the slowest dynamics. To preserve performance throughout the operating range, a co-design approach is adopted wherein the weighting filter parameters are optimized jointly with the controller gains [29, 30].

The weighting filters can be expressed as rational functions composed of gains and integrators, enabling their parameters to serve as tunable variables within the structured synthesis framework. In the case of the sensitivity weighting filter $W_{S_{o,\eta}}(s)$ in Equation 21, the frequency parameter $\omega_{S_{o,\eta}}$ is set as tunable so the bandwidth can be optimized. To incorporate this into the optimization, a transfer function of the form in Equation 28 is used to impose an $\mathcal{H}_\infty$ maximization constraint [30]. The variable N must be assigned a large value to ensure numerical stability and is set to 1000.

$$O_{S_{o,\eta}} = \frac{i_{S_{o,\eta}}}{o_{S_{o,\eta}}} = \frac{N}{1 + N\omega_{S_{o,\eta}}} \tag{28}$$

The optimization framework employed here enables multi-objective control design by minimizing selected soft constraints while ensuring strict satisfaction of all hard constraints [11]. Within this framework, $\|O_{S_{o,\eta}}\|_\infty$ is treated as a soft constraint, allowing the sensitivity bandwidth to be maximized as much as the other constraints allow. For the reference shape of $S_{o,\eta}$, upper and lower bounds are still required. An upper limit of 6 dB and a lower limit of −50 dB are imposed, in line with the design for the quadcopter with identical motor properties. Regarding input disturbance rejection, the integral behavior of the INDI controller means that constraining $S_o G$ is not strictly necessary and is therefore omitted to simplify the tuning process.

Stability margins: The disk margins at the plant output $\dot{\boldsymbol{\Omega}}$ are again constrained to improve the multi-loop disk margins for the actuator input and plant output. A minimum GM of 6.99 dB and a minimum PM of 41.8° are enforced for all design points, corresponding to an $\alpha_{\max}$ of 0.764.

Model following: To achieve reference model tracking, the bandwidth ω_M of the weighting filter $W_M(s)$ is co-designed in the same manner as $S_{o,\eta}$. A lower limit of −90 dB and an upper limit of 0 dB are applied. The lower bound is lower than the −50 dB used previously because the upper singular value of M falls below

this level for certain design points. The reference model $T_{\text{ref}}(s)$ is defined individually for each design point after tuning the feedback controller, following the procedure described in subsection III.C.

C. Gain-scheduled Controller Tuning and Analysis

The gain-scheduled controller was synthesized at 30 linearly spaced design points corresponding to actuator time constants between 10 ms and 80 ms. The resulting $S_{o,\eta}$ responses obtained after tuning the feedback controller are shown in Figure 22. Smaller actuator time constants yield larger $S_{o,\eta}$ bandwidths, demonstrating that the co-design approach effectively adapts the bandwidth to the actuator dynamics.

The result of the model-following constraint is illustrated in Figure 23. Frequency responses with higher crossover frequencies, which indicate closer matching to the reference model, correspond to lower values of τ. Although higher bandwidth generally implies improved model tracking, all step responses consistently exhibit overshoot values between 4.71 % and 4.99 %. This demonstrates that combining the method for reference model selection with the co-design of $W_M(s)$ yields consistent characteristics across all design points.

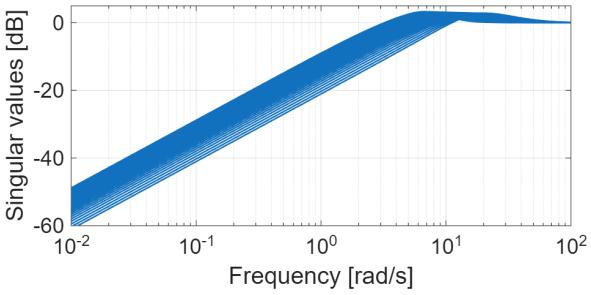


Figure 22. $S_{o,\eta}$ frequency response for each design point.

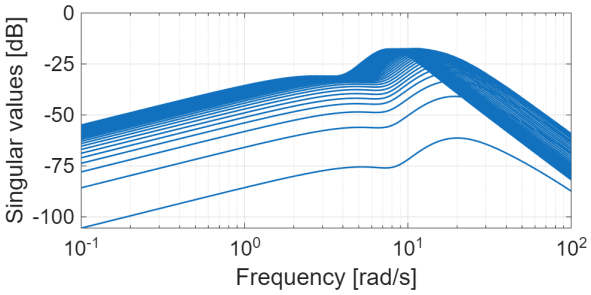


Figure 23. M frequency response for each design point.

The resulting values for K_η and K_Ω are shown in Figure 24, illustrating a smooth decrease as τ increases. Figure 25 shows the identified pole and zero for the feedforward filter, indicating that lead is added across the entire range except at the design point with the smallest τ. Although the schedules of the pole and zero are more complex than those of the gains, the use of 30 design points captures them adequately.

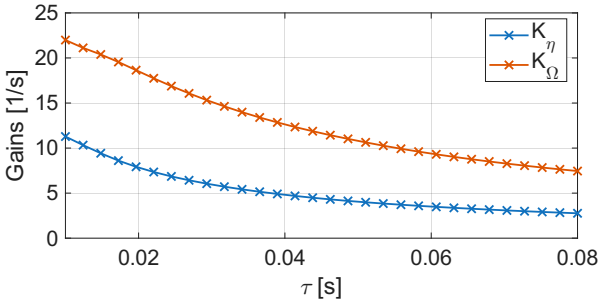


Figure 24. Tuned K_η and K_Ω for different values of τ.

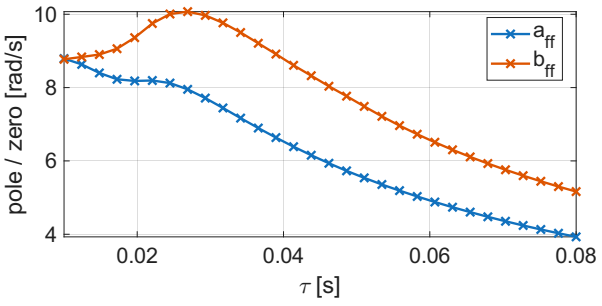
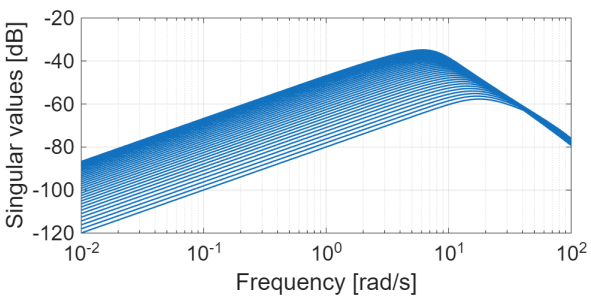


Figure 25. Tuned b_{ff} and a_{ff} for different values of τ.

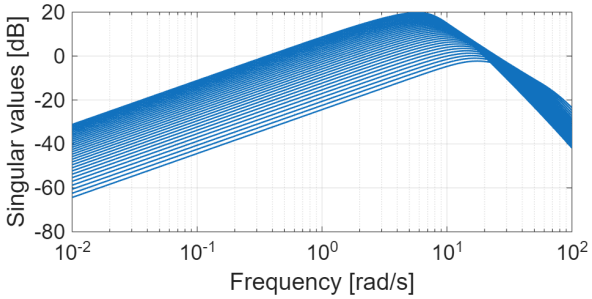
D. Frequency Domain Analysis

To evaluate the closed-loop behavior of the gain-scheduled controller, frequency-domain responses relevant to disturbance rejection are analyzed. Figure 26a shows the frequency response from the angular acceleration

disturbance $\boldsymbol{d}_{\dot{\Omega}}$ to the attitude output $\boldsymbol{\eta}$ for all design points. Slower actuators exhibit a lower crossover frequency, reflecting reduced disturbance rejection performance resulting from the smaller plant bandwidth. For applications in which disturbance rejection is critical, slower actuator dynamics may therefore require higher-order control structures such as those proposed in [8]. Since the present study focuses on more general performance requirements, the reduced disturbance rejection capability for slow actuators is considered acceptable. Similar trends are observed in the frequency response S_oG between the input disturbance $\boldsymbol{d_i}$ and attitude output $\boldsymbol{\eta}$ shown in Figure 26b. As the degradation of output and input disturbance rejection performance is comparable, not explicitly constraining S_oG still yields a balanced solution.



(a) Frequency response from $d_{\dot{\Omega}}$ to η.



(b) Frequency response from d_i to η.

Figure 26. Frequency responses relevant to disturbance rejection for all design points.

The frequency responses relevant to noise attenuation at the actuator input and plant output are provided in the appendix in Figure 26. These results show that the gain-scheduled controller has little influence on overall noise-rejection behavior, and the conclusions from subsection IV.D remain valid. This makes sense, as the main challenge lies in achieving roll-off at high frequencies, which is not influenced by changes in the outer-loop controller gains.

E. Stability Margins

As with the symmetric quadcopter configuration, the stability margins are evaluated to assess the robustness of the closed-loop system. The smallest nominal and worst-case stability margins for all design points are summarized in Table 2. It was found that the upper and lower bounds of the μ-analysis used to compute the worst-case disk margin limits did not converge. Therefore, a range between the theoretical lower bound and the heuristically determined upper bound of the worst-case disk margins is presented. This behavior results from the increase in the number of uncertain parameters from eight to twenty relative to the simplified model used previously, which can yield less tight bounds [31]. It should also be noted that the worst-case classical margins are based on heuristically found uncertain realizations and therefore represent upper bounds.

The results indicate that the nominal stability margin requirements of 4 dB gain margin (GM) and 35° phase margin (PM) are satisfied. The worst-case classical gain and phase margins also meet the required GM of 2 dB and PM of 17.5°, accounting for the corresponding disk-based gain margins of the angular rate and angular acceleration. The lower bounds of the worst-case disk phase margins do suggest that the actual worst-case classical phase margins may not fully satisfy R2. However, these lower bounds represent theoretical limits, and due to the 20 uncertain parameters, reaching such conditions in practice could be unlikely. To avoid making the design overly conservative for these conditions, the slightly reduced lower bounds are still considered acceptable as long as robust stability is maintained. The multi-loop disk margin evaluated at the actuator input and angular acceleration output yields a gain margin of 3.01 dB and a phase margin of 19.52°, thereby satisfying requirement R3.

Table 2. Smallest nominal and worst-case (S–T) based disk margins and classical stability margins across the design range.

Broken Loop	Nominal				Worst-case			
	DGM [dB]	DPM [°]	GM [dB]	PM [°]	DGM [dB]	DPM [°]	GM [dB]	PM [°]
ϕ / θ / ψ output	± 9.55	± 53.16	13.12	63.07	± 5.69–6.72	± 30.91–40.46	8.49	62.63
p / q / r output	± 6.99	± 41.80	-671.04	41.81	± 2.41–2.82	± 15.70–18.31	-658.00	18.81
$\dot{p}$ / $\dot{q}$ / $\dot{r}$ output	± 6.99	± 41.80	-497.36	41.81	± 2.41–2.82	± 15.72–18.31	-489.20	18.81
m_{c_i} input	± 8.65	± 49.47	∞	51.70	± 2.16–2.43	± 14.07–15.79	4.91	26.67

F. Robust Stability Analysis

To analyze robustness with respect to the modeled uncertainty in greater depth, a μ-analysis was performed on the uncertain closed-loop system for each design point. The resulting upper and lower bounds are presented in Figure 27. As with the worst-case disk margins, convergence between the upper and lower bounds was not achieved. The upper bound indicates robust stability, with a maximum μ value of 0.82. Although the nominal stability margins are similar, this value is significantly larger than that obtained for the symmetric quadcopter, indicating reduced robustness against the full uncertainty model. This result can be explained by the increase in uncertain parameters from eight to twenty, which substantially expands the uncertainty domain. Although this margin is smaller, the gap between the upper and lower bounds suggests that the result may be conservative, as the true value of μ can lie anywhere between them. In this context, designing for a smaller upper bound could lead to an overly conservative controller. The next section presents simulations based on structured gridding of uncertain parameters and Monte Carlo simulations to evaluate the practical performance under uncertainty.

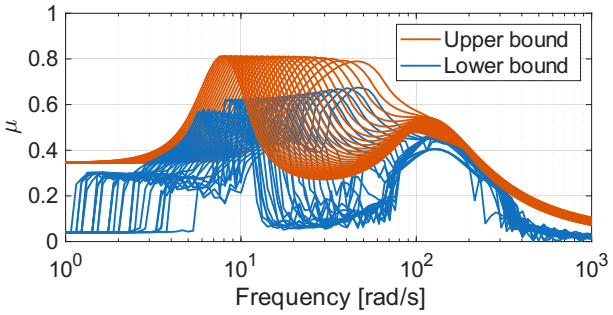


Figure 27. Upper and lower bound of μ for each design point.

VII. Gain-Scheduled Controller Results

This section describes the validation of the synthesized gain-scheduled controller through nonlinear simulation and experimental flights.

A. Simulation Results

To assess the nominal and robust tracking performance of the gain-scheduled controller, a roll reference tracking task was simulated for each design point under both nominal and uncertain conditions. Because the system includes 16 parameters with parametric uncertainty, a full structured grid approach considering the minimum, nominal, and maximum values would result in over 10^7 realizations. This number was too large to simulate, and therefore a Monte Carlo simulation was used. In this approach, individual control effectiveness parameters and motor time constants were randomly sampled from five linearly spaced values

within their respective bounds over 1000 iterations. The uncertain realizations from a structured grid based on equal motor properties, as well as the realization causing the largest $\mathcal{H}_\infty$ norm in $S_{o,\eta}$, were also simulated to increase the covered uncertainty space. Although this method does not guarantee that the absolute worst-case scenario is considered, the combination of the identified worst case with the set of uncertain realizations should provide a good representation of practical performance.

The nominal roll response and the bounds corresponding to all uncertain realizations are presented in Figure 28. For clarity, the 30 actuator time constant values, are grouped into five sets of six. Under uncertainty, the initial step response exhibits additional overshoot, which becomes more pronounced for larger τ values, reaching a peak of 11.8 % for the slowest actuators. Nevertheless, after this initial peak the bounds converge again and no significant degradation in performance is observed.

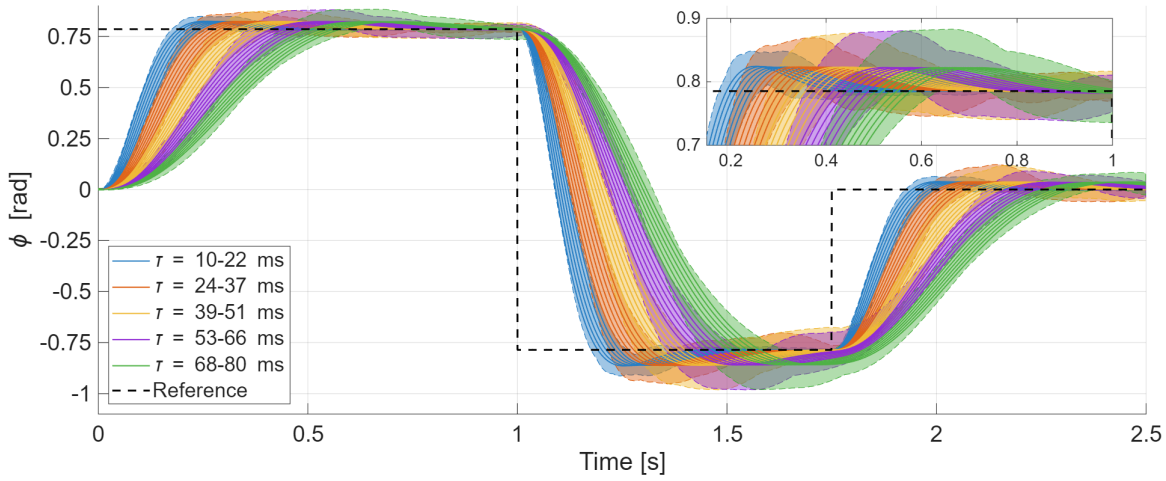


Figure 28. Simulated roll response to a doublet input with bounds under uncertainty for the entire gain-schedule domain.

In contrast to the quadcopter with identical motor properties, individual uncertainties in the control effectiveness coefficients may break the symmetry of the thrust distribution. This asymmetry leads to coupling between roll, pitch, and yaw, as control inputs on one axis no longer generate purely decoupled torques but also induce secondary moments around the other axes. From the simulation, it was found that this coupling caused pitch and yaw angles of at most 4.5° during the down step. This corresponds to a maximum coupling effect of 5 % relative to the roll angle reference difference of 90° at that point.

Figure 29 shows the combined bounds for all motors, with distinct spikes during step inputs that become more pronounced for faster actuators. For reference, the nominal output of motor 1 is included. It aligns with some peaks but not all, since different motors produce opposing control actions during the steps. Under nominal conditions reasonable control behavior is observed, with increasing amplitude for faster actuators. This trend is expected, as faster actuators enable more aggressive control actions. Under uncertainty, a band can be observed between approximately 0.4 and 0.58. This arises from unequal control effectiveness coefficients across the body axes, which can yield equilibrium states with nonuniform motor setpoints. Consequently, the steady-state motor values during hover are not always equal to $0.5\omega_{\max}$ but fall somewhere within this range. This imbalance can significantly influence control actuation due to the nonlinear relationship between thrust and motor angular rate ω, described by $T = k\omega^2$. The effect is particularly evident when the motor setpoint decreases, as indicated by the pronounced downward peaks. For the fastest actuators, the minimum motor speed reaches approximately 10 % of the maximum spin rate, narrowly avoiding saturation.

The actuator limits being close to saturation suggests that under these uncertainties, the current gain-

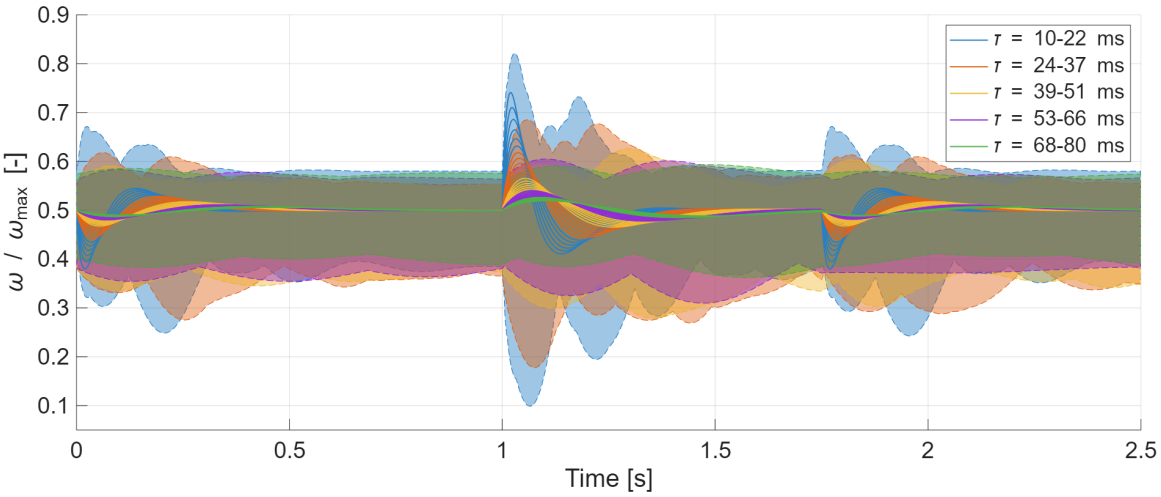


Figure 29. Simulated nominal response of motor 1 to a doublet input, with bounds under uncertainty for all motors.

scheduled controller may be unsuitable for platforms with low control effectiveness and fast actuators. However, analyzing only the attitude-control loop provides an incomplete picture, as altitude control in practice would increase overall actuator usage during the doublet maneuver, mitigating some nonlinear effects. Furthermore, the nominal hover operating point at $0.5\omega_{\max}$, corresponding to 25% thrust, represents a high-performance configuration such as the experimental platform. Other platforms with a larger thrust at hover will experience weaker nonlinear effects as the control action is overall higher. Nevertheless, these findings indicate that extending the gain-scheduling approach to account for variations in control effectiveness, for instance by introducing an additional constraint on the frequency response between $\boldsymbol{r_\eta}$ and $\boldsymbol{m_c}$, could help prevent actuator saturation in faster systems.

B. Experimental Results

To validate the performance of the gain-scheduled controller under real-world conditions, flight tests were conducted that combined the online identification procedure proposed by Blaha et al. [3] with the gain-scheduled controller. Using this method, the drone can be thrown into the air without prior system knowledge and rapidly identifies its motor parameters, which are then used within the INDI controller to stabilize the platform. After stabilization, the controller gains and feedforward filter parameters were determined from the maximum identified actuator time constants using linear interpolation of the gain schedules. Subsequently, a doublet maneuver in roll was performed to assess tracking performance.

The experiments were conducted using the platform described in subsection II.F, with the actuator dynamics slowed to five time constants, $\tau_{\text{exp}} = [17, 28, 40, 60, 80]$ ms, using a lag filter. Five throws were performed for each time constant, totaling 25 flights. For altitude control, knowledge of each motor's thrust effectiveness along the z-axis f_z was required. These values were taken from ground-truth bench test data rather than identified online, as the focus of these experiments was on attitude control [3]. For unknown reasons, the maximum motor rotational rate, $\omega_{\max}$, was severely misidentified during the experiments. While this issue was not observed in the original work, it led to significantly misidentified coefficients in the present experiments, since the control effectiveness terms are scaled with $\omega^2_{\max}$. As this made the drone uncontrollable, the value was instead hardcoded based on bench test results to enable the experiments to proceed.

Table 3 summarizes the ratios between the identified control effectiveness parameters ($C_{x_{\text{id}}}$) and actuator time constants (τ_{id}) relative to their nominal values across all throws. The results show that the identified

control effectiveness parameters can vary substantially, up to two to three times their nominal values, while the actuator time constants are consistently underestimated. The underlying cause of these discrepancies remains unclear. However, offline system identification using the same method yielded values much closer to nominal, suggesting a potential software-related issue. Although the experimental outcomes do not strictly fall within the defined uncertainty bounds, they still provide valuable insight into the system behavior under real conditions.

Table 3. Range of the ratio between the actual motor parameters and their identified values for all throws.

Motor	$C_{\dot{p}}/C_{\dot{p}\,\mathrm{id}}$ [%]	$C_{\dot{q}}/C_{\dot{q}\,\mathrm{id}}$ [%]	$C_{\dot{r}}/C_{\dot{r}\,\mathrm{id}}$ [%]	τ/τ_{id} [%]
1	149.0–203.3	182.3–269.1	211.9–446.9	65.0–87.1
2	181.8–235.6	195.7–313.9	14.4–161.4	71.8–90.7
3	171.6–220.6	118.5–198.0	192.3–422.5	80.7–90.6
4	127.3–171.8	107.7–163.4	22.4–107.7	62.2–80.8

The roll response to the doublet input after each throw is shown in Figure 30. For a time constant of 80 ms, the platform was poorly controllable with the estimated control coefficients, and thus only one flight was performed. For actuator time constants of 40 ms and below, excessive overshoot can be observed. This behavior aligns with the simulation results in Figure 28, where underestimated time constants and overestimated control effectiveness lead to increased overshoot. Figure 31 further illustrates this by comparing the experimental and simulated responses for one flight per time constant. For time constants above 40 ms, a clear mismatch between simulation and experiment is evident. Although the exact cause remains unclear, manual flight observations indicated that these uncertainties led to poor controllability and noticeable coupling effects, which may have influenced the results. When the nominal control coefficients were used with the same gains, flight performance improved significantly. This suggests that the extreme uncertainties are the primary cause, but this should be further investigated in future work.

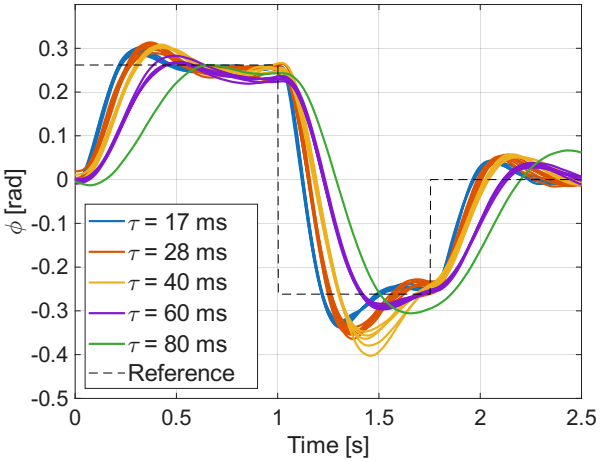


Figure 30. Roll response to a doublet input after online identification of INDI parameters during a throw for different actuator speeds.

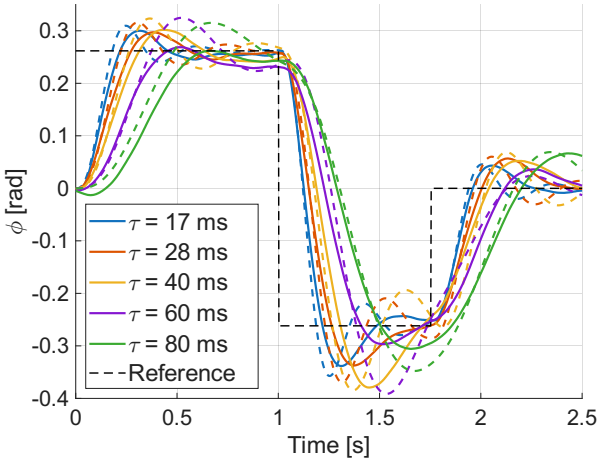


Figure 31. Experimental (solid) against simulated (dashed) roll response to a doublet input for different actuator speeds.

VIII. Conclusion and Future Work

This work presented the robust controller design for the outer loop of an INDI-controlled quadcopter. Both a controller specific for a symmetric quadcopter and a gain-scheduled controller over a wide range of actuator time constants were presented. The controllers were shown to have desired properties with regard to reference tracking, disturbance rejection, sensor noise attenuation, control signal attenuation, and good stability margins. Simulation and experimental validation of the controller for the symmetric quadcopter showed that with the proposed controller and tuning procedure, satisfactory robust performance was achieved. Some unexplained mismatches between simulation and reality, particularly related to steady-state error during non-zero reference tracking, were found and should be investigated further. Regarding the gain-scheduled controller, good robust performance under uncertainty was shown in simulation but actuator saturation for faster time constants could be a potential concern if control effectiveness is not high enough.

The gain-scheduled controller was further validated in practice by combining it with online parameter identification of all motor parameters. Since the online identification of these parameters deviated significantly from the nominal values obtained from bench tests and offline identification, it is difficult to draw definitive conclusions regarding the performance of the gain-scheduled controller. Nevertheless, for actuator time constants below 40 ms the tracking performance exhibited behavior consistent with the simulation results. This result is promising, as it demonstrates robustness of the controller to uncertainties larger than anticipated and suggests that the good nonlinear simulation results translate to real-world performance. For actuator time constants above 40 ms, the simulations no longer align with the experimental results and the quadrotor was poorly controllable. However, this appears to be caused by the extreme uncertainties as performance based on nominal parameters was significantly better. Further experiments under more realistic uncertainty levels are needed to fully validate the controller's robustness for slower actuators.

For future work, the steady-state error observed during experiments could be further investigated by incorporating additional effects, such as aerodynamics, into the model. Although the nominal model is applicable to any multirotor using this INDI system and equal gains, the current uncertainty analysis is limited to a quadrotor. Extending the analysis to configurations with additional rotors or unequal gains in different axes could provide insights into their effects on robustness. Furthermore, several uncertain parameters in the INDI controller, such as the B_2 matrix and the mapping from thrust setpoint to actual motor setting, were neglected in this study, and their effects under uncertainty could be explored. The 20 % uncertainty for the control effectiveness values was also partially based on intuition, and future work could explore a more systematic approach to obtain the uncertainty bounds. Finally, extending the gain schedule to account for control effectiveness could reduce the risk of actuator saturation for platforms with limited control authority, while extending this methodology to an outer position control loop would enable a complete control framework.

References

[1] Hassanalian, M., and Abdelkefi, A., "Classifications, Applications, and Design Challenges of Drones: A Review," *Progress in Aerospace Sciences*, Vol. 91, 2017, pp. 99–131. doi: 10.1016/j.paerosci.2017.04.003.

[2] Guebsi, R., Mami, S., and Chokmani, K., "Drones in Precision Agriculture: A Comprehensive Review of Applications, Technologies, and Challenges," *Drones*, Vol. 8, No. 686, 2024. doi: 10.3390/drones8110686.

[3] Blaha, T. M., Smeur, E. J. J., and Remes, B. D. W., "Control of Unknown Quadrotors from a Single Throw," *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2024, pp. 10350–10355. doi: 10.1109/IROS58592.2024.10801514.

[4] Blaha, T. M., Smeur, E. J. J., Remes, B. D. W., and de Visser, C. C., "Flying a Quadrotor with Unknown Actuators and Sensor Configuration," , Sep. 2024. doi: 10.48550/arXiv.2409.01080.

[5] 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.

[6] Sieberling, S., Chu, Q. P., and Mulder, J. A., "Robust Flight Control Using Incremental Nonlinear Dynamic Inversion and Angular Acceleration Prediction," *Journal of Guidance, Control, and Dynamics*, Vol. 33, No. 6, 2010, pp. 1732–1742. doi: 10.2514/1.49978.

[7] 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.

[8] Hachem, M., Roos, C., Miquel, T., and Bronz, M., "Improving Incremental Nonlinear Dynamic Inversion Robustness Using Robust Control in Aerial Robotics," , 2025. doi: 10.48550/arXiv.2501.07223.

[9] Cardenas, J. G. R., "INDI and $\mathcal{H}_\infty$ LSDP in Quadrotors," Master's Thesis, Delft University of Technology, Delft, Oct. 2024.

[10] Apkarian, P., and Noll, D., "Nonsmooth $\mathcal{H}_\infty$ Synthesis," *IEEE Transactions on Automatic Control*, Vol. 51, No. 1, 2006-01, pp. 71–86. doi: 10.1109/TAC.2005.860290.

[11] Apkarian, P., Gahinet, P., and Buhr, C., "Multi-Model, Multi-Objective Tuning of Fixed-Structure Controllers," *2014 European Control Conference (ECC)*, 2014-06, pp. 856–861. doi: 10.1109/ECC.2014.6862200.

[12] Apkarian, P., Dao, M. N., and Noll, D., "Parametric Robust Structured Control Design," *IEEE Transactions on Automatic Control*, Vol. 60, No. 7, 2015, pp. 1857–1869. doi: 10.1109/TAC.2015.2396644.

[13] Nguyen, D.-T., Saussié, D., and Saydy, L., "Robust Self-Scheduled Fault-Tolerant Control of a Quadrotor UAV," *IFAC-PapersOnLine*, Vol. 50, No. 1, 2017, pp. 5761–5767. doi: 10.1016/j.ifacol.2017.08.1141.

[14] Theodoulis, S., Proff, M., and Marchand, C., "Robust Design for Highly Agile Missile Autopilots," *2020 28th Mediterranean Conference on Control and Automation (MED)*, 2020, pp. 67–72. doi: 10.1109/MED48518.2020.9183304.

[15] Theodoulis, S., and Proff, M., "Robust Flight Control Tuning for Highly Agile Missiles," *AIAA SciTech Forum*, American Institute of Aeronautics and Astronautics, 2021. doi: 10.2514/6.2021-1568.

[16] Rhenman, F., and Theodoulis, S. T., "Robust Gain Scheduled Longitudinal Control of the ADMIRE Using Parametric Robust $\mathcal{H}_\infty$ Control," , 2025. doi: 10.2514/6.2025-2245.

[17] van Beers, J. J., and de Visser, C. C., "Peaking into the Black-box: Prediction Intervals Give Insight into Data-driven Quadrotor Model Reliability," *AIAA SCITECH 2023 Forum*, American Institute of Aeronautics and Astronautics, 2023. doi: 10.2514/6.2023-0125.

[18] Blaha, T. M., Smeur, E. J. J., and Remes, B. D. W., "A Survey of Optimal Control Allocation for Aerial Vehicle Control," *Actuators*, Vol. 12, No. 7, 2023, p. 282. doi: 10.3390/act12070282.

[19] Simplicio, P., Pavel, M., van Kampen, E., and Chu, Q., "An Acceleration Measurements-based Approach for Helicopter Nonlinear Flight Control using Incremental Nonlinear Dynamic Inversion," *Control Engineering Practice*, Vol. 21, No. 8, 2013, pp. 1065–1077. doi: 10.1016/j.conengprac.2013.03.009.

[20] Skogestad, S., and Postlethwaite, I., *Multivariable Feedback Control: Analysis and Design*, 2nd ed., John Wiley, Chichester, England, 2005.

[21] Ziemer, R., Tranter, W., and Fannin, D., *Signals and Systems: Continuous and Discrete*, Macmillan, 1993.

[22] Bates, D., and Postlethwaite, I., *Robust Multivariable Control of Aerospace Systems*, DUP Science, 2002.

[23] SAE International, "Aerospace—Flight Control Systems—Design, Installation and Test of Piloted Military Aircraft, General Specification For," Standard AS94900, 2007.

[24] Berrios, M., Berger, T., Tischler, M., Juhasz, O., and Sanders, F., "Hover Flight Control Design for UAS Using Performance-Based Disturbance Rejection Requirements," *Proceedings of the AHS 73rd Annual Forum*, 2017. doi: 10.4050/F-0073-2017-12202.

[25] Seiler, P., Packard, A., and Gahinet, P., "An Introduction to Disk Margins [Lecture Notes]," *IEEE Control Systems Magazine*, Vol. 40, No. 5, 2020-10, pp. 78–95. doi: 10.1109/MCS.2020.3005277.

[26] Ogata, K., *Modern Control Engineering*, 5th ed., Pearson, 2010.

[27] The MathWorks, *Robust Control Toolbox User Guide*, The MathWorks, Inc., 1 Apple Hill Drive, Natick, MA 01760-2098, r2020a ed., Mar. 2020.

[28] Stein, G., "Respect the Unstable," *IEEE Control Systems Magazine*, Vol. 23, No. 4, 2003, pp. 12–25. doi: 10.1109/MCS.2003.1213600.

[29] Gonzalez, J. A. P., "Commande Robuste Structurée : Application Co-design Mécanique / Contrôle d'Attitude d'un Satellite Flexible," , 2016.

[30] Pérez, C. A., Theodoulis, S., Sève, F., and Goerig, L., "Automatic Weighting Filter Tuning for Robust Flight Control

Law Design," *IFAC-PapersOnLine*, Vol. 55, No. 16, 2022-01-01, pp. 400–405. doi: 10.1016/j.ifacol.2022.09.057.

[31] Balas, G. J., Doyle, J. C., Glover, K., Packard, A., and Smith, R., *Analysis and Synthesis Toolbox User's Guide*, MathWorks, Natick, MA, 1993. User's Guide.

Appendix

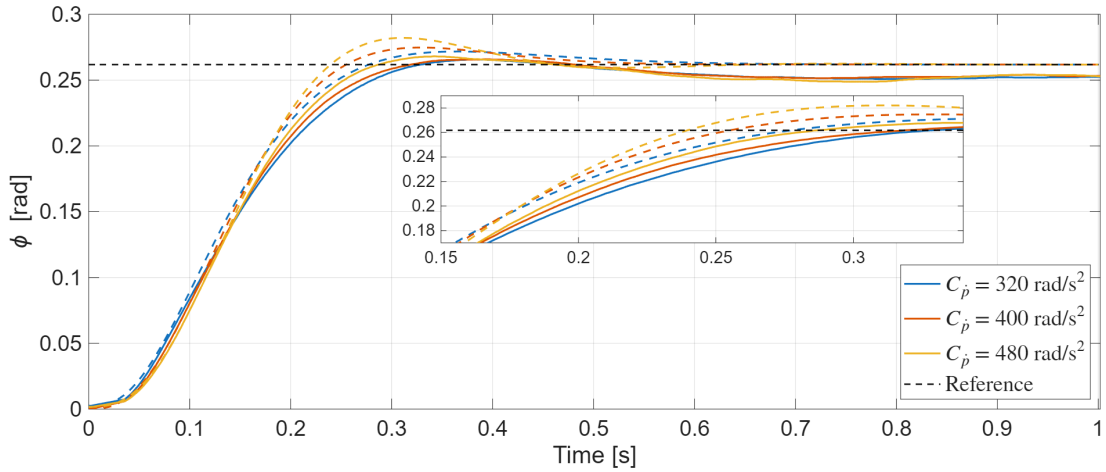
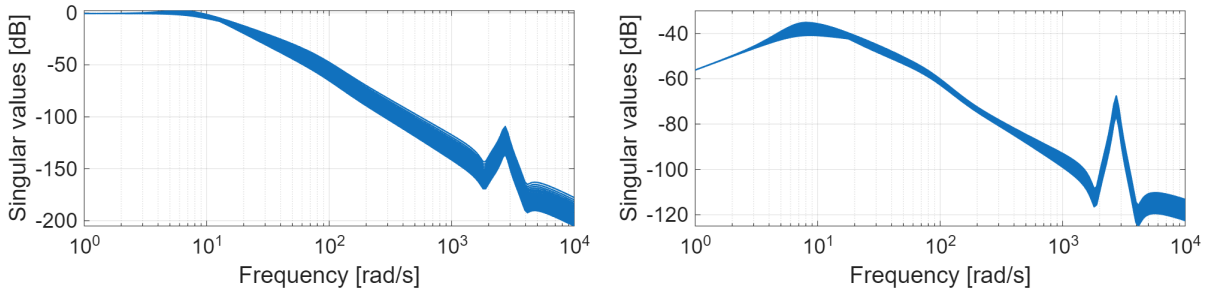


Figure 32. Experimental (solid) and simulated (dashed) roll response with uncertain parameters for $\tau = 28$ ms and different $C_{\dot{p}}$.



(a) Frequency response from the gyroscope noise n_Ω to the angular rate output Ω.

(b) Frequency response from the gyroscope noise n_Ω to motor setpoint m_c.

Figure 33. Frequency responses relevant to noise attenuation for all design points.

5

Conclusions & Recommendations

5.1. Conclusions

The research objective was to devise a method for obtaining a robust attitude controller for a quadcopter utilizing an INDI control system for the angular acceleration loop with parameters that are identified online. To solve this, the main research question that was investigated is: How can gain-scheduling be used for robust control of a multicopter utilizing an INDI control system for the angular acceleration loop with parameters that are identified online?

To this end, the signal-based $\mathcal{H}_\infty$ closed-loop shaping method was employed to design both a robust controller for a symmetric quadcopter and a gain-scheduled controller suitable for a wide range of actuator dynamics. For the symmetric quadrotor, it was found that a two-degree-of-freedom controller, combining feedback and feedforward components, effectively ensured robustness against uncertainties in identified parameters while maintaining low complexity. Good stability margins, robust stability, and disturbance rejection characteristics were found based on linear analysis. Additionally, consistent performance under uncertainty was validated through nonlinear simulations. These findings were further supported by flight experiments, where actuators were intentionally slowed down, and incorrect control coefficients were applied to the INDI controller. These showed consistent performance and a clear improvement of the feedforward controller over pure feedback control.

Combining co-design with the structured, signal-based $\mathcal{H}_\infty$ synthesis framework proved to be an effective method for gain-scheduling this controller against the actuator time constant. A controller was synthesized for time constants between $10\,\text{ms}$ and $80\,\text{ms}$ for which this approach optimized the controller to leverage additional plant bandwidth for optimal flight performance and disturbance rejection, while maintaining sufficient stability margins. Robust stability was also confirmed, and nonlinear simulations showed acceptable tracking performance under uncertainty for the entire time constant range. However, faster actuators were found to be at risk of saturation which could be solved in future work by extending the gain-schedule to include control effectiveness.

The gain-scheduled controller was further validated in practice by combining it with online parameter identification of all motor parameters. Since the online identification of these parameters was severely off from the nominal values obtained from bench tests and offline identification, it is difficult to draw definitive conclusions regarding the performance of the gain-scheduled controller. Nevertheless, for actuator time constants below $40\,\text{ms}$, the tracking performance does not exhibit poor behavior for all design points and it aligns well with the simulation results. This result is promising, as it demonstrates robustness of the controller to uncertainties larger than anticipated and suggests that the good nonlinear simulation results translate to real-world performance. For actuator time constants above $40\,\text{ms}$, the simulations no longer align with the experimental results and the quadrotor was poorly controllable. However, this appears to be caused by the extreme uncertainties as performance based on nominal parameters was significantly better. Further experiments under more realistic uncertainty levels are needed to fully validate the controller's robustness for slower actuators.

5.2. Recommendations

For future work, the steady-state error observed during experiments could be further investigated by incorporating additional effects, such as aerodynamics, into the model. Although the nominal model is applicable to any multirotor using this INDI system and equal gains, the current uncertainty analysis is limited to a quadrotor. Extending the analysis to configurations with additional rotors or unequal gains in different axes could provide insights into their effects on robustness. Furthermore, several uncertain parameters in the INDI controller, such as the B_2 matrix and the mapping from thrust setpoint to actual motor setting, were neglected in this study, and their effects under uncertainty could be explored. The $20\,\%$ uncertainty for the control effectiveness values was also partially based on intuition, and future work could explore a more systematic approach to obtain the uncertainty bounds. Finally, extending the gain schedule to account for control effectiveness could reduce the risk of actuator saturation for platforms with limited control authority, while extending this methodology to an outer position control loop would enable a complete control framework.

References

[1] M. Hassanalian and A. Abdelkefi, "Classifications, Applications, and Design Challenges of Drones: A Review," Progress in Aerospace Sciences, vol. 91, pp. 99–131, 2017. DOI: 10.1016/j.paerosci.2017.04.003

[2] R. Guebsi, S. Mami, and K. Chokmani, "Drones in Precision Agriculture: A Comprehensive Review of Applications, Technologies, and Challenges," Drones, vol. 8, no. 686, 11 2024. DOI: 10.3390/drones8110686

[3] T. M. Blaha, E. J. J. Smeur, and B. D. W. Remes, "Control of Unknown Quadrotors from a Single Throw," in 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2024, pp. 10 350–10 355. DOI: 10.1109/IROS58592.2024.10801514

[4] T. M. Blaha, E. J. J. Smeur, B. D. W. Remes, and C. C. de Visser, Flying a Quadrotor with Unknown Actuators and Sensor Configuration, Sep. 2024. DOI: 10.48550/arXiv.2409.01080

[5] E. J. Smeur, G. C. de Croon, and Q. Chu, "Gust disturbance alleviation with incremental nonlinear dynamic inversion," in 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2016, pp. 5626–5631. DOI: 10.1109/IROS.2016.7759827

[6] G. Hoffmann, H. Huang, S. Waslander, and C. Tomlin, "Quadrotor Helicopter Flight Dynamics and Control: Theory and Experiment," in AIAA Guidance, Navigation and Control Conference and Exhibit, Hilton Head, South Carolina: American Institute of Aeronautics and Astronautics, Aug. 20, 2007. DOI: 10.2514/6.2007-6461

[7] R. R. Da Costa, Q. P. Chu, and J. A. Mulder, "Reentry Flight Controller Design Using Nonlinear Dynamic Inversion," Journal of Spacecraft and Rockets, vol. 40, no. 1, pp. 64–71, Jan. 2003. DOI: 10.2514/2.3916

[8] S. Sieberling, Q. P. Chu, and J. A. Mulder, "Robust Flight Control Using Incremental Nonlinear Dynamic Inversion and Angular Acceleration Prediction," Journal of Guidance, Control, and Dynamics, vol. 33, no. 6, pp. 1732–1742, Nov. 2010. DOI: 10.2514/1.49978

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

[10] T. Pollack and E.-J. van Kampen, "Robust Stability and Performance Analysis of Incremental Dynamic-Inversion-Based Flight Control Laws," Journal of Guidance, Control, and Dynamics, vol. 46, no. 9, pp. 1785–1798, 2023. DOI: 10.2514/1.G006576

[11] R. van 't Veld, E. van Kampen, and Q. P. Chu, "Stability and Robustness Analysis and Improvements for Incremental Nonlinear Dynamic Inversion Control," in 2018 AIAA Guidance, Navigation, and Control Conference, AIAA-2018-1127, Kissimmee, FL, USA: American Institute of Aeronautics and Astronautics, 2018. DOI: 10.2514/6.2018-1127

[12] R. A. Cordeiro, A. S. Marton, J. R. Azinheira, J. R. H. Carvalho, and A. Moutinho, "Increased Robustness to Delay in Incremental Controllers Using Input Scaling Gain," IEEE Transactions on Aerospace and Electronic Systems, vol. 58, no. 2, pp. 1199–1210, 2022. DOI: 10.1109/TAES.2021.3123215

[13] Y. Huang, Y. Zhang, D. M. Pool, O. Stroosma, and Q. Chu, "Time-Delay Margin and Robustness of Incremental Nonlinear Dynamic Inversion Control," Journal of Guidance, Control, and Dynamics, vol. 45, no. 2, pp. 394–404, 2022. DOI: 10.2514/1.G006024

[14] B. Widrow and M. Kamenetsky, "On the Efficiency of Adaptive Algorithms," in Least-Mean-Square Adaptive Filters, John Wiley & Sons, Ltd, 2003, pp. 1–34. DOI: 10.1002/0471461288.ch1

[15] E. J. J. Smeur, G. C. H. E. de Croon, and Q. Chu, "Cascaded Incremental Nonlinear Dynamic Inversion for MAV Disturbance Rejection," Control Engineering Practice, vol. 73, pp. 79–90, Apr. 2018. DOI: 10.1016/j.conengprac.2018.01.003

[16] E. Lavretsky and K. A. Wise, "Robust Adaptive Control," in Robust and Adaptive Control: With Aerospace Applications, E. Lavretsky and K. A. Wise, Eds., Cham: Springer International Publishing, 2024, pp. 469–506. DOI: 10.1007/978-3-031-38314-4_11

[17] F. L. Lewis, D. L. Vrabie, and V. L. Syrmos, "Optimal Control of Continuous-Time Systems," in Optimal Control, John Wiley & Sons, Ltd, 2012, ch. 3, pp. 110–176. DOI: 10.1002/9781118122631.ch3

[18] A. Srivastava, S. Indu, and R. Sharma. "Design And Flight Testing Of LQRi Attitude Control For Quadcopter UAV." arXiv: 2404.12261 [cs].

[19] P. Foehn and D. Scaramuzza, "Onboard State Dependent LQR for Agile Quadrotors," in 2018 IEEE International Conference on Robotics and Automation (ICRA), May 2018, pp. 6566–6572. DOI: 10.1109/ICRA.2018.8460885

[20] J. Gadewadikar, F. Lewis, K. Subbarao, and B. M. Chen, "Structured H-Infinity Command and Control-Loop Design for Unmanned Helicopters," Journal of Guidance, Control, and Dynamics, vol. 31, no. 4, pp. 1093–1102, Jul. 2008. DOI: 10.2514/1.31377

[21] F. L. Lewis, D. L. Vrabie, and V. L. Syrmos, "The Tracking Problem and Other LQR Extensions," in Optimal Control, John Wiley & Sons, Ltd, 2012, ch. 4, pp. 177–212. DOI: 10.1002/9781118122631.ch4

[22] N. Koksal, M. Jalalmaab, and B. Fidan, "Adaptive Linear Quadratic Attitude Tracking Control of a Quadrotor UAV Based on IMU Sensor Data Fusion," Sensors, vol. 19, no. 1, p. 46, 1 Jan. 2019. DOI: 10.3390/s19010046

[23] D. Bates and I. Postlethwaite, Robust Multivariable Control of Aerospace Systems. DUP Science, 2002, 220 pp.

[24] G. Zames, "Feedback and Optimal Sensitivity: Model Reference Transformations, Multiplicative Seminorms, and Approximate Inverses," IEEE Transactions on Automatic Control, vol. 26, no. 2, pp. 301–320, 1981. DOI: 10.1109/TAC.1981.1102603

[25] J. Doyle, K. Glover, P. Khargonekar, and B. Francis, "State-Space Solutions to Standard H_2 and $\mathcal{H}_\infty$ Control Problems," IEEE Transactions on Automatic Control, vol. 34, no. 8, pp. 831–847, Aug. 1989. DOI: 10.1109/9.29425

[26] P. Gahinet and P. Apkarian, "A Linear Matrix Inequality Approach to $\mathcal{H}_\infty$ Control," International Journal of Robust and Nonlinear Control, vol. 4, no. 4, pp. 421–448, 1994. DOI: 10.1002/rnc.4590040403

[27] P. Apkarian and D. Noll, "The $\mathcal{H}_\infty$ Control Problem is Solved," AerospaceLab Journal, vol. Issue 13, 11 pages, 2017. DOI: 10.12762/2017.AL13-01

[28] P. Apkarian and D. Noll, "Nonsmooth $\mathcal{H}_\infty$ Synthesis," IEEE Transactions on Automatic Control, vol. 51, no. 1, pp. 71–86, Jan. 2006. DOI: 10.1109/TAC.2005.860290

[29] D. Schoon, "Review of $\mathcal{H}_\infty$ Static Output Feedback Controller Synthesis Methods," M.S. thesis, Delft University of Technology, Delft, Jan. 2024.

[30] Y. Huang, T. Pe, A. P. Popov, H. Werner, and F. Thielecke, "Control of a Two-Load-Path Trimmable Horizontal Stabilizer Actuator of an Aircraft — Comparison of $\mathcal{H}_\infty$ Design Approaches," in 49th IEEE Conference on Decision and Control (CDC), 2010, pp. 4863–4868. DOI: 10.1109/CDC.2010.5716948

[31] H. Kwakernaak, "Robustness optimization of linear feedback systems," in The 22nd IEEE Conference on Decision and Control, 1983, pp. 618–624. DOI: 10.1109/CDC.1983.269592

[32] J. Sefton and K. Glover, "Pole/Zero Cancellations in the General $\mathcal{H}_\infty$ Problem with Reference to a Two Block Design," Systems & Control Letters, vol. 14, no. 4, pp. 295–306, 1990. DOI: 10.1016/0167-6911(90)90050-5

[33] M. Tsai, E. Geddes, and I. Postlethwaite, "Pole-Zero Cancellations and Closed-Loop Properties of an H/Sup Infinity / Mixed Sensitivity Design Problem," in 29th IEEE Conference on Decision and Control, 1990, 1028–1029 vol.2. DOI: 10.1109/CDC.1990.203754

[34] H. Kwakernaak, "Robust Control and $\mathcal{H}_\infty$-Optimization—Tutorial Paper," Automatica, vol. 29, no. 2, pp. 255–273, 1993. DOI: 10.1016/0005-1098(93)90122-A

[35] L. i. y. u. Cao and Y. O. I. C. H. I. Hori, "Mixed Sensitivity Optimization to Avoid Pole/Zero Cancellation," Automatica, vol. 33, no. 7, pp. 1379–1385, Jul. 1, 1997. DOI: 10.1016/S0005-1098(97)00029-0

[36] K. Glover and D. McFarlane, "Robust Stabilization of Normalized Coprime Factor Plant Descriptions with $\mathcal{H}_\infty$ Bounded Uncertainty," IEEE Transactions on Automatic Control, vol. 34, no. 8, pp. 821–830, Aug. 1989. DOI: 10.1109/9.29424

[37] M. Baskın and K. Leblebicioğlu, "Robust Attitude Controller Design for an Uncommon Quadrotor With Big and Small Tilt Rotors," Journal of Mechanisms and Robotics, vol. 15, no. 031003, Feb. 21, 2023. DOI: 10.1115/1.4056826

[38] F. Biertümpfel, H. Pfifer, and J. Theis, "Robust Space Launcher Control with Time-Varying Objectives," Journal of Guidance, Control, and Dynamics, vol. 47, no. 5, pp. 934–944, 2024. DOI: 10.2514/1.G007632

[39] S. Pineau, S. Theodoulis, M. Zasadzinski, and M. Boutayeb, "Autopilot Design for Dual-Spin Projectiles using Incremental Nonlinear Dynamic Inversion," in 2022 30th Mediterranean Conference on Control and Automation (MED), Jun. 2022, pp. 444–449. DOI: 10.1109/MED54222.2022.9837217

[40] J. Thomas, J. Currie, and D. Wilson, "Automatic Weight Selection for Mixed Sensitivity Optimization Control of a Quadrotor," in 2017 24th International Conference on Mechatronics and Machine Vision in Practice (M2VIP), Nov. 2017, pp. 1–6. DOI: 10.1109/M2VIP.2017.8211477

[41] M. Hachem, C. Roos, T. Miquel, and M. Bronz, Improving Incremental Nonlinear Dynamic Inversion Robustness Using Robust Control in Aerial Robotics, 2025. DOI: 10.48550/arXiv.2501.07223 arXiv: 2501.07223 [cs.RO].

[42] D. C. McFarlane and K. Glover, Eds., Robust Controller Design Using Normalized Coprime Factor Plant Descriptions (Lecture Notes in Control and Information Sciences). Berlin/Heidelberg: Springer-Verlag, 1990, vol. 138. DOI: 10.1007/BFb0043199

[43] D. McFarlane and K. Glover, "A Loop-Shaping Design Procedure Using $\mathcal{H}_\infty$ Synthesis," IEEE Transactions on Automatic Control, vol. 37, no. 6, pp. 759–769, Jun. 1992. DOI: 10.1109/9.256330

[44] S. Skogestad and I. Postlethwaite, Multivariable Feedback Control: Analysis and Design, 2nd. Chichester, England: John Wiley, 2005.

[45] M. Chen and M. Huzmezan, "A Simulation Model and $\mathcal{H}_\infty$ Loop Shaping Control of a Quad Rotor Unmanned Air Vehicle," in Proceedings of the IASTED International Conference on Modelling, Simulation and Optimatization, Jan. 2003, pp. 320–325.

[46] K. Turkoglu and E. Jafarov, "$\mathcal{H}_\infty$ Loop Shaping Robust Control vs. Classical PI (D) Control: A Case Study on the Longitudinal Dynamics of Hezarfen UAV," in Proceedings of the 2nd WSEAS International Conference on Dynamical Systems and Control, Oct. 2006. DOI: 10.13140/2.1.1687.3921

[47] J. Li, Z. Yang, Q. Zhang, C. Xu, and H. Xu, "A $\mathcal{H}_\infty$ Loop Shaping Control for a Quadcopter with Tilting Rotor," in 2018 IEEE CSAA Guidance, Navigation and Control Conference (CGNCC), 2018, pp. 1–7. DOI: 10.1109/GNCC42960.2018.9018663

[48] J. G. R. Cardenas, "INDI and $\mathcal{H}_\infty$ LSDP in quadrotors," M.S. thesis, Delft University of Technology, Delft, Oct. 2024.

[49] P. Seiler, A. Packard, and P. Gahinet, "An Introduction to Disk Margins [Lecture Notes]," IEEE Control Systems Magazine, vol. 40, no. 5, pp. 78–95, 2020-10. DOI: 10.1109/MCS.2020.3005277

[50] S. Theodoulis and M. Proff, "Robust Flight Control Tuning for Highly Agile Missiles," in AIAA SciTech Forum, American Institute of Aeronautics and Astronautics, 2021. DOI: 10.2514/6.2021-1568

[51] J. Doyle, "Analysis of Feedback Systems with Structured Uncertainties," IEE Proceedings D Control Theory and Applications, vol. 129, no. 6, pp. 242–250, Nov. 1982. DOI: 10.1049/ip-d.1982.0053

[52] A. Hamza, A. H. Mohamed, and A. El-Badawy, "Robust $\mathcal{H}_\infty$ Control for a Quadrotor UAV," in AIAA SCITECH 2022 Forum, 2022. DOI: 10.2514/6.2022-2033

[53] A. Hossam and A. El-Badawy, "Mu-Based Trajectory Tracking Control for a Quad-Rotor UAV," Control Theory and Technology, vol. 20, no. 4, pp. 536–554, Nov. 1, 2022. DOI: 10.1007/s11768-022-00114-x

[54] T. Pollack, "Advances in Dynamic Inversion-based Flight Control Law Design: Multivariable Analysis and Synthesis of Robust and Multi-Objective Design Solutions," Delft University of Technology, 2024. DOI: 10.4233/UUID:28617BA0-461D-48EF-8437-DE2AA41034EA Accessed: Feb. 3, 2025.

[55] K. Masuda and K. Uchiyama, "Flight Controller Design Using μ-Synthesis for Quad Tilt-Wing UAV," in AIAA Scitech 2019 Forum, 2019. DOI: 10.2514/6.2019-1918

[56] S. Panza, M. Lovera, M. Sato, and K. Muraoka, "Structured μ-Synthesis of Robust Attitude Control Laws for Quad-Tilt-Wing Unmanned Aerial Vehicle," Journal of Guidance, Control, and Dynamics, vol. 43, no. 12, pp. 2258–2274, 2020. DOI: 10.2514/1.G005080

[57] B. Ata and M. C. Gencal, "Comparison of Optimization Approaches on Linear Quadratic Regulator Design for Trajectory Tracking of a Quadrotor," Evolutionary Intelligence, vol. 17, no. 5-6, pp. 3225–3240, Oct. 2024. DOI: 10.1007/s12065-024-00928-5

[58] P. Feyel, G. Duc, and G. Sandou, "Enhancing Optimal Weight Tuning in $\mathcal{H}_\infty$ Loop-Shaping Control with Particle Swarm Optimization," in IJCCI 2013 - Proceedings of the 5th International Joint Conference on Computational Intelligence, Sep. 2013.

[59] Y. Kanokmedhakul, S. Bureerat, N. Panagant, T. Radpukdee, N. Pholdee, and A. R. Yildiz, Metaheuristic-assisted Complex $\mathcal{H}_\infty$ Flight Control Tuning for the Hawkeye Unmanned Aerial Vehicle: A Comparative Study, 2024. DOI: 10.1016/j.eswa.2024.123428

[60] C. A. Pérez, S. Theodoulis, F. Sève, and L. Goerig, "Automatic Weighting Filter Tuning for Robust Flight Control Law Design," IFAC-PapersOnLine, 18th IFAC Workshop on Control Applications of Optimization CAO 2022, vol. 55, no. 16, pp. 400–405, Jan. 1, 2022. DOI: 10.1016/j.ifacol.2022.09.057

[61] F. J. Rhenman and S. Theodoulis, "Robust Gain Scheduled Longitudinal Control of the ADMIRE Using Parametric Robust $\mathcal{H}_\infty$ Control," in AIAA SCITECH 2025 Forum. DOI: 10.2514/6.2025-2245

[62] B. Lian, W. Xue, F. L. Lewis, H. Modares, and B. Kiumarsi, "Background on Integral and Inverse Reinforcement Learning for Feedback Control," in Integral and Inverse Reinforcement Learning for Optimal Control Systems and Games, Cham: Springer Nature Switzerland, 2024, pp. 11–36. DOI: 10.1007/978-3-031-45252-9_2

[63] R. Howard, Dynamic Programming and Markov Processes. Technology Press of Massachusetts Institute of Technology, 1960.

[64] D. Kleinman, "On an Iterative Technique for Riccati Equation Computations," IEEE Transactions on Automatic Control, vol. 13, no. 1, pp. 114–115, Feb. 1968. DOI: 10.1109/TAC.1968.1098829

[65] D. Vrabie, O. Pastravanu, M. Abu-Khalaf, and F. L. Lewis, "Adaptive Optimal Control for Continuous-Time Linear Systems Based on Policy Iteration," Automatica, vol. 45, no. 2, pp. 477–484, 2008. DOI: 10.1016/j.automatica.2008.08.017

[66] H. Modares and F. L. Lewis, "Linear Quadratic Tracking Control of Partially-Unknown Continuous-Time Systems Using Reinforcement Learning," IEEE Transactions on Automatic Control, vol. 59, no. 11, pp. 3051–3056, Nov. 2014. DOI: 10.1109/TAC.2014.2317301

[67] H. Modares, F. L. Lewis, and Z.-P. Jiang, "Optimal Output-Feedback Control of Unknown Continuous-Time Linear Systems Using Off-policy Reinforcement Learning," IEEE Transactions on Cybernetics, vol. 46, no. 11, pp. 2401–2410, 2016. DOI: 10.1109/TCYB.2015.2477810

[68] T. McLain, "Successive Galerkin Approximation Algorithms for Nonlinear Optimal and Robust Control," International Journal of Control, vol. 71, Feb. 1998. DOI: 10.1080/002071798221542

[69] H.-N. Wu and B. Luo, "Neural Network Based Online Simultaneous Policy Update Algorithm for Solving the HJI Equation in Nonlinear $\mathcal{H}_\infty$ Control," IEEE Transactions on Neural Networks and Learning Systems, vol. 23, no. 12, pp. 1884–1895, Dec. 2012. DOI: 10.1109/TNNLS.2012.2217349

[70] M. Abu-Khalaf, F. L. Lewis, and J. Huang, "Policy Iterations on the Hamilton–Jacobi–Isaacs Equation for $\mathcal{H}_\infty$ State Feedback Control With Input Saturation," ResearchGate, Jan. 2007. DOI: 10.1109/TAC.2006.884959

[71] K. G. Vamvoudakis and F. Lewis, "Online Solution of Nonlinear Two-Player Zero-Sum Games Using Synchronous Policy Iteration," in 49th IEEE Conference on Decision and Control (CDC), Dec. 2010, pp. 3040–3047. DOI: 10.1109/CDC.2010.5717607

[72] R. S. Sutton and A. G. Barto, Reinforcement Learning: An Introduction, 1st ed. MIT Press, 1998.

[73] W. J. Rugh and J. S. Shamma, "Research on Gain Scheduling," Automatica, vol. 36, no. 10, pp. 1401–1425, 2000. DOI: 10.1016/S0005-1098(00)00058-3

[74] H. Lhachemi, D. Saussié, and G. Zhu, "Gain-Scheduling Control Design in the Presence of Hidden Coupling Terms via Eigenstructure Assignment: Application to a Pitch-axis Missile Autopilot," in 2016 American Control Conference (ACC), 2016, pp. 4041–4046. DOI: 10.1109/ACC.2016.7525556

[75] A. Milhim, Y. Zhang, and C.-A. Rabbath, "Gain Scheduling Based PID Controller for Fault Tolerant Control of Quad-Rotor UAV," in AIAA infotech@Aerospace 2010, 2010. DOI: 10.2514/6.2010-3530

[76] I. Sadeghzadeh, A. Mehta, A. Chamseddine, and Y. Zhang, "Active Fault Tolerant Control of a quadrotor UAV based on gainscheduled PID control," in 2012 25th IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), Apr. 2012, pp. 1–4. DOI: 10.1109/CCECE.2012.6335037

[77] J. Qiao, Z. Liu, and Y. Zhang, "Gain Scheduling PID Control of the Quad-Rotor Helicopter," in 2017 IEEE International Conference on Unmanned Systems (ICUS), 2017, pp. 1594–1601. DOI: 10.1109/ICUS.2017.8278414

[78] D.-T. Nguyen, D. Saussié, and L. Saydy, "Robust Self-Scheduled Fault-Tolerant Control of a Quadrotor UAV," IFAC-PapersOnLine, 20th IFAC World Congress, vol. 50, no. 1, pp. 5761–5767, Jul. 2017. DOI: 10.1016/j.ifacol.2017.08.1141

[79] D.-T. Nguyen, D. Saussié, and L. Saydy, "Fault-Tolerant Control of a Hexacopter UAV based on Self-Scheduled Control Allocation," in 2018 International Conference on Unmanned Aircraft Systems (ICUAS), Jun. 2018, pp. 385–393. DOI: 10.1109/ICUAS.2018.8453440

[80] D.-T. Nguyen, D. Saussié, and L. Saydy, "Universal Adaptive Fault-Tolerant Control of a Multicopter UAV," IFAC-PapersOnLine, 21st IFAC World Congress, vol. 53, no. 2, pp. 9340–9347, Jan. 2020. DOI: 10.1016/j.ifacol.2020.12.2390

[81] H. Lhachemi, D. Saussié, and G. Zhu, "A Robust and Self-Scheduled Longitudinal Flight Control System: A Multi-Model and Structured $\mathcal{H}_\infty$ Approach," in AIAA Guidance, Navigation, and Control Conference, National Harbor, Maryland: American Institute of Aeronautics and Astronautics (AIAA), Jan. 2014. DOI: 10.2514/6.2014-0601

[82] S. Theodoulis, M. Proff, and C. Marchand, "Robust Design for Highly Agile Missile Autopilots," in 2020 28th Mediterranean Conference on Control and Automation (MED), 2020, pp. 67–72. DOI: 10.1109/MED48518.2020.9183304

[83] A. M. Simões and V. M. G. B. Cavalcanti, "Missile Autopilot Design via Structured Robust Linear Parameter-Varying Synthesis," Journal of Guidance, Control, and Dynamics, vol. 46, no. 8, pp. 1649–1656, 2023. DOI: 10.2514/1.G007580

[84] F. R. López-Estrada, J.-C. Ponsart, D. Theilliol, Y. Zhang, and C.-M. Astorga-Zaragoza, "LPV Model-Based Tracking Control and Robust Sensor Fault Diagnosis for a Quadrotor UAV," Journal of Intelligent & Robotic Systems, vol. 84, no. 1, pp. 163–177, Dec. 1, 2016. DOI: 10.1007/s10846-015-0295-y

[85] A. Serirojanakul and M. Wongsaisuwan, "Optimal Control of Quad-rotor Helicopter Using State Feedback LPV Method," in 2012 9th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, 2012, pp. 1–4. DOI: 10.1109/ECTICon.2012.6254219

[86] D. Rotondo, F. Nejjari, and V. Puig, "Model reference quasi-lpv control of a quadrotor uav," in 2014 IEEE Conference on Control Applications (CCA), 2014, pp. 736–741. DOI: 10.1109/CCA.2014.6981428

[87] D. Rotondo, F. Nejjari, and V. Puig, "Robust Quasi-LPV Model Reference FTC of a Quadrotor UAV Subject To Actuator Faults," Int. J. Appl. Math. Comput. Sci., vol. 25, no. 1, pp. 7–22, Mar. 2015. DOI: 10.1515/amcs-2015-0001

[88] J. Mota Campos, D. Cardoso, and G. Raffo, "A Robust Adaptive $\mathcal{H}_\infty$ Controller for Full Flight Envelope of a Convertible UAV," Congresso Brasileiro de Automática - CBA, vol. 3, Oct. 2022. DOI: 10.20906/CBA2022/3515

[89] A. Megretski and A. Rantzer, "System analysis via integral quadratic constraints," IEEE Transactions on Automatic Control, vol. 42, no. 6, pp. 819–830, Jun. 1997. DOI: 10.1109/9.587335

[90] G. M. Vinco, O. Sename, G. Strub, and S. Theodoulis, "Linear Parameter-Varying Polytopic Modeling and Control Design for Guided Projectiles," Journal of Guidance, Control, and Dynamics, vol. 47, no. 3, pp. 433–447, 2024. DOI: 10.2514/1.G007726

[91] S. Chaabani and N. Azouz, "Quasi-LPV Approach for the Stabilization of an Innovative Quadrotor," Modelling, vol. 6, no. 3, 2025. DOI: 10.3390/modelling6030060

[92] J. G. Ziegler and N. B. Nichols, "Optimum Settings for Automatic Controllers," 1942.

[93] S. Gaikwad, S. Dash, and G. Stein, "Auto-Tuning PID Using Loop-Shaping Ideas," in Proceedings of the 1999 IEEE International Conference on Control Applications, vol. 1, Aug. 1999, 589–593 vol. 1. DOI: 10.1109/CCA.1999.806712

[94] S.-E.-I. Hasseni and L. Abdou, "Decentralized PID Control by Using GA Optimization Applied to a Quadrotor," Journal of Automation, Mobile Robotics and Intelligent Systems, pp. 33–44, Jul. 2018. DOI: 10.14313/JAMRIS_2-2018/9

[95] K. Khuwaja, N.-u.-Z. Lighari, I. C. Tarca, and R. C. Tarca, "PID Controller Tuning Optimization with Genetic Algorithms for a Quadcopter," Recent Innovations in Mechatronics, vol. 5, no. 1, pp. 1–7. Apr. 2018. DOI: 10.17667/riim.2018.1/11

[96] A. Noordin, M. A. M. Basri, Z. Mohamed, and A. F. Z. Abidin, "Modelling and PSO Fine-tuned PID Control of Quadrotor UAV," International Journal on Advanced Science, Engineering and Information Technology, vol. 7, no. 4, pp. 1367–1373, Aug. 2017. DOI: 10.18517/ijaseit.7.4.3141

[97] J. Moreno-Valenzuela, R. Pérez-Alcocer, M. Guerrero-Medina, and A. Dzul, "Nonlinear PID-Type Controller for Quadrotor Trajectory Tracking," IEEE/ASME Transactions on Mechatronics, vol. 23, no. 5, pp. 2436–2447, Oct. 2018. DOI: 10.1109/TMECH.2018.2855161

[98] X. Li, J. Wu, C. Qi, and P. Cong, "Deep Reinforcement Learning and L1 Adaptive Control Algorithm-Based Attitude Control of Fixed-Wing UAVs," Lecture Notes in Electrical Engineering, vol. 1010 LNEE, pp. 2273–2285, 2023. DOI: 10.1007/978-981-99-0479-2_212

[99] F. L. Lewis, "L1 Adaptive Control Theory: Guaranteed Robustness with Fast Adaptation," IEEE Control Systems Magazine, vol. 31, no. 5, pp. 112–114, 2011. DOI: 10.1109/MCS.2011.941837

[100] S. Abdelmaksoud, M. Mailah, and A. Abdallah, "Improving Disturbance Rejection Capability for a Quadcopter UAV System Using Self-Regulating Fuzzy PID Controller," in Proceedings of: 2020 International Conference on Computer, Control, Electrical, and Electronics Engineering, 2021. DOI: 10.1109/ICCCEEE49695.2021.9429661

[101] V. Tran, F. Santoso, M. Garratt, and I. Petersen, "Fuzzy Self-Tuning of Strictly Negative-Imaginary Controllers for Trajectory Tracking of a Quadcopter Unmanned Aerial Vehicle," IEEE Transactions on Industrial Electronics, vol. 68, no. 6, pp. 5036–5045, 2021. DOI: 10.1109/TIE.2020.2988219

[102] D.-T. Nguyen, D. Saussie, and L. Saydy, "Design and Experimental Validation of Robust Self-Scheduled Fault-Tolerant Control Laws for a Multicopter UAV," IEEE/ASME Transactions on Mechatronics, vol. 26, no. 5, pp. 2548–2557, Oct. 2021. DOI: 10.1109/TMECH.2020.3042333